SESSION

PROTEIN CLASSIFICATION, STRUCTURE PREDICTION, AND COMPUTATIONAL STRUCTURAL BIOLOGY

Chair(s)

TBA

ISBN: 1-60132-427-8, CSREA Press ©

ISBN: 1-60132-427-8, CSREA Press ©

Bidirectional Representation and Backpropagation Learning

Olaoluwa Adigun and Bart Kosko
Department of Electrical Engineering
Signal and Image Processing Institute
University of Southern California

Abstract— *The backpropagation learning algorithm extends to bidirectional training of multilayer neural networks. The bidirectional operation gives a form of backward chaining or backward inference from a network output. We first prove that a fixed three-layer network of threshold neurons can exactly represent any finite permutation function and its inverse. The forward pass gives the function value. The backward pass through the same network gives the inverse value. We then derive and test a bidirectional version of the backpropagation algorithm that can learn bidirectional mappings or their approximations.*

Keywords: bidirectional associative memory, backpropagation learning, function representation, backward inference

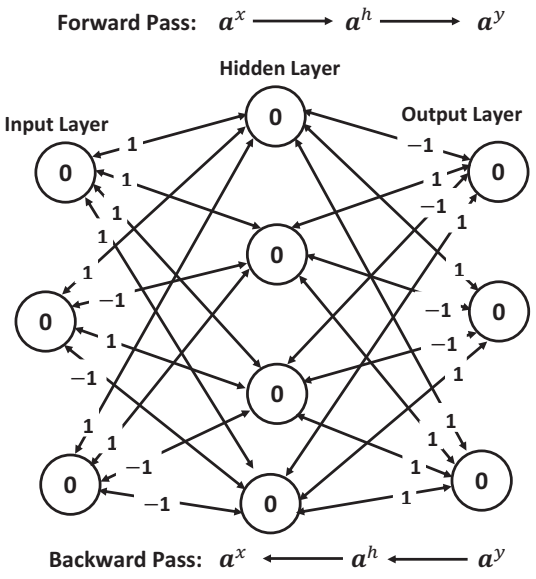


Fig. 1: *Bidirectional Representation of a Permutation Function. This 3-layer bidirectional threshold network exactly represents the invertible 3-bit bipolar permutation function f in Table 1. The forward pass feeds the input vector x to the input layer and passes it through the weighted links and the hidden layer of threshold neurons (each with zero threshold) to the output layer. The backward pass sends the output bit vector y back through the same weighted links and threshold neurons. The network computes $y = f(x)$ on the forward pass and the inverse value $f^{-1}(y)$ on the backward pass.*

1. Bidirectional Backpropagation

We show that bidirectional backpropagation (B-BP) training endows a multilayered neural network $N\colon \mathbb{R}^n \to \mathbb{R}^p$ with a form of backward inference. The forward pass gives the usual predicted neural output $N(x)$ given a vector input x. The output vector value $y = N(x)$ in effect answers the *what-if* question that x poses: What would we observe if x occurred? What would be the effect? Then the backward pass answers the *why* question that y poses: Why did y occur? What type of input would cause y? Feedback convergence to a resonating bidirectional fixed-point attractor [1], [2] gives a long-term or equilibrium answer to both the what-if and why questions.

This bidirectional approach to neural learning applies to big data because the BP algorithm [3], [4], [5] scales linearly with training data. BP has time complexity $O(n)$ for n training samples because the forward pass is $O(1)$ while the backward pass is $O(n)$. So the new B-BP algorithm still has only $O(n)$ complexity. This linear scaling does not hold in general for most machine-learning algorithms. An example is the quadratic complexity $O(n^2)$ of support-vector kernel methods [6].

We present the bidirectional results in two parts. The first part proves that there exist fixed-weight multilayer threshold networks that can exactly represent some invertible functions. Theorem 1 shows that this holds for all finite bipolar (or binary) permutation functions. Figure 1 shows such a bidirectional 3-layer network of zero-threshold neurons. It exactly represents the 3-bit permutation function f in Table 1 where $\{-,-,+\}$ denotes $\{-1,-1,1\}$. So f is a self-bijection that rearranges the 8 vectors in the bipolar hypercube $\{-1,1\}^3$. The forward pass converts the input bipolar vector $(1,1,1)$ into the output bipolar vector $(-1,-1,1)$. The backward pass converts $(-1,-1,1)$ into $(1,1,1)$ over the *same* fixed synaptic connection weights. These same weights and neurons convert the other 7 input vectors in the first column of Table 1 to the corresponding 7 output vectors in the second column and conversely.

Theorem 1 requires 2^n hidden neurons to represent a permutation function on the bipolar hypercube $\{-1,1\}^n$. Using so many hidden neurons is neither practical nor necessary. The representation in Figure 1 uses only 4 hidden neurons. It is just one example of a representation that uses fewer than 8 hidden neurons. We seek instead an efficient learning

algorithm that can learn bidirectional representations (or at least approximations) from sample data.

The second part extends the BP algorithm to just this bidirectional case. This takes some care because training the same weights in one direction tends to overwrite or undo the BP training in the other direction. The B-BP algorithm solves this problem by minimizing a joint error. It found representations of the permutation in Table 1 that needed only 3 hidden neurons.

The learning approximation also improves by adding more hidden neurons. Figure 2 shows the effect of training with 100 hidden neurons. Figure 3 shows how the B-BP training error falls off as the number of hidden neurons grows when learning the 5-bit permutation in Table 2.

2. Bidirectional Function Representation of Bipolar Permutations

This section proves that there exists multilayered neural networks that can exactly bidirectionally represent some invertible functions. We first define the network variables. The proof uses threshold neurons while the B-BP algorithm uses soft-threshold logistic sigmoids for hidden neurons and uses identity activations for input and output neurons.

A bidirectional neural network is a multilayer network $N\colon X \to Y$ that maps the input space X to the output space Y *and conversely* through the same set of weights. The backward pass uses the transpose matrices of the weight matrices that the forward pass uses. Such a network is a bidirectional associative memory or BAM [1], [2].

The forward pass sends input vector x through weight matrix $\mathbf{W}$ from the input layer to the hidden layer and then on through matrix $\mathbf{U}$ to the output layer. The backward pass sends the output y from the output layer back through the hidden layer to the input layer. Let I, J, and K denote the respective number of input, hidden, and output neurons. Then the $I \times J$ matrix $\mathbf{W}$ connects the input layer to the hidden. The $J \times K$ matrix $\mathbf{U}$ connects the hidden layer to the output layer.

Table 1: 3-Bit Bipolar Permutation Function f.

Input x	Output t
$[+\,+\,+]$	$[-\,-\,+]$
$[+\,+\,-]$	$[-\,+\,+]$
$[+\,-\,+]$	$[+\,+\,+]$
$[+\,-\,-]$	$[+\,-\,+]$
$[-\,+\,+]$	$[-\,+\,+]$
$[-\,+\,-]$	$[-\,-\,-]$
$[-\,-\,+]$	$[+\,-\,-]$
$[-\,-\,-]$	$[+\,+\,-]$

The hidden-neuron input o_j^h has the affine form

$$o_j^h = \sum_{i=1}^{I} w_{ji} a_i^x(x^i) + b_j^h \tag{1}$$

where weight w_{ji} connects the i^{th} input neuron to the j^{th} hidden neuron, a_i^x is the activation of the i^{th} input neuron, and b_j^h is the bias term of the j^{th} hidden neuron. The activation a_j^h of the j^{th} hidden neuron is a bipolar threshold:

$$a_j^h(o_j^h) = \begin{cases} -1 & \textit{if } o_j^h \leq 0 \\ 1 & \textit{if } o_j^h > 0\ . \end{cases} \tag{2}$$

The B-BP algorithm in the next section uses soft-threshold bipolar logistic functions for the hidden activations because such sigmoid functions are differentiable. The proof below also modifies the hidden thresholds to take on binary values in (13) and to fire with a slightly different condition.

The input o_k^y to the k^{th} output neuron from the hidden layer is also affine:

$$o_k^y = \sum_{j=1}^{J} u_{kj} a_j^h + b_k^y \tag{3}$$

where weight u_{kj} connects the j^{th} hidden neuron to the k^{th} output neuron . Term b_k^y is the additive bias of the k^{th} output neuron. The output activation vector $\mathbf{a}^y$ gives the predicted outcome or target on the forward pass. The k^{th} output neuron has bipolar threshold activation a_k^y:

$$a_k^y(o_k^y) = \begin{cases} -1 & \text{if } o_k^y \leq 0 \\ 1 & \text{if } o_k^y > 0\ . \end{cases} \tag{4}$$

The forward pass of an input bipolar vector $\mathbf{x}$ from Table 1 through the network in Figure 1 gives an output activation vector $\mathbf{a}^y$ that equals the table's corresponding target vector $\mathbf{y}$. The backward pass feeds $\mathbf{y}$ from the output layer back through the hidden layer to the input layer. Then the backward-pass input o_j^{hb} to the j^{th} hidden neuron is

$$o_j^{hb} = \sum_{k=1}^{K} u_{kj} y^k + b_j^h \tag{5}$$

where y^k is the output of the k^{th} output neuron. The backward-pass activation of the j^{th} hidden neuron a_j^{hb} is

$$a_j^{hb}(o_j^{hb}) = \begin{cases} -1 & \textit{if } o_j^{hb} \leq 0 \\ 1 & \textit{if } o_j^{hb} > 0\ . \end{cases} \tag{6}$$

The backward-pass input o_i^{xb} to the i^{th} input neuron is

$$o_i^{xb} = \sum_{j=1}^{J} w_{ji} a_j^{hb} + b_i^x \tag{7}$$

where b_i^x is the bias for the i^{th} input neuron. The input-layer activation $\mathbf{a}^x$ gives the predicted value for the backward pass. The i^{th} input neuron has bipolar activation

$$a_i^x(o_i^{xb}) = \begin{cases} -1 & \textit{if } o_i^{xb} \leq 0 \\ 1 & \textit{if } o_i^{xb} > 0\ . \end{cases} \tag{8}$$

We can now state and prove the bidirectional representation theorem for bipolar permutations. The theorem also applies to binary permutations because the input and output neurons have bipolar threshold activations.

Theorem 1: Exact Bidirectional Representation of Bipolar Permutation Functions. *Suppose that the invertible function $f\colon \{-1,1\}^n \to \{-1,1\}^n$ is a permutation. Then there exists a 3-layer bidirectional neural network $N\colon \{-1,1\}^n \to \{-1,1\}^n$ that exactly represents f in the sense that $N(x) = f(x)$ and $N^{-1}(x) = f^{-1}(x)$ for all x.*

Proof: The proof strategy picks weight matrices $\mathbf{W}$ and $\mathbf{U}$ so that only one hidden neuron fires on both the forward and the backward pass. So we structure the network such that any input vector $\mathbf{x}$ fires only one hidden neuron on the forward pass and such that the output vector $\mathbf{y} = \mathbf{N}(\mathbf{x})$ fires only the same hidden neuron on the backward pass.

The bipolar permutation f is a bijective map of the bipolar hypercube $\{-1,1\}^n$ into itself. The bipolar hypercube contains the 2^n input bipolar column vectors $\mathbf{x_1}, \mathbf{x_2}, \dots, \mathbf{x_{2^n}}$. It likewise contains the 2^n output bipolar vectors $\mathbf{y_1}, \mathbf{y_2}, \dots, \mathbf{y_{2^n}}$. The network will use 2^n corresponding hidden threshold neurons. So $J = 2^n$.

Matrix $\mathbf{W}$ connects the input layer to the hidden layer. Matrix $\mathbf{U}$ connects the hidden layer to output layer. Define $\mathbf{W}$ so that each row lists all 2^n bipolar input vectors and define $\mathbf{U}$ so that each column lists all 2^n transposed bipolar output vectors:

$$\mathbf{W} = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{x_1} & \mathbf{x_2} & \vdots & \vdots & \mathbf{x_{2^n}} \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

$$\mathbf{U} = \begin{bmatrix} \cdots & \mathbf{y_1}^T & \cdots \\ \cdots & \mathbf{y_2}^T & \cdots \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots \\ \cdots & \mathbf{y_{2^n}}^T & \cdots \end{bmatrix}$$

We now show that this arrangement fires only one hidden neuron and that the forward pass of any input vector $\mathbf{x_n}$ gives the corresponding output vector $\mathbf{y_n}$. Assume that every neuron has zero bias.

Pick a bipolar input vector $\mathbf{x}_m$ for the forward pass. Then the input activation vector $\mathbf{a}^x(\mathbf{x}_m) = (a_1^x(x_m^1), \dots, a_n^x(x_m^n))$ equals the input bipolar vector $\mathbf{x}_m$ because the input activations (8) are bipolar threshold functions with zero threshold. So $\mathbf{a}^x$ equals $\mathbf{x}_m$ because the vector space is bipolar $\{-1,1\}^n$.

The hidden layer input $\mathbf{o}^h$ is the same as (1). It has the matrix-vector form

$$\mathbf{o}^h = \mathbf{W}^T \mathbf{a}^x \tag{9}$$

$$= \mathbf{W}^T \mathbf{x}_m \tag{10}$$

$$= (o_1^h,\ o_2^h, ...,\ o_n^h, ...,\ o_{2^n}^h)^T \tag{11}$$

$$= (\mathbf{x}_1^T \mathbf{x}_m,\ \mathbf{x}_2^T \mathbf{x}_m, \dots,\ \mathbf{x}_j^T \mathbf{x}_m, ...,\ \mathbf{x}_{2^n}^T \mathbf{x}_m)^T \tag{12}$$

from the definition of $\mathbf{W}$ since o_j^h is the inner product of $\mathbf{x}_j$ and $\mathbf{x}_m$.

The input o_j^h to the j^{th} neuron of the hidden layer obeys $o_j^h = n$ when $j = m$ and $o_j^h < n$ when $j \neq m$. This holds because the vectors $\mathbf{x}_j$ are bipolar with scalar components in $\{-1,1\}$. The magnitude of a bipolar vector in $\{-1,1\}^n$ is $\sqrt{n}$. The inner product $\mathbf{x}_j^T \mathbf{x}_m$ is maximum when both vectors have the same direction. This occurs when $j = m$. The inner product is otherwise less than n.

Now comes the key step in the proof. Define the hidden activation a_j^h as a *binary* (not bipolar) threshold function where n is the threshold value:

$$a_j^h(o_j^h) = \begin{cases} 1 & \text{if } o_j^h \geq n, \\ 0 & \text{if } o_j^h < n\ . \end{cases} \tag{13}$$

Then the hidden layer activation $\mathbf{a}^h$ is the *unit* bit vector $(0,\ 0,\ ...,\ 1,\ ...,\ 0)^T$ where $a_j^h = 1$ when $j = m$ and where $a_j^h = 0$ when $j \neq m$. This holds because all 2^n bipolar vectors $\mathbf{x}_m$ in $\{-1,1\}$ are distinct and so exactly one of these 2^n vectors achieves the maximum inner-product value $n = \mathbf{x}_m^T \mathbf{x}_m$.

The input vector $\mathbf{o}^y$ to the output layer is

$$\mathbf{o}^y = \mathbf{U}^T\ \mathbf{a}^h \tag{14}$$

$$= \sum_{j=1}^{J} \mathbf{y}_j\ a_j^h \tag{15}$$

$$= \mathbf{y}_m \tag{16}$$

where a_j^h is the activation of the j^{th} hidden neuron. The activation $\mathbf{a}^y$ of the output layer is:

$$\mathbf{a}^y(o_j^y) = \begin{cases} 1 & \text{if } o_j^y \geq 0 \\ -1 & \text{if } o_j^y < 0\ . \end{cases} \tag{17}$$

The output layer activation leaves $\mathbf{o}^y$ unchanged because $\mathbf{o}^y$ equals $\mathbf{y}_m$ and because $\mathbf{y}_m$ is a vector in $\{-1,1\}^n$. So

$$\mathbf{a}^y = \mathbf{y}_m\ . \tag{18}$$

So the forward pass of an input vector $\mathbf{x}_m$ through the network yields the desired corresponding output vector $\mathbf{y}_m$ where $\mathbf{y}_m = f(\mathbf{x}_m)$ for bipolar permutation map f.

Consider next the backward pass over N.

The backward pass propagates the output vector $\mathbf{y}_m$ from the output layer to the input layer through the hidden layer. The hidden layer input $\mathbf{o}^h$ has the form (5) and so

$$\mathbf{o}^h = \mathbf{U}\ \mathbf{y}_m \tag{19}$$

where $\mathbf{o}^h = (\mathbf{y}_1^T\mathbf{y}_m,\ \mathbf{y}_2^T\mathbf{y}_m, ...,\ \mathbf{y}_j^T\mathbf{y}_m, ...,\ \mathbf{y}_{2^n}^T\mathbf{y}_m)^T$.

The input o_j^h of the j^{th} neuron in the hidden layer o_j^h equals the inner product of $\mathbf{y}_j$ and $\mathbf{y}_m$. So $o_j^h = n$ when $j = m$ and $o_j^h < n$ when $j \neq m$. This holds because again the magnitude of a bipolar vector in $\{-1, 1\}^n$ is $\sqrt{n}$. The inner product o_j^h is maximum when vectors $\mathbf{y}_m$ and $\mathbf{y}_j$ lie in the same direction. The activation $\mathbf{a}^h$ for the hidden layer has the same components as (13). So the hidden-layer activation $\mathbf{a}^h$ again equals the unit bit vecgtor $(0,\ 0,\ ...,\ 1,\ ...,\ 0)^T$ where $a_j^h = 1$ when $j = m$ and $a_j^h = 0$ when $j \neq m$.

Then the input vector $\mathbf{o}^x$ for the input layer is

$$\mathbf{o}^x = \mathbf{W}\ \mathbf{a}^h \tag{20}$$

$$= \sum_{j=1}^{J} \mathbf{x}_j\ \mathbf{a}^h \tag{21}$$

$$= \mathbf{x}_m\ . \tag{22}$$

The i^{th} input neuron has a threshold activation that is the same as

$$\mathbf{a}^x(o_i^x) = \begin{cases} 1 & if\ o_i^x \geq 0 \\ -1 & if\ o_i^x < 0 \end{cases} \tag{23}$$

where o_i^x is the input of i^{th} neuron in the input layer. This activation leaves $\mathbf{o}^x$ unchanged because $\mathbf{o}^x$ equals $\mathbf{x}_m$ and because the vector $\mathbf{x}_m$ lies in $\{-1, 1\}^n$. So

$$\mathbf{a}^x = \mathbf{o}^x \tag{24}$$

$$= \mathbf{x}_m\ . \tag{25}$$

So the backward pass of any target vector $\mathbf{y}_m$ yields the desired input vector $\mathbf{x}_m$ where $f^{-1}(\mathbf{y}_m) = \mathbf{x}_m$. This completes the backward pass and the proof. ■

.

3. Bidirectional BP Learning

We now develop the new bidirectional BP algorithm for learning bidirectional function representations or approximations. Bidirectional BP training minimizes both the error function for the forward pass and the backward pass. The forward-pass error E_f is the error at the output layer. The backward-pass error E_b is the error at the input layer. Bidirectional BP training combines these two errors.

The forward pass sends the input vector $\mathbf{x}$ through the hidden layer to the ouput layer. We use only one hidden layer. There is no loss of generality in using any finite number of them. The hidden-layer input values o_j^h are the same as (1). The j^{th} hidden activation a_j^h is the *bipolar* logistic that shifts and scales the ordinary logistic:

$$a_j^h(o_j^h) = \frac{2}{1 + e^{-2o_j^h}} - 1 \tag{26}$$

and (3) gives the input o_k^y to the k^{th} output neuron. The hidden activations can also be logistic or any other sigmoidal function. The activation for an output neuron is the identity function:

$$a_k^y = o_k^y \tag{27}$$

where a_k^y is the activation of k^{th} output neuron. The error function for the forward pass was the squared error E_f

$$E_f = \frac{1}{2}\sum_{k=1}^{K}(y_k - a_k^y)^2\ . \tag{28}$$

where y_k is the value of k^{th} neuron in the output layer. Ordinary unidirectional BP updates the weights and other network parameters by propagating the error from the output layer back to the input layer.

The backward pass sends the output vector $\mathbf{y}$ from the output layer to the input layer through the hidden layer. The input to j^{th} hidden neuron o_j^h is the same as (5). The activation a_j^h for j^{th} hidden neuron is:

$$a_j^h = \frac{2}{1 + e^{-2o_j^h}} - 1\ . \tag{29}$$

The input o_i^x for the i^{th} input neuron is the same as (7). The activation at the input layer is the identity function:

$$a_i^x = o_i^x\ . \tag{30}$$

A nonlinear sigmoid (or Gaussian) activation can replace the linear function.

The backward-pass error E_b is

$$E_b = \frac{1}{2}\sum_{i=1}^{I}(x_i - a_i^x)^2\ . \tag{31}$$

The partial derivative of the hidden-layer activation in the forward direction is

$$\frac{\partial a_j^h}{\partial o_j^h} = \frac{\partial}{\partial o_j^h}\Big(\frac{2}{1 + e^{-2o_j^h}} - 1\Big) \tag{32}$$

$$= \frac{4e^{-2o_j^h}}{(1 + e^{-2o_j^h})^2} \tag{33}$$

$$= \frac{2}{1 + e^{-2o_j^h}}\Big[2 - \frac{2}{1 + e^{-2o_j^h}}\Big] \tag{34}$$

$$= (a_j^h + 1)(1 - a_j^h)\ . \tag{35}$$

Let $a_j^{h'}$ denote the derivative of a_j^h with respect to the inner-product term o_j^h. We again use the superscript b to denote backward pass. Then the partial derivative of E_f with respect to weight u_{kj} is

$$\frac{\partial E_f}{\partial u_{kj}} = \frac{1}{2}\frac{\partial}{\partial u_{kj}}\sum_{k=1}^{K}(y_k - a_k^y)^2 \tag{36}$$

$$= \frac{\partial E_f}{\partial a_k^y}\ \frac{\partial a_k^y}{\partial o_k^y}\ \frac{\partial o_k^y}{\partial u^{kj}} \tag{37}$$

$$= (y_k - a_k^y) \times 1 \times a_k^y\ . \tag{38}$$

The partial derivative of E_f with respect to w_{ji} is

$$\frac{\partial E_f}{\partial w_{ji}} = \frac{1}{2}\frac{\partial}{\partial w_{ji}}\sum_{k=1}^{K}(y_k - a_k^y)^2 \tag{39}$$

$$= \Big(\sum_{k=1}^{K}\frac{\partial E_f}{\partial a_k^y}\ \frac{\partial a_k^y}{\partial o_k^y}\ \frac{\partial o_k^y}{\partial a_j^h}\Big)\ \frac{\partial a_j^h}{\partial o_j^h}\ \frac{\partial o_j^h}{\partial w_{ji}} \tag{40}$$

$$= \sum_{k=1}^{K}(y_k - a_k^y)u_{kj} \times a_j^{h'} \times x_i\ . \tag{41}$$

The partial derivative of E_f with respect to the bias b_k^y of the k^{th} output neuron is

$$\frac{\partial E_f}{\partial b_k^y} = \frac{1}{2}\frac{\partial}{\partial b_k^y}\sum_{k=1}^{K}(y_k - a_k^y)^2 \tag{42}$$

$$= \frac{\partial E_f}{\partial a_k^y}\ \frac{\partial a_k^y}{\partial o_k^y}\ \frac{\partial o_k^y}{\partial b_k^y} \tag{43}$$

$$= (y_k - a_k^y) \times 1 \times 1\ . \tag{44}$$

The partial derivative of E_f with respect to the bias b_j^h of the j^{th} hidden neuron is

$$\frac{\partial E_f}{\partial b_j^h} = \frac{1}{2}\frac{\partial}{\partial b_j^h}\sum_{k=1}^{K}(y_k - a_k^y)^2 \tag{45}$$

$$= \Big(\sum_{k=1}^{K}\frac{\partial E_f}{\partial a_k^y}\ \frac{\partial a_k^y}{\partial o_k^y}\ \frac{\partial o_k^y}{\partial a_j^h}\Big)\ \frac{\partial a_j^h}{\partial o_j^h}\ \frac{\partial o_j^h}{\partial b_j^h} \tag{46}$$

$$= \sum_{k=1}^{K}(y_k - a_k^y)u_{kj} \times a_j^{h'} \times 1\ . \tag{47}$$

The partial derivative of E_b with respect to w_{ji} is

$$\frac{\partial E_b}{\partial w_{ji}} = \frac{1}{2}\frac{\partial}{\partial w_{ji}}\sum_{k=1}^{K}(x_i - a_i^x)^2 \tag{48}$$

$$= \frac{\partial E_b}{\partial a_i^x}\ \frac{\partial a_i^x}{\partial o_i^x}\ \frac{\partial o_i^x}{\partial w^{ji}} \tag{49}$$

$$= (x_i - a_i^x) \times 1 \times a_i^x\ . \tag{50}$$

The partial derivative of E_b with respect to u_{kj} is

$$\frac{\partial E_b}{\partial u_{kj}} = \frac{1}{2}\frac{\partial}{\partial u_{kj}}\sum_{i=1}^{I}(x_i - a_i^x)^2 \tag{51}$$

$$= \Big(\sum_{i=1}^{I}\frac{\partial E_b}{\partial a_i^x}\ \frac{\partial a_i^x}{\partial o_i^x}\ \frac{\partial o_i^x}{\partial a_j^{hb}}\Big)\ \frac{\partial a_j^{hb}}{\partial o_j^{hb}}\ \frac{\partial o_j^{hb}}{\partial u_{kj}} \tag{52}$$

$$= \sum_{i=1}^{I}(x_i - a_i^x)w_{ji} \times a_j^{hb'} \times y_k\ . \tag{53}$$

The partial derivative of E_b with respect to the bias b_i^x of the i^{th} input neuron is:

$$\frac{\partial E_b}{\partial b_i^x} = \frac{1}{2}\frac{\partial}{\partial b_i^x}\sum_{i=1}^{I}(x_i - a_i^x)^2 \tag{54}$$

$$= \frac{\partial E_b}{\partial a_i^x}\ \frac{\partial a_i^x}{\partial o_i^x}\ \frac{\partial o_i^x}{\partial b_i^x} \tag{55}$$

$$= (x_i - a_i^x) \times 1 \times 1\ . \tag{56}$$

The partial derivative of E_b with respect to the bias b_j^h of the j^{th} hidden neuron is

$$\frac{\partial E_b}{\partial b_j^h} = \frac{1}{2}\frac{\partial}{\partial b_j^h}\sum_{i=1}^{I}(x_i - a_i^x)^2 \tag{57}$$

$$= \Big(\sum_{i=1}^{I}\frac{\partial E_b}{\partial a_i^x}\ \frac{\partial a_i^x}{\partial o_i^x}\ \frac{\partial o_i^x}{\partial a_j^{hb}}\Big)\ \frac{\partial a_j^{hb}}{\partial o_j^{hb}}\ \frac{\partial o_j^{hb}}{\partial b_j^h} \tag{58}$$

$$= \sum_{i=1}^{I}(x_i - a_i^x)w_{ji} \times a_j^{hb'} \times 1\ . \tag{59}$$

Bidirectional BP training minimizes the *joint* error E of the forward and backward passes. The joint error E sums the forward error E_f and backward error E_b:

$$E = E_f + E_b\ . \tag{60}$$

Then the partial derivative of E with respect to u_{kj} is

$$\frac{\partial E}{\partial u_{kj}} = \frac{\partial E_f}{\partial u_{kj}} + \frac{\partial E_b}{\partial u_{kj}} \tag{61}$$

$$= (y_k - a_k^y)a_k^y + \sum_{i=1}^{I}(x_i - a_i^x)w_{ji}a_j^{hb'}y_k \tag{62}$$

from (36) and (51) . The partial derivative of the joint error E with respect to the weight w_{ji} is

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial E_f}{\partial w_{ji}} + \frac{\partial E_b}{\partial w_{ji}} \tag{63}$$

$$= \sum_{k=1}^{K}(y_k - a_k^y)u_{kj}a_j^{h'}x_i \tag{64}$$

$$+ (x_i - a_i^x)a_i^x \tag{65}$$

from (39) and (48) .

The partial derivative of E with respect to b_j^h gives

$$\frac{\partial E}{\partial b_j^h} = \frac{\partial E_f}{\partial b_j^h} + \frac{\partial E_b}{\partial b_j^h} \tag{66}$$

$$= \sum_{k=1}^{K}(y_k - a_k^y)u_{kj} \times a_j^{h'} \tag{67}$$

$$+ \sum_{i=1}^{I}(x_i - a_i^x)w_{ji} \times a_j^{hb'}\ . \tag{68}$$

from (45) and (57) .

The error for the input neuron bias is E_b only because $\mathbf{x} = \mathbf{o}^x$ for the forward pass. The error for the output neuron bias is E_f only for output neuron bias because $\mathbf{y} = \mathbf{o}^y$ for the backward pass. Then

$$\frac{\partial E}{\partial b_i^x} = \frac{\partial E_b}{\partial b_i^x} = x_i - a_i^x \tag{69}$$

$$\frac{\partial E}{\partial b_k^y} = \frac{\partial E_f}{\partial b_k^y} = y_k - a_k^y \ . \tag{70}$$

Then B-BP training updates the parameters as

$$u_{kj}^{(n+1)} = u_{kj}^{(n)} - \eta \frac{\partial E}{\partial u_{kj}} \tag{71}$$

$$w_{ji}^{(n+1)} = w_{ji}^{(n)} - \eta \frac{\partial E}{\partial w_{ji}} \tag{72}$$

$$b_i^{x(n+1)} = b_i^{x(n)} - \eta \frac{\partial E}{\partial b_i^x} \tag{73}$$

$$b_j^{h(n+1)} = b_j^{h(n)} - \eta \frac{\partial E}{\partial b_j^h} \tag{74}$$

$$b_k^{y(n+1)} = b_k^{y(n)} - \eta \frac{\partial E}{\partial b_k^y} \tag{75}$$

where η is the learning rate. The partial derivatives are from (61)−(70). Algorithm 1 summarizes the B-BP algorithm.

4. Simulation Results

We tested the bidirectional BP algorithm on a 5-bit permutation functions in 3-layer networks with different numbers of hidden neurons. The B-BP algorithm produced either an exact representation or approximation. We report results for learning a permutation function from the 5-bit bipolar vector space $\{-1, 1\}^n$. The hidden neurons used bipolar logistic activations. The input and output neurons used identity activations. Table 2 displays the the permutation test function that mapped $\{-1, 1\}^5$ to itself. We compared the forward and backward forms unidirectional BP with bidirectional BP. We also tested to see whether adding more hidden neurons improved network approximation accuracy.

The simulations used 18,000 samples for network training and 2,000 separate samples for testing. Forward-pass of (standard) BP used E_f as its error while backward-pass BP used E_b as its error. Bidirectional BP combined both E_f and E_b for its joint error. We computed testing error for forward pass and backward pass. Each plotted error value averaged 20 runs.

Figure 2 shows the results of running the three types of BP learning on a 3-layer network with 100 hidden neurons. The training error falls along both directions as the training progresses. This in not the case for the unidirectional cases of forward BP and backward BP training. Forward training and backward training perform well only for function approximation in their preferred direction and not in the opposite direction.

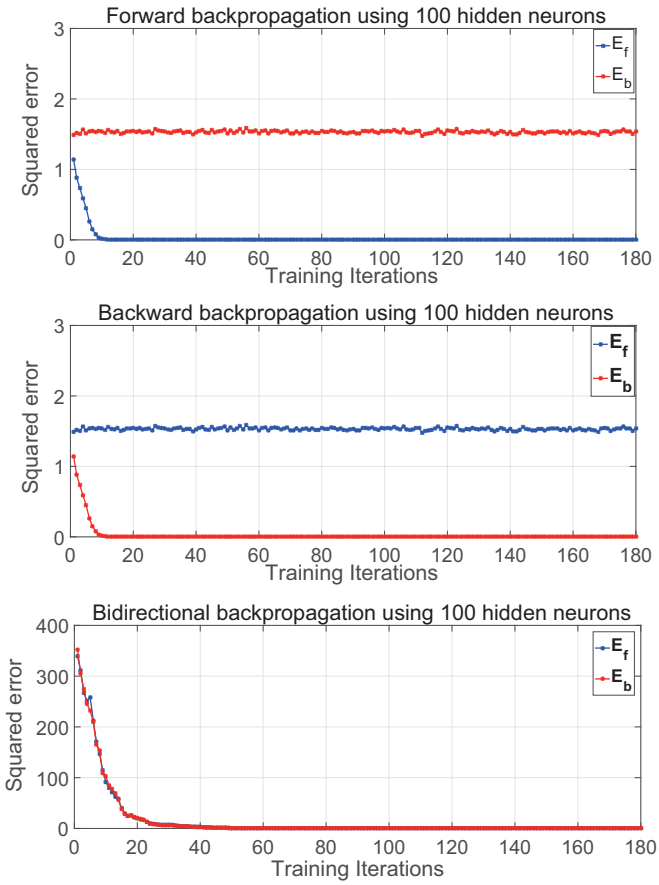


Fig. 2: *Training-set squared error using 100 hidden neurons with forward BP training, backward BP training, and bidirectional BP training. Forward BP tuned the network with respect to E_f only. Backward BP training tuned it respect to E_b only. Bidirectional BP training combined E_f and E_b to update the network parameters.*

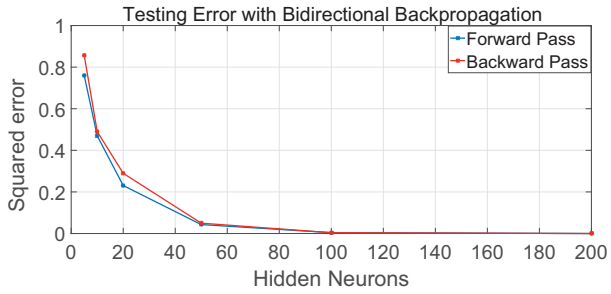


Fig. 3: *B-BP training error for the 3-bit permutation in Table 2 with different numbers of hidden neurons. The two curves describe the training error for the forward and backward passes through the 3-layer network. Each test used 2000 samples. The number of hidden neurons varied from 5, 10, 20, 50, 100, to 200.*

Table 3 shows the forward-pass test errors for learning 3-layer neural networks as the number of hidden neurons grows. We again compared the three forms of BP for the network training–two forms of unidirectional BP and

Table 2: 5-Bit Bipolar Permutation Function.

Input x	Output t	Input x	Output t
$[---\,--]$	$[+++\,--]$	$[+---\,-]$	$[+---\,-]$
$[---\,-+]$	$[+++\,-+]$	$[+---\,+]$	$[+---\,-]$
$[---\,+-]$	$[-+--\,+]$	$[+--+\,-]$	$[---\,---]$
$[---\,++]$	$[----\,+]$	$[+--+\,+]$	$[--++\,+]$
$[--+\,--]$	$[+-+-\,+]$	$[+-+-\,--]$	$[--++\,-]$
$[--+\,-+]$	$[--+-\,+]$	$[+-+-\,-+]$	$[++--\,+]$
$[--++\,-]$	$[+--+\,-]$	$[+-++\,-]$	$[---+\,-]$
$[--++\,+]$	$[++-+\,-]$	$[+-++\,+]$	$[++--\,-]$
$[-+--\,-]$	$[-+--\,---]$	$[++--\,--]$	$[-++\,+-]$
$[-+--\,+]$	$[-+--\,++]$	$[++--\,-+]$	$[+++\,+-]$
$[-+-+\,-]$	$[+-+-\,--]$	$[++-+\,--]$	$[+--+\,+]$
$[-+-+\,+]$	$[+---\,+]$	$[++-+\,-+]$	$[-+-+\,-]$
$[-++-\,-]$	$[++-+\,++]$	$[+++-\,--]$	$[+-++\,+]$
$[-++-\,+]$	$[--+-\,--]$	$[+++-\,-+]$	$[+++++]$
$[-+++\,-]$	$[-+++\,--]$	$[++++\,--]$	$[+-++\,-]$
$[-+++\,+]$	$[--+-\,---]$	$[++++\,-+]$	$[---+\,+]$

Table 3: Forward-Pass Testing Error E_f

	Backpropagation errors		
Hidden Neurons	Forward	Backward	Bidirectional
5	0.6032	1.2129	0.7600
10	0.1732	1.4259	0.4700
20	0.0068	1.4031	0.2307
50	1.5×10^{-4}	1.5811	0.0430
100	2.0×10^{-6}	1.4681	0.0043
200	5.0×10^{-8}	1.6061	4.0×10^{-6}

Table 4: Backward-Pass Testing Error E_b

	Backpropagation errors		
Hidden Neurons	Forward	Backward	Bidirectional
5	1.2070	0.6016	0.8574
10	1.4085	0.1584	0.4900
20	1.2384	0. 0023	0.2895
50	1.3157	1.2×10^{-4}	0.0498
100	1.4681	3.6×10^{-6}	0.0039
200	1.7837	8.7×10^{-8}	9.0×10^{-6}

bidirectional BP. The forward-pass error for forward BP fell significantly as the number of hidden neurons grew. The forward-pass error of backward BP decreased slightly as the number of hidden neurons grew and gave the worst performance. Bidirectional BP performed well on the test set. Its forward-pass error also fell significantly as the number of hidden neurons grew. Table 4 shows similar error-versus-hidden-neuron results for the backward-pass error

The two tables jointly show that the unidirectional forms of BP performed well only in one direction while the B-BP algorithm performed well in both directions. Hence we propose using only the B-BP algorithm for learning bidirectional function representations or approximations.

Data: $\boldsymbol{T}$ input vectors $\{\boldsymbol{x}_1,\ldots,\ \boldsymbol{x}_T\}$, $\boldsymbol{T}$ target vectors $\{\boldsymbol{y}_1,\ldots,\ \boldsymbol{y}_T\}$ such that $f(\boldsymbol{x}_i) = \boldsymbol{y}_i$. Number of hidden neurons $\boldsymbol{J}$. Batch size $\boldsymbol{S}$ and number of epochs $\boldsymbol{R}$. Choose the learning rate $\boldsymbol{\eta}$.

Result: Bidirectional network for function f.

Initialize: Randomly select the weights of $\boldsymbol{W}^{(0)}$ and $\boldsymbol{U}^{(0)}$. Randomly pick the bias weights for input, hidden, and output neurons $\{\boldsymbol{b}^x,\ \boldsymbol{b}^h,\ \boldsymbol{b}^y.\}$

while **epoch** $\boldsymbol{r:0\longrightarrow\ R}$ **do**

Initialize: $\Delta\boldsymbol{W}=\boldsymbol{0},\ \Delta\boldsymbol{U}=\boldsymbol{0},\ \Delta\boldsymbol{b}^x=\boldsymbol{0},\ \Delta\boldsymbol{b}^y=\boldsymbol{0},\ \Delta\boldsymbol{b}^z=\boldsymbol{0}$

while **batch_size** $\boldsymbol{l:1\longrightarrow\ L}$ **do**

- Pick input vector $\boldsymbol{x}$ and its corresponding target vector $\boldsymbol{y}$.
- Compute hidden activation $\boldsymbol{a}^h$ and output activation $\boldsymbol{a}^y$ for forward pass.
- Compute hidden activation $\boldsymbol{a}^h$ and output activation $\boldsymbol{a}^y$ for backward pass.
- Compute the derivatives with respect to $\boldsymbol{W}$ and $\boldsymbol{U}$: $\boldsymbol{\nabla_W}E$ and $\boldsymbol{\nabla}_UE$.
- Compute the derivatives with respect to the bias weights: $\boldsymbol{\nabla_{b^x}}E$, $\boldsymbol{\nabla_{b^y}}E$, and $\boldsymbol{\nabla_{b^y}}E$
- Compute change in weights: $\Delta\boldsymbol{W}=\Delta\boldsymbol{W}+\boldsymbol{\nabla_W}E$ and $\Delta\boldsymbol{W}=\Delta\boldsymbol{W}+\boldsymbol{\nabla}_UE$
- Compute change in bias weights: $(\Delta\boldsymbol{b^x}=\Delta\boldsymbol{b^x}+\boldsymbol{\nabla_{b^x}}E)$, $(\Delta\boldsymbol{b^y}=\Delta\boldsymbol{b^y}+\boldsymbol{\nabla_{b^x}}E)$ and $(\Delta\boldsymbol{b^z}=\Delta\boldsymbol{b^z}+\boldsymbol{\nabla_{b^z}}E)$.

End

Update: $\boldsymbol{W^{(r+1)}=\ W^{(r)}-\frac{\eta}{L}\Delta U}$

$\boldsymbol{U^{(r+1)}=\ U^{(r)}-\frac{\eta}{L}\Delta U}$

$\boldsymbol{b^{x(r+1)}=\ b^{x(r)}-\frac{\eta}{L}\Delta b^x}$

$\boldsymbol{b^{y(r+1)}=\ b^{y(r)}-\frac{\eta}{L}\Delta b^y}$

$\boldsymbol{b^{z(r+1)}=\ b^{z(r)}-\frac{\eta}{L}\Delta b^z}$

End

Algorithm 1: *The Bidirectional BP Algorithm*

5. Conclusions

We have shown that a bidirectional multilayer network can exactly represent bipolar or binary permutation mappings if the network uses enough threshold or sigmoidal neurons. The proof requires an exponential number of hidden neurons for exact representations but much simpler representation exist in general. The new B-BP algorithm allows bidirectional learning of sampled functions. It can often find efficient bidirectional representations or approximations with a smaller set of hidden neurons.

References

[1] B. Kosko, "Bidirectional associative memories," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 18, no. 1, pp. 49–60, 1988.

[2] B. Kosko, *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*. Prentice Hall, 1991.

[3] D. Rumelhart, G. Hinton, and R. Williams, "Learning representations by back-propagating errors," *Nature*, pp. 323–533, 1986.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[5] M. Jordan and T. Mitchell, "Machine learning: trends, perspectives, and prospects," *Science*, vol. 349, pp. 255–260, 2015.

[6] S. Y. Kung, *Kernel methods and machine learning*. Cambridge University Press, 2014.

Processing and Analysis of Large Data Sets Using High Performance Computing: Beyond Experimental Data

Brian Panneton[1], Brian Henz[2], Pritesh Patel[3], James Adametz[4]

[1]Technical and Project Engineering, Army Research Laboratory, Aberdeen Proving Ground, MD, USA
[2]Simulation Sciences Branch, Army Research Laboratory, Aberdeen Proving Ground, MD, USA
[3]Ad hoc Research Associates, Aberdeen Proving Ground, MD, USA
[4]QED Systems LLC, Aberdeen Test Center Test Technology Directorate, Aberdeen Proving Ground, MD USA

Abstract - *Efficiently processing and analyzing big data is gaining importance within the military setting. As systems-of-systems evaluation produces an expanding set of data in terms of volume, source, and range of types, the processing and analysis becomes more complex. The Army Research Laboratory (ARL) and Aberdeen Test Center (ATC) previously partnered to employ High Performance Computing (HPC) to address this problem. This paper describes the expansion of the distributed processing framework which now allows processing and analysis of data from simulation and emulation in addition to live experimentation and real-world sources. New modules have been added to the distributed framework to enable the processing of data from a variety of sources, which allows for a more in-depth analytical view of the event.*

Keywords: Parallel data reduction, Network Analysis, High Performance Computing, Emulation Data Reduction

1 Introduction

Communication systems deployed on the battlefield are critical for tactical operations. These systems must maintain stability and usability within challenging environments, more so than in the commercial field. The systems must go through stringent verification and validation testing before being deployed. In order for this process to occur, large-scale experimental field tests are conducted daily, recording multiple terabytes of many types of data, including network, geospatial, video, and temperature. The experimental data must then be reduced and restructured into a usable data model for analysts to examine and determine the success of the systems under test.

Dealing with a month-long event means the data size scales upwards quickly. Typically, analysts and evaluators want data in a queryable form ready by the next morning. Next-day turnaround time for processing and analytics is crucial, allowing problems to be addressed quickly to prevent further loss of required information. A collaboration between the Army Research Laboratory (ARL) and the Aberdeen Test Center (ATC) produced a distributed, python-based data reduction framework (Data Parser) which runs on High Performance Computing (HPC) resources at ARL's DoD Supercomputing Resource Center (DSRC) [1]. This system is able to achieve the necessary results in a reduced time frame, often in less time than the required time-table for the experimental test.

The Data Parser framework was designed to allow virtually any type of data to be processed in a scalable and parallel way. Originally, the focus was on tactical communications data, but as test types and evaluation goals changed, the framework evolved. The addition of Controller Area Network (CAN) [2], Analog/Digital sensor, video, and other types of data allowed for many new types of analysis to be performed. Newer tests such as Joint Land Tactical Vehicle (JLTV) [3], Autonomous Mobility Applique System (AMAS) [4] and various robotics systems are evaluated based on how the vehicular system interacts with the operator and the environment, more than how the vehicular communications system performs.

Currently, deployed systems are validated and verified by analyzing data from experimental results. However, many of these deployed systems have software models that can be studied within simulation and emulation environments. Comparing data from software models and deployed systems allows simulation and emulation testing to aid in design for future live experimentation and real-world events. The Data Parser has been updated to allow cross-validation between the models in the virtual realm and the physically deployed systems. Simulation and emulation drastically reduce costs in comparison to live

experimentation. Proving virtual tests using models are comparable to the physical tests will open the way for more testing to be conducted in a controlled and repeatable virtual environment. The tester is able to manipulate the tests in ways that are cost prohibitive or otherwise difficult to achieve in physical tests.

The following sections will detail the current advancements and capabilities of processing live experimental data, the work required to transition from processing experimental and real-world data to processing emulation data, the current state of analytics for both emulation and experimentation and finally where this work is heading in the future.

2 Capabilities and advancements

This section details the current state of the Data Parser as well as additions that have been made to it. Due to the initial design choices, the Data Parser framework is easily adaptable to new projects. As the requirements change the Data Parser is able to grow to accommodate them.

2.1 Framework overview

Processing big data in an HPC environment is slightly different than processing on commodity hardware. The HPC clusters generally use a large, highly optimized, distributed file system. This allows easy access to the data from each node and reduces the complexity of moving data. In addition, the HPC environment is meant to be shared among a variety of projects and problem spaces, so a scheduler, such as PBS,[1] LSF[2], or Slurm[3] is employed to queue and manage the system. Taking this into account, most modern big data frameworks will not run efficiently within an HPC environment. The development of a custom data processing framework allows data processing to be run on such a system.

The Data Parser Framework was focused on portability between HPC environments, scalability and ease of development for the user. The Message Passing Interface (MPI) is used for all Data Parser communications. Due to the nature of the shared disk infrastructure, control messages, rather than data results, are generally sent over MPI.

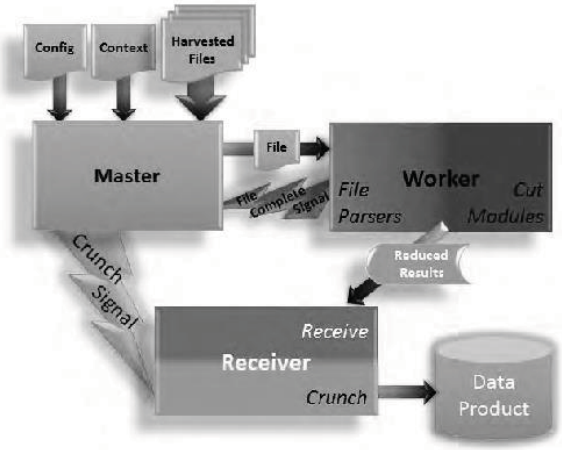


Figure 1 High Level Data Flow in Data Parser

The framework was built using a distributed producer-consumer paradigm. Figure 1 gives a high level view of how the Data Parser processes data. First, the Master parallelizes and distributes the input data to Workers, or MPI rank dedicated to data processing, on a per file basis. On each Worker there are File Parsers and Cut Modules, or producers and consumers respectively. The File Parsers read and break up each file into subsections of data called cuts. Cut Modules subscribe to certain types of cuts and receive them as they are parsed from the file. Cut Modules organize and reduce the data based on what results a user is looking for. These results can then be sent to a Receiver. A Receiver is a Worker dedicated to data aggregation for one Cut Module. The described data flow is referred to as the Process Stage. After the Process Stage, the data can once again be parallelized and reduced in what is referred to as the Crunch Stage. This second stage is where post-processing tasks occur, such as producing graphs or tables for later use. The combination of one Process and one Crunch stage make up a Phase. Phases can depend on each other forming logical data workflows. A more in depth overview of how the framework works can be found by reading the ARL Technical Report titled *High-Bandwidth Tactical Network Data Analysis in a High-Performance-Computing (HPC) Environment: HPC Data Reduction Framework* [5]

1 The Portable Batch System (PBS) is a job scheduler and resource allocation system for HPC created by Altair.
2 The Platform Load Sharing Facility (LSF) is a job scheduler and resource allocation system for HPC created by IBM.
3 The Simple Linux Utility for Resource Management (Slurm) is a job schedule and resource allocation system for HPC created through a collaborative effort between Lawrence Livermore National Laboratory, SchedMD, Linux NetworX, Hewlett-Packard and Groupe Bull.

The amount of time needed to reduce and process data is highly dependent on the type of input, the number of consumers and the size of the HPC cluster. The typical input files produced from testing at ATC are about 1/2 GB in size with the aggregate size being around 1-2 TB. The average test normally runs 25 Cut Modules over 6 phases. In the pre-HPC reduction methods, the reduction and processing time for each terabyte of data took approximately 60 hours. Using the Data Parser on the HPC system with 256 cores has reduced this time frame to under 10 hours. Scaling has allowed processing of data above 40 terabytes using 1024 cores, however this was with a smaller set of Cut Modules. Scalability above this range has not been tested since larger data sets are less frequent.

2.2 Current advancements

The data processing framework has undergone several advancements to enhance its capabilities. New modules have been developed to process additional types of data allowing for more robust cross-correlations and better insight into the observations from any given experiment. Adding new processing to the framework to combine virtual, experimental and real-world data allows for a more in-depth study of the systems under test. In general, new Cut Modules are built independent of the system under test allowing for reuse of the same processing pipelines.

2.2.1 Transition to emulation data

ARL and US Army Communications-Electronics Research, Development and Engineering Center's (CERDEC) Space and Terrestrial Communications Directorate (S&TCD) both use a modular framework called the Extend-able Mobile Ad-hoc Network Emulator (EMANE) to support Mobile Ad-hoc Network (MANET) emulations. Emulation provides real-time modeling of the data link and physical layers for MANETs in order to predict connectivity and provide a testbed for analyzing network protocols and applications that can be run unmodified in the emulation environment [6]. The emulation testbed provides a repeatable virtual test environment. This repeatability is not realizable under live exercise conditions. Configuration of a MANET emulation experiment includes provisioning a virtual machine, linux container (LXC), or physical hardware for each networked platform or device, configuring and executing daemons (EMANE, GPS, etc), and configuring the communication network. Post emulation analysis of the systems under test requires real time collection of data from multiple processes and collection points within the testbed. Having control of the testbed and its configuration, coupled with knowledge of how the data will be processed within the Data Parser, allows the data to be collected in a way that is easy to ingest for later analysis.

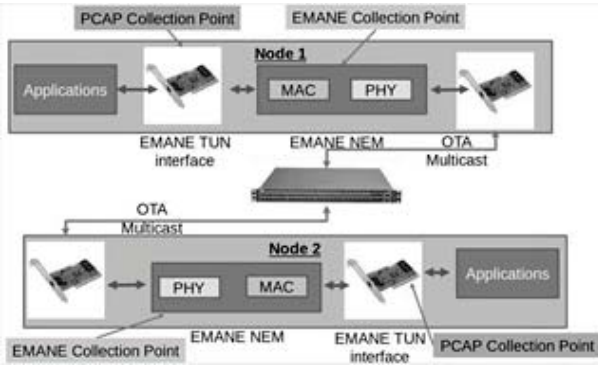


Figure 2 Data Collection Points from MANET Emulation

2.2.1.1 Configuration and data collection from MANET emulation

In Figure 2, the emulation testbed configuration is illustrated with data collection points identified. The collection points are located at the EMANE TUN[4] interface to collect packet capture (PCAP) data. These locations capture all of the ingress and egress traffic to and from applications and traffic generators running on the emulated devices. Depending on the radio model, the captured data may include routing messages for daemons such as the optimized link state routing (OLSR) daemon. These messages may be absent from the packet capture if the routing is configured in the EMANE radio model. The PCAP collection point will capture all UDP and TCP traffic that is terminated or generated by the applications for

```
tcpdump -p -n -B 8192 -x -s 0 -i emane0 -C 500 -Z root
        -w ${TEMP_OUTDIR}/node-${NODEID}-emane0-inbound.pcap inbound
tcpdump -p -n -B 8192 -x -s 0 -i emane0 -C 500 -Z root
        -w ${TEMP_OUTDIR}/node-${NODEID}-emane0-outbound.pcap outbound
```

Figure 3 Sample TCPDUMP Script

[4] A TUN is a software defined virtual network interface [11]

EMANE "shared code" models, including waveform applications such as routing in the EMANE Network Emulation Modules (NEM). The EMANE Over-the-air (OTA) Manager Channel is not recorded since it is the back-haul network for EMANE itself. Packets are sent and received faster than the configured latency on this channel.

The EMANE collection point can be provided data through the TestPoint software created by AdjacentLink LLC [7]. Testpoint probes the EMANE NEM at set intervals and records a predefined list of counters and tables from the running NEM.

2.2.1.2 Collecting PCAP data

The framework's PCAP File Parser uses the Device-Tap- Direction (DTD) concept [8] in order to avoid knowing detailed configuration of the system under test. The DTD is relatively easy to create within the virtual machines or LXCs (virtual nodes). Each provisioned virtual node needs to be uniquely identified within the network and is generally identified within the hostname by a unique numeric identifier (ie: thufir-n001). This numeric identifier can be used as the device in the DTD format. Some tests may use network-specific identifiers such as the Virtual Large Area Network (VLAN) tag instead. The collection point, which is mainly used for packet matching within the Data Parser framework, is commonly set to the EMANE TUN interface. In some tactical scenarios, analysts may be interested in how different parts of the overall network topology are performing. Performance can be derived from capturing packet data on both sides of a network device such as an in-line encrypter. The collection point configuration enables unique matching patterns where multiple communication channels may overlap. The packet's direction, inbound or outbound, is determined using a heuristic such as matching the interface's source Ethernet address or the packet's destination IP.

Record packets on the system is generally done with tcpdump. The lines shown in Figure 3 separately record 500MB PCAP files using the DTD format. There is a limit set on the size of the PCAPs strictly because it makes processing the data faster within the Data Parser framework due to distributing the data on a per file basis. The 500MB size comes from the normal size of the experimentally collected data files and is not a restriction.

Collecting PCAP data can also be done with the Automated Performance Assessment Framework Innovation (APAFI) [12] sniffer which monitors inbound and outbound packets. The APAFI sniffer writes PCAP files but has other features to perform real-time analysis that may be wanted.

In addition to the APAFI sniffer, data collection can be done using the Riverbed AppResponse Xpert (ARX) appliance to capture all incoming and outgoing traffic between physical and virtual nodes.

Once the PCAP files are collected and in the DTD format, they can be loaded and run through the communications processing Cut Modules which look at the different network protocols in depth including IP, TCP, and UDP. For instance, the TCP modules allow full flow calculations from host-to-host rather than round-trip. The packet data can also be combined with other data types to better analyze the system under test.

2.2.1.3 Collecting GPS data

The majority of the EMANE-related tests rely on geospatial positioning in order to calculate the propagation loss of a network over terrain. EMANE generates time-synchronized mobility events which get fed into the GPS daemon running on the virtual machine. The current experimental data has GPS times, space and position information recorded once a second. To produce data in a similar fashion, a simple process is started that polls the GPS daemon on the virtual node once per second and outputs a comma separated values (CSV) file. The generated CSV files can be ingested by the CSV File Parser and the data can be analyzed.

2.2.2 Processing instrumented emulated systems

TestPoint allows the insertion of probes into an EMANE emulation or the surrounding environment in order to record details about the events that took place. This data is recorded in a database as a look-up table in addition to being saved as a raw data file. Two new Cut Modules and a File Parser were added to the Data Parser Framework to handle TestPoint data.

The Cut Module called TestPoint Emane-Loss collects and plots network statistics from the Physical Layer and Data Link Layer of the Open Systems Interconnection (OSI) model. Statistics collected include forward pathloss, signal-to-noise ratio, data-rate over time and queue overflows. These lower level statistics can greatly improve evaluation of a system-under-test and are generally difficult to record accurately within a live experimental test.

The Cut Module called TestPoint Cpu-Mem collects and plots system statistics from the virtual nodes. The focus of this module is CPU and Memory usage of the overall emulated hardware. This information allows an analyst to assess the performance of the system under test during planned events such as stress testing of the network.

The TestPoint File Parser has been created to parse the database look-up table, extract the raw output for individual Probe Messages, and convert them into cuts. The two new Cut Modules subscribe to these cuts. This automated process allows the Data Parser to be able to processes any new data types that TestPoint can generate.

2.3 Test analytics

The Data Parser framework was originally designed to accommodate the US Army's Network Integration Evaluation (NIE) tests. As detailed in the previous section, many advancements and improvements have been made. The following section gives a quick look into some of the other tests that use the Data Parser framework for their analytics.

2.3.1 Vehicular and robotics test analysis

The original purpose for the processing framework was to perform reduction and analysis of tactical network communications data and render a database model for the analysts and evaluators to derive and extract key performance metrics. With the success of this framework, ATC has decided to enhance its capabilities to include vehicular and robotics test processing. To that end, new modules are being developed to allow the system to interpret and render Controller Area Network (CAN [1]), analog sensors, JAUS [2], video sources, device log files and other types of vehicular data. These will provide new and rich sources of information that can be later correlated with each other.

Much of the vehicular test analysis focuses on reliability of systems and subsystems. Within a vehicle, there can be a multitude of devices that can malfunction. It is difficult to record the precise time of when a malfunction occurs and the malfunction can sometimes remain undetected until a manual inspection occurs. By adding CAN data to the list of information being processed, the behavior of subsystems within each vehicle can be observed as they communicate with each other. Quick determination of when a particular device issues specific error codes or stops communicating with its peers is done by examining the reduced data.

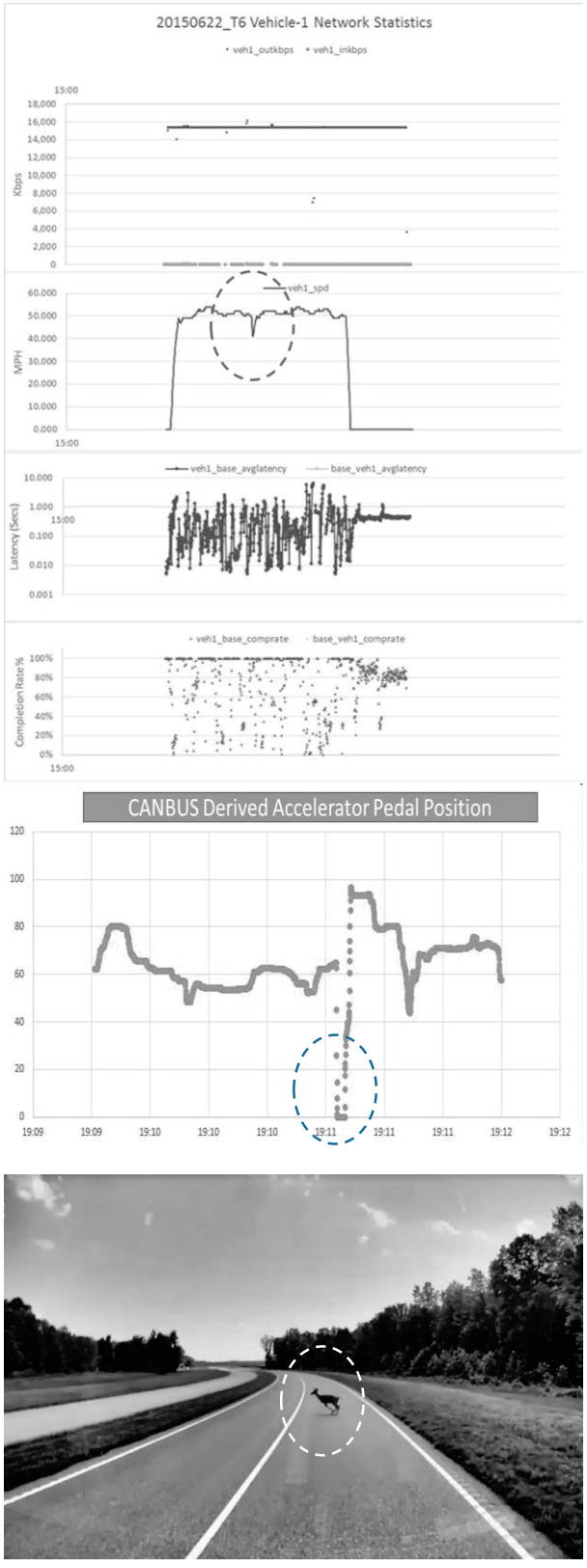


Figure 4 Cross-Domain Vehicular Anomaly Analysis

Future efforts related to robotics test and evaluation will employ a new JAUS module (currently under development). This new module will allow analysis of a vast amount of information on what a robotic entity is sensing and how it is reacting to its environment.

New video processing modules have been added which allow for indexing the video streams recorded from several sources into a queryable form that will allow the analysts and evaluators to quickly view pertinent video sources related to the vehicle at a specified time. Video processing that aides in object motion detection, and other computationally intensive video analysis techniques is also going to be incorporated.

ATC envisions adding new framework modules that will enable correlation over many sources to scan for anomalies in the data. This data can then be investigated in detail by the analysts to extract as much information as is possible for all sources available to support analytical objectives. A sample of this kind of cross-domain analysis is demonstrated in Figure 4. The network statistics charts (top section) includes a vehicle speed section (second line trace down from the top). The vehicle was supposed to be maintaining a relatively constant speed around the test track. The analysis indicates an unexpected momentary drop in vehicle speed (highlighted in the dotted circle). The CAN bus data (middle section of the figure) was queried for the vehicle at that time, and it was shown that the operator removed pressure from the accelerator (middle dotted circle). Querying the database for the corresponding video at the time showed exactly why the operator slowed down (a deer in his path).

2.3.2 MODESTA analysis

An up-and-coming capability from CERDEC S&TCD called Modeling, Emulation, Simulation Tool for Analysis (MODESTA) is being developed for system-of-systems experimentation, validation and assessments in a controlled lab environment. Analytical objectives of experimentations conducted using MODESTA are primarily driven to evaluate Tactical Applications and Waveform behaviors. In the current Army's tactical network architecture there are about six different radio waveforms, three different network security enclaves, numerous application hosts, and various network devices. With such layers of complex networks and devices, there are many data collection points. In order to evaluate the entire system-of-systems, all data collection points must be monitored to accurately analyze, evaluate, and troubleshoot system performance and integration issues.

MODESTA utilizes an updated version of the Command, Control, Communications, and Computers (C4) Data Model to store the resulting correlated data. The C4 Data Model was originally designed by Army Test and Evaluation Command (ATEC) and has been updated to enable storage of data generated from the diverse MODESTA environment. The MODESTA analysis team has leveraged the C4 Data Model to conduct deep dive analysis on multi-layered tactical waveform networks. For end-to-end system-of-systems performance assessments, it is important to track every packet as it traverses various ingress and egress collection points at all nodes in its path. Packet correlation of traversal events throughout large emulated virtual networks is computationally intensive. The smart hash-based pattern matching [9] performed in searching through the datasets at all virtual nodes reduces the complexity of end-to-end packet correlation. This end-to-end look enables analysts to deep-dive into performance-related issues. Having the ability to process the data in parallel on a large cluster also enables delivery of data products to analysts within a reasonable time frame.

2.4 Future ideas

The Data Parser updates have improved the framework's capabilities and now allows for processing of simulation and emulation data in addition to live experimental and real-world data. Though the framework works well in its current state, there are more improvements that are planned in the future. Reducing the memory footprint of Cut Modules and optimizing their algorithms will increase scalability. In addition, the possibility exists for a shift to a community-maintained big data ecosystem. Getting it to run well on an HPC cluster in comparison to commodity hardware would be the first hurdle to cross. Luckily, Lawrence Livermore National Laboratory's Magpie [10] is addressing this issue. The benefit would be a significant drop in maintenance and an increase in capabilities. Using a system that includes software such as Apache Spark allows for the use of utilities and libraries like its Machine Learning Library (MLlib) with minimal extra effort.

3 Conclusions

Having the capability to process large data sets of varying data types quickly and efficiently is important to the

Army. The analysis gathered from the tests helps validate and verify the systems in the field that assist the soldiers. Being able to follow the same processing pipeline on all data sources can drastically reduce the time and cost of testing new devices. The big-data field is growing rapidly and will continue to do so in the near future. The amount of data will only increase from this point on. Thus having reduction and processing methods in place will only be beneficial.

4 References

[1] B. Panneton, G. Besack, J. Adametz, B. Tauras K. Renard, "Processing and Analysis of Large Data Sets from High Bandwidth Tactical Networking Experiments Using High Performance Computing," *The International Test and Evaluation Association Journal*, vol. 36, no. 3, Sept 2015.

[2] CAN in Automation. (2016) CAN lower- and higher-layer protocols. [Online]. http://www.can-cia.org/can-knowledge/

[3] Andrew Feickert, "Joint Light Tactical Vehicle (JLTV): Background and Issues for Congress," LIBRARY OF CONGRESS CONGRESSIONAL RESEARCH SERVICE, Washington, DC, PDF CRS-RS22942, 2013.

[4] Joey Cheng. (2014, Sept) Defense Systems. [Online]. https://defensesystems.com/articles/2014/09/02/marines-army-amas-autonomous-convoys.aspx

[5] J. Adametz B. Panneton, "High-Bandwidth Tactical-Network Data Analysis in a High-Performance-Computing (HPC) Environment: HPC Data Reduction Framework," Army Research Labs, Aberdeen Proving Ground, Md, ARL-CR-0777, 2015.

[6] Naval Research Laboratory. (2016, Feb) Extendable Mobile Ad-hoc Network Emulator (EMANE). [Online]. http://www.nrl.navy.mil/itd/ncs/products/emane

[7] Steven Galgano, TestPoint Data Collection Framework, 2014, Rev. 1.2.

[8] B. Panneton K. Renard, "A Standard for Command, Control, Communications and Computers (C4) Test Data Representation to Integrate with High Performance Data Reduction," Army Research Laboratory, Aberdeen Proving Ground, Md, ARL-TR-7329, 2015.

[9] Panneton, Brian C; et. al., "High-Bandwidth Tactical Network Data Analysis in an HPC Environment: Packet-Level Analysis," Aberdeen Proving Ground, MD, 2015.

[10] Robin Goldstone, "Data Intensive Computing Solutions," Lawrence Livermore Nation Laboratory, http://computation.llnl.gov/projects/lc-big-data-leadership. [Online]. http://computation.llnl.gov/projects/lc-big-data-leadership

[11] TCPDUMP.ORG. (2015, Sept) TCPDUMP. [Online]. http://www.tcpdump.org/manpages/tcpdump.1.html

[12] Florian Thiel. (2000) Universal TUN/TAP Device Driver. [Online]. https://www.kernel.org/doc/Documentation/networking/tuntap.txt

Event Sequence Detection over Interval-Based Event Streams

Salah Ahmed, Olga Poppe, and Elke A. Rundensteiner
Computer Science, Worcester Polytechnic Institute, Worcester, MA, USA

Abstract— *Event stream applications from stock trend analytics to infection spread prevention must extract event sequence dependencies from event streams composed of interval-based events such as stock trades, records of shipments in storage, etc. The task of finding all possible sequences, which corresponds to Kleene closure, is not only NP Hard in CPU time, but also profoundly plagued by exponential memory usage. To tackle this challenge, we encode the incoming event stream as a directed acyclic graph called Compact Event Stream graph (CEStream or CES graph), the nodes of which are intervals denoting the events and the edges represent the non-overlapping relationships among the event-intervals. The CEStream data is captured in an interval tree data structure, called the CEStream index, to support efficient interval driven lookup. We analyze the state-of-the-art algorithms that leverage the CEStream index to compute longest event sequences with respect to their CPU time and memory consumption. Based on this analysis and thorough experimental study, we design the CES Fusion strategy which memorizes intermediate results computed on CEStream partitions and combines these intermediate results to construct the final results without memorizing them. CES Fusion is shown to achieve the best performance among all solutions, often exceeding 10 fold CPU time for feasible solutions.*

Keywords: Compact Event Stream, Longest Sequence, Kleene Closure, Stream Partitioning, Stream Optimization

1. Introduction

Background. Stock market analysis is a very important and widely used stream processing application for the financial sector. Finding the actual rising or falling trend of a particular stock or finding the relationship among different stocks is critical yet time consuming event processing application [14]. From an in-time forecast, traders can gain huge profits or can save themselves from huge losses. In Sep 29, 2008 the biggest fall in the US stock market of about $1.2 trillion caused huge financial losses to most people in the United States [2], [1]. That financial breakdown rapidly devolved into a global crisis resulting in a number of bank failures in Europe and sharp reductions in the value of stocks and commodities worldwide [3]. An intelligent event processing engine with rapid response time and the capability to handle the detection of stock trend patterns over the past transactions could potentially be utilized to help analysts to detect such looming trends to take appropriate actions.

Challenges. CEP applications like stock trend analysis face the following challenges:

High-rate event streams. Events arrives at very high rate from many different event sources, yet applications processing events require responsiveness.

Exponential CPU and memory cost. Finding longest sequences is NP-Hard in CPU, while also being affected by exponential memory.

Incremental evaluation of complex event patterns. Evaluation of Kleene closure on-the-fly degrades the responsiveness by causing reevaluation.

State-of-the-art. The best known solution for finding longest sequences in point-based event streams is described in [4], [21]. It requires exponential memory since it stores partial sequences by creating different runs for non-determinism. It eventually fails as huge numbers of sequences are generated, as confirmed by our experimental findings in Section 6.

Proposed Approach. In this work we focus on events with their occurrence time modeled as a time interval. Our approach detects the longest event sequences in high-rate interval based event streams specified as complex event pattern queries expressed by Kleene closure. We map the interval based events in a stream to a compact representation called compact event stream(CEStream). We design the CES Fusion solution, where we partitioned the CEStream using a linear time partitioning algorithm based on a robust cost model. The results of our study show that the compact representation of the stream, along with cost model based partitioning and a hybrid of memorized and non-memorized traversal strategies, produces results almost 10 times faster on average than the state-of-the-art approaches [4], [21]. More specifically, our contributions are:

1) We develop a model, called CEStream, for compactly representing interval-based event streams as a directed graph. The edges are formed by the overlapping relationships among the intervals of the events.
2) We design a novel approach, CES Fusion, for generating sequences, taking memory constraints into account. It uses the advantages of memoization and of the faster traversal of non-memoization in a hybrid approach.
3) We conduct an experimental evaluation of CES Fusion along with several key competitors. Our experiments demonstrate that the CES Fusion approach generates

results almost 10 times faster than the state-of-the-art approaches for a rich variety of scenarios.

Outline. We provide background in Section 2. In Section 3, we describe the base approach based on an interval index. In Section 4 we introduce the data structure and algorithms for the CEStream based approaches. While in Section 5 we describe the CES Fusion solution. Section 6 contains the experimental evaluation of CES Fusion compared to other techniques. Related work is presented in Section 7 and we conclude in Section 8.

2. Preliminaries

Time Intervals. Time is represented by a linearly ordered *set of time points* $(\mathbb{T}, \leq)$, where $\mathbb{T} \subseteq \mathbb{Q}^+$ and $\mathbb{Q}^+$ denotes the set of non-negative rational numbers. The *set of time intervals* is $\mathbb{TI} = \{[start, end] \mid start \in \mathbb{T}, end \in \mathbb{T}, start \leq end\}$. That is, a time interval is described by two time points $start$ and end indicating its bounds.

Event Stream. An *event* is a message indicating that something of interest happens in the real world. Each event e belongs to a particular *event type* E, denoted $e.type = E$. An event type E is described by a *schema* which specifies the set of *event attributes* and the domains of their values.

An event e has an *occurrence time interval* (also known as *application time* [18]) $e.interval = [e.start, e.end] \in \mathbb{TI}$ assigned by the event source.

Events are sent by event producers (e.g., sensors) to event consumers (e.g., an event stream processing engine) on an input *event stream I*.

Event Relationships. Events arriving on high-rate event streams might overlap, be contained in other events or even coincide. Two events e_1 and e_2 *overlap* if $e_1.start \leq e_2.end$ and $e_2.start \leq e_1.end$. Event e_1 is *contained* in another event e_2 if $e_1.start \leq e_2.start \leq e_2.end \leq e_1.end$. Events e_1 and e_2 *coinside* if $e_1.time = e_2.time$. Contained and coinciding event can be viewed as special types of overlapping events.

Two events form a *sequence* (also known as event sequence [20]) if they do not overlap. A sequence T is called *longest* if no event can be added to it. That is, an event which is not in T overlaps at least one event in T. In Figure 1, the sequence $[e_1e_2e_3e_4e_8]$ is a longest event sequence, while $[e_1e_2e_7e_8]$ is not a longest sequence because e_3 or e_5 can be added to it without breaking the non-overlapping relationship.

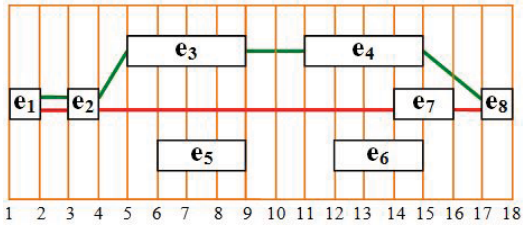


Fig. 1: Event, Overlapping Events and Longest Event Sequence

Event Sequence Pattern. To avoid reinventing the wheel, we reuse the event pattern syntax from [4], [20]. The event pattern syntax in Listing 1 captures all the rising sequence patterns in stock trades in the past 30 days. All stocks matched so far are denoted by $a[\,]$. The pattern uses a Kleene plus operator to extend the match. The predicate requires that the price of the current event $a[i]$ exceeds the price of the previously selected events $a[1, ..., i\text{-}1]$.

```
1 PATTERN SEQ(STOCK+ a[ ])
  WHERE a[i].price = a[i-1].price
3 WITHIN 30 days
  RETURN a[ ]
```

Listing 1: Event sequence pattern

Computing Kleene+ that returns all possible pattern matches has exponential CPU cost [21]. To save resources and thus improve system responsiveness, our approach detects the set of all *longest* event pattern matches (called longest event sequences, short LES). An application can then construct the final matches using any number of predicates as a light-weight post-processing step as long as the information about the events is maintained. It is also straight forward to push the predicates into the matching process [9].

Table 1 summarizes the notations used in the remainder of this paper.

Problem Statement. Given an input event stream I composed of interval-based events and an event sequence pattern p composed of Kleene closure and predicates as in Listing 1, our goal is to detect *all longest event sequences* or LES matched by the pattern p in a system with a fixed memory availability, while *minimizing the CPU costs* of the detection process.

3. Incremental LES Detection

As base case, we first introduce the classical strategy for sequence detection. For each new event we compare the event with each of the existing sequences to find the overlapping events. If the new event does not overlap with any event in an existing sequence, we add the new event to the end of that sequence. Otherwise, we extract the partial sequence without the overlapping events and add the new event to the end of the partial sequence to create a new sequence. Then we compare the new sequence with all the existing sequences to find duplication. If the new sequence is unique we save that to the list of sequences.

Table 2 shows the LESs before and after the arrival of event e7 in Figure 2. We notice that the new list of sequences contains duplicate sequences (shown in red). Removal of the duplicates must thus be undertaken - a very costly operation.

Algorithm in Listing 2 describes the incremental LES detection process.

Suppose N denotes the number of events, K the number of sequences, M the average number of overlapping events

Notation	Meaning
E	a set of all events
T	an interval tree
e	an event
L	the set of all LES in E
L'	the set of all LES constructed using e
S	a LET found so far
s'	a new LET constructed using e
overlapping(e, s)	the set of events in s overlapping e
overlapping(e,T)	the set of events in T overlapping e
following(e,T)	an event which is the immediately following event of e in T
successors(S,T)	the set of events in T which are successors of the events in S
predecessors(S,T)	the set of events in T which are predecessors of the events in S
firstEvent(S)	first event from the set of events in S based on the start time
lastEvent(S)	last event from the set of events in S based on the end time
size(S)	the number of events in the set of events S
Limit	minimum number of events in a partition
time[]	array to store all cut points
CESs[]	array to store the partitions of CES

Table 1: Notation

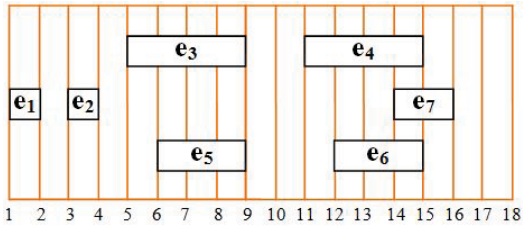


Fig. 2: e7 arrives

and L the average length of a sequence. Then the overall complexity of the algorithm in Listing 2 is $O(NK\ KL) = O(NK^2L) \approx O(N^2K^2) = O(N^2(3^{N/3})^2)$. For 100 events, K can be $3^{32} * 4 \approx 7412$ Trillion. The memory requirement can be 92000 Terabytes. Clearly this is unrealistic. This insight motivates us to explore better approaches.

4. CEStream Approach

We propose to represent the events as a directed acyclic graph called Compact Event Stream (CEStream). That is, each event is a node and edges model the *direct* non-overlapping interval relationships between the events. Figure 3 shows the CEStream representation of the events in Figure 1. In our context we work with interval-based events, and thus propose to maintain all interval based events of the CEStream in an *Interval Tree* data structure to quickly find the overlapping relationships among the events. Given an interval tree, we can find an overlapping interval for a given time period in $O(log\ N)$ time where N is the total number of intervals in the tree.

In general the CEStream approach is two-stepped approach, namely:

1) Construction of the CEStream Structure

Existing List	New List After Arrival of e_7	
$[e_1e_2e_5e_4]$	$[e_1e_2e_5e_4]$	$[e_1e_2e_5e_7]$
$[e_1e_2e_3e_4]$	$[e_1e_2e_3e_4]$	$[e_1e_2e_3e_7]$
$[e_1e_2e_5e_6]$	$[e_1e_2e_5e_6]$	$[e_1e_2e_5e_7]$
$[e_1e_2e_3e_6]$	$[e_1e_2e_3e_6]$	$[e_1e_2e_3e_7]$

Table 2: New List of LES

```
IncrementalLESDetection(E)}{
  /* input : set of events E */
  // output: set of all longest sequences L
  for each event e ∈ E
    if L = ∅
    then L ← L ∪ e
    else L' = ∅
      for each s ∈ L
        // find the overlapping event of the
        // new event e in each of the
        // existing sequences
        O ← overlapping(e,s)
        if O = ∅
        then // extend s by adding e
          s ← s ∪ e
        else // create new sequence
          s' = (s − O) ∪ e
          if s' ∉ L' // remove duplicates
          then L' ← L' ∪ s'
          end if
       end if
      end for
      L ← L ∪ L' // save new sequences
    end if
  end for
  return L
}
```

Listing 2: Incremental LES Detection Algorithm

2) Application of graph traversals in the CEStream Structure to form all result sequences.

Step 1: Basic Concept of CEStream construction. Whenever an event arrives, we find the predecessors and successors of the new event based on the overlapping relationship of the events in the interval tree. We update the graph by updating the successors of the predecessors and the predecessors of the successors. To find the predecessors of a new event we first find all the overlapping events of the new event, then we find the predecessors of those overlapping events, which produces the predecessors of the new event. Finding successors of a new event is similar, i.e., it is equal to the process of the successors of the overlapping events of the new event. Listing 3 illustrates the algorithm for constructing CEStream, the compact event stream structure.

Step 2: Apply Graph Traversals. After the compact graph has been constructed, then we can apply any standard graph traversal algorithm to find all the longest event sequences. We choose Depth First Search(DFS) as the traversal algorithm.

Discussion of CEStream Approach. Let N be the number of events, K the number of sequences and M the average number of overlapping events. Then the overall complexity

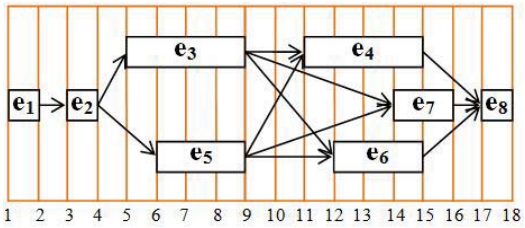


Fig. 3: CEStream graph from events in Fig 1

```
ConstructCESGraph(E,T){
  // input : set of events E and
  //         interval tree T
  // output: interval tree T with events with
  //         their successor and predecessor
  //         relationships in compact graph

  for each e ∈ E
    // find events overlapping with e
    O ← overlapping(e,T)
    //find successors of overlapping events
    S ← successors(O,T)
    // find predecessors of overlapping events
    P ← predecessors(O,T)
    // update successors of the new event
    e.successors ← S
    // update predecessors of the new event
    e.predecessors ← P
    insert e into T
  end for
  return T
}
```

Listing 3: CES Graph Construction Algorithm

of the construction of the CEStream as shown in Listing 3 is *O(N M log N)*. The complexity of depth first search (DFS) is *O(K)*. So the total complexity is *O(N M log N + K)*.

DFS depends on the number of sequences. So, for higher values of N, the value of K is always higher than NMlogN. The experiments indicate that the construction time of CEStream graph is negligible compared to the execution time of DFS.

Table 3 illustrates the memory requirement for two different versions of DFS traversal: DFS with memoization and DFS without memoization. Section 6 contains the details of the experiments. DFS without memoization does not store all the sequences, so it has very little memory requirement.

Algorithm	Memory Requirement
DFS with Memoization	O(NK)
DFS with No Memoization	O(N)

Table 3: Memory Requirement

Table 4 summarizes our key observations for CEStream approach. We find:

1) The CEStream graph construction time is very low and negligible compared to the time for DFS.

#	Memorize	# of Sequences	CPU Time	Memory
1	**Yes**	**< 200**	**Low**	**Low**
2	Yes	> 400	Low	Overflow
3	No	< 200	Moderate	Low
4	**No**	**> 400**	**High**	**Low**

Table 4: Summary of Experimental Observations with CEStream

2) For the number of sequences less than 200, DFS with memoization performed significantly better than DFS without memoization.
3) For the number of sequences greater than 800000, DFS with memoization always crashes with the out-of-memory error.
4) A trade-off between CPU time and memory consumption can be achieved by combining the advantages of rows 1 and 4 in Table 4 as will be discussed below.

5. CES Fusion Approach

Based on our initial findings described above we develop a three-step process called CEStream Fusion (CES Fusion):

1) Partition the bigger CEStream graph into smaller CEStream graphs.
2) Apply DFS with memoization to each of the smaller CEStream graphs.
3) Apply DFS without memoization to generate final result sequences by stitching together partial results from smaller CEStream graphs.

The idea is to divide the graph into many smaller graphs. We then find all the LESes of the smaller CEStreams graphs. Lastly, we combine the results of smaller CEStreams graphs to generate all final LESes.

Definition 5: A *cut* is a time point which divides a CEStream graph into two smaller CEStreams graph. If the cut overlaps with k events it is called a *k-cut*. Partitions of CEStream formed by *0-cut* are called *primitive partitions* and denoted by P_1, P_2, etc. A partition formed by two or more consecutive primitive partitions is called a *compound partition*. It is denoted by listing its constituent primitive partitions consecutively, e.g., $P_1P_2P_3$.

Figure 4 applies a *0-cut* at time 10 for the graph of Figure 3. This creates two *partitions* P_1 and P_2. P_1 consists of events e_1, e_2, e_3 and e_5; and P_2 consists of events e_4, e_6, e_7 and e_8. The *compound partition* formed by P_1 and P_2 is P_1P_2.

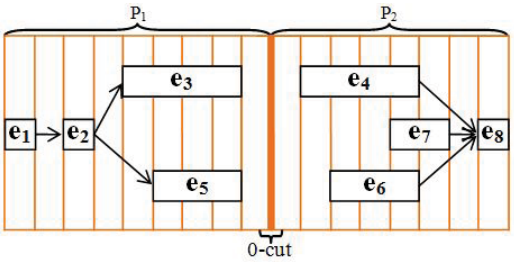


Fig. 4: Cut Point and Partitions

Algorithm	CPU Time	Memory
Non-Partitioned	|V|+|E|	$\|V\|^2.K$
Partitioned	|V|+|E|	$\|V_1\|^2.K_1 + \|V_2\|^2.K_2$

Table 5: Comparison of Partitioned and Non-Partitioned CEStream

P_1 has two Longest Event Sequences (LES)), $[e_1e_2e_3]$ and $[e_1e_2e_5]$. P_2 has 3 LESes, $[e_4e_8]$, $[e_6e_8]$ and $[e_7e_8]$. All sequences in P_1 are then combined with all sequences in P_2. So, there are 2 x 3 = 6 longest event sequences. If there are p groups with number of LESes n_1, n_2, ..., n_p respectively, then the total number of longest event sequences in the system would be n_1 x n_2 x ...x n_p. For this paper we consider 0-cut points, while the generalization of our approach to k-cut is a subject for future work.

Step 1. Partitioning the CEStream graph. We first find all the 0-cut points and try to merge consecutive partitions as long as the estimated number of sequences is below a specified limit, which can be the number of events or the number of sequences. For the purpose of this paper we choose number of sequences as the limit and fixed that to 200. This heuristics has been chosen based on the observations in Table 4 that the DFS with memoization performs better when the number of sequences in a CESteam graph is below 200. Listing 4 displays the algorithm used as foundation to partition the graph.

Step 2. Apply DFS with memoization to each partition. We apply the DFS with memoization to each of the smaller CEStream graphs to construct the partial sequences. The sequences of the smaller CEStream graphs are stored in a list of indexed by the partition's index.

Step 3. Apply DFS without memoization. We traverse the list of sequences of the smaller graphs using DFS without memoization to construct the final sequences. For each sequence of each of the partitions is joined with each of the sequence of the following partition. It directly outputs the longest sequence when the traversal reaches the last partition.

The search space of all possible partitions is exponential ($O(2^c)$) in the number of cut points, c.

Example: For five cut points, the solution space lattice is shown in Figure 6. Each node in the lattice is a solution. Figure 5 gives 0-cut partitions with cut points A, B, C, D and E. The top number in a partition denotes the estimated number of sequences in that partition and the bottom number the estimated average length of a sequence. Figure 6 demonstrates the branches traversed by the partitioning algorithm. The top level node containing ABCDE denotes that this solution contains all the five cut-points. In the next level, each solution contain four cut-points, which means that one of the cut-points is removed to merge two primitive partitions. So, in the second level, the algorithm first removes cut-point A (since 12 x 14 < 200) to merge the partitions P_1 and P_2. Then it tries to remove B, but cannot remove (dashed line) B, because the cost of the solution partition formed by merging three primitive partition P_1, P_2 and P_3

```
1  PartitionCES(T, limit, cut){
     /* input : interval tree T} */
3    //         limit of each partition
     //         cut value: e.g.,0 for 0-cut
5    // output: all the cut-points
     i ← -1
7    // initialize cut points
     cuts ← null
9    prev_time ← firstEvent(T).start

11   // jump to the next event of the last
     // event of the current partition
13   e ← following(lastEvent(
       overlapping(firstEvent(T),T),T),T)
15
     while e ≠ null
17       // check size of the cut
       if size(overlapping(e.start-1,T)) = cut
19       and
         // check limit of the partition
21       size(overlapping( prev_time,
         e.start-1,T)) = limit
23     then i ← i+1
         // save the cut point
25       cuts[i] ← e.start-1
         prev_time ← cuts[i]
27     end if

29     // jump to the next event of the last
       // event of the current partition
31     e ← following(lastEvent(
           overlapping(e,T),T),T)
33   end while
     return cuts
35 }
```

Listing 4: CEStream Partitioning Algorithm

by removing cut-points A and B, exceeds the allowed cost (12 x 14 x 15 > 200). Similarly it removes cut-point C, but cannot remove D, and finally removes E. The final solution contains the cut-points B and D with partitions P_1P_2, P_3P_4 and P_5P_6.

Comparison of Memory Consumption: Table 5 lists the memory requirements for the partitioned and the non-partitioned approaches. If there are 100 events and 10 partitions, each having 10 events and 10 sequences, the memory requirement for partitioned CEStream is 10^2 x 10 x 10 = 10,000, whereas for non-partitioned CEStream its 100^2 x 10^{10} = 100,000,000,000,000 (100 Trillion).

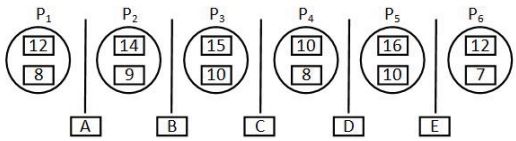


Fig. 5: Five cut-points with six partitions

Complexity of the Partitioning CES Algorithm. Let *N* be the number of events, *M* the average number of overlapping events and *P* the number of partitions. The

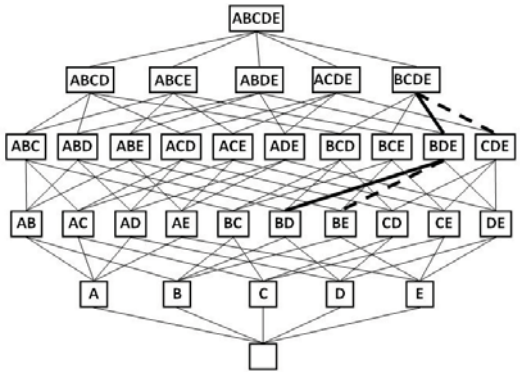


Fig. 6: Path taken by partitioning algorithm

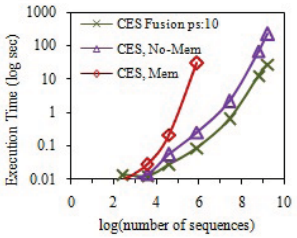


Fig. 7: Execution Times for Different Approaches

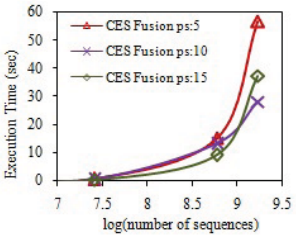


Fig. 8: Execution Times for Different Partitionings

loop in line 14 in Listing 4 iterates P times and in each iteration it finds the next set of overlapping events in the next partition. When a partition satisfying the limit and cut parameters is found, the algorithm records that cut point and skips all the events in the already considered partitions. The overall complexity of the algorithm in Listing 4 is $O(P\ M\ logN)$. Using an experimental study, we observed that bounding the number of sequences in a partition to below 200 produces near optimal solution within O(c) time, where c is the number of 0-cut points.

6. Experimental Evaluation

6.1 Experimental setup

The platform of the experiments was Intel Core i7 processor with 2GHz clock speed, 4 cores and 8 GB RAM. We have used Java as the programming language. Using Java we created an Event Generator. The start time and the interval of each event can be varied randomly or kept fixed within a window. We implement the algorithms for CES construction, CES traversals, CES graph partitioning. The performance metrics were number of sequences generated and the CPU response time. For the experiments we use fixed window length of 1000. We tested with both fixed and variable length of the interval of the events.

6.2 Experiments

The experiment in Figure 7 exhibits the execution time of the three approaches: CES Fusion, CES-with-memoization (CES Mem) and CES-without-memoization (CES No-Mem). CES-Mem fails after about a million sequences because of the out-of-memory problem. CES-without-memoization works, but it is evident that the time for CES-without-memoization is extremely high compared to the CES Fusion approach.

Figure 8 compares the execution times of three different partitioning of the CES Fusion approach. Although the CES Fusion approach outperforms any non-partitioning approaches, there is significant difference between the different partitioning solutions based on partition size (ps). ps=15 and ps=10 performed better than ps=5. ps=5 had more partitions since it allowed approximately 5 events in a partition. This indicates that when the number of partitions decreases the execution time also decreases.

From the experiments we found that the CEStream approach with memoization could handle up to approximately 800000 sequences of average length of 7. Above that limit, CEStream with memoization fails due to out-of-memory. The CEStream approach without memoization could handle unlimited number of sequences, however, generating approximately 1.7 billion sequences of average length of 8, took 224 seconds. Surprisingly the CES Fusion approach produced 1.7 billion sequences of average length of 8, in just 28 seconds. This is 224/28= 8 times faster on average than the CEStream approach without memoization. The state-of-the-art approaches are equivalent to the CEStream approach with memoization. Even if we consider the state-of-the-art to be comparable to CEStream without memoization, the CES Fusion is 8 times faster on average and in many occasions CES Fusion is more than 10 times faster. For 800000 sequences the CES Fusion took only 54ms, and CEStream with memoization took 30138 ms. So CES Fusion is 30138/54 = **558 times** faster than CEStream with memoization.

Figure 11 combines Figures 9 and 10. Figure 12 manifests the actual execution time for different partitions of CES Fusion. From Figure 11 and Figure 12 it is evident that the cost model reflects the actual execution time very well. When the number of partitions is near 4 to 5, both the estimated cost of DFS and the estimated cost of generating the sequences were lowest. This is also visible in Figure 11, the total estimated cost. The actual execution time is found to be lowest when the number of partitions is about 4 to 5. With our cost model we achieved at most around 75% accuracy to find the near optimal solution in linear time by the partitioning algorithm.

7. Related Work

Our work is inspired by [4] evaluating complex queries including Kleene closure. But their method of generating all possible sequences is susceptible to memory overflow since they store all previous sequences. For every non-deterministic branch they create a new run and that leads to memory burst. Active databases in [10], [11], [6], [15], [16],

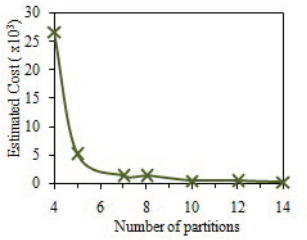


Fig. 9: Number of Partitions vs Cost of DFS

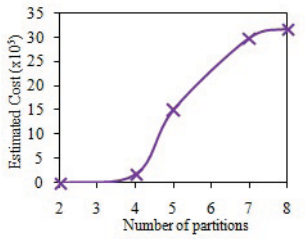


Fig. 10: Number of Partitions vs Cost of Sequence Generation

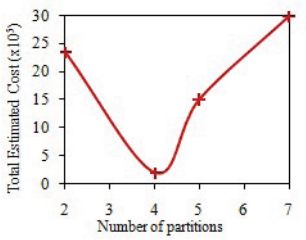


Fig. 11: Number of Partitions vs Total Cost

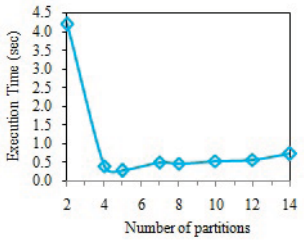


Fig. 12: Number of Partitions vs Execution Time

[22] handle interval based operators like sequencing and Kleene closure. Cayuga [7], [8] supports patterns containing Kleene closure, but does not support partitioning.

Most event processing systems [13], [17], [19], [20] do not focus on Kleene closure. Our work significantly extends NFA^b [4] by optimizing memory and CPU consumption. SASE+ [12] is a good language for pattern matching, but no implementation details are given.

CEDR [5] shows a temporal model based on the duration of events and their out-of-order relationships. We instead focus on partitioning the event stream and optimizing the consumption of memory and CPU.

8. Conclusion

Our proposed model of the CEStream Structure for compactly encoding partial max-sequences solves the problem of storage. Partitioning the huge CEStream into smaller CEStreams takes advantage of memoization and reduces CPU time and memory usage. The result generation by non-memoization has the advantage of handling a huge number of events. For interval based event streams, using 0-cut concept, we show that CES Fusion is more than 10 times faster than the state-of-the-art approaches for many occasions. The processing of out-of-order events is promising future work. Managing sliding windows and caching for sliding windows and for the search space are ongoing future work. The processing of partial sequences of each CEStream can be done in parallel in the cloud based environments using Storm, S4 etc.

Acknowledgments

Supported by NSF grants IIS 1018443 and IIS 1343620.

References

[1] Money Morning: Stock Market Crash History: The Dow's 10 Biggest One-Day Plunges. http://moneymorning.com/2014/02/13/stock_market_crash_history_dows_10_biggest_one_day_plunges/, 2014. [Online; February 13, 2014].

[2] CNN Money. http://money.cnn.com/2008/09/29/markets/markets_newyork/, 2016. [Online; September 29, 2008].

[3] Wikipedia. https://en.wikipedia.org/wiki/Stock_market_crash, 2016. [Online; 1 March 2016].

[4] J. Agrawal et al. Efficient pattern matching over event streams. In *Proc. of Int. Conf. on Management of data*, SIGMOD '08, pages 147–160. ACM, 2008.

[5] R. S. Barga, J. Goldstein, M. H. Ali, and M. Hong. Consistent streaming through time: A vision for event stream processing. In *CIDR*, pages 363–374, 2007.

[6] S. Chakravarthy, V. Krishnaprasad, E. Anwar, and S.-K. Kim. Composite events for active databases: Semantics, contexts and detection. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, pages 606–617, 1994.

[7] A. Demers, J. Gehrke, M. Hong, M. Riedewald, and W. White. Towards expressive publish/subscribe systems. In *Advances in Database Technology - EDBT 2006*, volume 3896, pages 627–644. Springer Berlin Heidelberg, 2006.

[8] A. Demers, J. Gehrke, and B. P. Cayuga: A general purpose event monitoring system. In *In CIDR*, pages 412–422, 2007.

[9] Y. Diao, N. Immerman, and D. Gyllstrom. Sase+: An agile language for kleene closure over event streams, 2007.

[10] S. Gatziu and K. Dittrich. Events in an active object-oriented database system. In *Rules in Database Systems*, Workshops in Computing, pages 23–39. Springer London, 1994.

[11] N. H. Gehani, H. V. Jagadish, and O. Shmueli. Composite event specification in active databases: Model & implementation. In *Proceedings of the 18th International Conference on Very Large Data Bases*, VLDB '92, pages 327–338, 1992.

[12] D. Gyllstrom, J. Agrawal, Y. Diao, and N. Immerman. On supporting kleene closure over event streams. In *Int. Conf on Data Engineering*, pages 1391–1393. IEEE, 2008.

[13] L. Harada and Y. Hotta. Order checking in a cpoe using event analyzer. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, pages 549–555. ACM, 2005.

[14] B. F. Jun Wang and D. Men. Data analysis and statistical behaviors of stock market fluctuations. *Journal of Computers*, 3:44–49, 2008.

[15] D. Lieuwen, N. Gehani, and R. Arlein. The ode active database: trigger semantics and implementation. In *Proceedings of the 12th International Conference on Data Engineering, 1996.*, pages 412–420, 1996.

[16] R. Meo, G. Psaila, and S. Ceri. Composite events in chimera. In *Advances in Database Technology*, volume 1057 of *Lecture Notes in Computer Science*, pages 56–76. 1996.

[17] S. Rizvi, S. R. Jeffery, S. Krishnamurthy, M. J. Franklin, N. Burkhart, A. Edakkunni, and L. Liang. Events on the edge. In *In SIGMOD*, pages 885–887, 2005.

[18] U. Srivastava and J. Widom. Flexible time management in data stream systems. In *Proc. of Symposium on Principles of Database Systems*, PODS, pages 263–274. ACM, 2004.

[19] F. Wang and P. Liu. Temporal management of rfid data. In *Proceedings of the 31st International Conference on Very Large Data Bases*, VLDB '05, pages 1128–1139. VLDB Endowment, 2005.

[20] E. Wu et al. High-performance Complex Event Processing over streams. In *Proc. of Int. Conf. on Management of data*, SIGMOD '06, pages 407–418. ACM, 2006.

[21] H. Zhang et al. On complexity and optimization of expensive queries in complex event processing. In *Proc. of Int. Conf. on Management of Data*, SIGMOD'14, pages 217–228. ACM, 2014.

[22] D. Zimmer and R. Unland. On the semantics of complex events in active database management systems. In *Proceedings of 15th International Conference on Data Engineering, 1999.*, pages 392–399, 1999.

Efficient Lossless Compression of 4D Hyperspectral Image Data

Hongda Shen, W. David Pan and Yuhang Dong
Dept. of Electrical and Computer Engineering
University of Alabama in Huntsville
Huntsville, AL 35899, USA
Emails:{hs0017, pand, yd0009}@uah.edu

Abstract— *Time-lapse hyperspectral imaging technology has been used for various remote sensing applications due to its excellent capability of monitoring regions of interest over a period of time. However, large data volume of four-dimensional hyperspectral imagery demands for massive data compression techniques. While conventional 3D hyperspectral data compression methods exploit only spatial and spectral correlations, we proposed a novel lossless compression algorithm that can achieve significant gains on compression efficiency by also taking into account temporal correlations inherent in the dataset. Experimental results demonstrated the effectiveness of the proposed algorithm.*

Keywords: Lossless compression, 4D hyperspectral image, temporal correlation, LMS, correntropy

1. Introduction

With more advanced remote sensing sensors being used, the spatial and spectral resolutions of the images captured by those sensors has increased rapidly, which naturally leads to large data volume. Hyperspectral imaging technology collects the image information across a wide-range electromagnetic spectrum with fine wavelength resolution. Hence, a hyperspectral image (HSI) is a three dimensional data cube with two spatial dimensions and one spectral dimension. Given the fact that most of remote sensing sensors collect data using either 12-bit or 16-bit precision, the size of a hyperspectral image cube is typically very large.

Time-lapse hyperspectral imagery is a sequence of 3D HSIs captured over the same scene but at different time stamps (often at a fixed time interval). Actually, time-lapse hyperspectral imagery can be considered as a 4D dataset whose size increases significantly with the total number of time stamps. Fig. 1 shows an illustration of one time-lapse hyperspectral image dataset. Each stack represents one 3D HSI. Furthermore, more stacks will be captured by the HSI sensor with the time. Particularly, in the extreme case of 4D HSI data streaming, the captured data volume accumulates very fast. This huge data volume does not only slow down the data transmission within the limited bandwidth condition but also requires more storage space which could be very expensive in many remote sensing applications. Data compression techniques provide a good solution to these problems. As captured images are most likely at high fidelity for the accuracy demanding applications, lossless compression is often chosen for these sensors over the lossy compression.

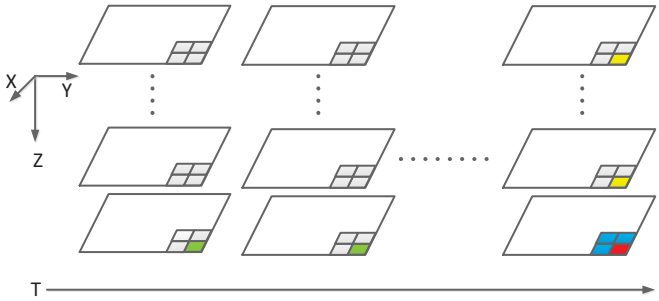


Fig. 1: A 4D time-lapse hyperspectral image dataset, where X and Y are the spatial directions, and Z is the spectral direction.

Large efforts have been made to develop a lossless compression algorithm for 3D HSI. LOCO-I [1] and 2D-CALIC [2] utilize spatial redundancy to reduce the entropy of prediction residuals. Since there exists strong spectral correlation, 3D methods including 3D-CALIC [3], M-CALIC [4], LUT [5] and its variants, SLSQ [6] and CCAP [7] take this spectral correlation into account and yield better compression performance. Also, some transform-based methods, such as SPIHT [8], SPECK [9], etc., can be easily extended to lossless compression even though they were designed for lossy compression. In addition to the goal of reducing the entropy of either prediction residuals or transform coefficients, low computational complexity is another influential factor because many sensing platforms have very limited computing resources. Therefore, a new method named as the “Fast Lossless" (FL) method, proposed by the NASA Jet Propulsion Lab (JPL) in [10], was selected as the core predictor in the CCSDS new Standard for Multispectral and Hyperspectral Data Compression [11], to deal with 3D HSI data compression.

This is a regular research paper. W.D. Pan is the contact author.

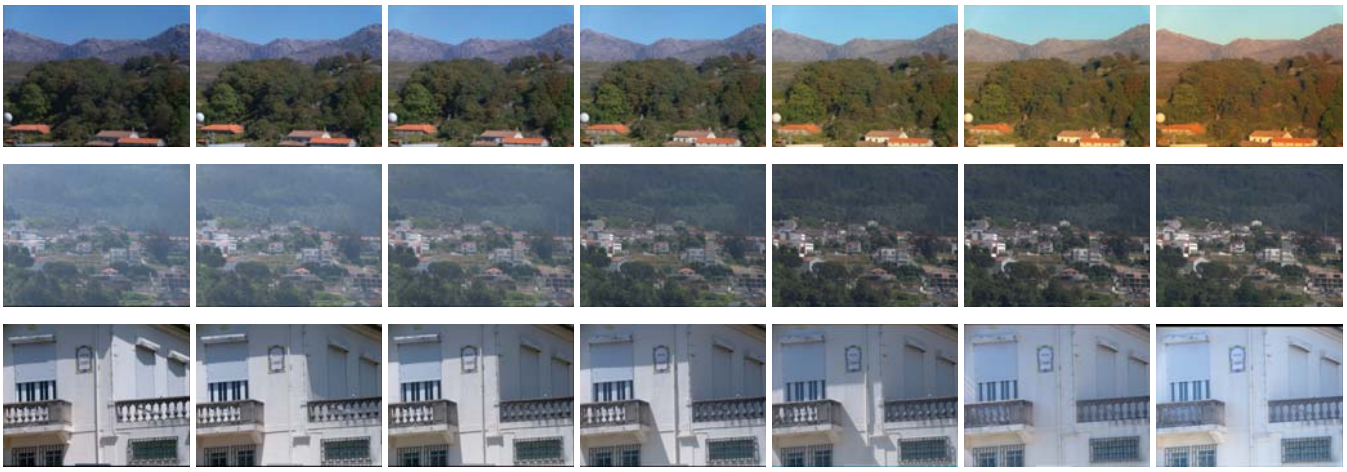


Fig. 2: Sample time-lapse hyperspectral image datasets at different time instants (from top to bottom: *Levada*, *Nogueiro* and *Gualtar*).

To give an idea on the 4D image datasets tested in this work, Fig. 2 shows the *Levada* sequence. Detailed information about the *Levada* sequence can be found in [12]. Note that only 2D color-rendered RGB images are shown in Fig. 2 instead of the actual HSI data for display purpose. Since time-lapse HSIs are captured over the same scene at different time instants with gradually changing natural illumination, there exists great similarity among these HSIs at each time instant marked in Fig. 2. Therefore, this temporal correlation can be further exploited to improve the overall compression efficiency. To the best of our knowledge, there is very few prior work on lossless compression of 4D time-lapse HSI data in the literature. [13] proposed a 4D lossless compression algorithm, albeit lacking details on the prediction algorithms used for prediction. On the other hand, in [14], a combination of Karhunen-Loeve Transform (KLT), Discrete Wavelet Transform (DWT) and JPEG 2000 has been applied to reduce the spectral and temporal redundancy of 4D remote sensing image data. However, it is a lossy compression method.

In this work, we conducted an information-theoretic analysis on the amount of compression achievable on 4D HSI based on conditional entropy, by taking into account spectral and temporal correlations. We then proposed a low-complexity correntropy-based least mean square (CLMS) learning algorithm, which was employed for the first time as a predictor to achieve higher data compression by better adapting to the underlying statistics of HSI data.

The rest of this paper is organized as follows. Section 2 introduces an information theoretic analysis framework for time-lapse HSI lossless compression. Section 3 reviews the CLMS learning algorithm. Furthermore, the proposed lossless compression engine based on CLMS learning is presented in detail. Experimental results are given in the Section 4. The paper is concluded in Section 5 with a discussion on the further work.

2. Problem Analysis

In order to evaluate the potential amount of compression we can achieve on the 4D dataset, we conducted an information-theoretic analysis. Let X_j^t be a 4D hyperspectral image source at the t^{th} time instant and j^{th} spectral band producing K different pixel values v_i $(i = 1, \cdots, K)$. Then the entropy of this source is computed based on the probabilities $p(v_i)$ of these values by

$$H(X_j^t) = -\sum_{i=1}^{K} p(v_i) \cdot log_2 \left[p(v_i)\right]. \quad (1)$$

If we assume that there are no dependencies between these pixel values for X_j^t, at least $H(X_j^t)$ bits must be spent on average for each pixel of this source. However, for the 4D hyperspectral images, this assumption does not hold given the existence of strong spectral and temporal correlations. The value of a particular pixel might depend on some other pixels from its spatial, spectral or temporal neighborhoods. Therefore, these correlations can be exploited to reduce the $H(X_j^t)$, i.e., less bits spent on average after compression. Furthermore, the conditional entropy of this time-lapse hyperspectral image source can be computed as follows:

$$H(X_j^t|C_j^t) = -\sum_{i=1}^{K} p(v_i|C_j^t) \cdot log_2 \left[p(v_i|C_j^t)\right]. \quad (2)$$

where C_j^t denoted as *context*, which represents a group of correlated pixels. As long as there is any correlation between the context C_j^t and the current pixel, $H(X_j^t|C_j^t) < H(X_j^t)$ always holds, in other words, fewer bits are required after compression.

The choice of context largely determines how much compression we can achieve by using prediction-based lossless

compression schemes. Intuitively, highly-correlated pixels are expected to be included into the context. Given the consideration that spectral and temporal correlations are typically much stronger than spatial correlation in hyperspectral images, our focus in this work is on spectral and temporal decorrelation. In fact, recent research [15], has shown that explicit spatial decorrelation is not always necessary to achieve good performance [16]. Also, in contrast to nonlinearity nature of spatial decorrelation, a linear prediction scheme is believed to be adequate for spectral and/or temporal prediction because of high degree of correlations [16]. In Section 4, we will investigate the actual compression gains using different combinations of context pixels.

3. The Algorithm

Linear prediction based lossless compression method uses a linear combination of those encoded pixels (causal context pixels) adjacent to the current pixel as its estimate. For 4D time-lapse HSI lossless compression, a linear prediction can be generalized as follows:

$$\widehat{x}_{m,n}^{t,j} = \mathbf{w}_{t,j}^T \mathbf{y}_{m,n}^{t,j}. \tag{3}$$

where $\widehat{x}_{m,n}^{t,j}$ represents an estimate of a pixel, $x_{m,n}^{t,j}$, at spatial location (m,n), j^{th} band and t^{th} time frame while $\mathbf{y}_{m,n}^{t,j}$ and $\mathbf{w}_{t,j}$ represent its causal context pixels and linear weights respectively. Note that only adjacent spectral bands and bands from previous time frames are included in this context as mentioned in Section 2.

Prediction residuals are generated by subtracting the actual pixel values from their estimates and then encoded using entropy coders such as Golomb-Rice Codes (GRC) [17] and Arithmetic Codes (AC) [18]. In order to produce accurate estimates, linear weights must be adapted to the local statistics of pixels in the time-lapse HSI data. Recently, learning algorithms have gained some success to optimize these weights in the applications of lossless compression of 3D HSI data [10], [19]. The FL method has been selected as a new standard by CCSDS for its low-complexity and effectiveness. The core learning algorithm of FL method is least mean square (LMS). Traditional LMS methods use mean square error (MSE) as the cost function. However, it is well known that MSE is the optimal cost function for Gaussian distributed signal [20], whereas the prediction residuals more likely follow a Laplacian or Geometric distribution [1]. So the performance of the conventional LMS predictor, for example, the FL method may degrade in presence of non-Gaussian signals, especially in those very structured regions of one image. Some similar observations have been noticed in [19]. To improve the robustness of the predictor, we introduce an adaptive learning based on the Maximum Correntropy Criterion (MCC) [20].

3.1 Correntropy-Based LMS (CLMS) Cost Function

Correntropy was developed as a local similarity measure between two random variables X and Y in [21], defined by:

$$V_\sigma(X,Y) = E\left[\kappa_\sigma(X-Y)\right], \tag{4}$$

where κ_σ is a positive definite kernel with kernel width controlled by the parameter σ, and the expectation $E(\cdot)$ is practically computed using sample arithmetic average. By following [20], we choose the normalized Gaussian kernel with variance σ as the kernel $\kappa_\sigma(\cdot) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(\cdot)^2}{2\sigma^2}}$.

In [21], Taylor series expansion was applied on the exponential term in the kernel in Eq. (4) so that the Correntropy can be viewed as a generalized correlation function containing even higher order moments of the error signal $X-Y$. Also, it is justified in [21] that localization introduced by the kernel can reduce the detrimental effects of outliers and impulsive noise, while second-order statistics, like MSE, may suffer from bias in these conditions. The good behavior of second order moment and fast convergence of the higher order moments are combined into this Correntropy measure. An adaptive filter was developed by replacing the conventional MSE with this Correntropy as the cost function. The detailed properties of Correntropy with derivation and analysis can be found in [21]. Assume we have a pair of random variables with a finite number of samples $\{d_i, y_i\}_{i=1}^N$ where N is the number of samples in each random variable. For example, d_i and y_i can be viewed as the actual pixel value and its estimate, respectively, in this work. Furthermore, the estimate y_i can be computed as $y_i = \mathbf{W}_i^T \mathbf{X}_i$, a linear weighted average of input vector $\mathbf{X}_i$. The Correntropy based cost function, at n^{th} time instant, can be written as:

$$J_n = \frac{1}{N\sqrt{2\pi}\sigma} \sum_{i=n-N+1}^{n} \exp\left[\frac{-\left(d_i - \mathbf{W}_n^T \mathbf{X}_n\right)^2}{2\sigma^2}\right], \tag{5}$$

where $\mathbf{W}_n$ is the filter weight at n^{th} time instant. To find the weight $\mathbf{W}$ to maximize this cost function analytically, iterative gradient descent method is used with a small learning rate μ. After computing the gradient of J_n with respect to $\mathbf{W}_n$, we obtain:

$$\mathbf{W}_{n+1} = \mathbf{W}_n + \frac{\mu}{N\sqrt{2\pi}\sigma^3} \sum_{i=n-N+1}^{n} \left[\exp\left(\frac{-{e_i}^2}{2\sigma^2}\right) e_i \mathbf{X}_i\right], \tag{6}$$

where $e_i = d_i - \mathbf{W}_n^T \mathbf{X}_n$. Inspired by the stochastic gradient, N is set to 1 to approximate the sum in Eq. (6). Therefore,

$$\mathbf{W}_{n+1} = \mathbf{W}_n + \frac{\mu}{\sqrt{2\pi}\sigma^3} \exp\left(\frac{-{e_n}^2}{2\sigma^2}\right) e_n \mathbf{X}_n, \tag{7}$$

which is very similar to the weight updating function of LMS. In fact, this Correntropy-induced updating function can be viewed as LMS with a self-adjusting learning rate,

which reflects the outlier rejection property of the Correntropy. It is worth noting that this CLMS is more robust to the outliers with almost no additional cost of algorithmic complexity compared to the conventional LMS.

In many remote sensing applications, images of scenes have complex structures as shown in Fig. 2. These structures will be reflected in the hyperspectral images as strong edges and corners. This non-linearity property of the image signal directly contributes to relatively larger residual values and consequently outliers for the residual data distribution because linear predictors cannot fully reduce the redundancy in these cases. To improve the compression performance, we utilize the CLMS to tackle structured regions in the HSI data for its outlier rejection property. On the other hand, convergence speed of the adaptive filtering plays a crucial role in the prediction. Slow convergence often leads to less accurate estimation in the prediction. CLMS provides a robust performance in the non-Gaussian condition with a faster convergence compared to the conventional LMS. Therefore, CLMS can help greatly enhance the prediction accuracy, which will further contribute to better compression.

3.2 CLMS Based Predictor

To eliminate spatial correlation effect, local mean subtraction is conducted in every band of these datasets. In Fig. 1, suppose the red pixel is the one we are predicting and the arithmetic average of the three blue pixels from its spatial causal neighborhood is computed and subtracted from the red pixel value. We apply this local mean subtraction to every pixel in the dataset. Denote N_s and N_t as the number of pixels from previous spectral bands at the current time instant (yellow pixels in Fig. 1) and the number of pixels from the same spectral bands from previous time frames (green pixels in Fig. 1), respectively. Then we define the aforementioned causal context as the feature vector $\mathbf{X}$ (shown in Eq. 7) for our CLMS learning algorithm. In order to simplify the expression, we replace $x^{t,j}_{m,n}$ defined in Eq. (3) with $x^{t,j}$. The causal context is constructed as $C^t_j = \left[x^{t,j-1}, x^{t,j-2} \cdots x^{t,j-N_s}, x^{t-1,j}, x^{t-2,j} \cdots x^{t-N_t,j}\right]$. Thus, for all pixels in one specific spectral band at any time frame, the number of their context pixels are always the same according to $N_s + N_t$ determined by the user.

For each band, we initialize the $\mathbf{w}_{t,j}$ to be all zeros and carry on the CLMS learning algorithm to all the pixels in a raster-scan order when it reaches the end of this band. Once the algorithm stops, the weight vector will be initialized again for the next band until the end of the entire HSI dataset. Algorithm 1 shows the details of the whole procedure of this CLMS predictor. We emphasize that all the residuals will be mapped to integers before sending to entropy encoder in a reversible manner.

Algorithm 1 CLMS Predictor

Initialize:
1) T (# of time frames)
2) B (# of spectral bands for each time frame)
3) $\mu = 0.3$ and $\sigma = 50$
4) Local mean subtracted data X
for t = 1:T **do**
  for b = 1:B **do**
    initialize: $\mathbf{w} = \mathbf{0}$
    for each pixel in this band **do**
      Output residual $x - \widehat{x}$ using Eq. (4).
      Updating $\mathbf{w}$ using Eq. (7).
    end for
  end for
end for

3.3 Entropy Coding

Both the Golomb-Rice code (GRC) and the arithmetic code (AC) have been widely used to encode the prediction residuals in hyperspectral image compression methods. Although AC may produce slightly better coding efficiency, GRC can yield comparable performance with accurate data modelling. Also, GRC is known for its simplicity and minimal memory capacity requirement while AC very often requires much more computations. Given the limited onboard computing power in most remote sensing applications, in this work, we employ GRC on prediction residuals of time-lapse HSI data to generate the final bit sequence for the data transmission and storage. The readers are referred to [10] for details of entropy coding.

4. Experimental Results

We conducted our experiment on three 4D time-lapse HSI test datasets, *Levada*, *Gualtar* and *Nogueiro*. Basic information of these three datasets are listed in the Table 1. Detailed information of these datasets can be found in [22]. Each single HSI has the same spatial size, 1024×1344, with 33 spectral bands. Both *Gualtar* and *Nogueiro* have nine time stamps while *Levada* has seven. Note that the original data for these datasets has been mapped into $[0, 1]$ and stored using "double" data format (64 bits). So we recover the data to its original precision by applying a linear remapping. Since our algorithm is a learning-based method which predicts the value regardless of the data scale, we believe these post-processed datasets are suitable for evaluating the lossless compression algorithm. While the size of a single dataset we tested is not very large, ranging from 454.78 MB (for 7 frames) to 584.71 MB (for 9 frames), the data can easily grow to a huge size with increased number of time frames and higher spatial and spectral resolutions. Besides, HSI data streaming can become a challenging task, where efficient data compression is essential.

There are two parameters to be determined in CLMS before the prediction: σ in kernel function (shown in Eq. 7) and initial learning rate μ. Small σ value will lead to relatively large actual learning rate and vice versa. We experimented with different parameters to achieve the best results. As a result, we fix μ and σ in Eq. (7) at 0.3 and 50 in our test.

Table 1: Datasets Used.

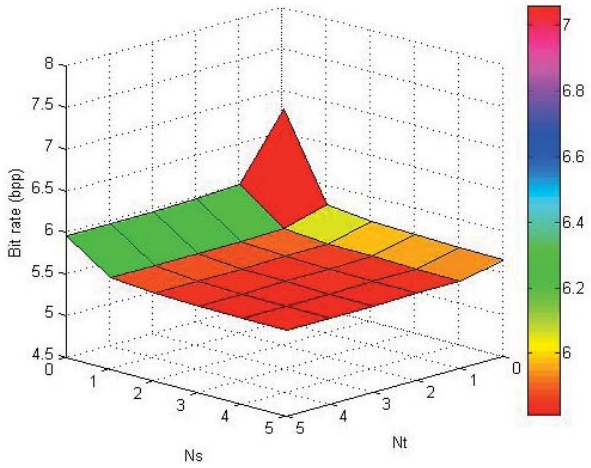
Dataset	Size	# of time frames	Precision(bits)
Levada	$1024 \times 1344 \times 33$	7	12
Gualtar	$1024 \times 1344 \times 33$	9	12
Noguerio	$1024 \times 1344 \times 33$	9	12

As discussed in Section 2, we applied our algorithm using different combinations of N_s and N_t causal pixels from spectral and temporal bands. Given the limited space, we only provide compression bit rate (bits/pixel) results for *Levada* in Table 2. As we can see, local mean subtraction without spectral and temporal decorrelation ($N_s = 0$ and $N_t = 0$), was effective in removing great deal of signal correlation as the bit rate drops from 12 to 7.0604 bits/pixel. This also indicates that it is not necessary to explicitly decorrelate spatially to achieve a competitive compression performance for time-lapse HSI data. More importantly, our algorithm can further compress 4D time-lapse HSI data by using spectral and temporal correlation. The bit rate has been reduced by approximately 1.2 bits by just adding one previous spectral band and the same spectral band from the previous time stamp in the context. Generally, the bit rate decreases, i.e., yielding less bits after compression, with more bands selected to form the learning context. Furthermore, if we fix either N_s or N_t and increase only N_t or N_s accordingly, the compression bit rate will drop as well. However, this performance improvement gradually becomes marginal as N_s or N_t increases. Fig. 3 shows three surface plots of bit rates on these test datasets, showing how the bit rate changes with different combinations of N_s and N_t.

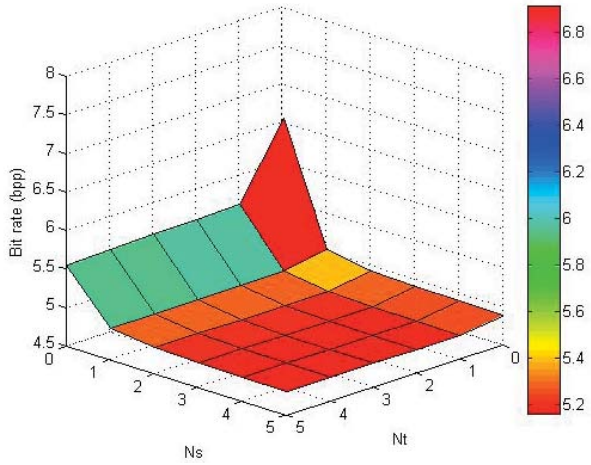
Table 2: Bit rates (bits/pixels) on "Levada".

N_s	$N_t = 0$	$N_t = 1$	$N_t = 2$	$N_t = 3$	$N_t = 4$	$N_t = 5$
0	7.0604	6.2858	6.2686	6.2431	6.2400	6.2382
1	6.0476	5.8985	5.8898	5.8813	5.8795	5.8787
2	5.9807	5.8570	5.8497	5.8414	5.8395	5.8388
3	5.9592	5.8433	5.8360	5.8279	5.8260	5.8252
4	5.9487	5.8359	5.8289	5.8211	5.8191	5.8184
5	5.9410	5.8308	5.8239	5.8162	5.8146	5.8138

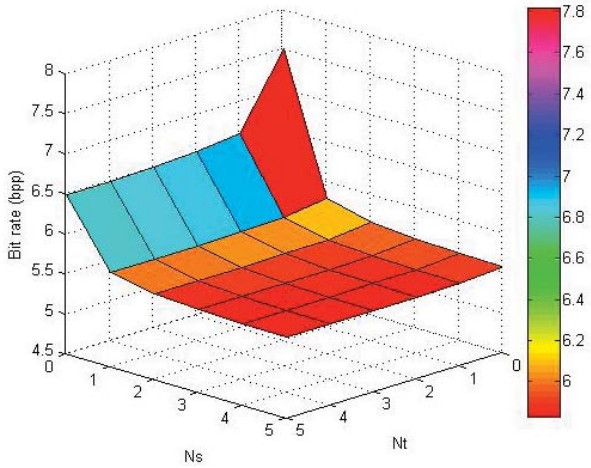
To further illustrate how bit rates respond to different combinations of N_s and N_t, we arbitrarily fix $N_s = 2$ and $N_t = 2$ separately and adjust another variable N_s or N_t from 0 to 5. The results of this experiment on three datasets have been plotted in Fig. 4. First, it is obvious that more bands used in the prediction will contribute to better



(a) *Levada*.



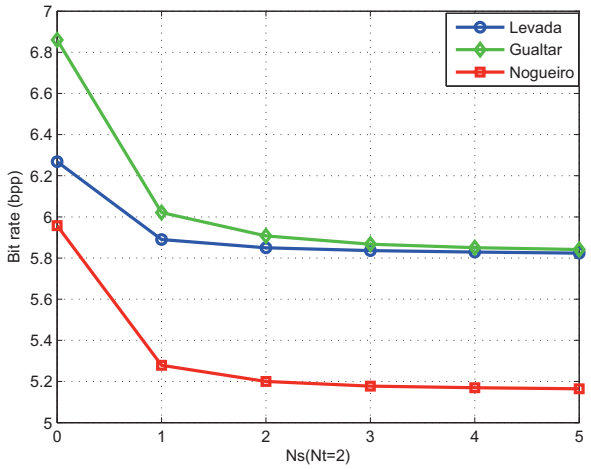
(b) *Nogueiro*.



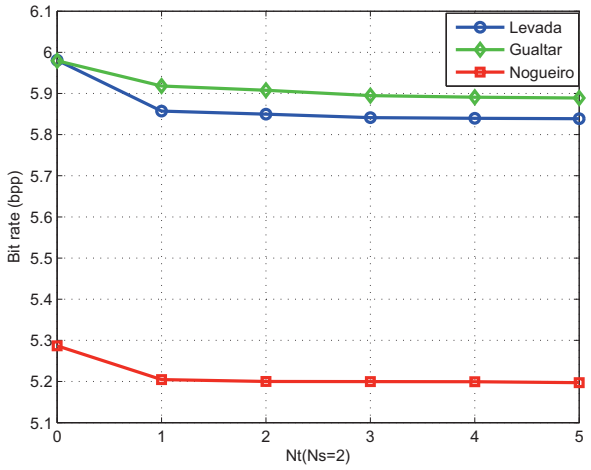
(c) *Gualtar*.

Fig. 3: Bit rate surface plots on three datasets.

prediction in terms of smaller bit rates. But this performance improvement decays really fast as what have we observed



(a) Bit rate vs. N_s.



(b) Bit rate vs. N_t.

Fig. 4: Bit rate changes with N_s and N_t.

in the Table 2. Moreover, we can find that the performance improvement caused by the spectral decorrelation is more noticeable than temporal decorrelation. We believe this is because for our test datasets spectral correlation is much stronger than temporal correlation especially each HSI in these 4D datasets is captured at approximately one hour interval which leads to less strong correlation temporally. If this imaging capture time interval is reduced, then adjacent HSIs will likely to share more similarities in statistics because of less illuminance condition change.

Overall, it is possible to increase N_s and N_t to achieve higher compression ratio. On the other hand, prediction using only one previous spectral band and/or the same spectral band but from last time instant will also yield good compression performance at a very low computational cost. This feature also provides great flexibility in compression performance and complexity.

5. Conclusions and Future Work

We have proposed a new predictive lossless compression algorithm for 4D time-lapse hyperspectral image data using a low-complexity Correntropy-induced LMS learning. The Correntropy based cost function seemed to be effective in capturing the non-linearity and non-Gaussian conditions of the prediction residuals of time-lapse HSI data. Experimental results have demonstrated the outstanding capability of this proposed algorithm to compress 4D time-lapse HSI data through spectral and temporal decorrelation.

Second, an information theoretic analysis based on conditional entropy has been made to provide a framework to guide and evaluate the actual compression. Increasing the number of previous bands involved in the prediction will absolutely yield better compression performance as long as they are correlated statistically with the current HSI band. We have seen the increasingly improved compression efficiency from the experimental results.

We will investigate how to fully utilize this proposed algorithm and analytic framework to handle HSI data streaming, which is more challenging but also in better need for compression. Additionally, ROI lossless compression of HSI has begun to gain attention from researchers. Recently, some work has been done to handle ROIs in HSI data. As long as ROIs can be identified accurately, we can compress the HSI data without any information loss at a high compression ratio which is comparable to lossy compression. Since our algorithm mainly utilizes spectral and temporal correlation in the prediction, it can be extended to the compression of ROIs in 4D time-lapse HSI data with minimal modifications.

References

[1] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS," *IEEE Trans. Image Process.*, vol. 9, no. 8, pp. 1309–1324, Aug. 2000.

[2] X. Wu and N. Memon, "Context-based lossless interband compression-extending CALIC," *IEEE Trans. Image Process.*, vol. 9, no. 6, pp. 994–1001, Jun 2000.

[3] E. Magli, G. Olmo, and E. Quacchio, "Optimized onboard lossless and near-lossless compression of hyperspectral data using CALIC," *IEEE Trans. Geosci. Remote Sens.*, vol. 1, no. 1, pp. 21–25, Jan 2004.

[4] X. Wu and N. Memon, "Context-based, adaptive, lossless image coding," *IEEE Trans. Commun*, vol. 45, no. 4, pp. 437–444, Apr 1997.

[5] J. Mielikainen, "Lossless compression of hyperspectral images using lookup tables," *IEEE Signal Process. Lett.*, vol. 13, no. 3, pp. 157–160, March 2006.

[6] F. Rizzo, B. Carpentieri, G. Motta, and J. A. Storer, "Low-complexity lossless compression of hyperspectral imagery via linear prediction," *IEEE Signal Process. Lett.*, vol. 12, no. 2, pp. 138–141, Feb. 2005.

[7] H. Wang, S. D. Babacan, and K. Sayood, "Lossless hyperspectral-image compression using context-based conditional average," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 4187–4193, Dec. 2007.

[8] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 243–250, Jun 1996.

[9] W. A. Pearlman, A. Islam, N. Nagaraj, and A. Said, "Efficient, low-complexity image coding with a set-partitioning embedded block coder," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 11, pp. 1219–1235, Nov 2004.

[10] M. Klimesh, "Low-complexity lossless compression of hyperspectral imagery via adaptive filtering," in *The Interplanetary Network Progress Report*, Jet Propulsion Laboratory, Pasadena, California, Nov. 2005, pp. 1–10.
[11] "Lossless multispectral & hyperspectral image compression CCSDS 123.0-B-1, Blue Book, May 2012," http://public.ccsds.org/publications/archive/123x0b1ec1.pdf, 2015 (accessed December 10, 2015).
[12] D. H. Foster, K. Amano, and S. M. Nascimento, "Time-lapse ratios of cone excitations in natural scenes," *Vision Research*, 2016.
[13] M. A. Mamun, X. Jia, and M. Ryan, "Sequential multispectral images compression for efficient lossless data transmission," in *2010 Second IITA Intl. Conf. on Geosci. Remote Sens.*, vol. 2, Aug 2010, pp. 615–618.
[14] J. Munoz-Gomez, J. Bartrina-Rapesta, I. Blanes, L. Jimenez-Rodriguez, F. Auli-Llinas, and J. Serra-Sagrista, "4D remote sensing image coding with JPEG2000," *Proc. SPIE*, vol. 7810, pp. 1–9, 2010.
[15] J. Mielikainen and B. Huang, "Lossless compression of hyperspectral images using clustered linear prediction with adaptive prediction length," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 6, pp. 1118–1121, Nov. 2012.
[16] E. Magli, "Multiband lossless compression of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 4, pp. 1168–1178, April 2009.
[17] S. Golomb, "Run-length encodings (corresp.)," *IEEE Trans. Inf. Theory*, vol. 12, no. 3, pp. 399–401, Jul. 1966.
[18] A. Moffat, R. Neal, and I. H. Witten, "Arithmetic coding revisited," in *Proc. Data Compression Conf.*, Mar 1995, pp. 202–211.
[19] H. Shen, W. D. Pan, and Y. Wang, "A novel method for lossless compression of arbitrarily shaped regions of interest in hyperspectral imagery," in *Proc. 2015 IEEE SoutheastCon*, April 2015.
[20] A. Singh and J. C. Principe, "Using correntropy as a cost function in linear adaptive filters," in *Intl. Joint Conf. on Neural Netw.*, June 2009, pp. 2950–2955.
[21] W. Liu, P. P. Pokharel, and J. C. Principe, "Correntropy: Properties and applications in non-Gaussian signal processing," *IEEE Trans. Signal Process*, vol. 55, no. 11, pp. 5286–5298, Nov 2007.
[22] "Time-lapse hyperspectral radiance images of natural scenes 2015," http://personalpages.manchester.ac.uk/staff/david.foster/Time-Lapse_HSIs/Time-Lapse_HSIs_2015.html, 2015 (accessed March 1, 2015).

String Vector based KNN for Index Optimization

Taeho Jo
Department of Computer and Information Communication Engineering, Hongik University, Sejong, South Korea

Abstract— *In this research, we propose the string vector based KNN as the approach to the index optimization. The task may be viewed into an instance of word classification and the problems in encoding words or texts into numerical vectors were solved by encoding texts into string vectors in the previous works on text mining tasks. Influence by the previous works, we encode words into string vectors, as well as texts, define the semantic operations on string vectors, and apply the modified version of the KNN to the index optimization which is mapped into a classification task. As the benefits from this research, we may expect the better performance and more compact representations than encoding texts or words into numerical vectors. Therefore, the goal of this research is to develop the index optimization system with the benefits.*

Keywords: Keyword Extraction, String Vector, K Nearest Neighbor

1. Introduction

Index optimization refers to the process of optimizing a list of words which indicates texts for maximizing the information retrieval efficiency and performance. We need to expand the important words by adding their semantically similar words for improving the performance and remove unimportant words for improving the efficiency. In this research, we view the index optimization into the classification task where each word is classified into important words as targets of expansion, neutral words as ones of inclusion, and unimportant words as ones of removal. We prepare the sample words which are labeled with one of the three classes and construct the classification capacity by learning them. In this research, we assume that the supervised learning algorithms are used as the approach to the task, even if other types of approaches are available.

We mention the problems with which this research needs to tackle with. In encoding texts or words into numerical vectors for using the traditional classifiers, many features are required for keeping the robust classifications[1]. Each numerical vector which represents a word or a text has usually zero values dominantly as its elements; the discriminations among numerical vectors get very weak [5][9]. Although we proposed previously that texts or words should be encoded into tables as alternative structured forms to numerical vectors, it is very expensive to carry out the computation on them[5][9]. Therefore, this research challenges against the above problems by encoding words into string vectors.

Let us mention what is proposed in this research, as its ideas. In this research, we encode the words into string vectors each of which consists of text identifiers as its elements. We define the similarity measure between string vectors; it corresponds to the cosine similarity between numerical vectors. We modify the KNN into the version where a string vector is given as the input data, and apply it to the classification task into which we interpret the index optimization. The scope of this research is restricted to the classification of words into one of the three categories; the process of expanding words semantically is set out of this research.

Let us mention the benefits which are expected from this research. From this research, it is expected to represent words with more compactness and efficiency than to do them into numerical vectors. The improve discriminations among string vectors are expected from this research by avoiding almost completely the sparse distributions. The improved performance is also expected by solving the problems from encoding words into numerical vectors. Therefore, the goal of this research is to implement the index optimization module for information retrieval system with the benefits.

This article is organized into the four sections. In Section 2, we survey the relevant previous works. In Section 3, we describe in detail what we propose in this research. In Section 4, we mention the remaining tasks for doing the further research.

2. Previous Works

Let us survey the previous cases of encoding texts into structured forms for using the machine learning algorithms to text mining tasks. The three main problems, huge dimensionality, sparse distribution, and poor transparency, have existed inherently in encoding them into numerical vectors. In previous works, various schemes of preprocessing texts have been proposed, in order to solve the problems. In this survey, we focus on the process of encoding texts into alternative structured forms to numerical vectors. In other words, this section is intended to explore previous works on solutions to the problems.

Let us mention the popularity of encoding texts into numerical vectors, and the proposal and the application of string kernels as the solution to the above problems. In 2002,

Sebastiani presented the numerical vectors are the standard representations of texts in applying the machine learning algorithms to the text classifications [1]. In 2002, Lodhi et al. proposed the string kernel as a kernel function of raw texts in using the SVM (Support Vector Machine) to the text classification [2]. In 2004, Lesile et al. used the version of SVM which proposed by Lodhi et al. to the protein classification [3]. In 2004, Kate and Mooney used also the SVM version for classifying sentences by their meanings [4].

It was proposed that texts are encoded into tables instead of numerical vectors, as the solutions to the above problems. In 2008, Jo and Cho proposed the table matching algorithm as the approach to text classification [5]. In 2008, Jo applied also his proposed approach to the text clustering, as well as the text categorization [9]. In 2011, Jo described as the technique of automatic text classification in his patent document [7]. In 2015, Jo improved the table matching algorithm into its more stable version [8].

Previously, it was proposed that texts should be encoded into string vectors as other structured forms. In 2008, Jo modified the k means algorithm into the version which processes string vectors as the approach to the text clustering[9]. In 2010, Jo modified the two supervised learning algorithms, the KNN and the SVM, into the version as the improved approaches to the text classification [10]. In 2010, Jo proposed the unsupervised neural networks, called Neural Text Self Organizer, which receives the string vector as its input data [11]. In 2010, Jo applied the supervised neural networks, called Neural Text Categorizer, which gets a string vector as its input, as the approach to the text classification [12].

The above previous works proposed the string kernel as the kernel function of raw texts in the SVM, and tables and string vectors as representations of texts, in order to solve the problems. Because the string kernel takes very much computation time for computing their values, it was used for processing short strings or sentences rather than texts. In the previous works on encoding texts into tables, only table matching algorithm was proposed; there is no attempt to modify the machine algorithms into their table based version. In the previous works on encoding texts into string vectors, only frequency was considered for defining features of string vectors. In this research, based on [10], we consider the grammatical and posting relations between words and texts as well as the frequencies for defining the features of string vectors, and encode words into string vectors in this research.

3. Proposed Approach

This section is concerned with encoding words into string vectors, modifying the KNN (K Nearest Neighbor) into the string vector based version and applying it to the keyword extraction, and consists of the four sections. In Section 3.1, we deal with the process of encoding words into string vectors. In Section 3.2, we describe formally the similarity matrix and the semantic operation on string vectors. In Section 3.3, we do the string vector based KNN version as the approach to the keyword extraction. In Section 3.4, we focus on the process of applying the KNN to the given task with viewing it into a classification task.

3.1 Word Encoding

This section is concerned with the process of encoding words into string vectors. The three steps are involved in doing so, as illustrated in Figure 1. A single word is given as the input, and a string vector which consists of text identifiers is generated as the output. We need to prepare a corpus which is a collection of texts for encoding words. Therefore, in this section, we will describe each step of encoding the words.

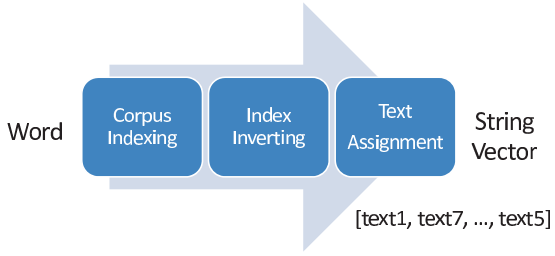


Fig. 1: Overall Process of Word Encoding

The first step of encoding words into string vectors is to index the corpus into a list of words. The texts in the corpus are concatenated into a single long string and it is tokenized into a list of tokens. Each token is transformed into its root form, using stemming rules. Among them, the stop words which are grammatical words such as propositions, conjunctions, and pronouns, irrelevant to text contents are removed for more efficiency. From the step, verbs, nouns, and adjectives are usually generated as the output.

The inverted list where each word is linked to the list of texts which include it is illustrated in Figure 2. A list of words is generated from a text collection by indexing each text. For each word, by retrieving texts which include it, the inverted list is constructed. A text and a word are associated with each other by a weight value as the relationship between them. The links of each word with a list of texts is opposite to those of each text with a list of words becomes the reason of call the list which is presented in Figure 2, inverted list.

Each word is represented into a string vector based on the inverted index which is shown in Figure 3. In this research, we define the features which are relations between texts and words as follows:

- Text identifier which has its highest frequency among the text collection
- Text identifier which has its highest TF-IDF weight among the text collection

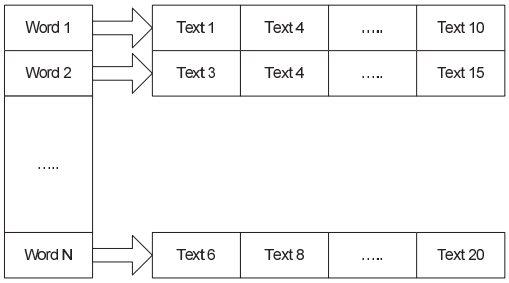


Fig. 2: The Inverted Index

- Text identifier which has its second highest frequency among the text collection
- Text identifier which has its second highest TF-IDF weight among the text collection
- Text identifier which has its highest frequency in its first paragraph among text collection
- Text identifier which has its highest frequency in its last paragraph among text collection
- Text identifier which has its highest TF-IDF weight in its first paragraph among text collection
- Text identifier which has its highest TF-IDF weight in its last paragraph among text collection

We assume that each word is linked with texts including their own information: its frequencies and its weights in the linked texts and their first and last paragraphs. From the inverted index, we assign the corresponding values which are given as text identifiers to each feature. Therefore, the word is encoded into an eight dimensional string vector which consists of eight strings which indicate text identifiers.

Let us consider the differences between the word encoding and the text encoding. Elements of each string vector which represents a word are text identifiers, whereas those of one which represents a text are word. The process of encoding texts involves the link of each text to a list of words, where as that of doing words does the link of each word to a list of texts. For performing semantic similarity between string vectors, in text processing, the word similarity matrix is used as the basis, while in word processing, the text similarity matrix is used. The relations between words and texts are defined as features of strings in encoding texts and words.

3.2 String Vectors

This section is concerned with the operation on string vectors and the basis for carrying out it. It consists of two subsections and assumes that a corpus is required for performing the operation. In Section 3.2.1, we describe the process of constructing the similarity matrix from a corpus. In Section 3.2.2, we define the string vector formally and characterize the operation mathematically. Therefore, this section is intended to describe the similarity matrix and the operation on string vectors.

3.2.1 Similarity Matrix

This subsection is concerned with the similarity matrix as the basis for performing the semantic operation on string vectors. Each row and column of the similarity matrix corresponds to a text in the corpus. The similarities of all possible pairs of texts are given as normalized values between zero and one. The similarity matrix which we construct from the corpus is the $N \times N$ square matrix with symmetry elements and 1's diagonal elements. In this subsection, we will describe formally the definition and characterization of the similarity matrix.

Each entry of the similarity matrix indicates a similarity between two corresponding texts. The two documents,d_i and d_j, are indexed into two sets of words, D_i and D_j. The similarity between the two texts is computed by equation (1),

$$sim(d_i, d_j) = \frac{2|D_i \cap D_j|}{|D_i| + |D_j|} \tag{1}$$

where $|D_i|$ is the cardinality of the set, D_i. The similarity is always given as a normalized value between zero and one; if two documents are exactly same to each other, the similarity becomes 1.0 as follows:

$$sim(d_i, d_j) = \frac{2|D_i \cap D_i|}{|D_i| + |D_i|} = 1.0$$

and if two documents have no shared words, $D_i \cap D_j = \emptyset$ the similarity becomes 0.0 as follows:

$$sim(d_i, d_j) = \frac{2|D_i \cap D_j|}{|D_i| + |D_j|} = 0.0$$

The more advanced schemes of computing the similarity will be considered in next research.

From the text collection, we build $N \times N$ square matrix as follows:

$$S = \begin{pmatrix} s_{11} & s_{12} & \dots & s_{1d} \\ s_{21} & s_{22} & \dots & s_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ s_{d1} & s_{d2} & \dots & s_{dd} \end{pmatrix}.$$

N individual texts which are contained in the collection correspond to the rows and columns of the matrix. The entry, s_{ij} is computed by equation (1) as follows:

$$s_{ij} = sim(d_i, d_j)$$

The overestimation or underestimation by text lengths are prevented by the denominator in equation (1). To the number of texts, N, it costs quadratic complexity, $O(N^2)$, to build the above matrix.

Let us characterize the above similarity matrix, mathematically. Because each column and row corresponds to its same text in the diagonal positions of the matrix, the diagonal elements are always given 1.0 by equation (1).

In the off-diagonal positions of the matrix, the values are always given as normalized ones between zero and one, because of $0 \leq 2|D_i \cap D_i| \leq |D_i| + |D_j|$ from equation (1). It is proved that the similarity matrix is symmetry, as follows:

$$s_{ij} = sim(d_i, d_j) = \frac{2|D_i \cap D_j|}{|D_i| + |D_j|} = \frac{2|D_j \cap D_i|}{|D_j| + |D_i|}$$
$$= sim(d_j, d_i) = s_{ji}$$

Therefore, the matrix is characterized as the symmetry matrix which consists of the normalized values between zero and one.

The similarity matrix may be constructed automatically from a corpus. The N texts which are contained in the corpus are given as the input and each of them is indexed into a list of words. All possible pairs of texts are generated and the similarities among them are computed by equation (1). By computing them, we construct the square matrix which consists of the similarities. Once making the similarity matrix, it will be used continually as the basis for performing the operation on string vectors.

3.2.2 String Vector and Semantic Similarity

This section is concerned with the string vectors and the operation on them. A string vector consists of strings as its elements, instead of numerical values. The operation on string vectors which we define in this subsection corresponds to the cosine similarity between numerical vectors. Afterward, we characterize the operation mathematically. Therefore, in this section, we define formally the semantic similarity as the semantic operation on string vectors.

The string vector is defined as a finite ordered set of strings as follows:

$$\mathbf{str} = [str_1, str_2, ..., str_d]$$

An element in the vector, str_i indicates a text identifier which corresponds to its attribute. The number of elements of the string vector, $\mathbf{str}$ is called its dimension. In order to perform the operation on string vectors, we need to define the similarity matrix which was described in Section 3.2.1, in advance. Therefore, a string vector consists of strings, while a numerical vector does of numerical values.

We need to define the semantic operation which is called 'semantic similarity' in this research, on string vectors; it corresponds to the cosine similarity on numerical vectors. We note the two string vectors as follows:

$$\mathbf{str}_1 = [str_{11}, str_{12}, ..., str_{1d}]$$
$$\mathbf{str}_2 = [str_{21}, str_{22}, ..., str_{2d}]$$

where each element, d_{1i} and d_{2i} indicates a text identifier. The operation is defined as equation (3.2.2) as follows:

$$sim(\mathbf{str}_1, \mathbf{str}_2) = \frac{1}{d}\sum_{i=1}^{d} sim(d_{1i}, d_{2i}) \qquad (2)$$

The similarity matrix was constructed by the scheme which is described in Section 3.2.1, and the $sim(d_{1i}, d_{2i})$ is computed by looking up it in the similarity matrix. Instead of building the similarity matrix, we may compute the similarity, interactively.

The semantic similarity measure between string vectors may be characterized mathematically. The commutative law applies as follows:

$$sim(\mathbf{str}_1, \mathbf{str}_2) = \frac{1}{d}\sum_{i=1}^{d} sim(d_{1i}, d_{2i})$$

$$= \frac{1}{d}\sum_{i=1}^{k} sim(d_{2i}, d_{1i}) = sim(\mathbf{str}_2, \mathbf{str}_1)$$

If the two string vectors are exactly same, its similarity becomes 1.0 as follows:

$$\text{if } \mathbf{str}_1 = \mathbf{str}_2 \text{with } \forall_i sim(d_{1i}, d_{2i}) = 1.0$$

$$\text{then } sim(\mathbf{str}_1, \mathbf{str}_2) = \frac{1}{d}\sum_{i=1}^{d} sim(d_{1i}, d_{2i}) = \frac{d}{d} = 1.0$$

However, note that the transitive rule does not apply as follows:

$$\text{if } sim(\mathbf{str}_1, \mathbf{str}_2) = 0.0 \text{ and } sim(\mathbf{str}_2, \mathbf{str}_3) = 0.0$$

$$\text{then, not always } sim(\mathbf{str}_1, \mathbf{str}_3) = 0.0$$

We need to define the more advanced semantic operations on string vectors for modifying other machine learning algorithms. We define the update rules of weights vectors which are given as string vectors for modifying the neural networks into their string vector based versions. We develop the operations which correspond to computing mean vectors over numerical vectors, for modifying the k means algorithms. We consider the scheme of selecting representative vector among string vectors for modifying the k medoid algorithms so. We will cover the modification of other machine learning algorithms in subsequent researches.

3.3 Proposed Version of KNN

This section is concerned with the proposed KNN version as the approach to the text categorization. Raw texts are encoded into string vectors by the process which was described in Section 3.1. In this section, we attempt to the traditional KNN into the version where a string vector is given as the input data. The version is intended to improve the classification performance by avoiding problems from encoding texts into numerical vectors. Therefore, in this section, we describe the proposed KNN version in detail, together with the traditional version.

The traditional KNN version is illustrated in Figure 3. The sample words which are labeled with the positive class or the negative class are encoded into numerical vectors. The similarities of the numerical vector which represents a novice

word with those representing sample words are computed using the Euclidean distance or the cosine similarity. The k most similar sample words are selected as the k nearest neighbors and the label of the novice entity is decided by voting their labels. However, note that the traditional KNN version is very fragile in computing the similarity between very sparse numerical vectors.

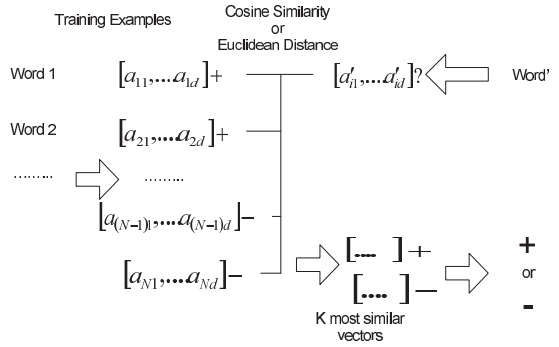


Fig. 3: The Traditional Version of KNN

Separately from the traditional one, we illustrate the classification process by the proposed version in Figure 4. The sample texts labeled with the positive or negative class are encoded into string vectors by the process described in Section 3.1. The similarity between two string vectors is computed by the scheme which was described in Section 3.2.2. Identically to the traditional version, in the proposed version, the k most similarity samples are selected, and the label of the novice one is decided by voting ones of sample entities. Because the sparse distribution in each string vector is never available inherently, the poor discriminations by sparse distribution are certainly overcome in this research.

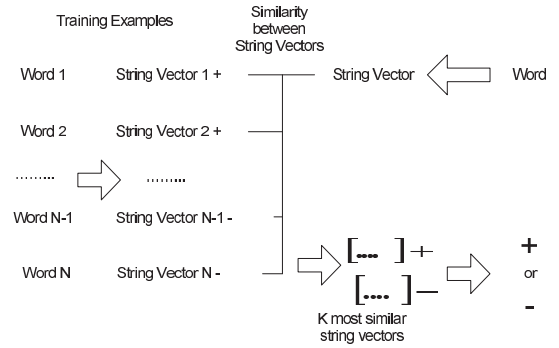


Fig. 4: The Proposed Version of KNN

We may derive some variants from the proposed KNN version. We may assign different weights to selected neighbors instead of identical ones: the highest weights to the first nearest neighbor and the lowest weight to the last one. Instead of a fixed number of nearest neighbors, we select any number of training examples within a hyper-sphere whose center is the given novice example as neighbors. The categorical scores are computed proportionally to similarities with training examples, instead of selecting nearest neighbors. We may also consider the variants where more than two variants are combined with each other.

Because string vectors are characterized more symbolically than numerical vectors, it is easy to trace results from classifying items in the proposed version. It is assumed that a novice item is classified by voting the labels of its nearest neighbors. The similarity between string vectors is computed by the scheme which is described in Section 3.2.2. We may extract the similarities of individual elements of the novice string vector with those of nearest neighbors labeled with the classified category. Therefore, the semantic similarities play role of the evidence for presenting the reasons of classifying the novice one so.

3.4 The Application to Index Optimization

This section is concerned with the scheme of applying the proposed KNN version which was described in Section to the index optimization task. Before doing so, we need to transform the task into one where machine learning algorithms are applicable as the flexible and adaptive models. We prepare the words which are labeled with 'expansion', 'inclusion' or 'removal' as the sample data. The words are encoded into tables by the scheme which was described in Section . Therefore, in this section, we describe the process of extracting words which belong to the two categories, 'expansion' and 'inclusion', from texts automatically using the proposed KNN with the view of the index optimization into a classification task.

In this research, the index optimization is viewed into a classification task, as shown in Figure 5. A text is given as the input, and a list of words is extracted by indexing the text. Each word is classified by the classifier into one of the three categories: 'expansion', 'inclusion', or, 'removal'. In the task, the text is mapped into words which are classified with 'expansion' or 'inclusion'. The similar words to one labeled with 'expansion' will be added from external sources.

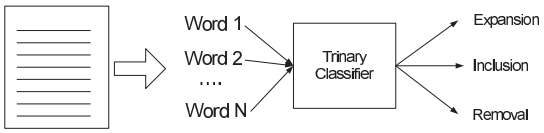


Fig. 5: View of Index Optimization into Classification Task

We need to prepare sample words which are labeled with one of the three categories, before classifying a novice one or ones. A text collection is segmented into sub-collections of content based similar words which are called domains, manually or automatically. We prepare sample words which are labeled manually, domain by domain. To each domain,

we assign and train a classifier with the words in the corresponding sub-collection. When a text is given as the input, the classifier which corresponds to the most similar domain is selected among them.

Let us consider the process where an section is given as the input and a list of essential words is extracted as the output. We nominate the classifier which corresponds to the subgroup which is closest to the given section with respect to its content. A list of words is extracted by indexing the section, and each word is encoded into their structured forms. The words are classified by the nominated classifier into one of the three categories, and we select ones which are labeled with 'expansion' or 'reservation' as the optimized index. The addition of external words which are semantically similar as ones labeled with 'expansion' is set as the subsequent task.

Even if the index optimization is viewed into an instance of word categorization, it needs to be distinguished from the topic based word categorization. The word categorization is given as a single multiple classification or multiple binary classifications, whereas the index optimization is done as a single triary classification or three binary classification tasks. In the word categorization, each word is classified semantically into one or some of the predefined topics, whereas in the index optimization, it is classified one of the three actions. In the word categorization, each word is classified by its meaning, whereas in the index optimization, it is classified by its importance to the given text. In the word categorization, when the given task is decomposed into binary classification tasks, a classifier is assigned to each topic, whereas, in the index optimization, a classifier is done to each domain.

4. Conclusion

Let us mention the remaining tasks for doing the further research. The proposed approach should be validated and specialized in the specific domains: medicine, engineering and economics. Other features such as grammatical and posting features may be considered for encoding words into string vectors as well as text identifiers. Other machine learning algorithms as well as the KNN may be modified into their string vector based versions. By adopting the proposed version of the KNN, we may implement the index optimization system as a real program.

5. Acknowledgement

This work was supported by 2016 Hongik University Research Fund.

References

[1] F. Sebastiani, "Machine Learning in Automated Text Categorization", pp1-47, ACM Computing Survey, Vol 34, No 1, 2002.

[2] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, "Text Classification with String Kernels", pp419-444, Journal of Machine Learning Research, Vol 2, No 2, 2002.

[3] C. S. Leslie, E. Eskin, A. Cohen, J. Weston, and W. S. Noble, "Mismatch String Kernels for Discriminative Protein Classification", pp467-476, Bioinformatics, Vol 20, No 4, 2004.

[4] R. J. Kate and R. J. Mooney, "Using String Kernels for Learning Semantic Parsers", pp913-920, Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, 2006.

[5] T. Jo and D. Cho, "Index based Approach for Text Categorization",International Journal of Mathematics and Computers in Simulation, Vol 2, No 1, 2008.

[6] T. Jo, "Single Pass Algorithm for Text Clustering by Encoding Documents into Tables", pp1749-1757, Journal of Korea Multimedia Society, Vol 11, No 12, 2008.

[7] T. Jo, "Device and Method for Categorizing Electronic Document Automatically", Patent Document, 10-2009-0041272, 10-1071495, 2011.

[8] T. Jo, "Normalized Table Matching Algorithm as Approach to Text Categorization", pp839-849, Soft Computing, Vol 19, No 4, 2015.

[9] T. Jo, "Inverted Index based Modified Version of K-Means Algorithm for Text Clustering", pp67-76, Journal of Information Processing Systems, Vol 4, No 2, 2008.

[10] T. Jo, "Representationof Texts into String Vectors for Text Categorization", pp110-127, Journal of Computing Science and Engineering, Vol 4, No 2, 2010.

[11] T. Jo, "NTSO (Neural Text Self Organizer): A New Neural Network for Text Clustering", pp31-43, Journal of Network Technology, Vol 1, No 1, 2010.

[12] T. Jo, "NTC (Neural Text Categorizer): Neural Network for Text Categorization", pp83-96, International Journal of Information Studies, Vol 2, No 2, 2010.

Data Quality Evaluation previous to Big Data Analytics

M. Mejia-Lavalle*, J. Perez-Ortega, A. Magadan-Salazar, E. Perez-Luna,
G. Reyes Salgado and D. Mujica Vargas
Departamento de Ciencias Computacionales, Centro Nacional de Investigación y Desarrollo Tecnológico,
Int. Internado Palmira S/N, 62490, Cuernavaca, Morelos, México
{mlavalle*, magadan, eduardo.perez, greyes, dantemv@cenidet.edu.mx}, jpo_cenidet@yahoo.com.mx

Abstract - *Big Data technology is a computing area with a great growth. Today it is common that we hear about databases with huge volumes of information and also we hear about Data Mining and Business Intelligence projects related with these huge databases. However, in general, little attention has been given to the quality of the data. Here we propose and present innovative metrics and schema designed to perform a basic task related to the Data Quality issue, this is, the diagnostic. The preliminary results that we obtained when we apply our approaches to Big Data encourage us to continue this research.*

Keywords: Data quality, Big data, Data cleansing.

1 Introduction

Big Data is one of the computing areas more active for research, especially since the devices for storing large volumes of data have become very efficient and inexpensive. Not many years ago we found that a database close to one Gigabyte (10^6 bytes) was considered very large. Currently it is common that we can hear about databases that store, for example, Terabytes (10^{12} bytes) and Yottabytes (10^{24} bytes) of information, and the trend is increasing.

But this high growth has been generally accompanied with little attention to the quality of data that these databases contain, being now more than ever true the old phrase at the beginning of the computer days that said "garbage in, garbage out". And while there is abundant literature on the Big Data subject, there are few concrete proposals for schemas that directly address the issue of data quality for very large databases.

Given this problem, in this paper we propose and present some metrics and a schema designed to perform one of the key tasks related to data quality, i.e. diagnostic, which involves measuring the level of quality in a database. In other words, our approach realizes a data quality evaluation, previous to initiate a Big Data analytics phase and it has been tested in a preliminary project and the results that we obtained have been satisfactory, as described in this article.

For developing these ideas, first we will address the issue of Big Data and data quality in general, and then we will present the ideas that we propose, describing new measures designed by us and used to establish an objective diagnosis of the data quality; also we will describe how these metrics operate joined to a modified machine learning algorithm, being our design developed in a generic way in order to work with various databases and platforms; finally we will summarizes the preliminary results and we will discuss the conclusions and the work to be performed in the immediate future.

2 Data Mining, Big Data and Quality

Nowadays, huge corporations are seeking to know more about their business process. They usually have enormous and valuable data repositories, but they do not know what to do with this data. It is common to hear the phrase: "worse than have too little (or any) data, is to have many data and not knowing what to do with it" [1].

Data Mining, Knowledge Discovery in Databases (KDD), Business Intelligence or Big Data Analytics, can be useful technologies to meet that challenge. These approaches are focused on transforming data into knowledge (or intelligence) to improve corporate central process. At the end, the term Big Data represents a computer discipline formed with tools emerged from Artificial Intelligence and Database technology, which the main purpose is to give people the information or knowledge that they need to do their jobs.

Before Big Data and after Data Mining, the term Business Intelligence (BI) was coined by Howard Dresner several years ago [2], to describe an emerging discipline concerned with the discovery of information (that was not known before) in a corporation. BI includes disciplines and tools like:

- Data Warehouses [3],
- On Line Analytical Processing (OLAP) and related methods (MOLAP, ROLAP, etc.) [4],
- Knowledge Discovery in Databases (KDD) and Data Mining [5],
- Artificial Intelligence areas and algorithms like, for example, Machine Learning, Intelligent Multi-Agents Systems, Artificial Neural Networks, Fuzzy Logic, Case Base Reasoning, Pattern Recognition, Genetic Algorithms, etc. [6],
- Statistical analysis,
- And, in general, any algorithm, tool or method that serve to transform data into knowledge.

It is predicted that, in the near future, BI will become a need of all huge corporations [2]. But, more recently the term "Big Data" has emerged. According to [7] Big Data can be characterized by: a) volume (large amounts of data), b) variety (includes different types of data), and c) velocity (constantly accumulating new data).

But maybe the first great challenge for Big Data is to manage information that contains data with the appropriate

quality. Speaking in a broad context Data Quality refers to conduct a thorough investigation of the data in the database. This research can be done before to the creation of the database or for those already in operation. It includes determining who are the users of the database, what they need, what is the essence of the business, what are the important variables, how often the information will be required, what level of detail is required, what levels of safety and risk is needed, etc. And, for those databases in operation, we need to measure the current quality of information, in order to know and improve that information.

The activities of defining, measuring, analyzing and improving the data in the database results in the total quality management data cycle, which sees information as a product and is a powerful methodology to develop and maintain databases that contain quality data which is required by the business and is based on the principles of quality proposed by Deming [8]. According to Hufford [9] Data Quality consists of five basic dimensions: completeness, validity, consistency, timeliness and accuracy, which together mean that the data are appropriate for a particular purpose.

Although data quality should be a starting point must for every computer system with databases, in practice this objective is not met in most of the cases. And even with a quality system in place, the experts agree in the sense that any large database can have a 100% quality, as mentioned by the international computer systems analyst company Gartner [10]. Thus, since we cannot achieve a perfect database which meets all the requirements expressed by the Data Quality theory, a remedy to ensure that a database is useful, initially, is to focus only on the dimension named "accuracy", identifying dirty data and diagnosing the quality of data in order to apply cleaning (data cleansing or data cleaning). This cleaning process can include removing those records or variables that, according to some criterion, are dirty, duplicate or un-useful. Another more sophisticated type of cleaning is by means of estimate statistically the possible value of dirty data based on data believed to be clean, or by inferring it [11].

A special form of data with noise is when the data is unknown, and then Kononenko [12] identifies several types: forgotten or lost, not applicable, irrelevant, or omitted in the design. Brazdil [13] has proposed ways of dealing with unknown values, and in particular Quinlan [14] has worked with top-down induction of decision trees techniques for the handling of unknown values, and has proposed up to seven different treatment schemes.

An important part of data cleaning is to check the consistency of records, i.e., detect whether there are cases with the same values of attributes (or similar) with different classes [15]. A special case is when the cleaning process is over non-numeric attributes, i.e., there are text descriptions, such as names of people, products, addresses, etc.: in that case the cleaning has to be developed based on a parser program to detect similarities and standardize and verify the data [16].

For the metrics and schema proposed here, we have used concepts from Big Data, BI, KDD, data mining, data quality and data cleaning described above to identify dirty data and thus obtain a general analysis of the database. These topics are detailed in the next Section.

3 Proposed Data Quality Diagnosis Metrics and Schema

Among the objectives of the schema that we present for the diagnostic of the quality of a very large database, we can include the following:

- Obtain an initial way of how to attack the problem,
- Get a general idea of the status of data (global view – focused on the business data),
- Measure data quality,
- Establish patterns of data quality,
- Detect critical points in the data, and
- Reach a starting point to develop the cleaning business rules to be applied to the data.

To describe the data quality evaluation schema that we developed, first we will discuss the metrics that we devised to obtain a numeric indicator of the quality level of the data, in an objective way. Then we will describe how this approach operates, being designed in a generic way to work with various databases and platforms. Finally we will discuss the preliminary results that we obtained by applying this schema to simulated large databases.

3.1 Metrics for Data Quality

There are a number of metrics designed to obtain an indication of the quality of the data. In particular we focused our research work on the dimension "accuracy" of data.

We seek for a metric that was simple, so it could be easily understood, yet robust, to be able to get data quality information at different levels of data aggregation, i.e. at the attribute level, the table level or at the database level. Additionally, we seek that our metric can accept a weighted schema (assigning costs depending on the importance of each attribute or table), and we seek that it was supported by the experience of other companies related in the data quality issue. We also seek that the metric may include different types of dirty data, from the most common, even those who are less frequent.

Our metric is based on the "Frequency check" that is used by: Cambridge Research Group [17], Knowledge Integrity Incorporated [18], Business Objects (recently acquired by SAP) [19], Group 1 [20] and Gartner [10]; all these are solid companies in the Information Technology and Big Data areas.

In our case, we define one error per each incorrect or missing data, and we sum all occurrences and we named like *"#incorrect*". The accumulated error is expressed as a percentage according to:

$$\%Error = \#incorrect \,/\, TD \tag{1}$$

where *TD* stands for "total data" and it is obtained in various ways, depending on the level of aggregation. For an attribute the variable *TD* is equal to the total number of records; for a

table the *TD* value is obtained by multiplying the number of attributes in the table by the number of records; for a database it is calculated by the sum of the "total data" of each table in the database.

In the event that a field has no data, an error is registered. In the case of an attribute with no data, it is assigned a 100% error to this attribute. In the case of a table with no data, also it is assigned a 100% error to this table.

To assign weights to the attributes or important tables, 100 points should be considered for all attributes of a table. Then these 100 points are distributed according to the importance of each attribute (representing the weight assigned for the user). If we have a total of 10 attributes, each would have 10 points if we want that all the attributes had the same weight. Thus, the weight serves as a factor that is applied to each attribute to obtain the value of "%Error" in a weighted schema. In other words, "% Error" reflects the fact that there are attributes with greater relevance than others. The same idea would be applied to the table level.

According to the above expressed, the quality is calculated as:

$$Quality = 100 - \%Error \quad (2)$$

Then, if "*Quality*" is 100% we have a perfect database and if "*Quality*" takes a value of 50% we can say that the database is wrong in a half of its data. The importance of this measure is that it permits to have an objective measure such that it is able to independently evaluate certain attributes of interest for a particular user, or evaluates a single table that is of particular importance, or shown, in a comprehensive manner, the quality of a complete database, all this depending on the special information needs of each user.

3.2 Data Quality Schema Description

As stated before, our schema allows for an automatic analysis of the data quality of a specific database, through three aggregation levels: a) Attribute, b) Table, and c) Database. Additionally the proposed approach based his diagnosis by means of identifying missing values (blanks), zero (never caught), repeated characters, dates and numbers out of range, etc. There are relationships among the several characteristic data blocks: the hierarchy is established in terms of how each characteristic data block interacts.

The central idea to search for and identify bad data is to conduct a count of the number of occurrences of each of the values of an attribute that occurs in the table: data that appear very infrequently can be considered as "suspicious dirty", and this basic idea is applied by us to numerical values and also to text values of an attribute. This idea is detailed paragraphs below.

An innovative feature is that our design seeks for flexibility, since it has the characteristic of being configurable to access various sources of data (platforms) to create various Business Intelligence rules that are capable of detecting suspicious quality in data.

This design has the ability to connect to various data sources by means of JDBC (Java Data Base Connectivity) technology or via ODBC (Open Data Base Connectivity). Additionally, the proposed schema manages business rules and they assist the diagnostic process, serving as indicators to identify incorrect or anomalous values. In our shema it is possible to define a business rule catalog, which can later be used in different "cases of diagnosis", relating each rule with multiple attributes to support the quality data diagnosing process.

The business rules are a particular type of production rules, traditionally used in Expert Systems. We design our schema like an Expert System Shell [21] in order to gain several advantages from this area, like: capability to create and increase expert knowledge by means of new production rules, include common sense knowledge, obtain permanent expertise, achieve easy to transfer and document rules, gain consistency, capability to verify knowledge and obtain expertise in an affordable way.

We define two types of business rules: for text data and numeric data. In the case of text data, the business rule can detect out of range data (only accepts a set of predefined valid descriptions), incorrect data, dates out of range, null data, data with repeated characters and missing data. For numeric data, the schema detects out of range values by grouping into a predefined quantity of intervals, being the first and last intervals (often with infrequent data) those that can be considered dirty-suspect.

For example, to create a business rule to detect strange symbols, null values and repeated characters, the user just has to select the "Text type" button, followed by the "Special characters" option and click the "Ok" button. To create a business rule to detect values out of range of a numeric attribute, the user only has to select the "Number type" button, then define a valid range and click the "Ok" button. Once defined and stored all the necessary business rules, the user has created a catalog of business rules, which may be applied to the attributes which she or he considers necessary and appropriate to link.

The schema also allows the user to create and store cases of diagnosis: this feature allows the user to easily run this pre-defined diagnoses cases, without necessity of rewriting the business rules. To do this, the user specifies a title of the event (diagnostic case), the period of data to analyze, the business rules assigned by attribute, and sets the data source, tables and attributes to diagnose.

After the execution of a "diagnosis case" the schema automatically generated three types of reports:

a) Frequency Values Report: is an outline of the analyzed data by means of a frequency list of values that each attribute has. If some assigned business rules is related,

the report also shows a column with the number of errors found by that rule.

b) List of rules applied, and

c) List of detail records where suspicious data or errors were detected.

We summarize the data quality diagnosis algorithm in Figure 1.

Given a database with *M* tables, and each table with *D* attributes and *N* instances,

1. Initialize variables *%Error*, *Quality*, *#incorrect*;
2. Assign user-expert estimated weights to attributes and tables;
3. For each *M*, *D* and *N*:

   Apply business rules from the knowledge base to numeric or text data

   If an error is detected, increment *#incorrect*,
4. Calculate global metrics *%Error*, *Quality* at different aggregation levels;
5. Print quality reports.

Fig. 1 Summarization of the proposed schema (data quality diagnostic phase).

At the moment to write this paper, the data cleaning phase is not applied yet. But the same scheme of business rules for diagnosis can be used for the data cleaning phase. In Figure 2 we show a possible algorithm proposed by us to infer unknown data, following the ideas from [14]. In particular we propose step C2 to use the well known ID3 algorithm applied to the unknown data problem.

3.3 Diagnosis results for simulated Big Data

The schema described here was used successfully to analyze and diagnose a large academic database. Our schema was capable of analyzed nearly 200 tables containing more than 2,000 attributes that represents about 2 billion data. With the prototype was able to detect whether there were attributes with errors, and if there were some tables more problematic than others. In general, we can say that the information obtained using the proposed diagnostic schema is appropriate to improve the quality of the data, like candidate users point out during the test period.

In particular, we consider that the results were successful because we can meet the initial project objectives like: a) To obtain an initial approach to the problem: at the beginning we don't know the databases data quality situation, and after apply the prototype we obtain a better idea of the dimensions of the problem and then it could be possible propose several future action schemes in order to increase database quality, b) To get a general idea of the status of data, detecting in a global view and focused on the business data, the reality of the data, c) To obtain a objective measuring of the data quality, i.e., a qualification or score, that represents a starting point to initiate a total quality management project, d) To establish a group of initial patterns of data quality, that can be enriched with the time instead to be lost, e) To detect critical points in the data that needs immediate attention, and f) To be able to have a starting point to develop the cleaning business rules to be applied to the data in order to increase in an automatic and human-like way the quality of the database.

A. Find the attribute that better divides the data set into homogeneous subsets: for each attribute, calculate the disorder or entropy according to the following formula:

$$E = \sum_{r} [Nr/Nt] \left[\sum_{c} \{-(Nrc/Nr) \log_2 (Nrc/Nr)\} \right]$$

Nr = number of examples in branch *r*

Nt = total number of examples in all branches

Nrc = total of examples in branch *r* of class *c*

B. The attribute which has the smallest value of *E* is taken as the root node of the tree (attribute-node) and there will be one branch for each value that the attribute has.

C. For each value of the attribute-node, select all the examples (rows) with the same attribute value. For each subset do the following:

C1. If all examples belong to the same class, the branch is labeled with the class.

C2. If the subset is empty, find the most similar example (smaller distance) to the current branch; if the distance is acceptable (according to certain threshold previously defined), label the branch with the class of the most similar example, otherwise label the branch as "unknown class".

C3. If the examples in the subset belong to different classes, go to step *A*, with this subset as the new data set.

D. If there are branches without labels, go to step *A*, otherwise finish.

Fig. 2 ID3- based algorithm to infer unknown data (data cleansign phase).

4 Conclusions and Future Work

We present a novel software schema for the diagnostic of the quality of data in large databases, in the context Big Data. In particular we describe an innovative measure in an objective way to measure this quality on the dimension "accuracy" of the data and able to obtain indices at different levels of data aggregation, i.e. at the attribute level, the table level or at the database level. The results obtained by applying this schema to a large academic database have been successful, because the prototype was capable to detect wrong data immersed in billions of data. With the data conveniently clean, we can now initiate Big Data analytics properly.

As future work, we see that it would be important add to the diagnosis schema the ability to create business rules to find dirty data in an inter-relationships among attributes way, i.e. to find when one or more data make that other data be "dirty "because they lack the proper context. To give a simple example, one can consider the case of an attribute or field of "personal names" that could be validated against the attribute of "sex of the person", so this require that the name of the person was appropriate to their gender, otherwise, would be marked as an error or a like a wrong captured data.

Additionally, we need aggregate a more complete inference mechanism to the prototype, in order to take more advantage from the Expert Systems ideas (i.e., symbolic reasoning) and can manage more sophisticated diagnosis schemas. Also it will be important add an explanation facility to justify how the schema reaches a particular data quality diagnostic.

5 References

[1] Richeldi, M. (1999). *A business intelligence solution for energy budget control*. Proceedings of the 3rd International Conference on the Practical Application of Knowledge Discovery and Data Mining, (167-82).

[2] McKay, L. (2008). *Business intelligence comes out of the back office.* CRM magazine, Jun .

[3] Gill, H. S. (1996). *Data warehousing,* Prentice Hall Hispano-americana, S.A.

[4] Brackett, M. H. (1996). *The data warehouse challenge,* John Wiley & Sons, Inc.

[5] Piatetsky-Shapiro, G. (1991). *Knowledge Discovery in Databases: An Overview*, In Knowledge Discovery in Databases, Piatetsky-Shapiro, G. eds., Cambridge, MA, AAAI/MIT.

[6] Turban, E., Aronson, J., Liang, T., Sharda, R. (2005). *Decision support and business intelligence systems*, Prentice Hall.

[7] Berman, J., (2013). Principles of Big Data, Elsevier Inc.

[8] Huang, K., Lee, Y., Wang, R. (1999) *Quality information and knowledge*. Prentice-Hall, NJ.

[9] Hufford, D., *Data warehouse quality*, DMReview, www. Dmreview.com/editorial/dmreview/

[10] Gartner, 4th Annual Enterprise Technologies Summit, Centro Banamex - Ciudad de México, Abril 1999.

[11] Ibarguengoytia, P. (1997) *Anytime probabilistic sensor validation.* PhD Thesis, ITESM, México.

[12] Kononenko, I. (1992) Combining decisions of multiple rules. In Boulay, B. (ed) *Artificial Intelligence (AIMSA)*, Elsevier science Pub, pp. 87-96.

[13] Brazdil, P., Bruha, I. (1992) A note on processing missing attribute values. *Canadian Conf. on AI, Workshop on Machine Learning*, Vancouver, B.C., Canada.

[14] Quinlan, J. (1989) Unknown attribute values in ID3. *Int. Conf. on Machine learning*, pp. 164-168.

[15] Bruha, I. (2000) From machine learning to knowledge discovery: survey of preprocessing and postprocessing. *Intelligent data analysis*, IOS Press 4: 363-374.

[16] Kimball, R. (1996) Dealing with dirty data, *DBMS on line.* www. dbmsmag.com/9609d14.html, Sept.

[17] http://research.microsoft.com/en-us/labs/cambridge/ [consulted on January 2016].

[18] http://knowledge-integrity.com/wpblog/ [consulted on March 2016].

[19] http://www.sap.com/solutions/sapbusinessobjects/index.epx [consulted on February 2016].

[20] http://www. G1.com/Support/ [consulted on October 2015].

[21] Waterman, D. (1986). A Guide to Expert Systems, Addison-Wesley Publishing Co.

SESSION

GENE EXPRESSION, REGULATORY NETWORKS, MICROARRAY, SEQUENCING, ALIGNMENT, AND RELATED STUDIES

Chair(s)

TBA

ISBN: 1-60132-427-8, CSREA Press ©

A Novel Control-flow based Intrusion Detection Technique for Big Data Systems

Santosh Aditham
Dept of Computer Science and Engineering
University of South Florida
Tampa, USA.

Nagarajan Ranganathan
Dept of Computer Science and Engineering
University of South Florida
Tampa, USA.

Abstract*— Security and distributed infrastructure are two of the most common requirements for big data software. But the security features of the big data platforms are still premature. It is critical to identify, modify, test and execute some of the existing security mechanisms before using them in the big data world. In this paper, we propose a novel intrusion detection technique that understands and works according to the needs of big data systems. Our proposed technique identifies program level anomalies using two methods - a profiling method that models application behavior by creating process signatures from control-flow graphs; and a matching method that checks for coherence among the replica nodes of a big data system by matching the process signatures. The profiling method creates a process signature by reducing the control-flow graph of a process to a set of minimum spanning trees and then creates a hash of that set. The matching method first checks for similarity in process behavior by matching the received process signature with the local signature and then shares the result with all replica datanodes for consensus. Experimental results show only 0.8% overhead due to the proposed technique when tested on the hadoop map-reduce examples in real-time.

***Keywords*-big data; intrusion detection; control-flow graph;**

I. Introduction

The architectures for big data systems rely on parallel execution techniques like mapreduce [1] for fast processing. With the growing popularity of real-time data processing in big data environments, there is a pressing need to re-imagine the traditional computing techniques. For example, data locality in popular big data system distributions like hadoop [2] and spark [3] is redefined as bringing compute to data instead of the traditional approach of the moving the data that needs to get processed. This trend of re-inventing the traditional methods do not necessarily transform to the security needs of big data. The security features implemented in big data systems are still based on traditional methods for systems based on general purpose machines. User authentication, multi-level data access control and logging are typically used for security in big data [4]. Data encryption is slowly being adopted in the big data field, but it is limited by big data properties like volume and velocity. As we covered in our previous work [5], big data security is premature and there is a lot of scope for improvement in this area. For instance, the current security standards for big data systems assume system-level consistency which is not necessarily true always. We demonstrated in our previous work [5] that big data platforms can be affected by insider attacks. In this work, we concentrate on detecting process-level intrusions within big data systems.

Intrusion detection systems (IDS) can identify malicious use based on their knowledge of possible threats or by learning from the behavior of programs. Knowledge-based IDS usually search a program for known threat signatures that are stored in a database. With new and zero-day attacks emerging regularly, it is impractical to have a pre-populated database of all possible threats. Even if it is assumed to have such a database, maintaining it would require a lot of resources and running search queries against it would be expensive. Behavior based IDS tries to model, analyze and compare application behavior to identify anomalies. This technique needs more resources and is more complex than signature-based IDS but it is more effective in a dynamically changing threat environment. Behavior based IDS generally use statistics and rules to detect anomalies. Figure 1 gives a taxonomy of the different types of IDS.

In today's internet age, a distributed implementation of IDS is needed for which aggregation, communication and cooperation are key factors of success. Distributed IDS gives centralized control and detects behavioral patterns even in large networks but it has to be employed at multiple levels: host, network and data [6]. Hence, using big data in general-purpose distributed IDS implementations is recommended for faster processing. In this work, we concentrate on IDS that can be used for security within big data systems. IDS within a big data system favors anamoly-based IDS when compared to knowledge-based IDS because of the naturally large and ever increasing scope of threats.

Using control-flow graphs for logic level intrusion detection is a commonly known idea [7], [8], [9]. For example, control-flow integrity [10] is a security mechanism that can identify misuse of application logic bugs, like buffer-overflow attacks. Though CFGs are generally sparse graphs, they can grow very big in size. Hence, it is important to design IDS techniques that can work with a reduced representation of CFGs. A Minimum Spanning Tree (MST) contains all vertices and only some paths of its source graph and the number of MSTs for sparse graphs is generally less. Hence, a set of MSTs extracted from a CFG can be used for IDS that detects program level anomalies.

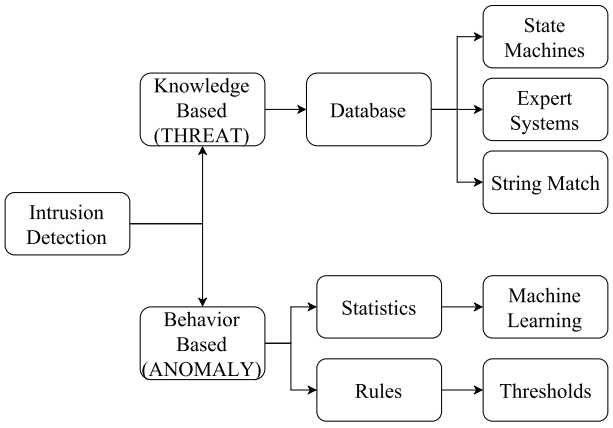


Figure 1: A taxonomy of Intrusion Detection Techniques

In this paper, we propose a control-flow based intrusion detection technique for big data systems. The proposed technique checks for program level anomalies in big data applications by analyzing and comparing the control-flow behavior of all processes running inside a big data system. The proposed intrusion detection technique is divided into two parts. First, the control-flow of each process running on a data node in the big data cluster is locally analyzed. This is done by extracting a set of MSTs from the instruction level CFG of a compiled program. The extracted set of MSTs are hashed and stored in an array called the *program signature*. Then, the stored program signature is encrypted and shared with other replica nodes that run the same program. In the second step, the received encrypted program signature is decrypted and matched with the local version to check for coherence. Matching two program signatures involves finding a perfect match for every MST in a signature within the set of MSTs of the other. The result of the matching step is then shared with replica nodes for consensus. Our technique is designed to be simple, scalable and efficient in identifying both control-flow and brute-force attacks.

The rest of this paper is organized as follows. Section II gives some background about big data systems, control-flow graphs and IDS. The various related works are also discussed here. Section III explains the proposed intrusion detection technique in detail. Experimental setup and results are thoroughly discussed in Section IV. Finally, Section V gives the conclusion and future work.

II. Background and Related Work

In this section, background about the three topics - big data systems, control-flow graphs and intrusion detection is provided. The related works are briefly outlined here.

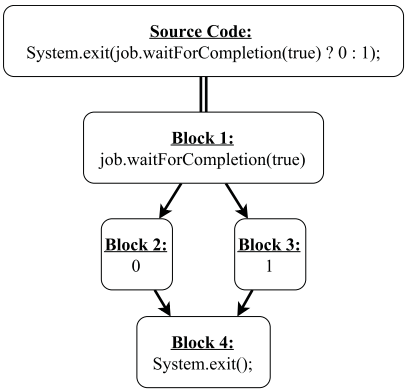
A. Big Data Systems

Big data systems are data driven and their work can be classified into 2 major tasks - writing user data to the disk for storage and; reading stored data when user requests for it. Typically, this data is quantified in units called *blocks*. For fast and fault-tolerant service, big data systems rely on replication of data blocks which in turn demands data consistency. Big data systems cannot afford to have read or write service-level inconsistency. The motivation for this work comes from a weak assumption in the big data community that the services used by a big data system to maintain data consistency are never attacked. It is our knowledge that this problem has not been widely addressed before.

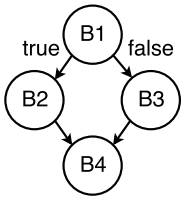
To propose an IDS for big data services, it is important to understand how the services work. For this, we picked 2 popular big data services - reads and writes. When a client (or user) wants to write a block, the namenode picks n data nodes from the big data cluster to complete this task where n is the replication factor of the cluster. First the namenode checks if the datanodes are ready. It sends a ready request to datanode1 which when ready, forwards that request to datanode2 and so on. When the namenode knows that all n datanodes are ready, it asks the client to start writing. The client only writes to datanode1 which is subsequently written on to datanode2, datanode3 and so on. In case of any failure, namenode orders a new datanode to maintain block replicas. When the client wants to read a block, namenode gives the client a list of all datanodes that have the block and the client picks first datanode. If there is a problem reading from datanode1, the client request gets forwarded to the next datanode that has a copy of the same block.

B. Control-flow Graphs

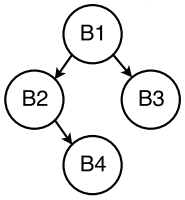
A control-flow graph (CFG) is a directed graph representation of a program and usually a sparse graph. CFGs include all possible control paths in a program. This makes CFG a great tool to obtain control-flow behavior of its process. Vertices in a CFG give the level of detail, such as instruction-level or basic block level, that cannot be further divided. Edges in CFG represent control jumps and are classified into two types - forward and backward. Branch instructions, function calls, conditional and unconditional jumps account for forward edges. Virtual calls and indirect function calls are also considered as forward edges but their destinations are difficult to determine. Loops and returns generally account for backward edges. The integrity among duplicate processes that run on replica nodes of a big data system can be verified with the information available in a CFG [11]. Similarity check between program logic of two programs can be performed by comparing their CFGs for isomorphism. There are many ways to check for such graph isomorphism [24], [25] but analyzing the similarity of two processes by conducting CFG level graph isomorphism is hard and time consuming. Graph isomorphism is a complex problem, sometimes known to be NP-complete as well [8]. To reduce the complexity of graph algorithms, CFGs can be reduced to trees or subgraphs before performing any



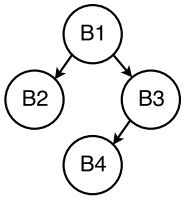
(a) Source code to basic blocks



(b) Basic-block CFG



(c) MSA of CFG



(d) Another MSA of CFG

Figure 2: Multiple MSAs of same CFG

coherence or integrity checks [12]. A CFG can be converted to a tree using methods such as Depth-first traversal. Several tree structures like Dominator Tree, Minimumm Spanning Tree (MST), Minimumm Spanning Arborescence (MSA) can be extracted form CFGs [13], [14], [15]. For this work, MST and MSA can be used interchangeably. CFGs can be broken into subgraphs using methods like k sub-graph matching and graph coloring. Some popular methods for graph reduction and graph comparison that can be found in the literature are given below (assume graphs to have n vertices and m edges):

- *Based on Edit Distance*: Using Smith-Waterman algorithm with Levenshtein distance to identify similarity between two graphs represented as strings [16]. The time complexity is O(nm).
- *Based on Traversal*: (a) A preorder traversal of a graph G where each node is processed before its descendants. (b) A reverse postorder in a DAG gives a topological order of the nodes [17].
- *Based on Dominator trees*: A data structure built using Depth First Search or using the method proposed by Tarjan in [18]. Tarjan's method has a time complexity of O((n+m)log(n+m)).
- *Based on Reachability*: Transitive reduction of a sparse graph to another graph with fewer edges but same transitive closure [19]. The time complexity is O(nm).

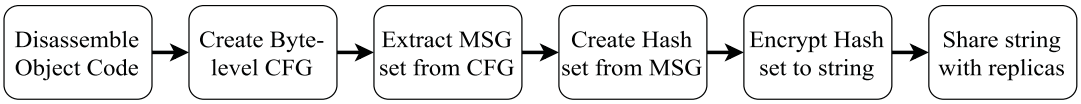
In this work, we chose to reduce a CFG to a set of MSTs because CFGs are generally sparse graphs and hence the size of the set of MSTs will be finite and small. Edmond's algorithm can be used to extract MSTs from a digraph [13], [14], [15]. Since an MST contains all vertices of its graph, there will be no loss in the program instruction data. Depending on the connectedness of the graph, the edge count will defer between the CFG and MST representation of a program. Figure 2 shows transformation of a line of java code to basic blocks of bytecode to CFG to set of MSAs. Vertices B1, B2, B3, B4 are the basic blocks formed from java bytecode. There exists an O(m + n log n) time algorithm to compute a min-cost arborescence [13]. Alternately, another approach for converting a CFG to MST using union find is used by popular compilers like llvm and gcc for security purposes [**?**]. One known disadvantage of using CFGs and MSTs for security is that dynamic link library calls cannot be verified.

C. Intrusion Detection Systems

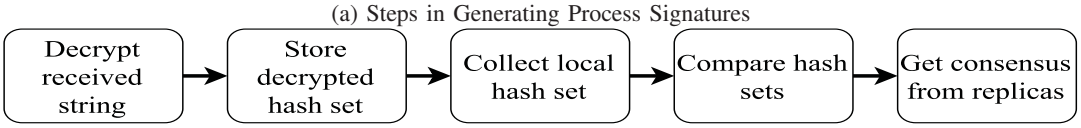
Traditionally, IDS checks for known malware in programs by performing signature matching on a threat database [20]. Signature match using exact string matching is limited in its scope. This is because variants of same attack will have different signatures. Recently, methods to detect new malwares using statistical machine learning have been proposed. Static analysis using CFG is another efficient way to detect intrusions but it is very complex [21]. Converting a CFG to a string and implementing string matching is another way to deal with this problem but the solution will not be polynomial. Also, CFG at basic block level can have basic block variants that look different but perform the same function. To deal with these shortcomings, many approximate matching techniques have been proposed. Tracing applications to get their CFG is another approach that is used in applications like xtrace, pivottrace etc [22], [23]. In case of big data systems, data nodes usually have the same processor architecture. Hence it can be assumed that there will be no variants when the CFG is constructed at byte-level. It is then sufficient to verify similarity among the CFGs of two processes to confirm coherence in the nodes of a big data system.

III. Proposed Technique

In this section, we describe our proposed two-step intrusion detection technique for big data systems. The first step involves capturing the control-flow of a process running on a datanode of the big data system. The second step involves process-level similarity check followed by consensus among replica datanodes.



(a) Steps in Generating Process Signatures



(b) Steps in Matching Process Signatures

Figure 3: Proposed Algorithm for Intrusion Detection

A. Generating Process Signatures

In this work, we emphasize on process level intrusion detection by observing coherence in the behavior of duplicate processes running on replica datanodes of a distributed big data system. To capture the program behavior, the first step is to identify a representation of the program that has the information we need and filters out all other data. We call this representation as the program signature. Since our goal is to identify intrusions from control-flow mismatch, our program signatures should contain all possible control flow information of a program.

Compiled source code of a program is generally used to generate static CFG. Since most big data frameworks use a virtual machine (like JVM), an instruction level CFG in this context is generated from java byte code. In this work, disassembled object code (DOC) from java byte code is used as input to generate the CFG at instruction level. It is important for the program signature to contain only the information that is necessary. Hence, every CFG is converted into a set of MSTs that are later used to generate the program signature. In this work, we propose the idea of representing a program by a set of MSTs/MSAs that can be extracted from a byte-level CFG using Edmonds algorithm. This set of MSTs that are extracted from a CFG are further filtered to only the set of edge-disjoint MSTs. There are many versions proposed for Edmonds algorithm [13], [14], [15] and for this work we used a version from NetworkX graph library [31] that generates edge disjoint spanning trees from the root vertex of a given digraph. Once a minimal representation of the logic in a program is obtained in the form of an MSA, it is converted into a string by listing the node list first followed by edge list, which is in accordance to the DOT format representation.

The length of a MST string in DOT format is dependent on program size. To make the comparison step faster, we convert the variable length MST strings of a program to fixed length strings using hashing. The extracted set of edge-disjoint MSTs are hashed using popular hashing algorithms like SHA or MD5 to generate a set of fixed-length hash strings. Since a sparse graph like CFG can have multiple MSAs, the program signature can be a single hash string or a set of hash strings. Having all possible MSAs in the program signature makes the graph similarity check more reliable. In the end, a *program signature* is a set of fixed-length strings.

Program signatures are encrypted before being shared with replica datanodes for tighter security. The private key for encryption is generated from a harcoded master key if we use secure hardware like the one proposed in our previous work [5]. Every datanode in a big data system runs the proposed *profiling method* for every running process and it includes all the steps involved in converting the compiled binary of a program to its program signature. A pictorial representation of the steps in profiling method is given in Figure 3.

B. Matching Process Signatures

Replication property of big data systems opens scope for new methods of implementing application logic level IDS techniques. Process similarity check among duplicate nodes of the cluster helps in checking for coherence among the replica datanodes while performing a write or read operation. When a process is scheduled to run on a datanode that hosts the primary copy of a data, a signature for that process is created by the *profiling method* (Step 1) of our proposed IDS technique and that signature string is shared with all replica datanodes. In the *matching method* (Step 2), these signatures received from other datanodes are decrypted and matched with the local versions of the same process. The results are shared with all other replica datanodes for consensus. For secure communication among datanodes, we intend to use the same secure communication protocol that was proposed in our previous work [5].

The most important part of the matching method is to check for similarity (or dissimilarity) between two program signatures. Generally, graph similarity check can be performed by checking node similarity and edge similarity. The following points are considered while comparing MSTs to check for similarity among programs:

- MSTs are sparse graphs obtained from byte-level CFGs. Hence, checking for path sensitivity is not exponential.
- All edges are assumed to have the same weight of 1.

- The total number of MSTs for a CFG is limited (by Cayley's formula [26]).
- By Edmonds theorem, a graph which is k-connected always has k edge-disjoint arborescences.
- Two MSTs are a perfect match if their node sets and edge sets match exactly.
- If edge set of one MST is a subset of the edge set of another MST, the source graphs of these MSTs are not similar.
- Two graphs are similar if for every MST in one graph there exists a perfect match in the set of MSTs of the other graph.
- Hashing algorithms like SHA1 or MD5 are quick and efficient.

Based on the points listed above, the following method is developed for graph similarity check. Let us consider 2 control-flow graphs G1 and G2. Let $<N1, E1>$ represent G1 where N1 is the node set of the graph G1 and E1 is the edge set of the graph. Similarly, $<N2, E2>$ represents G2 where N2 is the node set of the graph G1 and E2 is the edge set of the graph. After employing a variation of Edmonds algorithm on these CFGs (such as finding all edge-disjoint MSTs), lets us assume that M1 $[<N1, E1'>]$ is the set of MST/MSA for G1 and M2 $[<N2, E2'>]$ is the set of MST/MSA for G2. In order to check for similarity in both graphs G1 and G2, we check if there is a perfect match in M2 for all MSTs in M1. In order to simplify the match function, we propose using a hash function on M1 and M2 that creates a unique hash for every MST. Let H1 be a set of hashes generated from M1 and H2 be the set of hashes from M2. If any hash in H1 does not exist in H2, we deduce that the graphs are not equal.

IV. Experimental Results

In this section, the experimental setup and experiments used for testing the proposed technique are provided. The results and some analysis are also provided.

A. Setup

An Amazon EC2 [27] m4.xlarge instance running Ubuntu 14.04 is used to generate MSTs (and their hashes) from CFGs using SageMath. The proposed technique was implemented and tested on an Amazon EC2 big data cluster of 5 t2.micro nodes - 1 master node, 1 secondary master node and 3 datanodes with a replication factor of 3. The list of softwares used in conducting our experiments are:

- **SageMath** [28] is a free open-source mathematics software system for mathematical calculations.
- **GraphML** [29] is a popular graph representation format which can used to represent both CFG and MST.
- **Graphviz** [30] is open source graph visualization software that takes input in DOT format and makes diagrams in useful formats.

Table I: List of Hadoop Map Reduce Examples

E.No	Name	Description
1	wordmean	A map/reduce program that counts the average length of the words in the input files.
2	pentomino	A map/reduce tile laying program to find solutions to pentomino problems.
3	distbbp	A map/reduce program that uses a BBP type formula to compute the exact bits of pi.
4	aggregate-wordcount	An Aggregate based map/reduce program that counts the words in the input files.
5	sec-ondarysort	An example defining a secondary sort to the reduce.
6	aggregate-wordhist	An Aggregate based map/reduce program that computes the histogram of the words in the input files.
7	ran-domwriter	A map/reduce program that writes 10 GB of random data per node.
8	teravali-date	Check the results of the terasort.
9	qmc	A map/reduce program that estimates the value of Pi using a quasi-Monte Carlo (qMC) method.
10	wordstan-darddevia-tion	A map/reduce program that counts the standard deviation of the length of the words in the input files.
11	wordme-dian	A map/reduce program that counts the median length of the words in the input files.
12	bbp	A map/reduce program that uses Bailey Borwein Plouffe to compute the exact digits of pi.
13	teragen	Generate data for the terasort.
14	sudoku	A Sudoku solver.
15	wordcount	A map/reduce program that counts the words in the input files.
16	multi-filewc	A job that counts words from several files.

- **NetworkX** [31] is a Python language software package that provides graph algorithms like Edmonds and VF2.
- **Control-flow graph factory** [32] is a software that generates CFGs from java bytecode (class file) and exports them to GraphML or DOT formats.

B. Experiments

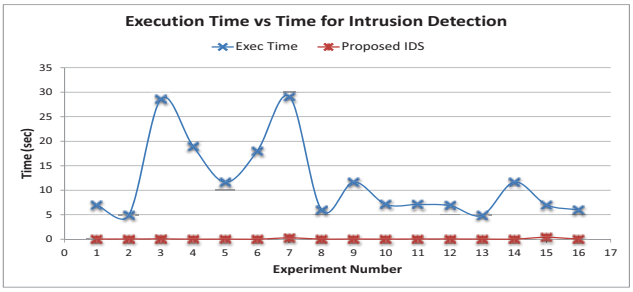
The proposed intrusion detection technique was tested using 16 hadoop map-reduce examples that can be found in all hadoop distributions. These examples cover a wide range of big data applications as listed in Table I. The class files of these examples are readily available in the hadoop distributions. First, control-flow graph factory [32] was used to generate control flow graphs from the class files. These graphs are stored in graphml format and given as input to a simple SageMath [28] script that uses NetworkX library [31] and computes the edge-disjoint MSAs and hashes them using MD5. A C++ application was used to implement encryption and secure communication needed for the proposed IDS technique. The implementation was based on framework from [5]. The hashes are fixed length strings and so we restrained to using a basic numeric key based left/right shift for encryption/decryption of messages. Since there are no benchmarks for some of these examples, we executed them with minimum input requirements.

Table II: Hadoop Map Reduce Examples - Program level time metrics in seconds

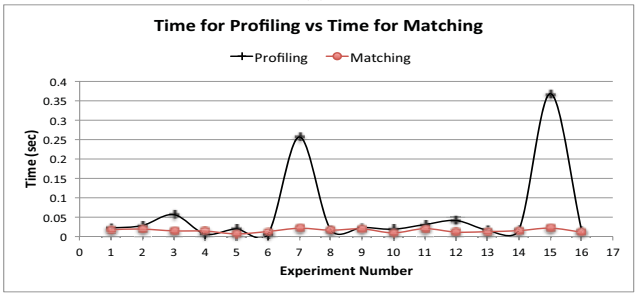
E.No	Example	Profiling method	CFG to MSA set	Hashing	Matching method	Avg Hash Match	Consensus	Proposed	Exec Time	% Time
1	wordmean	0.0216	0.0216	7.89E-05	0.0190	0.0002	0.0187	0.0407	6.988	0.58%
2	pentomino	0.0288	0.0288	8.70E-05	0.0196	0.0013	0.0182	0.0485	4.914	0.99%
3	distbbp*	0.0567	0.0567	6.29E-05	0.0150	0.0019	0.0130	0.0718	28.58	0.25%
4	aggregatewordcount	0.0070	0.007	5.70E-05	0.0145	0.0002	0.0143	0.0215	19.002	0.11%
5	secondarysort*	0.0199	0.0199	5.10E-05	0.0072	0.0018	0.0054	0.0272	11.657	0.23%
6	aggregatewordhist	0.0066	0.0066	4.20E-05	0.0135	0.0012	0.0122	0.0201	18.024	0.11%
7	randomwriter	0.2561	0.2561	8.58E-05	0.0217	0.0025	0.0191	0.2779	29.111	0.95%
8	teravalidate	0.0181	0.0181	5.20E-05	0.0169	0.0001	0.0168	0.0351	5.958	0.59%
9	qmc*	0.0238	0.0238	7.39E-05	0.0202	0.0015	0.0186	0.0440	11.657	0.38%
10	wordstandarddeviation	0.0193	0.0193	7.89E-05	0.0098	0.0021	0.0076	0.0292	7.112	0.41%
11	wordmedian	0.0312	0.0312	6.20E-05	0.0208	0.0020	0.0187	0.0520	7.028	0.73%
12	bbp	0.0415	0.0415	9.08E-05	0.0118	0.0003	0.0115	0.0534	6.865	0.78%
13	teragen	0.0169	0.0169	5.51E-05	0.0131	0.0023	0.0108	0.0301	4.905	0.61%
14	sudoku*	0.0177	0.0177	5.60E-05	0.0156	0.0006	0.0150	0.0334	11.657	0.29%
15	wordcount	0.3672	0.3672	6.99E-05	0.0221	0.0023	0.0197	0.3893	7.034	5.54%
16	multifilewc	0.0159	0.0159	5.20E-05	0.0118	0.0001	0.0116	0.0277	5.963	0.47%
Average Values		0.0593	0.0592	6.59E-05	0.0158	0.0013	0.0144	0.07516	11.657	0.81%

C. Results

Table II, Figures 4a and 4b show the results of our experiments. Figure 4a shows the comparison between the time taken to run the hadoop map-reduce examples on a big data cluster and the time taken to run the proposed intrusion detection technique. The execution times for some examples (represented by * in table II) are inconsistent among multiple runs. We can notice from table II that only 0.81% of time taken to execute an example is needed to analyze it for intrusion detection. The time needed to run the proposed detection technique includes (a) time taken to create CFG for the main method from the class file; (b) time taken to extract MST set from CFG; (c) time taken to hash the MSTs and encrypt them and; (d) time taken to check for similarity among duplicate processes by comparing the program signatures. All of these values can be found in table II. The last row of this table gives the average values. It can be noticed from Figure 4b that the time required by the proposed technique is influenced by the profiling method trying to extract MSAs from CFG, particularly when there are more than one MSAs for a CFG. Though the matching method performance is directly proportional to the square of the size of the number of edge-disjoint MSAs in a CFG i.e. $O(n^2)$ worst case complexity, we observed that it is rare to have more than a couple of edge-disjoint MSAs in a CFG because of the sparse nature of CFG.



(a)



(b)

Figure 4: A time comparison between (a) Proposed IDS technique and run-time for map-reduce examples. (b) Profiling and matching methods of the proposed IDS technique.

V. Conclusion and Future Work

In this paper, we introduced a novel approach to detect program level intrusions in big data systems with help of control flow analysis. The main idea is to use the replication property of big data systems and check for coherence in program behavior among replica datanodes. Behavior of a program is modeled by extracting a MSA set representation of its CFG. Similarity check among duplicate programs is performed by a complete matching among hashed sets of MSAs. Experiments were conducted on real-world hadoop map-reduce examples and it is observed that the proposed technique takes only 0.8% of execution time to identify intrusions. The naturally sparse nature of CFGs helps in achieving this low overhead. For future work, we would like to explore graph string matching and compare the proposed

matching method (step2) with other graph isomorphism techniques.

REFERENCES

[1] Dean, Jeffrey, and Sanjay Ghemawat. "MapReduce: simplified data processing on large clusters." Communications of the ACM 51.1 (2008): 107-113.

[2] White, Tom. "Hadoop: The definitive guide." O'Reilly Media, Inc., 2012.

[3] Zaharia, Matei, et al. "Spark: cluster computing with working sets." Proceedings of the 2nd USENIX conference on Hot topics in cloud computing. 2010.

[4] OMalley, Owen. "Integrating kerberos into apache hadoop." Kerberos Conference. 2010.

[5] Aditham, Santosh, and Nagarajan Ranganathan. "A novel framework for mitigating insider attacks in big data systems." Big Data (Big Data), 2015 IEEE International Conference on. IEEE, 2015.

[6] Tan, Zhiyuan, et al. "Enhancing big data security with collaborative intrusion detection." Cloud Computing, IEEE 1.3 (2014): 27-33.

[7] Bruschi, Danilo, Lorenzo Martignoni, and Mattia Monga. "Detecting self-mutating malware using control-flow graph matching." Detection of Intrusions and Malware & Vulnerability Assessment. Springer Berlin Heidelberg, 2006. 129-143.

[8] Nagarajan, Vijay, et al. "Matching control flow of program versions." Software Maintenance, 2007. ICSM 2007. IEEE International Conference on. IEEE, 2007.

[9] Dullien, Thomas, and Rolf Rolles. "Graph-based comparison of executable objects (english version)." SSTIC 5 (2005): 1-3.

[10] Abadi, Martn, et al. "Control-flow integrity principles, implementations, and applications." ACM Transactions on Information and System Security (TISSEC) 13.1 (2009): 4.

[11] Amighi, Afshin, et al. "Provably correct control flow graphs from Java bytecode programs with exceptions." International Journal on Software Tools for Technology Transfer (2015): 1-32.

[12] Gold, Robert. "Reductions of Control Flow Graphs." World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering 8.3 (2014): 417-424.

[13] Gabow, Harold N., et al. "Efficient algorithms for finding minimum spanning trees in undirected and directed graphs." Combinatorica 6.2 (1986): 109-122.

[14] Uno, Takeaki. An algorithm for enumerating all directed spanning trees in a directed graph. Springer Berlin Heidelberg, 1996.

[15] J. Edmonds, Optimum branchings, J. Res. Natl. Bur. Standards 71B (1967), 233240.

[16] Bunke, Horst. "On a relation between graph edit distance and maximum common subgraph." Pattern Recognition Letters 18.8 (1997): 689-694.

[17] Sharir, Micha. "A strong-connectivity algorithm and its applications in data flow analysis." Computers & Mathematics with Applications 7.1 (1981): 67-72.

[18] Georgiadis, Loukas, Robert Endre Tarjan, and Renato Fonseca F. Werneck. "Finding Dominators in Practice." J. Graph Algorithms Appl. 10.1 (2006): 69-94.

[19] Tarjan, Robert E., and Mihalis Yannakakis. "Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs." SIAM Journal on computing 13.3 (1984): 566-579.

[20] Pathan, Al-Sakib Khan, ed. The state of the art in intrusion prevention and detection. CRC press, 2014.

[21] Wagner, David, and Drew Dean. "Intrusion detection via static analysis." Security and Privacy, 2001. S&P 2001. Proceedings. 2001 IEEE Symposium on. IEEE, 2001.

[22] Wang, William. End-to-end Tracing in HDFS. Diss. Carnegie Mellon University Pittsburgh, PA, 2011.

[23] Mace, Jonathan, Ryan Roelke, and Rodrigo Fonseca. "Pivot tracing: dynamic causal monitoring for distributed systems." Proceedings of the 25th Symposium on Operating Systems Principles. ACM, 2015.

[24] Koutra, Danai, et al. Algorithms for graph similarity and subgraph matching. Technical Report of Carnegie-Mellon-University, 2011.

[25] Cordella, Luigi P., et al. "A (sub) graph isomorphism algorithm for matching large graphs." Pattern Analysis and Machine Intelligence, IEEE Transactions on 26.10 (2004): 1367-1372.

[26] Shor, Peter W. "A new proof of Cayley's formula for counting labeled trees." Journal of Combinatorial Theory, Series A 71.1 (1995): 154-158.

[27] Amazon, E. C. "Amazon elastic compute cloud (Amazon EC2)." Amazon Elastic Compute Cloud (Amazon EC2) (2010).

[28] Sage Mathematics Software (Version 4.0), The Sage Developers, 2016, http://www.sagemath.org.

[29] Brandes, Ulrik et al. Graph Markup Language (GraphML). CRC (2013).

[30] Emden R. Gansner and Stephen C. North. "An open graph visualization system and its applications to software engineering." SOFTWARE - PRACTICE AND EXPERIENCE 30.11 (2000): 1203-1233.

[31] Aric A. Hagberg, Daniel A. Schult and Pieter J. Swart, Exploring network structure, dynamics, and function using NetworkX, in Proceedings of the 7th Python in Science Conference (SciPy2008), Gel Varoquaux, Travis Vaught, and Jarrod Millman (Eds), (Pasadena, CA USA), pp. 1115, Aug 2008

[32] Alekseev, Sergej, Peter Palaga, and Sebastian Reschke. "Bytecode Visualizer." Control Flow Graph Factory. N.p., 2008. Web. 24 Mar. 2016.

Big Data to Optimise Product Strategy in the Electronics Industry

Nawaz Khan[1], Vijayalakshmi Subbiah[1], Elli Georgiadou[1] and Angela Repanovich[2]
[1]School of Science and Technology, Middlesex University, NW4 4BT, London.
[2]Mechatronics Department, Transilvania University of Brasov, Romania.

Abstract - *This research identifies the success factors for new product development and competitive advantage as well as argues how big data can expedite the process of launching a new product initiative. By combining the research findings and the patterns of background theories, an inquisitive framework for the new product development and competitive advantage is proposed. This model and framework is a prototype, which with the aid of scenario recommends the parsimonious and an unified way to elucidate the requisite of the market analysis, organizational potential and customer insights for product strategy and competitive advantage.*

Keywords: product strategy, customer insights, market analysis, organizational potential, innovation.

1 Introduction

Transistor by transistor, the electronic industry is literally changing the universe. Companies whose core business revolves around the innovation of new products always reliies on a strategy with an empirical research for its competitive advantage. In an epitome, leaders who win every quarter and every year, and decade after decade, in all environments, and against the best competitors are skilled at transferring their archetype and come up with product strategy which determines the direction of the new product [18]. In order to launch a new product in the marketplace a firm needs a well-planned product strategy and must be supported by large scale data that existed in the public domain.

1.1 Definition of Product Strategy

The encompassment of a product strategy is to assure the success of any organisation by executing specific tasks at a perfect time and it should acquire the significant purpose of the product [46].This can only be achieved in today's world by adhering to big data and its analytics as it gives an opportunity to capture the decisions made by organisations about product within particular markets. Big data also can lead to the decisions of determining the improvement of products to satisfy market requirements and determine in which way to gain competitive advantage [45].

The sheer scale of big data and high frequency of mature data adding power to the product strategy development by combining decision with the management of the different levels of a product, product platforms, product lines and individual products [29].The success of newly developed products can be measured in real-time due to the existence of the big data, i.e. 'now casting'. The product strategy idealises the basis for executing a product roadmap and apparently the product releases [4]. However, a company is able to explicitly contemplate more on a segmented market explicitly and set features if they are properly aligned with the big data. The responsibility of product strategy is to make a superior relationship between the firm's product development and its product strategy [30]. McGarth and MacMillan [31] asserted that the firms and their products are plighted in the captured markets from a competence enhancing perspective. Gathering instant insights from digital big data is the resultant consequence of making imperative decisions in overseeing new product development [47].

The development and accomplishment of the product strategy takes place within an intricate market situation and success can be profoundly influenced by external artefacts (digital data about the competition, the economy, and even regulation); in this paper we propose a model for formulating a realistic product strategy that harnesses information and insights of big data and ensures collaborative innovation. Subsequently, this research is organised around questions :

1) Why some products win and some products fail the competitive advantage in the electronics industry?
2) Can the success factors for new product development and competitive advantage be achieved with the Big Data Analytics?
3) How can companies achieve competitive advantage?

2 Dimensions of Product Strategy

A product strategy focuses on forecasting at the growth stage of the product life cycle to ensure competitive advantage [28] .

Figure 2.1: Strategic Questions-Element of product strategy

Figure 2.1 illustrates the challenges to new product development in the electronics industry.

While deriving a product strategy, the above factors as well as the process listed in the figure 2.2 has to be taken in account.

3 Background : Electronics Industry and Role of Big Data

The electronics world was not just the result of effort of some years or decades, it is rather the result of the hard work of great minds since ages, i.e. Moore's Law.

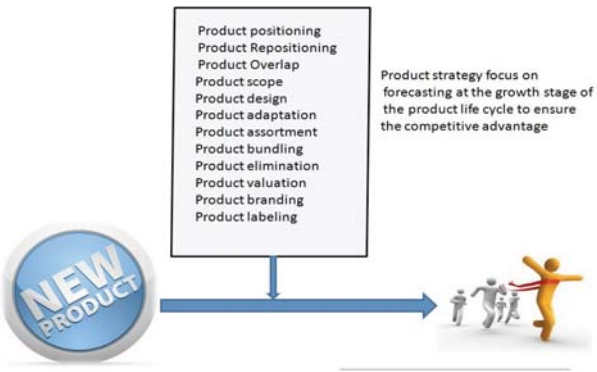


Figure 2.2: Forecasting Product Strategy

Innovation is the creative development of a particular product, service, idea, environment, or process with the ultimate goal of attracting customers and extracting value from its commercialization [39]. Decisive discrepancy must continue swiftly to avoid being overtaken by historically known competitors, as well as those who have yet to appear on the business radar [12]. The endurance of the company is highly implausible, if the rate of change outside the organization surpasses that inside the company, survival is highly unlikely [41]. In this decade, failure to address environmental pressures has brought thousands of companies to slash product augmentation programmes, reduce the size of the workforce, merge with other companies, or close down entirely. To subsist and succeed in the hypercompetitive global marketplace, corporations, therefore, must produce a steady stream of innovation [22]. Every enterprise is required to have a "channel" filled with upcoming innovation releases and must do everything within its power to abbreviate the time required for the development of new innovations.

3.1 Effect on Economy

The rapid increase in the efficient productivity in the electronics industry surpluses the economy which expanded the supply chain and moved the industry globally and has developed a future opportunities in the global economy[38]. Collaborative bodies such as the Materials Research Society continue to feature improvements and innovations [6]. Subsequently, the reduced cost of manufacturing and the increased consistency of new technology nodes have resulted in abundant improvement in the equity and operating profits of the semiconductor industry and, as a result, the electronics sector [26].

The in-depth implications of Moore's Law are seen in the growth of social media technologies and cloud computing , which require reinforced computing capabilities and are directly accountable for the demand for more elements on a single chip. The significance of this law is emphasized by the fact that it has caused a technological advancement and diaspora from microelectronics to nanoelectronics and fabricated an industry segment -- nanotechnology -- that is experiencing exponential growth. Regardless of reports that the law may be "slowing down," it remains the guiding law of the industry today [20].From from medical to transportation, from entertainment to adventures, communication to education and financial aspects all over electronics is the main tool behind the development [34]. The essence of development, especially in times of economic crisis, is innovation.

3.2 Advancement of Electronic Industry

Product innovation throughout the electronics industry has unfold into a highly maneuvered interdependence of technologies, materials, and design methods, modelling tools, and manufacturing process development. Organizations that launched superior high quality products such as Apple and Microsoft have been accredited with adopting the right product strategy for their products [20]. The quest for competitive advantage is already renovating the competitive landscape which will oblige companies to change the way they think about processes, technologies, products, and business models. That competitive advantage will stand them in good stead, because feasibility will always be an integral part of innovation and development [13].

3.3 Types of consumer electronics product

Figure 3.1 illustrates the types of consumer electronics in today's world. According to [36], miniaturization, convergence, digitization are the main factors that help in the growth of the electronic industry to come up with innovations with high value products and maintain competitive advantage.

3.4 Big data: Customer centric insights and innovations

Offering the right product to the right customer is what makes a business successful. The most ideal path for companies to achieve a competitive advantage is through innovation. Some product newness is better than no newness. According to [9], a marketing strategy supported by market insight characterises how a firm plans to compete in a preferred market research and the marketing strategy consists of the product strategy which helps to decide what the company wants to offer to the customers. Hence, product strategy is viewed as a core component of the overall marketing strategy. While formulating a marketing strategy, consumer insights and satisfaction is actually the main goal and a strategy that does not meet the needs of the consumers is a poor product strategy [40].

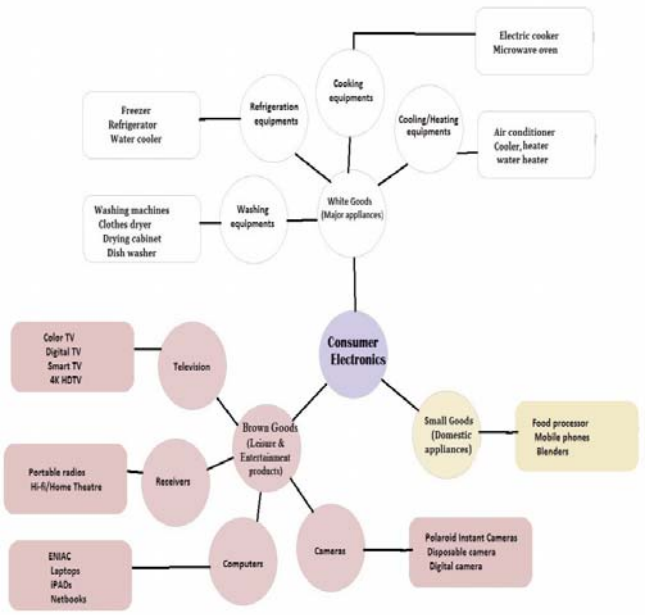


Figure 3.1: Types of consumer electronics types

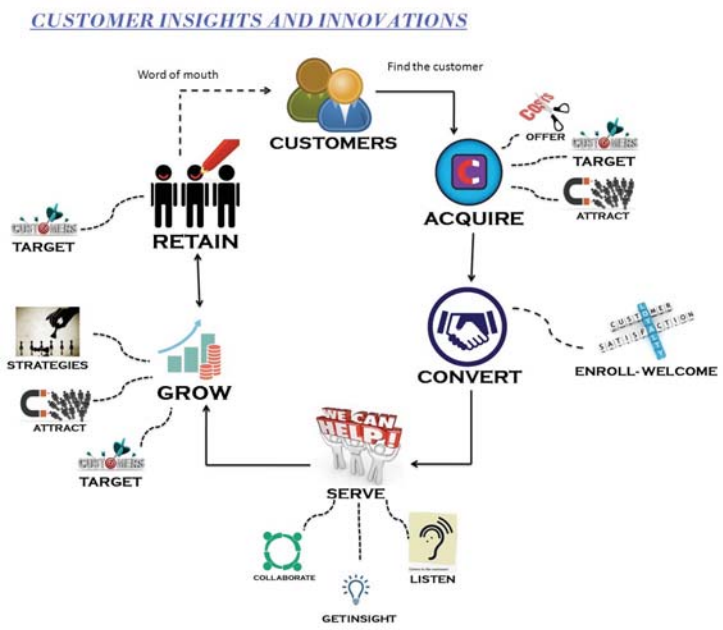


Figure 3.2: Customer centric innovation

Consequently, the association between product strategy and customer satisfaction have been a focal perception for both academics and practitioners, in perspective of the fact that repeat purchase tend to decrease [42].

Customer lifecycle [27] illustrates that in order to make a product success, it is vital to make the customer analytics persuasive across the life cycle of a product (Fig. 3.2).

Gundersen et al. [10] asserts that customer contentment is used up as an evaluative discernment about a particular product or service. In the core of marketing and product strategies customer satisfaction is essential. Therefore many companies are engaged to amend, assess and implement product strategies to increase customer gratification and upgrade share of customers in view of the positive outcome on the economic execution of the company. Angelova and Zekiri [1] point out that customer satisfaction is the outcome considered by consumers that have experienced a company's product strategy that have met their expectations. As part of business strategies, product strategies focus on the market, customer insights, customer satisfaction and their relations [43]. Furthermore the most significant purposes of building a Product strategy is to understand and increase customer satisfaction level which in turn take the firm to achieve competitive advantage.

4 Product strategy, Product Maturity, Competitive advantage

Without an appropriate implementation great strategies are nothing but simply void [35]. In simple words, better to implement efficiently a second class strategy than to devastate a first class strategy by unproductive implementation. Less than half of designed strategies get implemented and every breakdown in the execution/implementation is the breakdown of strategy formulation as suggested by [11],[32],[33]. For many of the electronic companies, creating/developing/inventing products are a focal point by which they adapt and sometimes even transform themselves in transforming the entire circumstances [50]. For example, Hewlett-Packard rehabilitated from an instruments company to a computer company through critical market analysis and new product development to achieve competitive advantage. Similarly, Intel transformed from a memory company to a microprocessor firm through product development [3]. Thus in the face of intense competitive advantage, a rapid technology advancement and customers' growing expectations, product innovation with a regulated critical market analysis and with a well-planned product strategy is the primary way in which firms actually adapt.

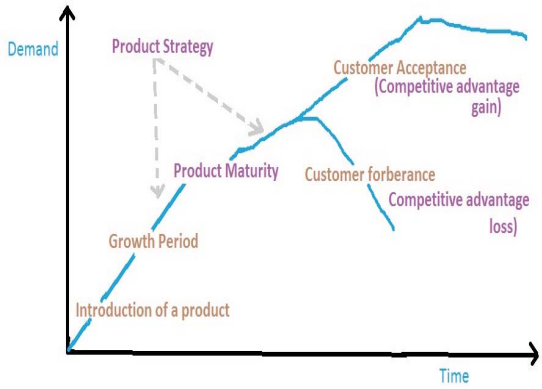


Fig: 4.1: Product strategy-Product Maturity and Competitive advantage

In recent years, fast adaptation has become a remarkable strategic competence for many organisations [7], [44]. Not surprisingly then, this similar theme of fast pace has become crucial in product innovation towards the competitive advantage The insight, known as Moore's Law, became the outstanding principle for the electronics industry, and a driving paradigm for innovation. As a co-founder, Gordon developed the path for Intel to generate the best ever faster, smaller, more reasonably priced transistors that drive our modern tools and equipments. Even half a century later, the enduring impact and remuneration are felt in many ways [19].

4.1 Companies losing their edge

Most companies strive hard to innovate new products to be successful in the competitive market and the few that do find it daunting to stay there. The dissimilarities can be interpreted in relation to the unsuccessful situation and the final decisions can be drawn [4]. History has shown that software projects are highly likely to be successful if they are extremely focused and built upon well-understood reliable technology [14]. For example, [8] tells us that "projects are unsuccessful too often just because the project scope was not completely acceptable and/or user requirements are not fully understood." [17] Tells us that "MIS projects and related procurements take place in a circumstance characterized by the following: Weak management progression and an enticement system that motivates overly optimistic quotients of the benefits that can be achieved from doing the project." [24] Proclaims that the main reason for a failure of project is the highly because of the high user expectations. [16] Tells that because of the lack of alignment between IT departments and business users a project tend to fail.

4.2 Breakdown in Innovation Management

The Innovation management breakdown is considered the second most frequent cause of development progression: some continual issues in managing the internal business processes for validating existing current products and services and developing the new ones [2]. The innovation breakdown is attributed where the revenue growth stalls, the issues are definitely not centred on individual product launch failures; given that most large organisations depend on business models that have boomed to develop chronological product innovations, when things go off beam here—at the heart of these organisations' most vital business process—exceptionally serious, multiyear issues result [23].

5 Survey and Framework Design

5.1 Survey on Usability dimensions and Principles

The questionnaire was designed and hosted online using "Smart-Survey", which provides both free and paid online tools and services for designing and hosting the survey questionnaire online. 12 interviews were conducted and 225 survey data were collected to determine several factors that contribute to successful product strategy development. The 12 interviewee are from various electronic companies with over 10 years of working experience with various product development departments and big data analytics. The aim of this research survey is to get the insight of the capability of big data in the electronic sector, product strategy development, and the competitive advantage from the company perspective. These factors are then collected and presented in the form of a framework.

5.2 Factors affecting successful product strategy

Failure of a product (Fig 5.1) is determined by various factors. These include: launching a product at the wrong time, no uniqueness or differentiation, poor marketing strategy, weak product positioning, misleading advertising, poor pricing strategy, weak product scope and design, flawed market analysis, insufficient quality, outdated technology, poor MDS (Marketed, Delivered, Serviced) and very weak product positioning.

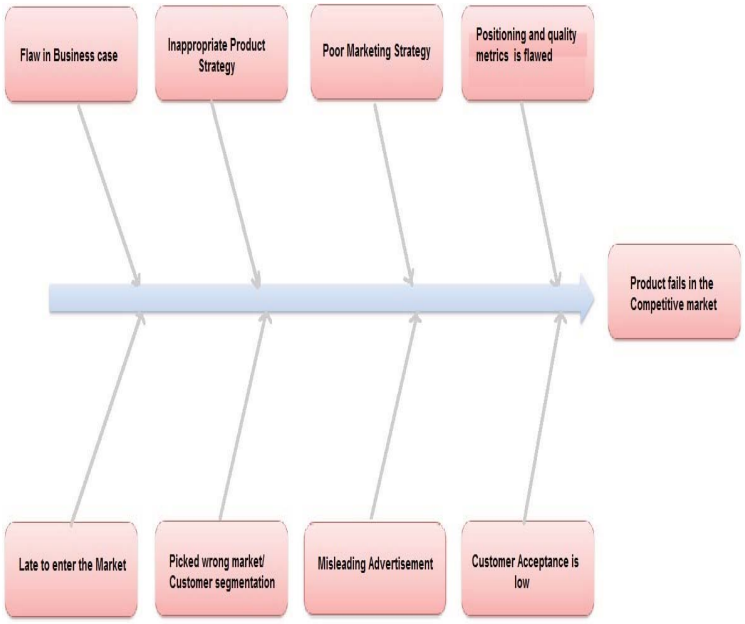


Figure 5.1: Reasons for product failure-Ishikawa diagram

The multiple convergent and the conceptual idea of parallel processing of products result in product failure [48].

Product differentiating strategy and cost leadership are the critical success factors for a product to master better competitiveness and achieve competitive advantage [37]. The objective of differentiation is to develop a position that potential customers see as unique.

5.3 Customer insights for deriving product strategy

Finding the customer touch-points from social analytics big data strategy. The list of customer touch points can vary depending on the business segment. Understanding the customer segments before and after product development as well as before, during and after purchase guides to know the customer touch-points. Formulating customer insights can happen with : 1. Gathering data (social media, transactional

systems, call records, marketing emails, service and support team) 2. Managing the data. 3. Converting the data into insights, 4. Translating insights into successful frontline action.

Knowing the customer lifecycle journey is vital and one of the critical success factors for a company to develop a new product to win the competitive advantage. A company can also analyse how effective the customer touch-points are for dynamic behaviour and moulding attitudes with respect to the touch-points of the competitors [22]. However, it must also be noted that all market segments are unique and hence, for example, Samsung cannot adopt a one size fits all strategy and instead, must approach each market differently. And hence Samsung derived its own product strategy- the Scattershot strategy.

5.4 Success factors for new product development and competitive advantage

Figure 5.2. shows the proposed categorisation of the identified success factors for new product development or product innovation and success factors for competitive advantage. Many researchers have acknowledged the critical success factors in product innovation. Here four main dimensions namely strategic, development (process) ,market, and organizational factors are presumed metrics to quantify the product level success resulting in competitive advantage.

Success factors	Strategic factors	Market factors	Process factors	Organizational factors
Success factors for New Product Development	Innovation strategy and competitive environment	Product Commercialization	Stage-gate process (milestones, checkpoints, stop/go decisions)	Good senior management
	Culture and behavior of Organization	Commercialization measured in terms of sales, distribution and promotion	Project methodologies (Total design,cycle-time excellence and phased development)	Organizational flexibility
	Product strategy portfolio management	Market analysis, Competitor analysis	Continuous assessment	Innovation management and Resources
	Investment in Research and Development	Customer integration and evaluation for robustness	Project management criteria (project efficiency, collaboration tools and communications)	Product champions (highly skilled and effective product development team)
Success factors for Competitive Advantage	Actual product performance (Robust, economic easy to use)	Market synergy/ uniqueness of product	New service development (after sales offerings)	Product differentiation and simultaneous development activities.
	Perception of product (Brand image, product positioning)	Customer insights	Invalidate customer research and competitors progress	Risk management
	Low cost operations (Considering location and buying power)	Product launch effectiveness	Cannibalization or Cross contamination to retain customers	Technology advancement
	Flexibility (Developing customized solutions)	Diagnose potential opportunities	LEAN thinking and sustainability	Change management/ adaptability

Figure 5.2: Success factors for product strategy

Cooper constructs the factors for new product performance NewProductDevelopment(NPD) in order of consequence as: NPD process, NPD strategy, organisation, and culture and management assurance. Cooper's ethics is Circumstantiated as being 'techno-centrism' in nature and declined to recognize the role of knowledge and other non-technical aspect of innovation [25]. The existence of new product development strategy is undoubtedly considered as the most important sign of a successful new product development [5].

5.5 "4-Level Venn" Business Framework

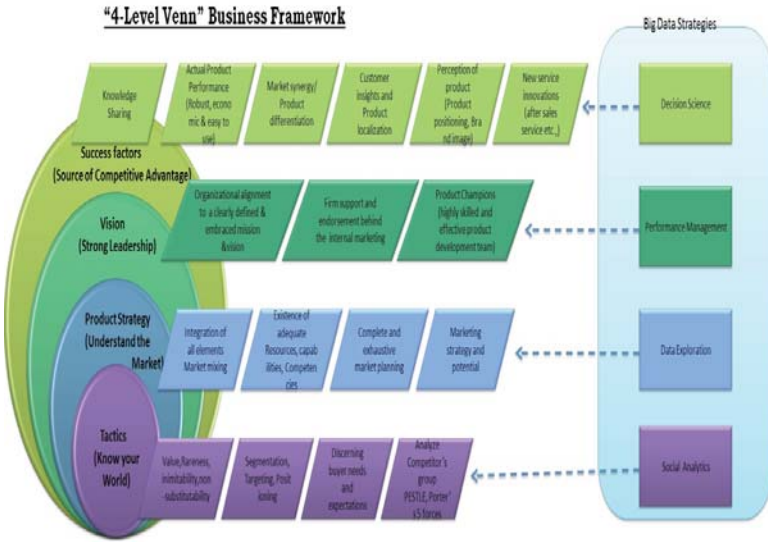


Figure: 5.3: 4-Level Venn Product Strategy Framework

Figure 5.3 proposes the "4-Level Venn" product strategy framework. The figure shows the key areas to concentrate on 'before', 'during' and 'after' a product development. The key areas are derived incorporating big data strategies. This framework would evoke the possibility of a product to achieve competitive advantage where big data analytics plays a major role by identifying the market trends, hidden patterns, customer preferences, unknown correlations, and other useful business information. The big data is capable of measuring both the transactional and non-transactional data and involved in the derivation of the key factors for New Product Development. The type of innovation makes the differences in any product launch When a product launch fails, it is not the product that fails but the management. It is important to be objective for a successful product launch. Today, big data is big business.

5.6 Stability Strategic Model

Fig.5.4 shows the proposed model which would be successful with the "4-Level Venn" business framework. A product developed with the '4-level Venn' business framework, art of innovation along with consumer economics will result in a high quality product. That Ace product launched in market is highly likely to win the competitive advantage. This model can be implemented througout the new product development but also during any upgrade, cannibalization, augmentation, cross-contamination of the existing product.

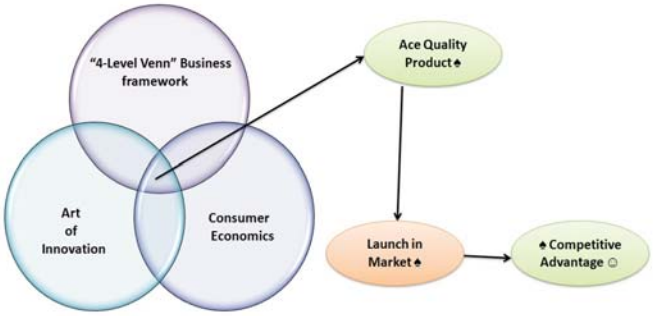


Figure 5.4: Stability Model

A successful business model results from its business level strategies that achieve a competitive advantage over rivals and generate superior performance in an industry [15].

6 Discussion

The article encompasses a detailed description of the product strategy and some other important elements that are linked to the subject matter. Also discussed how customer insight is a key factor for the new product development. Few examples are discussed for the success and failure product strategy. With the responses from the research findings, "4-Level Venn" business framework and Stability Strategic Model is proposed which is the profitable business model focuses on the various aspects of a product (planning, pre-launch and post-launch) to competitive advantage. The proposed model has been evaluated with a Scenario (Sony failed in competitive positioning). Though Sony was doing well in the market with all its new innovations and strong product champions, it failed to look around the work to maintain its competitive advantage. If Sony is to regain its competitive advantage, they need to get back to creating ace innovative products that consumers identify as unique and provides value. This could be attained by knowing its market position and competitors' analysis. Consumers have much more choosing power and competition is fierce [49]. For aiming at the larger profit margins, Sony is suggested to forcefully concentrate on the business segments and strategies development as per the Stability Strategic Model. This would take advantage of their R&D department without waiting for individual consumers to come around to their product innovation. Along these lines, they should seek ways to incentivize their engineers to be exceptionally creative for growth hacking which would skyrocket the product, its value and differentiation than their competitors [16]. Hence the proposed model could be the profitable business model which further develops an appropriate strategy for Sony that allows being distinctive and regaining its competitive advantage.

This research also has analyzed how that strategy draws competitive advantage. The survey findings has certainly enlightened for the investigation of the success and failure factors of a product and why a company fall short to achieve competitive advantage as well as the success factors for competitive advantage. The research findings and the literatures from different sources been the guiding path to achieve the objectives in a well defined sequence of this thesis paper. The critical factors for the failure of the product has been concluded relating the internal (like strategies, resources, organisational ability) and external factors (competitors, market analysis, current trends, customer insights) for the new product development.. To achieve competitive advantage any company should go on in an extra mile to capture the market, and hence the success factors for Competitive Advantage (CA) are drawn clearly. With respect to the research findings it is implicit that the vision of the company should me more strategic and is highly significant in overseeing the company's objectives. Knowing the competitor is the first step in knowing the position of any company. Underestimating the competitor's power is the first step towards losing CA. Customer insights; product differentiation and cost leadership are the most critical metrics to develop a product strategy roadmap which automatically leads to CA. While performing this research, it is undoubtedly nailed that product failure is not failure of a product rather failure of the management as a whole, which poorly perform the analysis of internal factors related to the existing market.

7 References

[1] Angelova, B. and Zekiri, J. (2011).Measuring Customer Satisfaction with Service Quality Using American Customer Satisfaction Model (ACSI Model).IJARBSS, 1(3), p.27

[2] Barnett, H. (1953). Innovation: the basis of cultural change. New York: McGraw-Hill.

[3] Burgelman, R., Christensen, C. and Wheelwright, S. (2009). Strategic management of technology and innovation. Boston: McGraw-Hill Irwin.

[4] Chesbrough, H., Vanhaverbeke, W. and West, J. (2006). Open innovation. Oxford: Oxford University Press.

[5] Cooper, R. (1999). The Invisible Success Factors in Product Innovation. Journal of Product Innovation Management, 16(2), pp.115-133

[6] Dess, G. (2012). Strategic management. New York: McGraw-Hill/Irwin.

[7] Eisenhardt, K. (1989). Building Theories from Case Study Research. Academy of Management Review, 14(4), pp.532-550.

[8] Field, T. (1997). When bad things happen to good projects. CIO magazine, 11,2, pp.54,56.

[9] Fortini-Campbell, L. (1992). Hitting the sweet spot, the consumer insight workbook. Chicago, ILL: Copy Workshop.

[10] Gundersen, M., Heide, M. and Olsson, U. (1996). Hotel Guest Satisfaction among Business Travelers: What Are the Important Factors?.Cornell Hotel and Restaurant Administration Quarterly, 37(2), pp.72-81

[11] Hambrick, D. and Cannella, A. (1989).Strategy Implementation as Substance and Selling.Academy of Management Executive, 3(4), pp.278-285.

[12] Hamel, G. (2000). Leading the revolution. Boston, Mass.: Harvard Business School Press.

[13] Harvard Business Review, (2009). Why Sustainability Is Now the Key Driver of Innovation. [online] Available at: https://hbr.org/2009/09/why-

sustainability-is-now-the-key-driver-of innovation [Accessed 2 Dec. 2015].
[14] Heerkens, G. (2002). Project management. New York: McGraw-Hill.
[15] Hill, C. and Jones, G. (2013). Strategic management. Mason, OH: South-Western, Cengage Learning, p.144.
[16] Hoffman, T. (2003). Value of Project Management Offices Questioned. Computerworld.
[17] Hulme, M. (1997). Procurement Reform and MIS Project Success. International Journal of Purchasing and Materials Management, 33(4), pp.2-7.
[18] Innovation Scientific, (2015). Welcome to a world where innovation is an applied science. [online] Available at: https://innovationscientific.com/ [Accessed 29 Oct. 2015].
[19] Intel, (2015).50 Years of Moore's Law. [online] Available at:http://www.intel.com/content/www/us/en/silicon-innovations/moores-law technology.html [Accessed 1 Dec. 2015].
[20] Investopedia, (2015).What is the growth rate of the electronics sector?. [online] Available at: http://www.investopedia.com/ask/answers/052515/what-growth-rate-electronics sector.asp [Accessed 3 Dec. 2015]
[21] i-SCOOP, (2015). The customer lifecycle journey as looked upon by Oracle - source. [online] Available at: http://www.i-scoop.eu/customer-experience/the-customer-lifecycle-journey as-looked-upon-by-oracle-source/ [Accessed 6 Jan. 2016]
[22] Kelley, T. and Littman, J. (2005).The ten faces of innovation. New York: Currency/Doubleda.
[23] Kmetovicz, R. (1992). New product development. New York: Wiley.
[24] Leicht, M. (1999). Managing User Expectations. University of Missouri St. Louis e-publication.
[25] Leonard, D. and Sensiper, S. (1998). The Role of Tacit Knowledge in Group Innovation. California Management Review, 40(3), pp.112-132.
[26] Lewis, W. (1955). The theory of economic growth. London: Allen &Unwin, p.44.
[27] Management, C. (2015).Customer Lifecycle Management. [online] Pitney Bowes. Available at: http://www.pitneybowes.com/us/customer-engagement-marketing/synchronized communications-execution/customer-lifecycle-management.html [Accessed 7 Dec. 2015].
[28] MaRS. (2016). Product strategy: setting your strategic vision for product offerings | Entrepreneur's Toolkit. [online] Available at: http://www.marsdd.com/mars-library/product-strategy-setting-your-strategic-vision-for-product-offerings/ [Accessed 27 May 2016].
[29] McGrath, M. (2001). Product strategy for high technology companies. New York: McGraw-Hill.
[30] McGrath, M. and McGrath, M. (1996). Setting the PACE in product development a guide to product and cycle time excellence. Boston, MA: Butterworth-Heinemann.
[31] McGrath, R. and MacMillan, I. (2000). The entrepreneurial mindset. Boston, Mass.: Harvard Business School Press.
[32] Miller, D. (2001). Successful change leaders: What makes them? What do they do that is different?.Journal of Change Management, 2(4), pp.359-368.
[33] Mintzberg, H. (1994). The rise and fall of strategic planning. New York: Free Press.
[34] Morishima, M. (1969). Theory of economic growth. Oxford: Clarendon P., p.156
[35] Okumus, F. (1999).A Review of Disparate Approaches to Strategy Implementation in Hospitality Firms.Journal of Hospitality & Tourism Research, 23[1], pp.21-39.
[36] Pollock, K., Jones, C. and Brown, T. (1994).Angler survey methods and their applications in fisheries management. Bethesda, Md.: American Fisheries Society.
[37] Porter, M. (1985). Competitive advantage. New York: Free Press.
[38] Rdniehaus.com, (2015). [online] Available at: http://www.rdniehaus.com/rdn/wpcontent/uploads/2015/07/Economic-Impact-of-STEP-on-the-Electronics Industry.pdf#page=52&zoom=auto,69,389 [Accessed 2 Dec. 2015].
[39] Rogers, E. (1983). Diffusion of innovations. New York: Free Press.
[40] Schendel, D. (2002). Strategic management journal. Chichester: J. Wiley.
[41] Slater, R. and Welch, J. (2004).Jack Welch on leadership. New York: McGraw-Hill.
[42] Smith, A., Bolton, R. and Wagner, J. (1999). A Model of Customer Satisfaction with Service Encounters Involving Failure and Recovery. Journal of Marketing Research, 36(3), p.356.
[43] Smith, W. (1956).Product Differentiation and Market Segmentation as Alternative Marketing Strategies. Journal of Marketing, 21[1], p.3.
[44] Stalk, G. and Hout, T. (1990). Competing against time. New York: Free Press
[45] Steinhardt, G. (2010). The product manager's toolkit. Heidelberg: Springer.
[46] Teece, D. (2009). Dynamic capabilities and strategic management. New York: Oxford University Press.
[47] Ulrich, K. and Eppinger, S. (2012). Product design and development. New York: McGraw-Hill/Irwin.
[48] Urban, G., Hauser, J. and Dholakia, N. (1987). Essentials of new product management. Englewood Cliffs, N.J.: Prentice-Hall.
[49] Williams, C. (2000). Management. Cincinnati, Ohio: South-Western College Pub.
[50] Womack, J., Jones, D. and Roos, D. (1990).The machine that changed the world. New York: Rawson Associates.

Effective Detecting Microblog Spammers Using Big Data Fusion Algorithm

Yang Qiao [1], Huaping Zhang [1*], Yanping Zhao [2], Yu Zhang [1], Yu Min [1]
[1]School of Computer Science, Beijing Inst.of Tech., Haidian, Beijing, China
[2]School of Management & Economics, Beijing Inst.of Tech., Beijing 100081, China
qiaoyang2014@nlpir.org kevinzhang@bit.edu.cn zhaoyp@bit.edu.cn zhangyu2014@nlpir.org
yumin2014@nlpir.org

Abstract - *The Spammers spread rumors and threaten social stability to get profit while resulting a serious impact. Most of the existing studies utilize machine learning techniques to detect spammers. While new trend of the Spammers is they are getting more intelligent too, evolving to evade existing detection features including to avoid being detected by performing like normals. In this paper, we design a Big Data Fusion algorithm to investigate the combination effects of multiple factors in detecting spammers with a series of comprehensive experimental studies. We grab a large amount of microblog data on the Internet and tested for 1.1TB of spammers' data which contains Weibo Microblog message of over 800,000 accounts. The results show that our new algorithm is much more effective than the existing detectors in that it is significantly improved in both the accuracy and the FP-rate by a large margin.*

Keywords: Social media, Spammer detection, Big data fusion algorithm

1 Introduction

Statistics show that the average time spent on social network sites are far more than other sites [1]. Take Twitter as an example, every day there are at least 65 million tweets were sent [2]. Especially in China, the social media like micro blog, Weibo Microblog in Sina.com [3], is also developing much more rapidly. Spammers in social media sites have utilized micro blog as the new platform in a convenient way to get high profits and to achieve illegal purposes [4]. The social media spammers can achieve their malicious goals such as sending rumors [5], spreading malware [5], hosting botnet command and launching other underground illicit activities [6]. These malicious acts even threaten social stability and national security. In February of 2010, thousands of Twitter users, such as the Press Complaints Commission, the BBC correspondent Nick Higham and the Guardian's head of audio Matt Wells, have seen their accounts hijacked after a viral phishing attack [7]. Many researchers along with engineers have devoted themselves to keep social media a spam-free online community. The representatives such as Sina Weibo Microblog to provide microblogging zombie clean-up plug-ins [8], Zinman et. al. [9] using the method of Naive Bayesian Model and Neural Networks for spammers detection, and Amleshwaram et. al. [10] using all aspects of the user's features in integrated social network.

However, spammers are evolving to evade existing detectors. Such as spammers will switch IP frequently while reposting to evade the detecting of IP address [11]. Or using tools to 'spin' their tweets so that they can have heterogeneous tweets with the same semantic meaning [12]. What's more, some spammers imitate the behavior of normal to avoid detection.

In this paper, we plan to design new detection features to detect evasive Weibo Microblog spammers through in-depth analysis of the evasion tactics utilized by current spammers. To achieve our research goals, we use blacklist and honeypot [13]to build our dataset.

Our contributions of this paper are as follows:

1) Set up a large Weibo Microblog data set of 1.1TB. Based on the data we set up analytics to counterpart the evasion tactics.
2) We evaluate the detection rates of two existing state-of-the-art solutions on our collected dataset.
3) We design a Big Data Fusion algorithm to investigate the combination effects of multiple factors in detecting spammers. According to our evaluation, while keeping decrease false positive rate, the detection rate significantly increases to at least 80% which are better than the existing methods.

2 Related Work

Generally, users that luring others to click on illegal links, deliberately distorting the facts, spreading advertising on social network are defined as spammers [14].

To identify distinguishable spammer characteristics, Ramchandran et. al. [15] study the network properties of email spam. Their analysis reveals a correlation between

* Huaping Zhang is the Corresponding author. (E-mail: kevinzhang@bit.edu.cn)

spammers and their physical locality (geographical IP or ASN) while the study also highlights BGP hijacking used for spam attacks. [16] is an extension of ideas from [17] where the authors employ supervised learning using network-level features to distinguish spam from ham. As the work in [18] suggests, Twitter based spam differs qualitatively from email spam.

The commonly used method to detect spammers is using machine learning methods [19] [20], such as detecting the release time distribution of the message to find abnormal [21] or use the user relationship of social network in community detection [22]. Most of the existing methods can be divided into 2 categories. As the examples of the first class[23,24,25], they extract the features of the spammers and normals to train machine-learning classifier as the detector of spammers. Based on the profile features, Lee [23] et al. develop machine learning based classifiers for identifying previously unknown spammers with high precision and a low rate of false positives. Benevenuto et al [24] identify a number of characteristics related to tweet content and user social behavior, they used these characteristics as attributes of machine learning process for classifying users as either spammers or non-spammers. Second types of methods such as [26] examine whether the use of URL blacklists would help to significantly stem the spread of Twitter spam. In addition to collecting training data, [23] and [27] also use social honey pot to collect spammers message. We also use a similar approach in this paper to collect spammers message.

3 Big Data Fusion Algorithm

In this part, based on the machine learning techniques to classifying accounts as spammers or regulars we design a big data fusion algorithm using the behaviors, the profile descriptions, and the content of the users from Sina-Weibo Microblog (like Twitter) in China. In the following section, we will describe and explain how we explore them to distinguish the significant features for effective detections in details.

3.1 Feature Set

In this part, we will explore feature set and the reasons for the selections and verify their discrimination ability.

3.1.1 Behavior-feature

Behavior-feature describes the habit of a user using Weibo Microblog. For example, users are accustomed to visiting Weibo Microblog at a specific time every day or maintain a stable posting frequency and so on. We build an auxiliary data set which contains 1000 normal users and 1200 spammers to evaluate the discrimination ability of features.

Posting Frequency: According our big data statistics' investigation of spammer operators, we find the irregular time span of using Weibo Microblog: the spammers are in a very unusual frequency pattern which is distinguishable in line with the user's habits. Due to the spammers in the absence of business they are often in an idle state, or in a very long time interval to posting a Weibo Microblog.

We set up an analytics as post frequency F_{post} of user v, computed by Eq. (1) to calculate the users posts per hour in some pre-assigned time slot (e.g. the last 2 months):

$$F_{post} = \frac{N_{post-dura}(v)}{|T_{dura}|} \quad (1)$$

where T_{dura} donates the time slot that needed to calculate posting frequency, $N_{post-dura}$ donate the number of the Weibo Microblogs posted in the time slot. To better show the distinguishable pattern between normals and spammers we draw the distributions of the F_{post}s in Fig 1.

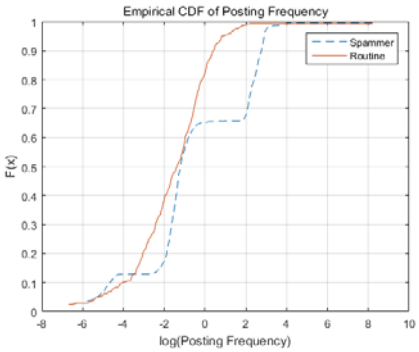


Fig. 1: the distributions of F_{post}

From Fig. 1 we can see that the distribution of spammers is a ladder shape curve(blue) which means spammers' posting frequency distributed in a few lengths of time spans. By comparison, the curve of normals is more smooth in line with the user's human habits. If the spammer still wants to evade, he needs to pay more by posting much more with limited financial support.

Ways to access Weibo Microblog(*Number of ways to post repost*): People have a lot of ways to access Weibo Microblog such as webpage or mobile client. In Fig. 2: we show some common ways

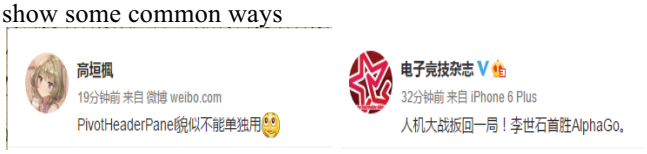


Fig. 2: ways to access Weibo Microblog

We define that *Ways to access Weibo Microblog* as the total number of the ways a user used to post original Weibo Microblog and Number of the ways to repost.

Because spammers need to post a large number of similar Weibo Microblogs in a given time period for some purposes, they need to use API or Weibo Microblog-repeater to release. In contrast, normals have a variety of ways to access Weibo Microblog but not the spammers. We draw the distributions of two:

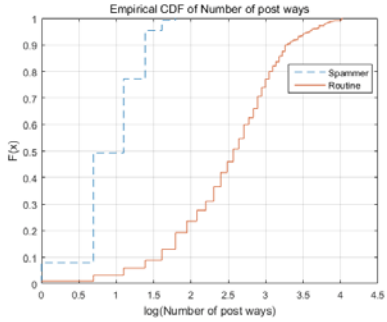


Fig. 3(a): *Ways to access Weibo Microblog(for post)*

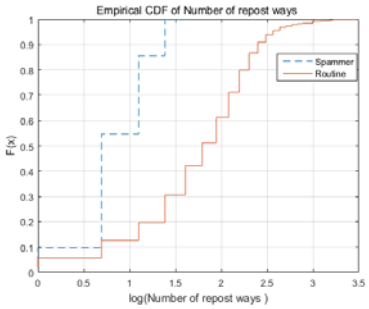


Fig. 3(b): *Ways to access Weibo Microblog(for repost)*

From Fig. 3 we can see that normals use more different ways to access Weibo Microblog than spammers.

Whether to participate in hot Weibo Microblog: Firstly, we define hot Weibo Microblog as having minimum of 100 reposts or comments on record. For spammers, the most common ways they used to make a profit is to repost or comment a target Weibo Microblog much more frequently. We get the evidence through social investigation and find that the purchase fee for a spammer account has the minimum committed consumption standard such as at least 100 times repost! That means spammers participate in hot Weibo Microblog more likely, because it comes to the core of their business, it's hard to avoid.

3.1.2 Content-based feature

In order to avoid the high computational complexity of the semantic analysis of every Weibo Microblog, we selected the following three content-based features as big data analytics.

Ratio of Original and Repost: Firstly, we define a user's total number of Weibo Microblog as N_{all}, the number of original Weibo Microblog is N_{ori}, the number of repost Weibo Microblog is N_{rp}.Then we calculate a user's ratio of Original and that of Repost:

$$R_{ori} = \frac{N_{ori}}{N_{all}}, R_{rp} = \frac{N_{rp}}{N_{all}} \quad (2)$$

Through big data analysis of spammers content we found spammers usually got lower ratio of Original Weibo Microblog and higher one of Repost than normals. As shown in Fig. 4.

We explain two main reasons for this situation: 1) Spammers are often assigned a target Weibo Microblog to be reposted so spammers' reposts are much more. 2)Posting original Weibo Microblogs needs more efforts so that less profit for spammers, and it will increase the probability of being detected if a spammer posts too often.

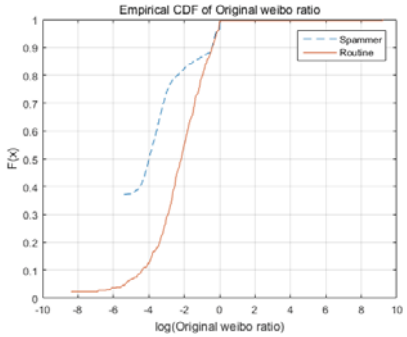


Fig. 4(a): Comparison the Ratio of Original

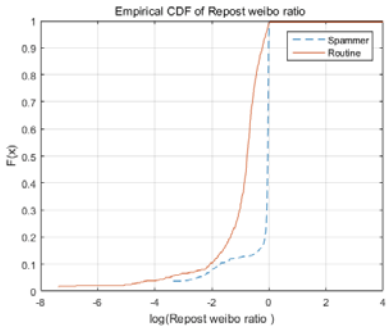


Fig. 4(b): Comparison the ratio of Repost

If spammers try to evade these two detection features, they will have to pay a high price. For example, spammers use websites like spin-bot to convert the target Weibo Microblog to a variety of forms like original Weibo Microblogs, they will pay a high time cost.

Average of @mention: @mention presents the public interaction between Weibo Microblog users, or the multiple layer repost feature of some users. According to our investigation, most of spammers' Weibo Microblogs only contain one @mention, which means their target Weibo Microblog are original. We define user v's Average of @mention as $A_{rp-at}(v)$:

$$A_{rp-at}(v) = \frac{1}{|N_{rp}(v)|} \cdot \sum_{u \in RP(v)} N_{rp-at}(u) \quad (3)$$

Where $N_{rp}(v)$ is the total number of user v's reposts, $RP(v)$ is the set of user u's repost, N_{rp-at} is the number of @mentions per repost of user v. We also draw the curve of the *Average number of @mention* distribution as Fig. 5 We can see that there are obvious differences in the curve of distributions between normals and spammers. As for evasion tactics, spammers can randomly @someone while reposting to evade detection, but that may also cause accusation and lead to accounts suspended.

3.1.3 Profile-based Feature

Profile features describe the basic user information. Due to the different purposes of using Weibo Microblog account, we choose the following profile-based features.

Total of Fans and Followings: Fans number and Followings number describe a user's popularity or level of attraction, which also show a user's level of activity in Weibo Microblog. We randomly select spammers and normals each of the 1000 people from the annotated corpus. We use the distribution in Fig. 6(a)(b) to show the difference between spammers and normals in this feature.

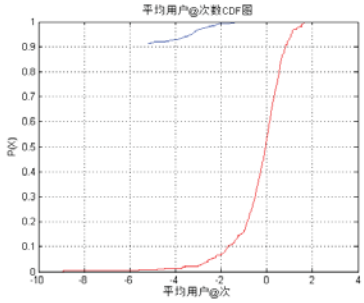


Fig. 5: Average of @ mentions

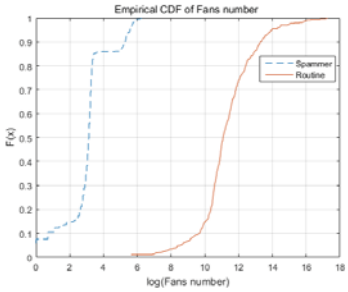


Fig. 6(a): Fans number

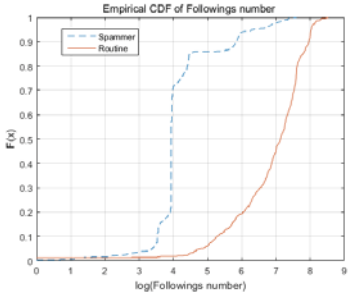


Fig. 6(b): Followings number

We can see that although the owner of spammers can raise the number of fans or followings by follow other spammers, there are still obvious difference between spammers and normals. Normal often get higher number than spammers both in fans number and followings number. This is also a good reflection of the difference between the account activity.

Ratio of fans number and followings number: Firstly, we use N_{fans} and $N_{follows}$ represent the Fans number and Followings number of a Weibo Microblog user. Then a user's ratio of fans number and followings number R_{fafo} can be calculated with the following formula:

$$R_{fafo} = \frac{N_{fans}}{N_{follows}} \quad (4)$$

The same as the last part, we draw the distribution of this feature in Fig. 7. We can see that most of the spammers' Ratio of fans number and followings number are less than 1, and normals are the opposite. We think this is because spammers sometimes are used to follow other account to making profit.

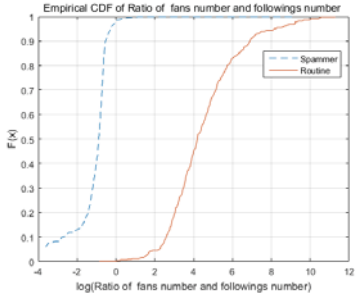


Fig. 7: Ratio of fans number and followings number

3.1.4 Big data Infusion Approach

We setup a big data infusion approach by incorporating our analytics and tactics to the existing state-of-the-art algorithms or methods, to improve the whole classifying or detecting efficacy.

4 Experiment and Evaluation

In this part, we will verify the validity of our new feature set through the experimental method. Based on this, we will analyze the impact of the classification model and the type of feature set on the detection results.

4.1 Experimental Data Preparation

We wrote a Sina Weibo Microblog crawler crawling users information and Weibo Microblog message for our experiment. In the process of collection, we use key words and posting time to identify a arousal event. In this way, we selected 10 arousal event that may contain spammers for crawling. Details about the crawling information can be seen in Table 1.

Table 1: Weibo Microblog accounts crawling information

CATEGORY	AMOUNT
TOTAL OF WEIBO ACCOUNTS	853,041
TOTAL OF WEIBO POSTS	142,304,427
TOTAL OF FANS' ACCOUNTS	130,334,187
TOTAL OF FOLLOWINGS' ACCOUNTS	115,675,345

Then, we need to identify Sina Weibo spammers from our crawled dataset. We randomly selected 10% of the accounts from each weibo arousal event and tag every account manually. What's more, we also bought spammers account on Internet authorities and collected their Sina Weibo information. Finally, we collect 20,000 spammers(15,000 through purchasing) and 20,000 normals to build each of our cross-validation test data set.

4.2 Evaluation of Big data Infused technique in Different Classifier

For the comparison the performance of the big data features infused classifiers, we selected 4 popular machine learning classifiers, including Logistic Regression[28],

SMO[29], AD tree[30] and Random Forest[31]. For each classifier we use 10-fold cross-validation to conduct evaluation.

In order to simulate the real situation, considering spammers detection is a imbalanced classes problem, we randomly selected 700,000 weibo posts from 720 normal accounts and 480 spammer accounts to build experiment dataset. In Fig. 7, we show the detection result of different classifiers:

Fig. 7: Detection result of different classifiers

As shown in Fig. 7 and details in Table 2, the bigdata infused classifier based on Random Forest has the highest detection accuracy of 90.08% which means the best performance in distinguishing spammers from normals. The highest recall rate of 0.931 was obtained based on the SMO method, so we can use SMO to detect more spammers. In addition, the SMO algorithm leads to a much higher FP-rate of 0.279 than the other methods. According to the ROC-Area mesure of overall performance, the detector using Random Forest got the best performance. Since it detects more spammers than other method.

In order to facilitate the comparison, we use Table 2 for further analysis:

Table 2: Detection result of different classifiers

分类器	Accuracy	Recall	F-Measure	Precision	FP-rate	ROC Area
Simple Logistic	84.83%	0.804	0.809	0.814	0.122	0.804
AD Tree	88.08%	0.844	0.861	0.879	0.078	0.953
SMO	80.50%	0.931	0.793	0.690	0.279	0.826
Random Forest	90.08%	0.856	0.874	0.892	0.069	0.962

By observing Table 2 we can find that:

1) *Low model accuracy does not mean that no use.* We can see from the ROC-Area column in Table 2 that detector using SMO get the lowest value. That is to say its classification result is the worst. But when we do not consider the classification accuracy and consider only to find more spammers, wo should also choose SMO. Because it get the highest recall rate.

2) *Decision trees are suitable for our feature sets.* The method based on decision tree (AD Tree and Random Forest) is superior to the other two methods in classification accuracy. So we think decision tree is more suitable for our feature set under normal circumstances.

3) *The effect of the algorithm in class unbalanced problem.* Threshold shift and Composition Technologies are two commonly used methods to improve the accuracy of the class imbalance problem. Simple Logistic and Random Forest belong to these two kinds of methods respectively. Therefore the two methods have higher accuracy.

In order to refine the analysis of the results, we study the cross relationship between the classification results of each classifier, the results are shown in Fig. 8:

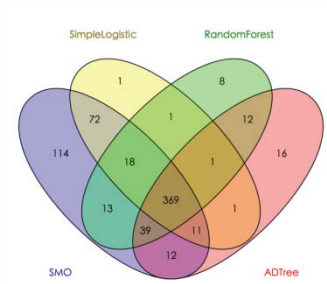


Table 3: Proportion of cross section

Classifier	Coincidence ratio
SMO	56.94%
Simple Logistic	77.84%
Random Forest	80.04%
AD Tree	80.04%

Fig. 8: cross relationship

Fig. 8 shows that the SMO algorithm and the other three algorithm results have obvious differences in the detection. What's more, 80% account are wrong classified by in the 114 account only detected by SMO. We also studied the accounts that were detected by the four methods, only 5 (1.4%)of them were identified as wrong classification. From Table 3 we see that the proportion of the coincidence part of all four algorithm is about 80% except SMO. So Using a variety of methods voting to determine the results is also a method to improve the accuracy of classification.

4.3 Comparison with Existing Strategies

In this part we implement two existing effective detection schemes [32, 33] and compare with our method. we also used the experiment dataset in 4.1. In order to ensure the fairness of the comparison, we choose the big data infused Logistic Regression method which was used both in [32] and [33]. Assuming that our proposed method is A, [32] is B, [33] is C. The comparison results are shown in Fig. 9.

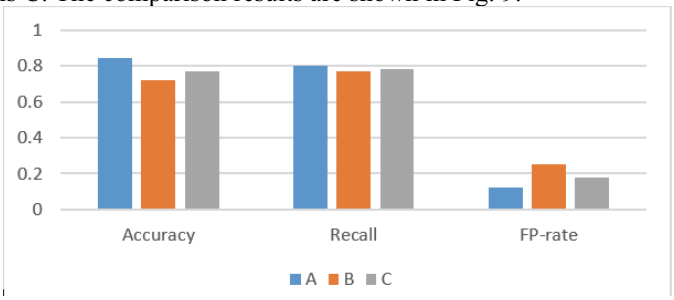


Fig. 9: Comparison of different detection methods

Compared with the method B and method C, our method improves the accuracy of 12% and 7% respectively. And there are also 1% and 4% optimizations on the recall rate. We think what we have in our ascension is:

1) Our feature set is *more abundant*, so higher accuracy can be obtained.

2) Feature *Bilateral Friend Ratio* in B can be evade by spammers by following other spammers easily, so it will lead to wrong classification.

3) Feature *Maximum number of Reposting* in C is an outdated detection feature. Through our investigation, the owner of

spammers control a large number of Weibo Microblog account so a spammer's *Maximum number of Reposting* is 1.A spammer don't need to repost a Weibo Microblog many times.So, this feature doesn't seem to work very well now.

4.4 Analysis and Evaluation of the Feature

In this part, we split and combine the feature subsets to analyze the detection results of different kinds of features. We define Behavior-features as feature set A, Content-features as feature set B, Profile-features as feature set C. In the following experiments, we tested the six feature sets: A, B, C, A+B, B+C, and A+C in Logistic Regression. Results are shown in the Fig. 10.

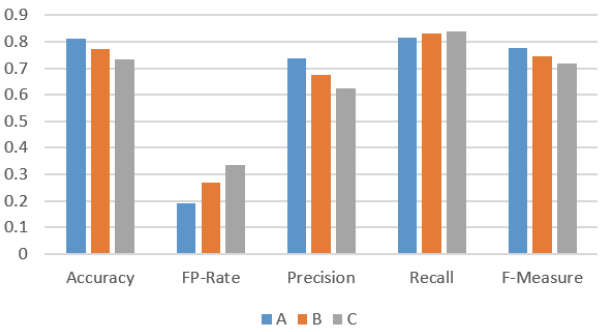


Fig. 10 (a): Comparison of different feature set

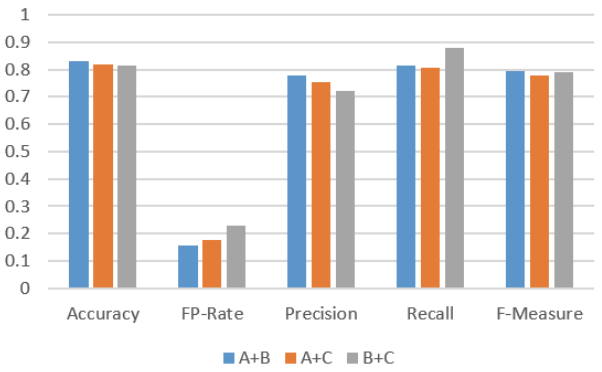


Fig. 10 (b): Comparison of different feature set

Firstly, we analyze the result in Table Fig. 10 (a), it compares the differences between different types of features:

1) The accuracy is decreased with the order of Behavior-features, Content-features, and Profile-features and the magnitude of the decline is even about 4%. On the other hand, it is also the most difficult to avoid the Behavior-features through our investigation. So we think that Behavior-features is more effective than other features in distinguishing between spammers and normals.

2) We can see from the Fig. 10 that the use of Profile-features can get a higher recall rate. But when we improved the recall about 1% using Profile-features only FP-Rate also increased by about 8%. This is the loss outweighs the gain. So we believe Profile-features should be combined with other features.

3) Same as accuracy, the F-Measure is also decreased with the order of Behavior-features, Content-features, and Profile-features. So we think as a whole Behavior-features is superior to Content-features and Profile-features on the detection of spammers.

Then, we analyze the detection result of two kinds of features in Fig. 10 (a). By comparing the Fig. 10 (a) and table 5(b) we can see:

1) By incorporating two kinds of features, the accuracy is increased by about 5%. At the same time, the recall rate is only decreased by 2% with the combination of Behavior-features and Profile-features.

2) By combine different kind of features we also get higher F-Measure, so it is very necessary to use a variety of features.

By comparing Fig. 10 and the result of our whole feature set we find that the accuracy is promoted most by adding Behavior-features followed by Content-features. This is also verified from one side that Behavior-features is superior to Content-features and Profile-features on the detection of spammers.

4.5 The promotion of new features

In order to further verify the correctness of the new big data analytics we propose, we analysed the detection result of the following two feature sets. The first set contains analytics we used that are also used in some of the previous studies including: *Original weibo ratio, Repost weibo ratio, Fans and Followings Ratio* The second set is our whole analytics set. The experimental results are shown in the following Table 4:

Table 4: The promotion of new features

	Without our Features			All Features		
Classifier	Accuracy	FP-Rate	F-Measure	Accuracy	FP-Rate	F-Measure
Simple Logistic	73.75%	0.356	0.728	84.83%	0.122	0.809
AD Tree	83.66%	0.116	0.834	89.08%	0.078	0.861
SMO	63.08%	0.161	0.409	80.50%	0.279	0.793
Random Forest	85.5%	0.095	0.843	90.08%	0.069	0.874

From the table 6, we can see that after adding the new analytics, the accuracy of each algorithm are improved at least 5%. At the same time, the FP-Rate are also improved. This observation implies that the improvement of the detection performance is indeed proportional to our newly designed big data infused analytics rather than the combination of several existing features.

5 Conclusion

In this paper, we design a novel big data infusion algorithm to detect Weibo spammers based on an in-depth analysis of the new evasion tactics utilized by social spammers. We collected a large amount of spammer data on the Internet and do the examination of two state-of-the-art solutions. Through the analysis of those evasion tactics and existing research design a multi-features fusion algorithm to detect spammers. According to our evaluation, while keeping an even lower false positive rate, the detection rate by using our new method increases over 10% than all existing detectors under four different prevalent machine learning classifiers. Finally, depending on the demand, we can choose different classification models or feature subsets in practical application.

6 References

[1] http://www.businessinsider.com

[2] Costolo: Twitter Now Has 190 Million Users Tweeting 65 Million Times A Day.

[3] https://en.wikipedia.org/wiki/Microblogging_in_China

[4] Biao Li, MN Zheng. Study on effect of online water army in the Communication of Network Opinion in the Era of Micro-blog[J]. CJC, 2012(10):30-36

[5] Yang C, Harkreader R C, Gu G. Die Free or Live Hard? Empirical Evaluation and New Design for Fighting Evolving Twitter Spammers[J]. IEEE Transactions on Information Forensics & Security, 2011, 8(8):1280 - 1293.

[6] http://xueshu.baidu.com/

[7] Twitter phishing hack hits BBC, Guardian and cabinet minister.

[8] http://app.Weibo Microblog.com/detail/776yQ

[9] Zinman A, Donath J. Is britney spears spam. In: Proc. of the 4th Conf. on Email and Anti-Spam (CEAS 2007). 2007. 1−10.http://ceas.cc/2007/

[10] Amleshwaram A A, Reddy N, Yadav S, et al. CATS: Characterizing automation of Twitter spammers[C]// Communication Systems and Networks (COMSNETS), 2013 Fifth International Conference on. IEEE, 2013:1-10.

[11] http://tech.qq.com/a/20101126/000325.htm

[12] https://spinbot.com/

[13] http://www.projecthoneypot.org/about_us.php

[14] GF Deng, GW tang. Network communication and social impact studies rumor [J]. Seeker, 2005, (10):88-90.

[15] Tseng CY, Sung PC, Chen MS. Cosdes: A collaborative spam detection system with a novel e-mail abstraction scheme. IEEE Trans. on Knowledge and Data Engineering, 2011,23(5):669−682. [doi: 10.1109/TKDE.2010.147]

[16] Kan Cheng, Liang Chen, Peidong Zhu. Interaction based on method for spam detection in online social networks [J]. Journal on Communications, 2015, 36(7):120-128.

[17] Sathawane KS, Tuteja RR. A robust spam detection system using a collaborative approach with an E-mail abstraction scheme and spam tree data structure. Int'l Journal of Computer Science and Applications, 2013,6(2):293−298.

[18] Hayati P, Chai K, Potdar V, Talevski A. HoneySpam 2.0: Profiling Web spambot behaviour. In: Proc. of the Principles of Practicein Multi-Agent Systems. Heidelberg: Springer-Verlag, 2009. 335−344. [doi: 10.1007/978-3-642-11161-7_23]

[19] https://en.wikipedia.org/wiki/Machine_learning

[20] MO Qian, YANG Ke. Overview of Web Spammer Detection[J]. Ruan Jian Xue Bao/ Journal of Software, 2014, 25(7): 1505-1526.http://www.jos.org.cn/1000-9825/4617.html.

[21] Hayati P, Chai K, Potdar V, Talevski A. Behaviour-Based Web spambot detection by utilising action time and action frequency. In:Taniar D, Gervasi O, Murgante B, Pardede E, Apduhan BO, eds. Proc. of the Computational Science and Its Applications (ICCSA2010). Heidelberg: Springer-Verlag, 2010. 351−360. [doi: 10.1007/978-3-642-12165-4_28]

[22] Hayati P, Potdar V, Talevski A, Chai K. Characterisation of Web spambots using self organising maps. Int ' l Journal of ComputerSystems Science & Engineering, 2011,26(2):87−96.

[23] Lee K, Caverlee J, Webb S. Uncovering social spammers: social honeypots machine learning[C]// Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2010:435-442.

[24] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida. Detecting Spammers on Twitter. InCollaboration, Electronic messaging, Anti-Abuse and Spam Confference (CEAS), 2010.

[25] Wang A H. Don't follow me: Spam detection in Twitter[C]. Security and Cryptography (SECRYPT), Proceedings of the 2010 International Conference on. IEEE, 2010:1 – 10.

[26] Grier C, Thomas K, Paxson V, et al. @spam: the underground on 140 characters or less[C]// In Ccs. ACM, 2010:27-37.

[27] Stringhini G, Kruegel C, Vigna G. Detecting spammers on social networks[C]. Computer Security Applications Conference. 2010:1-9.

[28] https://en.wikipedia.org/wiki/Logistic_regression

[29] https://en.wikipedia.org/wiki/Sequential_minimal_optimization

[30] https://en.wikipedia.org/wiki/Alternating_decision_tree

[31] https://en.wikipedia.org/wiki/Random_forest

[32] Wang K, Xiao Y, Xiao Z. Detection of Internet water army in social network[C]//Proc. of the 2014 Int'l Conf. on Computer, Communications and Information Technology (CCIT 2014). Amsterdam: Atlantis Press. 2014: 189-192.

[33] Lin C, He J, Zhou Y, et al. Analysis and identification of spamming behaviors in sina Weibo Microblog microblog[C]//Proceedings of the 7th Workshop on Social Network Mining and Analysis. ACM, 2013: 5.

A Preliminary Report on Infusing Data-Enabled Active Learning in Undergraduate CS Mathematics and Statistics Courses

Carl Pettis[1], **Rajendran Swamidurai**[1], **and Ash Abebe**[2]
[1]Mathematics and Computer Science, Alabama State University, Montgomery, AL, USA
[2]Statistics, Auburn University, Auburn, AL, USA

Abstract - *This paper presents an experience in designing and implementing a big data based learning model for undergraduate computer science mathematics and statistics courses. Industry demands workers who can retrieve useful information from very complex, unstructured data. The mathematics courses offered for computing majors at universities help students develop the logical thinking and problem-solving skills while the statistics courses introduce students to methods of collection, organization, analysis, and interpretation of data; however, big data analytics requires new mathematical and statistical methods and algorithms developed specifically for use with big data. We describe the design and implementation of infusing big data analytics in existing computer science undergraduate mathematics and statistics courses.*

Keywords: Big data; big data analytics; active learning.

1 Introduction

Recent advances in technology, such as e-commerce, smart phones, and social networking, are generating new types of data on a scale never seen before—a phenomenon known as "big data." [1]. Industry demands workers that can retrieve useful information from very complex, unstructured data. The current undergraduate mathematics courses help students develop the logical thinking and problem-solving skills while the statistics courses introduce students to methods of collection, organization, analysis, and interpretation of data; however, big data analytics requires new mathematical and statistical methods and algorithms developed specifically for use with big data. We strongly believe that equipping students with such skills greatly improves their employability.

The U.S. Bureau of Labor Statistics (BLS), Occupational Outlook Handbook [2] highlights our claim. The report states, "The amount of digitally stored data will increase over the next decade as more people and companies conduct business online and use social media, smartphones, and other mobile devices. As a result, businesses will increasingly need analytics to analyze the large amount of information and data collected. Analyses will help companies improve their business processes, design and develop new products, and even advertise products to potential customers."

A recent survey of senior Fortune 500 and federal agency business and technology leaders by the Harvard Business Review [3] found that "85% of the organizations surveyed had funded Big Data initiatives underway or in the planning stage". The same survey reports that 70% of the respondents plan to hire data scientists, but nearly all report finding employees skilled in big data analytics as challenging to impossible. The nature of academic research is also transforming from model-driven to data driven. For instance, NASA is collaborating with Amazon Web Services Inc. (AWS) to make a large collection of NASA climate and Earth science satellite data publicly available to researchers in an effort to "grow an ecosystem of researchers and developers who can help us solve important environmental research problems" [8]. Higgs bosons were discovered recently by clever algorithms that mined terabytes of data for their signature. While STEM careers in academia and industry are increasingly requiring technical skills for dealing with the analysis of "big data", undergraduate courses in mathematics and statistics fall short of providing adequate training to students in data-driven methods that integrate theory and computation.

This paper presents an experience in infusing, teaching, and assessing big data modules in various undergraduate mathematics and statistics courses that immerses students in real-world big data practices through active learning. Our courses walked students through producing working solutions by having them perform a series of hands-on big data exercises developed specifically to apply cutting-edge industry techniques with each mathematics and statistics course module.

2 Related Works

Universities are waking up to the need for developing skills in big data analytics. Several universities now have graduate level courses focused on big data. Some have masters programs in data science; however, there is very little

evidence of big data concepts being integrated in undergraduate mathematics and statistics courses. Exceptions are the National Science Foundation (NSF) funded EXTREEMS-QED project at the College of William and Mary and a senior big data projects course offered by the Department of Mathematical Sciences of the University of Montana [9].

3 Big Data and Mathematics

3.1 Linear Algebra

Several of the methods used in big data analytics such as feature extraction, clustering, and classification involve the manipulation of large matrices. The important topics in big data analytics should have learn include understanding the relationships between matrix decomposition and principal components analysis for dimension reduction, application of eigenvectors (e.g. in Google's PageRank method), performing multiplication of very large matrices using block decomposition methods, measuring distance between objects represented as vectors (e.g. Jaccard, Hamming, cosine), and understanding the relationship between projections and least squares optimization for regression and clustering.

3.2 Discrete Mathematics

Linked data are usually represented by a graph (vertices and edges). Notions such as centrality, shortest path, and reachability can be derived from the graph using graph analytics. A widely used practical application of large graph analytics is the internet search engine. Topics such as visualizing big data as graphs (e.g. the World Wide Web), computation for strongly connected large graphs (e.g. PageRank for strongly connected graphs), matching in bipartite graphs (e.g. Internet advertising), and social networks and hubs are very important for big data analytics.

3.3 Differential Equations

Differential equations explain the underlying dynamics in spatiotemporal pattern formation and detection, disease modeling, image visualization, processing, and analysis, etc. Topics important for big data analytics include numerical solutions systems of differential equations, nonlinear differential equations and stability, and using observed data to refine solutions.

3.4 Probability and Statistics

Statistical methods make up the majority of methods employed for understanding big data and making inferences. Some topics to be considered for big data analytics are Markov processes and the Markov transition matrix (e.g. Web surfing), correlations in high dimensional data, the Bonferroni Principle, and Monte Carlo simulation.

3.5 Modern Geometry

The use of geometry and topology is an emerging area of research in big data analytics. Currently, the methods are used for exploratory data analysis in high dimensional spaces. When exploring big data analytics in this area, one should learn the topics such as the geometry of data, visualization, and recovering low dimensional structures from high dimensional data.

4 Integrating Big Data Analytics in Existing CS Mathematics Courses

To facilitate active learning, the methods were included in two-part modules. The first part focused on theoretical and conceptual ideas behind the methods under discussion and the second part had hands-on experimentation using simulation experiments as well as real data. The initial set of courses in which we integrated big data analysis methods were chosen using two criteria: suitability of material for pedagogical integration of big data methods and impact on all computing majors. Instructors may eventually choose to expand the integration of methods to other mathematics courses in the future. The initial set of courses included:

- Introduction to Linear Algebra
- Differential Equations
- Probability and Statistics
- Modern Geometry

4.1 Introduction to Linear Algebra

Linear algebra concepts such as feature extraction, clustering, and classification involving the manipulation of large matrices are extensively used in big data analytics; therefore, this is a natural course to start introducing students to big data analytics.

Problem-solving is at the heart of computer science, whether it is games or working with data, we are trying to create tools to help us solve whole categories of problems. We have created a one-week big data module, which introduces the idea of an "algorithm" as a set of instructions used to solve a problem. This sets the context for our discussion of searching and matrix multiplication algorithms, which is used in Google PageRank.

The instructional unit was divided into the following three day lectures:

Day-1: 1) *Lecture:* Introduction to algorithms, 2) *Hands-on activity:* For the hands-on activity the students were grouped into pairs. Each group gets a deck of random number of play cards and is asked to find a specific "key card" in the deck. While one student searches, the other records the algorithm in plain language, and 3) *Assignment:* Rewrite the algorithm they wrote during the hands-on activity

with Pseudocode and implement it in a high-level programming language.

Day-2: 1) *Lecture:* Introduction to Matrix Multiplication, 2) *Hands-on activity:* Each group is to calculate a product of 2 NxN matrices and write down the steps in plain language, and 3) *Assignment:* Rewrite the algorithm they wrote during the hands-on activity with Pseudocode and implement it in a high-level programming language.

Day-3: 1) *Lecture:* Introduction to analysis of algorithms, 2) *Hands-on activity:* Each group is to compute the complexity of their matrix multiplication algorithm created on day 2, and 3) *Assignment:* Complexity analysis of PageRank Algorithm – The Mathematics of Google Search.

4.2 Differential Equations

Differential equations deal with applications making use of differentials. In order to introduce big data in this course, we felt it would be important to discuss the connection between data assimilation and big data. Data assimilation involves comparing a previous model of a state with newly obtained real observations and using this information to update the numerical model of the system.

The following three lectures were added to the existing differential equations course to infuse the big data concept:

- The connection between data assimilation and the discretization of the model state in the first lecture of the Big Data module. After a review of differential equations that cannot be solved by previously introduced methods, we have introduced numerical methods to approximate solutions of those equations. We used Euler's Method to solve linear differential equations.
- In the second lecture, we used the MatLab software to solve systems of ODEs (ordinary differential equations), effectively reinforcing the idea that discretization is a first step toward making a model or function suitable for numerical evaluation and implementation on a computer.
- In the third lecture, we introduced the basics of Monte Carlo simulation and as an exercise to teach the students how to draw a random number according to basic distributions using MatLab or another computer program.

4.3 Probability and Statistics

Statistical methods make up the majority of methods employed for understanding big data and making inferences.

We have created the following one-week module to enhance learning and expose students to big data:

- *Lecture:* The instructor presented the class with a formal definition of "Big Data" that best fits a statistical viewpoint. A brief review of the topics covered during the semester that are necessary for an understanding of the big data labs that follow was given. In addition, the students were introduced to the Python open-source statistical program. Python is a general-purpose programming language, and is more flexible and powerful than R, which is commonly used by statisticians for data analysis and modeling. Therefore, Python was selected as the instruction language.

- *First lab module:* In this lab, the main contents included random number generation as well as calculation of probabilities and expectations using Monte Carlo simulation. The lab used both simple and complicated examples. For simple examples, students were asked to compare the results of simulation experiments with the corresponding analytical solutions obtained using hand calculation.

- *Second lab module:* In this lab, the main contents included graphical visualization for some real data. Many datasets are publically available from sites such as kaggle.com and data.gov. Graphical visualization ranges from simple graphics such as histogram, boxplot, and scatterplot to advanced graphics such as PCA projection plots, trellis plots, maps, etc. were used. Students explored some real data using graphics to investigate and discover information from the real data.

- *Take-home project:* Students used simulation examples relevant to the real world including (a) gambling games, (b) biological evolution, (c) finance, (d) social network, (e) forensic science, etc. Depending on the students programming background, some template codes that are amenable to plug-and-play experimentation were provided to facilitate the activity and reduce the effort of writing a program. In this case, students were asked to examine and manipulate the python code.

4.4 Modern Geometry

The following are sketches of a one-week module consisting of outlines for three lectures and assignments for the big data topics in modern geometry.

Lecture-1: *A1-Intuitive introduction to topology and homology:* Define topology in terms of continuous and continuously invertible mappings and illustrate by examples of topologically equivalent spaces. Approach homology in terms of the number of holes of different dimensional spaces and give a formal definition of simplicial homology and a computation. *Assignments:* Visually classify different spaces first topological and then in terms of topology. Then do a

simple calculation of homology using linear algebra over the rational numbers.

A2-Bar codes: Introduce the idea of looking, for finite sets of points, of the geometric set where we expand the points to disks of radius 'r'. Define the image of homology classes under a continuous mapping and give examples of this. Study how its topology and homology changes as we change 'r'. Visually identify which cycles persists. *Assignments:* Given a set of data points and radii visually determine which cycles persist.

A3-Intuitive discussion of big data in general: Give a definition of big data and tell how it is used. Discuss how bar codes in particular might be used to study it. *Assignment:* Write a short report on some aspect of big data.

Lecture-2: *B1-Dealing with pictures on computers:* Show students how to upload pictures from computers and how to write programs to modify these pictures by mathematical transformations. *Assignment:* Take a picture with a cell phone and upload it to a computer. Then write a mathematical program to invert it. Lastly, write a mathematical program to change the colors.

B2-Projective transformations and vision: Review projective geometry and projective transformations and tell what geometric properties are preserved by projection and what change. Also tell how to represent a given 3-D scene, given in 3-dimensional coordinates, as it would be viewed from any angle. *Assignment:* Write out the formulas or a program to show how a simple 3-D scene (without overlaps) would be seen or displayed on a screen as viewed from a given angle.

B3-Experimentation with computers: Discuss some other types of transformations such as Mobius transformations, inversions, and conformal mappings. Relate this to the theory of map projections. *Assignment:* Take the picture stored in the computer and transform it by these mappings.

Lecture-3: *C1-General discussion of the problem of reconstructing a scene in 3-dimensions from flat pictures*: Give examples from the Internet where several different views of the same scene are available. Archaeologists use this to reconstruct what ancient buildings may have been like, such as temples. Real estate agents and companies can use it to give a 3-D model of a house they are selling. Tourist bureaus might want to give a visual tour of a city. Architects might want to have a 3-D model of what they are planning. Next discuss Magic Eye pictures (random dot stereograms), in which students can focus at the right distance on a picture of apparently random sets and see a 3-D scene. *Assignment*: Find an example of this 3-D reconstruction on the Internet.

C2-Mathematical discussion of the reconstruction problem in terms of projective geometry: Discuss aspects of the situation such as how a computer might divide the scene into objects (this ties in somewhat to the first module and topology). How could the computer tell that it is looking at the same objects when pictures are taken from different angles?

C3-In class assignments: Suppose Mathematically you are given a set of points in two flat pictures with a labelling of which point corresponds to which and necessary information about the points of view. Students would be asked to reconstruct the 3-D coordinates of each point.

5 Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 1436871. We are thankful for the discussion and contribution to the learning modules provided by the participants of the Big Data Analytics Workshop held at Alabama State University (ASU) on November 13, 2015.

6 Conclusions

We have infused one-week big data modules into three of our existing core undergraduate mathematics and statistics course and evaluated its effectiveness through pre- and post-tests. The modules were taught using examples that were worked through interactively during class. The students then worked on a programming assignment that incorporated the new instructional concept into concepts previously taught. This allowed the instructors to evaluate the students on their performance and to give feedback as to how they might improve. We feel the courses were a moderate success, but indicated there was room for improvement. A formal evaluation of these course modules is underway; we are quite optimistic that the big data based learning model will prove to be an effective approach to reinvigorating mathematics and statistics education for undergraduate computer science students.

7 References

[1] Sara Royster (2013), Working with big data, Occupational Outlook Quarterly, 57, 3, 2-10

[2] Bureau of Labor Statistics, U.S. Department of Labor, Occupational Outlook Handbook, 2014-15 Edition, Mathematicians, on the Internet at http://www.bls.gov/ooh/math/mathematicians.htm

[3] http://blogs.hbr.org/2012/11/the-big-data-talent-gap-no-pan/

[4] Labor Force Characteristics by Race and Ethnicity, 2012, BLS Reports, October 2013, http://www.bls.gov/cps/cpsrace2012.pdf

[5] http://www.aps.org/programs/education/statistics/aamajors.cfm

[6] Hankerson, Darrel; Harris, Greg A. ; Johnson, Peter D., Jr. Introduction to information theory and data compression. Second edition. Chapman & Hall/CRC, Boca Raton, FL, 2003.

[7] http://blogs.hbr.org/2012/11/the-big-data-talent-gap-no-pan/

[8] http://www.nasa.gov/press/2013/november/nasa-brings-earth-science-big-data-to-the-cloud-withamazon-web-services

[9] http://cas.umt.edu/math/

[10] http://www.ibmbigdatahub.com/blog/addressing-big-data-skills-gap

[11] http://www.data.gov

[12] http://aws.amazon.com/datasets

[13] http://figshare.com

[14] http://www.kdnuggets.com/datasets/index.html

The Impact of Macroeconomic Factors on Saudi Stock Market (Tadawul) Prices

Mu'tasem Jarrah[1, 2], **Naomie Salim**[1]
[1]Faculty of Computing, Universiti Teknologi Malaysia, Malaysia
[2]Faculty of Computing & Information Technology, King Abdulaziz University, Saudi Arabia

Abstract *- Financial markets play a crucial role in the foundation of the stable and efficient financial system of an economy. This research paper identifies the factors affecting the performance of the stock market in Saudi Arabia, known as aggregate (Tadawul), while the index is called Tadawul All Stock Index (TASI). The movements in the stock market can be quite volatile and sometimes movements in share prices can seem divorced from economic factors [1]. However, there are certain underlying factors which have a strong influence over the movement of share prices and the stock market in general. In this paper, we discuss the factors affecting the Saudi stock market, where the main factors have been identified, together with their own sub factors.*

It can be discerned that generally, shares will be in greater demand when investors have the prospect of earning more dividends. Therefore, factors which make firms more profitable will tend to cause a rise in stock markets.

Keywords: Tadawul, TASI, Macroeconomic Factors.

1 Introduction

Saudi companies began operating in the mid-1930s, when a new company was founded as the first Arabic automotive stock company in Saudi Arabia, and by 1975 there were about 14 joint-stock companies. This has led to rapid economic growth in the Saudi capital, mergers with foreign banks in the 1970s, and newly established large numbers of companies and banks.

However, the Saudi Stock Exchange remained unofficial until the early eighties of the last century, when the government began its consideration of a regulated market for trading and the creation of the necessary regulations for that purpose. As it was, in 1984, a ministerial committee under the Ministry of Finance and National Economy, the Ministry of Commerce and the Saudi Arabian Monetary Agency was formed in order to organise and develop the market. The Saudi Arabian Monetary Agency is a government agency concerned with organising and monitoring the market until the Capital Market Authority was established in 2003 under the "Capital Market Law", which oversees the regulation and control of the financial market through the instructions and rules designed to protect investors and ensure fairness and efficiency in the market.

There are eight sectors listed on the Saudi stock market: Banks & Financial Services, Petrochemical Industries, Cement, Energy & Utilities, Agriculture & Food Industries, Telecommunication & Information Technology, Hotel & Tourism, Insurance, Industrial Investment, Building & Construction, Real Estate Development, Transport and Media and Publishing.

The sectors mentioned above include a number of companies listed on the Saudi stock market, amounting to 170 different companies which fall in different sectors, and because of the potential profit and loss of the shares of these companies, this highlights the existence of external factors and internal factors that affect returns and stock prices [2].

In this research paper, we will mention factors that have different effects on stock returns and which are divided into main factors (Political, Oil price, Company news and Industry performance, Performance and Investor sentiment and Economic factors), as well as sub factors that will be discussed later with the related details [3].

2 Literature Review

Previous studies. There have been several studies that focused on the factors affecting the stock market prices or index.

Durga, Sultan and Bokkasam (2014) in their research examined the three important factors influencing the returns in the Saudi Stock Exchange (TASI) based on the macroeconomic variables of the Saudi economy. The dependent variable (a factor or reason that is applied to see its impact on the result) taken here is the Saudi index that is Tadawul All Stock Index (TASI). Meanwhile, the three independent variables (a result that is measured by the impact of the independent variable) considered for this study were Oil WTI, Saudi Exports and Price Earnings Ratio. Correlation analysis revealed that Saudi Exports and PE Ratio were found to be highly correlated with the TASI. The researchers were able to confirm that the TASI was positively correlated with the three economic variables considered, i.e., Oil WTI, Saudi Exports and PE ratio. Since the three independent variables were significantly correlated with the dependent variable, the step-wise regression established the significant importance each of these three variables had in predicting the TASI [3].

Next, **Almazari (2014)** investigated the internal factors that affected the profitability of the banking sector. This study's main objective was to compare the profitability of the Saudi

and Jordanian banks by using internal factors for estimation. The researcher concluded on the following: The findings of this study reflected the actual status of the sample banks. Since then, very few empirical studies had been undertaken investigating the characteristics of internal factors affecting the profitability of banks in Saudi Arabia and Jordan [2].

Besides that, **Arouri and Fouquau (2009)** examined the short-run relationships between oil prices and Gulf Cooperation Council (GCC) stock markets. Since GCC countries were major world energy market players, their stock markets might be susceptible to oil price shocks. To account for the fact that stock markets might respond non-linearly to oil price shocks, both linear and non-linear relationships were analysed. The findings illustrated that there were significant links between the two variables in Qatar, Oman and UAE. Thus, stock markets in these countries reacted positively to oil price increases. For Bahrain, Kuwait and Saudi Arabia, it was found that oil price changes did not affect stock market returns. The researchers concluded that: First, the relationships between oil price changes and stock market returns in GCC countries could be expected to vary from one economic sector to another. A sector analysis of this link would be informative. Second, the same approach applied in this article could be used to examine the effects of other energy products, such as natural gas. Third, further research could examine the links of causality binding oil and stock markets in GCC countries and other oil-exporting countries [4].

In addition, **Aurangzeb (2012)** identified the factors affecting the performance of stock markets in South Asia. The data used in this study was collected from the period of 1997 to 2010 of three South Asian countries, i.e., Pakistan, India and Sri Lanka. Regression results indicated that foreign direct investment and exchange rates had significant positive impact on the performance of stock markets in South Asian countries while interest rates had negative and significant impact on the performance of stock markets in South Asia. Results also demonstrated the negative but insignificant impact of inflation on stock market performance in South Asia. It was recommended that in order to take full advantage of the stock market and carry on with the international markets, well managed macroeconomic policies were necessary in which interest rates and inflation rates were thoroughly monitored and there was a need to reduce the value as much as possible. This would give confidence to the investors as well as the industries. It was also recommended that some extra benefits were given to foreign investors as it was observed that the influence of foreign investors was strong in the region [5].

Research conducted by **Al-Abedallat and Al Shabib (2012**) aimed to study the effect of the change in investment and gross domestic product (GDP) on the Amman Stock Exchange Index, through the study of the relationship between the change in the investment and the rate of growth in gross domestic product (GDP) and the movement of the Amman Stock Exchange Index from 1990 - 2009. To test its hypotheses, the study used statistical analysis (SPSS), and chose multiple regression to analyse the relationship between the dependent variable (Amman Stock Exchange Index) and the independent variables (investment and GDP).

The study concluded that there was a relationship between two macroeconomic indicators (investment and GDP) and the Amman Stock Exchange Index, and between each of them separately and the stock index, which meant that the movement of prices in the Amman Stock Exchange was affected by the movement of these two variables, and there was the effect of both variables on the movement of the Amman Stock Exchange Index. The impact of the change in investments was greater than the impact of change in GDP on the Amman Stock Exchange index [9].

3 Problem statement

Many individuals, entrepreneurs, companies and governments are investing large sums of money in the financial markets, nevertheless, the majority of market participants are not well informed about how to deal with the stock market and thus, they have a tendency to act irrationally. The lack of information at the right time and the cost of obtaining new information can cause a state of imbalance in the performance of the market and the performance of investors.

The movements of the stock market are difficult to understand and predict. This creates the need for empirical analysis, which can help in the understanding and prediction of the stock market and to help predict potential stock market prices. Hence, this study attempts to address the gap in the literature through the relationship between the stock prices of analysing and influencing internal and external factors in the Saudi stock market.

4 Data Collection

4.1 Data Collection

The data for this study was collected from 500 reports which were published from September 2015 to February 2016. By using daily data to portray a larger view of the relationship and data filtration criteria, the data covered 150 day observations for each variable. Data regarding TASI and the sectors have been extracted from the official website of the Saudi Stock Exchange (www.tadawul.com.sa) and from the website of Aljazira Capital (http://www.aljaziracapital.com.sa). Aljazira Capital was chosen as a source as it is a Saudi Closed Joint Stock company operating under the regulatory supervision of the Capital Market Authority and it specialises in securities business and providing dealing, underwriting, managing, arranging, advisory and custody services.

The aim of this paper is to investigate the effects of macroeconomic determinants on the performance of the Saudi stock market. It proposes to identify the factors affecting stock market prices through daily, weekly and monthly economic reports that specialise in Saudi financial market affairs analysis.

4.2 Brief Description of Factors

4.2.1 Political

The political factor is a major cause for the stability of the economy in general and the stock prices in the financial market in particular. This is because this factor also affects neighbouring countries. Non-political stability and exposure to political shocks such as demonstrations have confirmed the changes in market volatility as a result of some domestic and international events that have an impact on the domestic economy and the financial market [11].

4.2.2 Oil price

The price of oil and the stock market work in the opposite direction, with an increase in oil price leading to the decrease of the returns in the stock market, and vice-versa [6].

4.2.3 Company news and Industry performance

Many events can cause the price of a stock to rise or fall, from specific news about a company's earnings to a change in how investors feel about the stock market in general. Here are some company specific factors that can affect the share price: News releases on earnings, profits, future estimated earnings, Announcement of dividends, Introduction of a new product, Product recalls, Securing a new large contract, Employee layoffs, Anticipated takeover or merger, Change of management and Accounting errors or scandals [11].

4.2.4 Performance and investor sentiment

Investor sentiment or confidence can cause the market to go up or down, which can cause stock prices to rise or fall. The general direction that the stock market takes can affect the value of a stock: [7]

- Bull market – a strong stock market where stock prices are rising and investor confidence is growing. It is generally tied to economic recovery or an economic boom, as well as investor optimism.

- Bear market – a weak market where stock prices are falling and investor confidence is fading. It often happens when an economy is in recession and unemployment is high, with rising prices.

4.2.5 Economic factors

4.2.5.1 Growing, Shrinking and Shocks

In summary, previous empirical research has suggested a connection between stock market development and economic growth, but this is far from definitive. Although the relationship postulated is a causal one, most empirical studies have addressed causality obliquely, if at all. Moreover, most studies have not adequately dealt with the fact that efficient markets should incorporate expected future growth in current period prices.

4.2.5.2 Interest rates

The bank can raise or lower interest rates to stabilise or stimulate the economy, otherwise known as monetary policy. If a company borrows money to expand and improve its business, higher interest rates will affect the cost of its debt. Moreover, this can also reduce company profits and the dividends it pays its shareholders. As a result, its share price may drop. Clearly, in times of higher interest rates, investments that pay interest tend to be more attractive to investors than stocks [10].

4.2.5.3 Economic outlook

If it looks like the economy is going to expand, stock prices may rise. Investors may buy more stocks thinking they will see future profits and higher stock prices. If the economic outlook is uncertain, investors may reduce their buying or start selling [8].

4.2.5.4 Inflation

Inflation means higher consumer prices and this often slows sales and reduces profits. Higher prices will also often lead to higher interest rates. For example, the Bank of Samba may raise interest rates to slow down inflation. These changes will tend to bring down stock prices. Commodities, however, may do better with inflation, so their prices may rise 10].

4.2.5.5 Deflation

Falling prices tend to mean lower profits for companies and decreased economic activity. Stock prices may go down, and investors may start selling their shares and move to fixed-income investments like bonds. Interest rates may be lowered to encourage people to borrow more. Whereby the goal is to increase spending and economic activity.

4.2.5.6 Changes in economic policy

If a new government comes into power, it may decide to make new policies. Sometimes these changes can be seen as good for business, and sometimes not. In addition, they may lead to changes in inflation and interest rates, which in turn may affect stock prices.

4.2.5.7 Value of the US dollar (increase or decrease)

Many Saudi companies sell products to buyers in other countries. If the US dollar rises, their customers will have to spend more to buy US goods. This can drive down sales, which in turn can lead to lower stock prices. When the price of the US dollar falls, it makes it cheaper for others to buy our products, therefore making it possible for stock prices to rise [11].

5 Data Analysis

The researchers in this study have analysed Aljazira Capital reports which are divided into three sections: daily, weekly and monthly, where they contain all the events and news related to companies listed divided into 13 sectors on the Saudi stock market (Tadawul), and subsequently connected them with the stock of the company. This enabled

the researchers to determine the impact of the news or event on the share price immediately, through events mentioned report of the Aljazira Capital analysis, and observation the impact of those events on the behavior of the stock market through the size of the buying and selling as well as the level of the change on the share price on that day.

The table below (Table 1) shows a range of events as a sample, which included reports and their impact on the various sectors of the Saudi stock market (Tadawul).

Table 1: Sample of events' impact on sectors

Event	Effect			sector
	Up	Down	Normal	
Occurrence of limited fire in one of the reservoirs for oily waste		yes		Petrochemical Industries
Cash dividend to shareholders of company	yes			Industries
Increase company's capital through issuance of bonus shares to company shareholders	yes			Industries
Negotiate for acquisition of foreign partner's share			yes	Building & Construction
Entering third partner as an investor	yes			Petrochemical Industries
Transformation from limited liability company to closed joint stock company	yes			Retail
Signature to get roundabout loan agreement		yes		Petrochemical Industries
Signed agreement with global investment company for organisational restructuring	yes			Industrial Investment
Recommendation of Board of Directors to increase capital through share offering	yes			Petrochemical Industries
Use of funds obtained from issuance of shares from IPO Subscription Process	yes			Insurance
Expiration of Memorandum of Understanding and signing of new Memorandum of Understanding	yes			Transport
Stop production in order to conduct periodic maintenance	yes			Petrochemical Industries
Signing of Memorandum of Understanding to renting a piece of land and prefabricated buildings	yes			Agriculture & Food
Signing of sale of exclusive franchise agreement	yes			Retail
Experimental operation of plant equipment before starting actual production	yes			Industrial
Signing of agreement on the purchase and modernisation of a fleet of buses	yes			Transport
Signed agreement to provide automated insurance system services	yes			Insurance
Modified legal form of company and converted to limited liability company		yes		Retail
Suspension to request rehabilitation of company because of failure to meet eligibility requirements	yes			Insurance
Transmission company's main administration offices from old		yes		Industrial

place to new address				
Signing of long-term loan	yes			Industrial Investment
Announcement of company's strategy for the next five years	yes			Health
Signing of credit facility agreement to ensure compatible with provisions of Islamic Sharia law		yes		Industrial
Signing contract to buy factory for production of medical supplies	yes			Industrial
Signing of Memorandum of Understanding for sale of land in an industrial area	yes			Banks
Stop production of ovens (for clinker - essential ingredient for cement) on temporary basis until market conditions improve		yes		Cement
Fire in rubber belt conveyor in limestone reservoir		yes		Cement
Postponement of replacement of old cement mills due to market conditions		yes		Cement
Memorandum of Understanding to buy stake in one of the founding companies of the manufacturing industries		yes		Power
Renewal of agency's contract to exploit franchise brand		yes		Transport
Occurrence to seek electrical circuit breaker in the room.	yes			Retail

Receive payment under account of insurance company as result of insured accident	yes			Petrochemical Industries
Reschedule instalments for outstanding loan balance	yes			Petrochemical Industries
Signed agreement to acquire partner owned by another	yes			Petrochemical Industries
Signing of credit facility agreement compatible with the provisions of Islamic Sharia law	yes			Banks
To approve company's request to renew rehabilitated to provide health insurance services	yes			Insurance
Buy all shares of another partner		yes		Industrial
Announcement of completion rate achieved in company's projects	yes			Cement
Announcement of accumulated losses		yes		Insurance
Announcement of decreases in net losses		yes		Industries
Announcement of earnings before Zakat	yes			Insurance
Signed Memorandum of Understanding	yes			Building & Construction
Signature for Islamic financial loan	yes			Hotel & Tourism
Negotiate for acquisition of foreign partner's share	yes			Hotel & Tourism
Announcement of sale of part of the shares owned by the company	yes			Transport Sector

Announcement of purchase of part of shares owned by partner and attach to company	yes			Transport Sector

5 Conclusions

Through the results that have been obtained from this research, it can be confirmed that there is a direct effect of the relationship between the direction of the stock price rise or fall, and all the events and news that exposed the facility.

Based on that and through follow-up news and events, an investor can decide to purchase or sell stocks at the right time.

6 References

[1] L. Kalyanaraman1 & B.Al Tuwajri1 "Macroeconomic Forces and Stock Prices: Some Empirical Evidence from Saudi Arabia"; International Journal of Financial Research, Sciedu Press, Vol. 5, Issue 1, (81—92), Jan. 2014.

[2] Ahmad Aref Almazaril "Impact of Internal Factors on Bank Profitability: Comparative Study between Saudi Arabia and Jordan"; Journal of Applied Finance & Banking, (name of publisher of the journal), Vol. 4, Issue 1, (125—140), 2014.

[3] Durga Prasad Samontaray, Sultan Nugali & Bokkasam Sasidhar "A Study of the Effect of Macroeconomic Variables on Stock Market: Saudi Perspective"; Research in Applied Economics (name of publisher of the journal), Vol. 6, Issue 2, (47—72), Sep, 2014.

[4] Mohamed El Hedi Arouri & Julien Fouquau "On the short term influence of oil price changes on stock markets in GCC countries: linear and nonlinear analyses"; Journal of Economics Bulletin, Vol. 2, Issue 9, (795-804), 2009.

[5] Aurangzeb "Factors Affecting Performance of Stock Market: Evidence from South Asian Countries"; International Journal of Academic Research in Business and Social Sciences, Vol. 2, Issue 9, (1-15), Sep. 2012.

[6] Muazu Ibrahim & Alhassan Musah. "An Econometric Analysis of the Impact of Macroeconomic Fundamentals on Stock Market Returns in Ghana"; International Journal of Financial Research (name of publisher of the journal), Vol. 5, Issue 4, (120—127), April, 2014.

[7] Kavitha S, Raja Vadhana P & Nivi A.N. "Big Data Analytics in Financial Market"; International Journal of Research in Engineering and Technology (name of publisher of the journal), Vol. 4, Issue 2, (120—127), Feb, 2015.

[8] Sattam Allahawiah & Sameer Al Amro "Factors affecting Stock Market Prices in Amman Stock Exchange: A Survey Study"; European Journal of Business and Management (name of publisher of the journal), Vol. 4, Issue 8, (236—245), 2012.

[9] A. Al-Abedallat & D. Al Shabib "Impact of the investment and gross domestic product on the Amman Stock Exchange index"; Investment Management and Financial Innovations, (name of publisher of the journal), Vol. 6, Issue 2, 2012.

[10] Sezgin Acikalin, Rafet Aktas & Seyfettin Unal "Relationships between stock markets and macroeconomic variables: an empirical analysis of the Istanbul Stock Exchange"; Investment Management and Financial Innovations, (name of publisher of the journal), Vol. 5, Issue 1, (8—16), 2008.

[11] Shawkat M. Hammoudeha, Yuan Yuana & Michael McAleer "Shock and volatility spillovers among equity sectors of the Gulf Arab stock markets"; The Quarterly Review of Economics and Finance 49; journal homepage: www.elsevier.com/locate/qref (829–842), 2009.

Bigdata platform based approach for defending against DDoS

Yoon Joo Chae, Nikitha Johnsirani Venkatesan, and Dong Ryeol Shin
{chaeyj, nikithajv, drshin}@skku.edu
Information & Communication Engineering, Sungkyunkwan University, Suwon, Gyeonggi Do, South Korea

Abstract - *Distributed denial-of-service (DDoS) is a rapidly growing problem. From the first known attack in 1999 to the highly publicized Operation Ababil, the DDoS attacks have a history of flooding the victim network with an enormous number of packets, hence exhausting the resources and preventing the legitimate users to access them. The variety of attacks are overwhelming now a days. However, the existing method for managing a number of computer with respect to an external invasion, such as hacking, is not strong. We look into the real time attacks of DDoS. Using a big data platform real-time processing capacity, we propose a way to manage a large number of individual computer log collecting, processing via the analysis at a public organization. We propose a way of former prediction by using system logs analysis. Therefore, to identify the malicious data, processing, analysis, are done using Logstash and Spark respectively.*

Keywords: DDoS, Spark, Logstash, Elastic Search and HUE

1 Introduction

DDoS attacks are one of the biggest challenge faced by security researchers on International scale. The losses caused by the security breach can cost up to several billions of dollars [10]. In a public sector, a hacker has managed to target multiple systems from a centralized computer. The existing defensive mechanisms for predicting the cyber-attacks are still in the basic level. Now a days hackers use sophisticated hacking system to invade the authorized systems of the users. The current security systems like even antivirus are failure in detecting the intrusion. Sometimes, outside intrusion such as downloading data from USB device is unavoidable. But, still using proper analysis of the data some of the malicious attacks can be overcome. Currently the attacks are of two types [11]. The first one is to send the malicious packet which is injected with virus called vulnerability attack as a running application. The second one is traditional method of draining the resources of the victim like input-output bandwidth, database bandwidth, CPU memory and the like.

Traditional defensive mechanisms try to rectify the virus after detecting it. For example, anti-virus which is installed in our system detect the virus after the virus got executed in the system. In this paper, using big data platform, we propose a way to detect the malicious virus before even entering the system. The framework manages the system logs from large number of computers thus eliminating the malicious attacks. The system logs are collected and processed and finally analyzed with the help of public organization. The real-time response measures in the cluster structure predict accidents in advance to allow flexible management about the incident.

The rest of the paper is organized as follows: Section 2 describes the DDoS architecture. Section 3 describes the related work done to defend the DDoS attacks and the real-time attacks happened in the organizations. We outlined the architecture and the experimental results in section 4. We also discussed the future work regarding this framework in section 5. Finally, section 6 concludes the paper.

2 DDoS Architecture:

A denial-of-service (DoS) is a type of attack in which the hackers prevent the authorized users from accessing the service [5]. A distributed denial-of-service (DDoS) is a kind of DoS attack. The difference between DoS and DDoS is that, DDoS involves multiple systems to target a single computer whereas DoS attack uses one system. The word "distributed" implies that an attack is focused within a team of disruptors who hack with a common goal of preventing the webservers from working normally. Normally, the source of attack is more than one, likely to be in thousands of unique IP address.

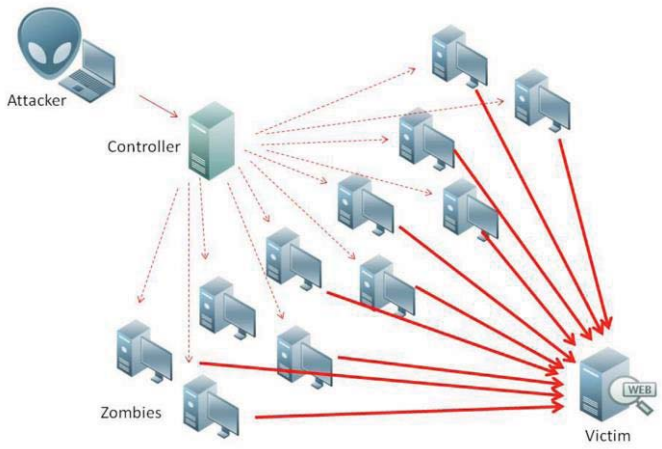


Figure 1. Overview of DDoS

The overview of the DDoS attack is depicted in figure 1. The attacker sets up the hierarchical attack architecture. Initially, an attacker selects one controller which is vulnerable to securities. After that, the zombies are selected following the same procedure as of the controller. But, the zombies are indirectly handled by the attacker through the controller. The zombies, otherwise called as selected agents perform the DDoS attacks by sending enormous amount of malicious traffic to the targeted system [7]. The controller and the zombies are commonly located in the external networks. Once the hacker successfully selected the controller and the zombies, he/she starts controlling the communication among the

controller, zombies and the targeted system. After completing the selection and the communication process, the attacker starts launching the DDoS attacks on the victim simultaneously. Normally, the communication between the controller, zombies and the victim will be encrypted for the safe information exchange.

There are many kinds of DDoS attacks [6]: Traffic attacks, Bandwidth attacks, Application attacks and the like. The DDoS attacks can be done by various techniques which are listed as follows [5]:

- Internet Control Message Protocol (ICMP) flood
- Teardrop attacks
- Peer-to-peer attacks
- Permanent denial-of-service attacks
- Application-layer floods.
- Nuke
- Slow Read attack
- Telephony denial-of-service(TDoS)

The above attacks are detailed in [5]. The main motivation of DDoS attacks are financial frauds, competitive rivalry, ideological hacktivism and extortion.
Virtually, all kind of resources that are connected to the Internet are vulnerable to DDoS attacks. Many existing systems are not capable of protecting the systems against these DDoS attacks. To protect the Internet, most of the organizations use security tools such as, Internet Service Providers (ISP), firewalls and secure web gateways. Although all these security tools act as first layer protection for the basic threats, they cannot give protection from advanced threats

3 Related works:

3.1 Intrusion practices in Institutions:

This section lists the known cyber-attacks on organizations which are made known to the public. The fields that are very often prone to DDoS attacks are listed below:

- Retail
- Communication
- Technology
- Health-care

We will discuss the real time scenarios of the cyber-attacks in the respective fields [8].

3.1.1 Retail:

In December 2013, 70 million individual's personal information was stolen along with their credit and debit card details from an organization named Target.
In October 2013, more than 9,000 credits cards were used fraudulently following the attack in Neiman Marcus. The employees who worked in the company were not even able to detect the attack for months because of the hacker's code.
In January 2014, 2.6 million customers in Michael's payment card were affected. The hackers targeted the POS system to gain access.
In May 2014, the contact and log-in information of 233 million customers was hacked in eBay along with their employee's details. Later, eBay requested all of its customers to change their password.
In September 2014, again 868,000 credit and debit card information was hacked from Goodwill Industries International. Malware infected the chain store through infected third party vendors.
In March 2012, the banks in South Korea named Jeju Bank, NongHyup, Shinhan Bank reported that Internet Banking servers were blocked temporarily. Some of the branches told that the computers were infected with virus and many important files had been erased [9].

3.1.2 Communication:

In January 2014, Yahoo mail reported that around 273 million accounts were hacked.
In April 2014, AT&T was hacked for two weeks completely from inside by someone who accessed all the users and social security information.
In June 2014, Feedly reported that 15 million users were temporarily affected by three DDoS attacks.
In September 2014, around 5 million usernames and passwords were hacked from Gmail users and they were released on Russian site.
In October 2014, the pictures of 200,000 users were hacked from snapsave. Snapsave is a third-party app for saving photos from Snapchat.

3.1.3 Technology

In June 2014, 100 million users from Evernote faced DDoS attacks.
In September 2014, Hackers used third party applications to access Apple user's online data storage. This attack leads to posting the celebrities personal pictures online.

3.1.4 Health care

In June 2014, credit and debit card information from Chang's restaurant was hacked and reported that they all sold online.
In August 2014, the personal data of 4.5 million patients were hacked from Community Health Services (CHS). CHS made a statement that all the patients who visited any of its branches might have their information hacked. They claim that malware used in the attack originated from China. The FBI warns eventhe other health care information might have been stolen.

4 System Architecture:

Figure 2 shows the real-time computer configuration Central Management System. We use the Logstash to collect system logs from Windows. The Logstash is used in Windows Management Instrumentation (WMI) to gather logs. Logstash is an open source platform which can process any data from

any source. It will centralize the data processing of all types and extend to custom log formats

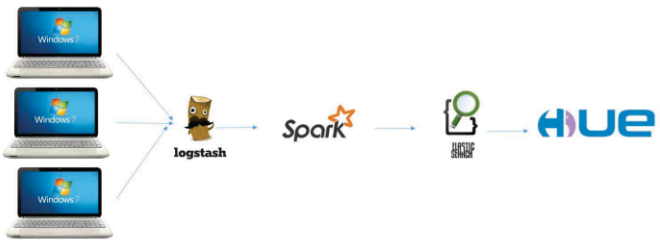


Figure 2. Architecture of the framework

WMI is a Microsoft implementation of the Web based enterprise management which gives the overall information. The data is collected and then sent to Logstash Server. We later use WMI in Apache Spark logs of collected system logs. Log data is transferred in real time to Spark for processing in memory. Now, the Apache Spark analyze the data for the malicious data. [3] Elastic Search stores the analysis data on the server Elastic Search. Finally, the analysis data is expressed in a common User Interface UI called HUE (Hadoop User Interface).

4.1 Experiments and Results:

We experimented the proposed framework in a server which contains 8 racks. The memory of the Rack is 16 core CPU * 8 (rack). Each rack has 32 GB memory. The server has 1 TB of storage size. The figure 3 depicts the results of the system logs from Logstash.

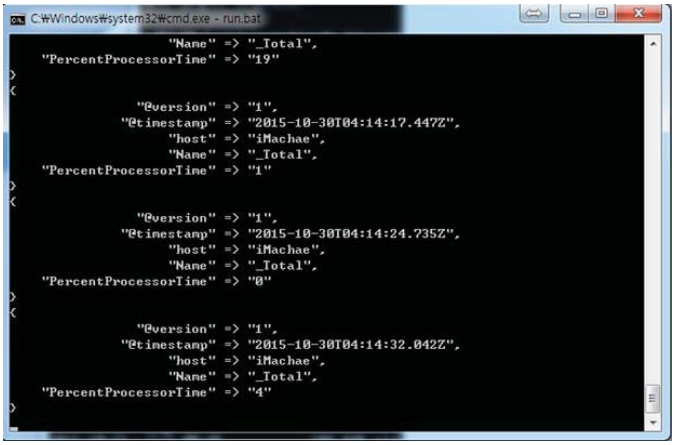


Figure 3. Logstash analysis

The analysis time took one hour in total and the log data size is 100 Mb. The interval between the analyses is 7 seconds.

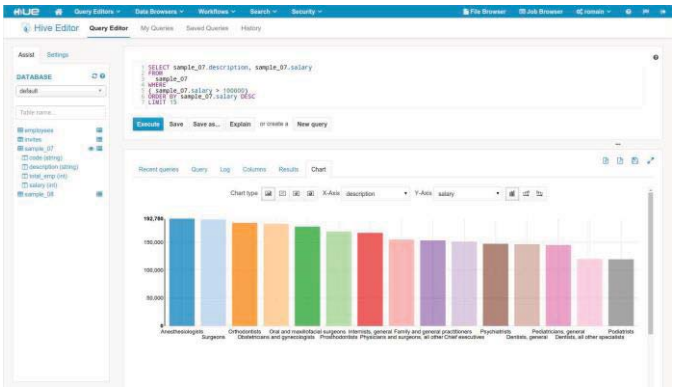


Figure 4. Results of the Analysis

The analysis results is done using HUE (Hadoop User Experience) and the results is given in figure 4.

5 Future work:

In the above work, since the analysis is done by spark in-memory cache, it can only process limited amount of data.
Real-time computer in the log file of the Windows WMI collected in real time by using the management system without passing has the categories of about 100. It collects logs for the individual machine (slave machine) to be used in public organizations in real time, it is expected to be with respect to the suspected role quickly diagnose and predict than the conventional method. In particular log , such as the (log of the operations access reservation using the Schedule service) and CPU (CPU management), network adapter management log contents of the job provided by WMI is after a certain period of time after infection , such as DDoS attacks respond in real time with respect to that work process and malicious , it can analyze the overall log management for individual computers, it is expected to be easier to manage than traditional methods of centralized management system.

6 Conclusion:

Conventional defensive method to DDOS attacks, usually response against such attacks in a scheduled treatment.
However, using the same platform, to examine the process, which are analyzed in advance with respect to a single computer that is managed by the public institution. In this paper, we have analyzed and checked the reservation process in advance before a hacking accident occurs. The framework can obtain information in advance without any timely schedule where we can block easily.

7 Reference:

[1] Zaharia, Matei, et al. "Spark: cluster computing with working sets." Proceedings of the 2nd USENIX conference on Hot topics in cloud computing. Vol. 10. 2010.

[2] Krishna, T. Lakshmi Siva Rama, T. Ragunathan, and Sudheer Kumar Battula. "Customized Web User Interface for Hadoop Distributed File System." Proceedings of the Second International Conference on Computer and Communication Technologies. Springer India, 2016.

[3] Kim Joo-hyuk, imjinsu., "The latest information security issues and encryption technology overseas study Trend", National Internet Development Agency of Korea, 2014 236 Proceedings of 2015

[4] Kim Ji Hoon., "DDoS Internet chaos", AhnLab Available at: http://www.ahnlab.com/kr/site/securityinfo/secunews/secuNewsView.do?menu_dist=3&seq=16241

[5] Available at: https://en.wikipedia.org/wiki/Denial-of-service_attack

[6] Peng, Tao, Christopher Leckie, and Kotagiri Ramamohanarao. "Detecting distributed denial of service attacks by sharing distributed beliefs." *Information Security and Privacy*. Springer Berlin Heidelberg, 2003.

[7] Lee, Keunsoo, et al. "DDoS attack detection method using cluster analysis. "*Expert Systems with Applications* 34.3 (2008): 1659-1665.

[8] Walters, Riley. "Cyber-attacks on us companies in 2014." *Heritage Foundation Issue Brief* 4289 (2014).

[9] Available at: https://en.wikipedia.org/wiki/2013_South_Korea_cyberattack

[10] Singh, Kamaldeep, et al. "Big data analytics framework for peer-to-peer botnet detection using random forests." *Information Sciences* 278 (2014): 488-497.

[11] Tripathi, Shweta, et al. "Hadoop based defense solution to handle distributed denial of service (DDoS) attacks." *Journal of Information Security* 4.3 (2013): 150.

Application of Deep Convolutional Neural Networks for Detecting Extreme Weather in Climate Datasets

Yunjie Liu[1],Evan Racah[1],Prabhat[1],Joaquin Correa[1],Amir Khosrowshahi[2],
David Lavers[3],Kenneth Kunkel[4],Michael Wehner[1],William Collins[1]
[1]Lawrence Berkeley Lab, Berkeley, CA, US
[2]Nervana Systems, San Diego, CA, US
[3]Scripps Institution of Oceanography, San Diego, CA, US
[4]National Oceanic and Atmospheric Administration, Asheville, NC, US

Abstract— *Detecting extreme events in large datasets is a major challenge in climate science research. Current algorithms for extreme event detection are build upon human expertise in defining events based on subjective thresholds of relevant physical variables. Often, multiple competing methods produce vastly different results on the same dataset. Accurate characterization of extreme events in climate simulations and observational data archives is critical for understanding the trends and potential impacts of such events in a climate change content. This study presents an application of Deep Learning techniques as alternative methodology for climate extreme events detection. Deep neural networks are able to learn high-level representations of a broad class of patterns from labeled data. In this work, we developed deep Convolutional Neural Network (CNN) classification system and demonstrated the usefulness of Deep Learning technique for tackling climate pattern detection problems. Coupled with Bayesian based hyper-parameter optimization scheme, our deep CNN system achieves 89%-99% of accuracy in detecting extreme events (Tropical Cyclones, Atmospheric Rivers and Weather Fronts).*

Keywords: Pattern Recognition, Deep Learning; Convolutional Neural Network; Climate Analytics; Extreme Events

1. Introduction

Extreme climate events (such as hurricanes and heat waves) pose great potential risk on infrastructure and human health. Hurricane Joaquin, for example, hit Carolina in early October 2015, and dropped over 2 feet of precipitation in days, resulted in severe flooding and economic loss. An important scientific goal in climate science research is to characterize extreme events in current day and future climate projections. However, understanding the developing mechanism and life cycle of these events as well as future trend requires accurately identifying them in space and time. Satellites acquire 10s of TBs of global data every year to provide us with insights into the evolution of the climate system. High resolution climate models produces 100s of TBs of data from multi-decadal run to enable us to explore future climate scenarios under global warming. Detecting extreme climate events in terabytes of data presents an unprecedented challenge for climate science.

Existing extreme climate events (e.g. hurricane) detection methods all build upon human expertise in defining relevant events based on evaluating of relevant spatial and temporal variables on hard and subjective thresholds. For instance, tropical cyclones are strong rotating weather systems that are characterized by low pressure and warm temperature core structures with high wind. However, there is no universally accepted sets of criteria for what defines a tropical cyclone [1]. The "Low" pressure and "Warm" temperature are interpreted differently among climate scientists, therefore different thresholds are used to characterize them. Researchers [2], [3], [4], [5], [6], [7] have developed various algorithms to detect tropical cyclones in large climate dataset based on subjective thresholding of several relevant variables (e.g. sea level pressure, temperature, wind etc.). One of the general and promising extreme climate event detecting software, Toolkit for Extreme Climate Analysis (TECA) [6], [7], is able to detect tropical cyclones, extra-tropical cyclones and atmospheric rivers. TECA utilizes the MapReduce paradigm to find pattern in Terabytes of climate data with in hours. However, many other climate extreme events do not have a clear empirical definition (e.g. extra-tropical cyclone and mesoscale convective system), which precludes the development and application of algorithms for detection and tracking. This study attempts to search for an alternative methodology for extreme events detection by designing a neural network based system that is capable of learning a broad class of patterns from complex multi-variable climate data, thus avoiding subjective threshold.

Recent advances in deep learning have demonstrated exciting and promising results on pattern recognition tasks, such as ImageNet Large Scale Visual Recognition Challenge [8], [9], [10] and speech recognition [11], [12], [13], [14]. Many of the state-of-art deep learning architectures for visual pattern recognition are based on the hierarchical feature learning convolutional neural network (CNN). Modern CNN systems tend to be deep and large with many hidden layers and millions of neurons, making them flexible in learning a broad class of patterns simultaneously from data. AlexNet (7

layers with 5 convolutonal layer and 2 fully connected layer) developed by [8] provides the first end to end trainable deep learning system on objective classification, which achieved 15.3% top-5 classification error rate on ILSVRC-2012 data set. On the contrary, previous best performed non-neural network based systems achieved only 25.7% top-5 classification error on the same data set. Shortly after that, Simonyan and Zisserman [9] further developed AlexNet and introduced an even deeper CNN (19 layers with 16 convolutional layer and 3 fully connected layer) with smaller kernel (filter) and achieved an impressively 6.8% top-5 classification error rate on ILSVRC-2014 data set. Szegedy et al.[10] introduced the "inception" neural network concept (network includes sub-network) and developed an even deeper CNN (22 layers) that achieved comparable classification results on ImageNet benchmark. Build on deep CNN, Sermanet et al. [15] introduced an integrated system of classification and detection, in which features learned by convolutional layers are shared among classification and localization tasks and both tasks are performed simultaneously in a single network. Girshick et al. [16] took a completely different approach by combining a region proposal framework [17] with deep CNN and designed the state of art R-CNN object detection system.

In this paper, we formulate the problem of detecting extreme climate events as classic visual pattern recognition problem. We then build end to end trainable deep CNN systems, following the architecture introduced by [8]. The model was trained to classify tropical cyclone, weather front and atmospheric river. Unlike the ImageNet challenge, where the training data are labeled natural images, our training data consist of several continuous spatial variables(e.g. pressure, temperature, precipitation) and are stacked together into image-like patches.

2. Related Work

Climate data analysis requires an array of advanced methodology. Neural network based machine learning approach, as a generative analysis technique, has received much attention and been applied to tackle several climate problems in recent year. Chattopadhyay et al. [18] developed a nonlinear clustering method based on Self Organizational Map (SOM) to study the structure evolution of Madden-Julian oscillation (MJO). Their method does not require selecting leading modes or intraseasonal bandpass filtering in time and space like other methods do. The results show SOM based method is not only able to capture the gross feature in MJO structure and development but also reveals insights that other methods are not able to discover such as the dipole and tripole structure of outgoing long wave radiation and diabatic heating in MJO. Gorricha and Costa [19] used a three dimensional SOM on categorizing and visualizing extreme precipitation patterns over an island in Spain. They found spatial precipitation patterns that traditional precipitation index approach is not able to discover, and concluded that three dimensional SOM is very useful tool on exploratory spatial pattern analysis. More recently, Shi et al. [20] implemented a newly developed convolutional long short term memory (LSTM) deep neural network for precipitation nowcasting. Trained on two dimensional radar map time series, their system is able to outperform the current state-of-art precipitation nowcasting system on various evaluation metrics. Iglesias et al. [21] developed a multitask deep fully connected neural network on prediction heat waves trained on historical time series data. They demonstrate that neural network approach is significantly better than linear and logistic regression. And potentially can improve the performance of forecasting extreme heat waves. These studies show that neural network as a generative method and can be applied on various climate problems. In this study, we explore deep CNN on solving climate pattern detection problem.

3. Methods

3.1 Convolutional Neural Network

A Deep CNN is typically comprised of several convolutional layers followed by a small amount of fully connected layers. In between two successive convolutional layers, subsampling operation (e.g. max pooling, mean pooling) is performed typically. Researchers have questioned about the necessity of pooling layers, and argue that they can be simply replaced by convolutional layer with increased strides, thus simplify the network structure [22]. In either case, the inputs of a CNN is *(m,n,p)* images, where m and n is the width and height of an image in pixel, p is the number of color channel of each pixel. The output of a CNN is a vector of q probability units (class scores), corresponding to the number of categories to be classified (e.g. for binary classifier q=2).

The convolutional layers perform convolution operation between kernels and the input images (or feature maps from previous layer). Typically, a convolutional layer contains k filters (kernels) with the size *(i,j,p)*. Where *i,j* is the width and height of the filter. The filters are usually smaller than the width m and height n of input image. p always equal to the number of color channel of input image (e.g. a color image has three channels: red, green, and blue). Each of the filters is independently convolved with the input images (or feature maps from previous layer) followed by non-linear transformation and generates k feature maps, which serve as inputs for the next layer. In the process of convolution, a dot product is computed between the entry of filter and the local region that it is connected to in the input image (or feature map from previous layer). The parameters of convolutional layers are these learnable filters. Sliding convolutional kernels across all the input will produce larger outputs for certain sub-regions than for others. This allows features to be extracted from inputs and preserved in the feature maps regardless of where the feature is located in

the input. The pooling layer subsamples the feature maps generated from convolutional layer over a *(s,t)* contiguous region, where *s,t* is the width and height of the subsampling window. This operation reduces the resolution of feature maps with the depth of CNN. All feature maps are high-level representations of the input data. The fully connected layer has connections to all hidden units in previous layer. If it is the last layer within CNN architecture, the fully connected layer also does the high level reasoning based on the feature vectors from previous layer and produce final class scores for image objects.

3.2 Hyper-parameter Optimization

Training deep neural network is known to be hard [23], [24]. Effectively and efficiently train deep neural network not only requires large amount of training data, but also requires carefully tuning model hyper-parameters (e.g. learning parameters, regularization parameters) [25]. The parameter tuning process, however, can be tedious and non-intuitive. Hyper-parameter optimization can be reduced to find a set of parameters for a network that produces the best possible validation performance. As such, this process can be thought of as a typical optimization problem of finding a set, x, of parameter values from a bounded set X that minimize an objective function $f(x)$, where x is a particular setting of the hyper-parameters and $f(x)$ is the loss for a deep neural network with a particular set of training and testing data as function of the hyper-parameter inputs. Training a deep neural network is not only a costly (with respect to time) procedure, but a rather opaque process regarding to how the network performance varies with respect to its hyper-parameter inputs. Because training and validating a deep neural network is very complicated and expensive, Bayesian Optimization (which assumes $f(x)$ is not known, is non-convex and is expensive to evaluate) is a well-suited algorithm for hyper-parameter optimization for our task at hand. Bayesian Optimization attempts to optimize $f(x)$ by constructing two things: a probabilistic model of $f(x)$ and an acquistion function that picks which point x in X to evaluate next. The probabilistic model is updated with Bayesian rule with a Gaussian prior. The acquisition function suggests hyper-parameter settings or points to evaluate by trying to balance evaluating parameter settings in regions, where $f(x)$ is low and points in regions where the uncertainty in the probabilistic model is high. As a result the optimization procedure attempts to evaluate as few points as possible [26], [25]. In this study, we use spearmint (https://github.com/JasperSnoek/spearmint) for performing network hyper-parameter optimization.

3.3 CNN Configuration

Following AlexNet [8], we developed a deep CNN which has totally 4 learnable layers, including 2 convolutional layers and 2 fully connected layers. Each convolutional layer is followed by a max pooling layer. The model is constructed based on the open source python deep learning library NOEN (https://github.com/NervanaSystems/neon). The configuration of our best performed architectures are shown in Table 1.

The networks are shallower and smaller comparing to the state-of-art architecture developed by [9], [10].The major limitations for exploring deeper and larger CNNs is the limited amount of labeled training data that we can obtain. However, a small network has the advantage of avoiding over-fitting, especially when the amount of training data is small. We also chose comparatively large kernels (filters) in the convolutional layer based on input data size, even though [9] suggests that deep architecture with small kernel (filter) is essential for state of art performance. This is because climate patterns are comparatively simpler and larger in size as compared to objects in ImageNet dataset.

One key feature of deep learning architectures is that it is able to learn complex non-linear functions. The convolutional layers and first fully connected layer in our deep CNNs all have Rectified Linear Unit (ReLU) activation functions [27] as characteristic. ReLU is chosen due to its faster learning/training character [8] as compared to other activation functions like Tanh.

$$f(x) = max(0, x) \tag{1}$$

Final fully connected layer has Logistic activation function as non-linearity, which also serves as classifier and outputs a probability distribution over class labels.

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2}$$

3.4 Computational Platform

We performed our data processing, model training and testing on Edison, a Cray XC30 and Cori, a Cray XC40 supercomputing systems at the National Energy Research Scientific Computing Center (NERSC). Each of Edison computing node has 24 2.4 GHz Intel Xeon processors. Each of Cori computing node has 32 2.3 GHz Intel Haswell processors. In our work, we mainly used single node CPU backend of NEON. The hyper-parameter optimization was performed on a single node on Cori with tasks fully parallel on 32 cores.

4. Data

In this study, we use both climate simulations and reanalysis products. The reanalysis products are produced by assimilating observations into a climate model. The spatial scale of both climate model simulation and reanalysis products covers the entire global. A summary of the data source and its temporal and spatial resolution is listed in Table 2. Ground truth labeling of various events is obtained via multivariate threshold based criteria implemented in TECA

Table 1: Deep CNN architecture and layer parameters. The convolutional layer parameters are denoted as <filter size>-<number of feature maps> (e.g. 5x5-8). The pooling layer parameters are denoted as <pooling window> (e.g. 2x2). The fully connected layer parameter are denoted as <number of units> (e.g. 2).

	Conv1	Pooling	Conv2	Pooling	Fully	Fully
Tropical Cyclone	5x5-8	2x2	5x5-16	2x2	50	2
Weather Fronts	5x5-8	2x2	5x5-16	2x2	50	2
Atmospheric River	12x12-8	3x3	12x12-16	2x2	200	2

Table 2: Data Sources

Climate Dataset	Time Frame	Temporal Resolution	Spatial Resolution (lat x lon degree)
CAM5.1 historical run	1979-2005	3 hourly	0.23x0.31
ERA-Interim reanalysis	1979-2011	3 hourly	0.25x0.25
20 century reanalysis	1908-1948	Daily	1x1
NCEP-NCAR reanalysis	1949-2009	Daily	1x1

Table 3: Dimension of image, diagnostic variables (channels) and labeled dataset size for extreme events considered in this study (PSL: sea surface pressure, U: zonal wind, V: meridional wind, T: temperature, TMQ: vertical integrated water vapor, Pr: precipitation)

Events	Image Dimension	Variables	Total Examples
Tropical Cyclone	32x32	PSL,V-BOT,U-BOT, T-200,T-500,TMQ, V-850,U-850	10,000 +ve 10,000 -ve
Atmospheric River	148 x 224	TMQ, Land Sea Mask	6,500 +ve 6,800 -ve
Weather Front	27 x 60	T-2m, Pr, PSL	5,600 +ve 6,500 -ve

[6], [7], and manual labeling by experts [28], [29]. Training data comprise of image patterns, where several relevant spatial variables are stacked together over a prescribed region that bounds an event. The dimension of the bounding box is based on domain knowledge of events spatial extent in real word. For instance, tropical cyclone radius are typically with in range of 100 kilometers to 500 kilometers, thus bounding box size of 500 kilometers by 500 kilometers is likely to capture most of tropical cyclones. The chosen physical variables are also based on domain expertise. The prescribed bounding box is placed over an event. Relevant variables are extracted within the bounding box from the climate model simulations or reanalysis products and stacked together. To facilitate model training, bounding box location is adjusted slightly such that all of events are located approximately at the center. Image patches are cropped and centered correspondingly. Because of the spatial dimension of climate events vary quite a lot and the spatial resolution of source data is non-uniform, final training images prepared differ in their size among the three types of events. The class labels of images are "containing events" and "not containing events", in other words, we formulate the problem as binary classification task. A summary of the attributes of training images is listed in Table 3.

5. Results and Discussion

Table 4 summarizes the performance of our deep CNN architecture on classifying tropical cyclones, atmospheric rivers and weather fronts. We obtained fairly high accuracy (89%-99%) on extreme event classification. In addition, the systems do not suffer from over-fitting. We believe this is mostly because of the shallow and small size of the architecture (4 learnable layers) and the weight decay regularization. Deeper and larger architecture would be inappropriate for this study due to the limited amount of training data. Fairly good train and test classification results also suggest that the deep CNNs we developed are able to efficiently learn representations of climate pattern from labeled data and make predictions based on feature learned. Traditional threshold based detection method requires human expert carefully examine the extreme event and its environment, thus come up with thresholds for defining the events. In contrast, as shown in this study, deep CNNs are able to

learn climate pattern just from the labeled data, thus avoiding subjective thresholds.

Table 4: Overall Classification Accuracy

Event Type	Train	Test	Train time
Tropical Cyclone	99%	99%	≈ 30 min
Atmospheric River	90.5%	90%	6-7 hour
Weather Front	88.7%	89.4%	≈ 30 min

5.1 Classification Results for Tropical Cyclones

Tropical cyclones are rapid rotating weather systems that are characterized by low pressure center with strong wind circulating the center and warm temperature core in upper troposphere. Figure 1 shows examples of tropical cyclones simulated in climate models, that are correctly classified by deep CNN (warm core structure is not shown in this figure). Tropical cyclone features are rather well defined, as can be seen from the distinct low pressure center and spiral flow of wind vectors around the center. These clear and distinct characteristics make tropical cyclone pattern relatively easy to learn and represent within CNN. Our deep CNNs achieved nearly perfect (99%) classification accuracy.

Figure 2 shows examples of tropical cyclones that are mis-classified. After carefully examining these events, we believe they are weak systems (e.g. tropical depression), whose low pressure center and spiral structure of wind have not fully developed. The pressure distribution shows a large low pressure area without a clear minimum. Therefore, our deep CNN does not label them as tropical cyclones.

Table 5: Confusion matrix for tropical cyclone classification

	Label TC	Label Non_TC
Predict TC	0.989	0.003
Predict Non_TC	0.011	0.997

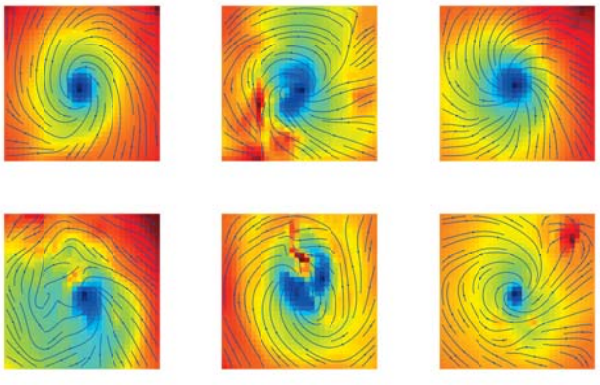


Fig. 1: Sample images of tropical cyclones correctly classified (true positive) by our deep CNN model. Figure shows sea level pressure (color map) and near surface wind distribution (vector solid line).

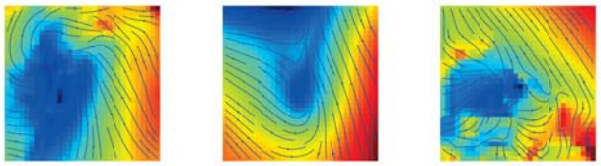


Fig. 2: Sample images of tropical cyclones mis-classified (false negative) by our deep CNN model. Figure shows sea level pressure (color map) and near surface wind distribution (vector solid line).

5.2 Classification Results for Atmospheric Rivers

In contrast to tropical cyclones, atmospheric rivers are distinctively different events. They are narrow corridors of concentrated moisture in atmosphere. They usually originate in tropical oceans and move pole-ward. Figure 3 shows examples of correctly classified land falling atmospheric rivers that occur on the western Pacific Ocean and north Atlantic Ocean. The characteristics of narrow water vapor corridor is well defined and clearly observable in these images.

Figure 4 are examples of mis-classified atmospheric rivers. Upon further investigation, we believe there are two main factors leading to mis-classification. Firstly, presence of weak atmospheric river systems. For instance, the left column of Figure 4 shows comparatively weak atmospheric rivers. The water vapor distribution clearly show a band of concentrated moisture cross mid-latitude ocean, but the signal is much weaker comparing to Figure 3. Thus, deep CNN does not predict them correctly. Secondly, the presence of other climate event may also affect deep CNN representation of atmospheric rivers. In reality, the location and shape of atmospheric river are affected by jet streams and extra-tropical cyclones. For example, Figure 4 right column shows rotating systems (likely extra-tropical cyclone) adjacent to the atmospheric river. This phenomenon presents challenge for deep CNN on representing atmospheric river.

Table 6: Confusion matrix for atmospheric river classification

	Label AR	Label Non_AR
Predict AR	0.93	0.107
Predict Non_AR	0.07	0.893

5.3 Classification Results for Weather Fronts

Among the three types of climate events we are looking at, weather fronts have the most complex spatial pattern. Weather fronts typically form at the interface of warm air and cold air, and usually associated with heavy precipitation due moisture condensation of warm air up-lifting. In satellite

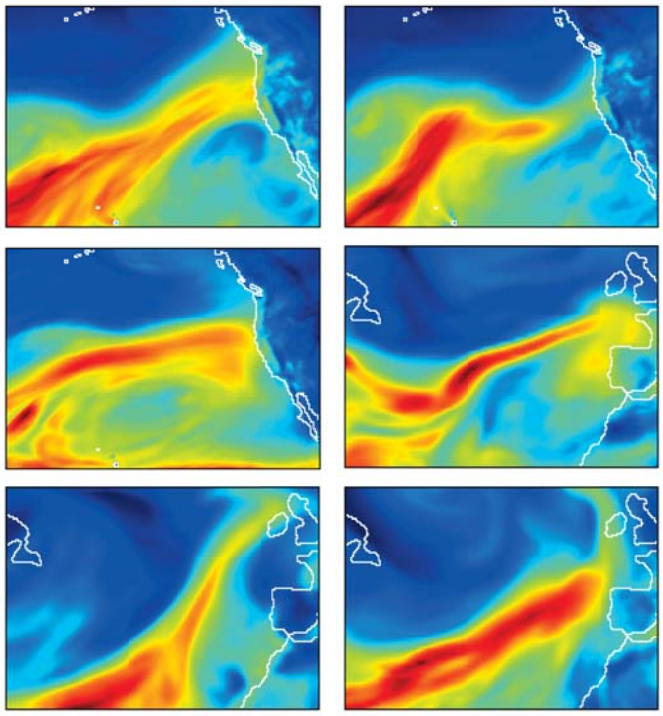


Fig. 3: Sample images of atmospheric rivers correctly classified (true positive) by our deep CNN model. Figure shows total column water vapor (color map) and land sea boundary (solid line).

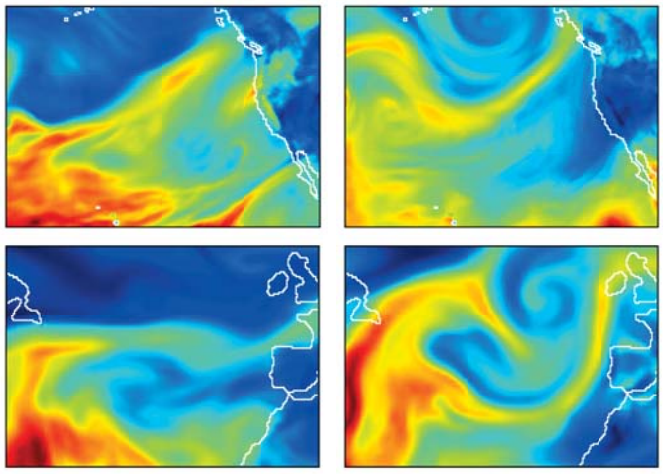


Fig. 4: Sample images of atmospheric rivers mis-classified (false negative) by our deep CNN model. Figure shows total column water vapor (color map) and land sea boundary (solid line).

images,a weather front is observable as a strip of clouds, but it is hardly visible on two dimensional fields such as temperature and pressure. In middle latitude (e.g. most U.S.), a portion of weather front are associated with extra-tropical cyclones. Figure 5 shows examples of correctly classified weather front by our deep CNN system. Visually, the narrow long regions of high precipitation line up approximately parallel to the temperature contour. This is a clear characteristics and comparatively easy for deep CNNs to learn.

Because patterns of weather fronts is rather complex and hardly show up in two dimensional fields, we decided to further investigate it in later work.

Table 7: Confusion matrix for weather front classification

	Label WF	Label Non_WF
Predict WF	0.876	0.18
Predict Non_WF	0.124	0.82

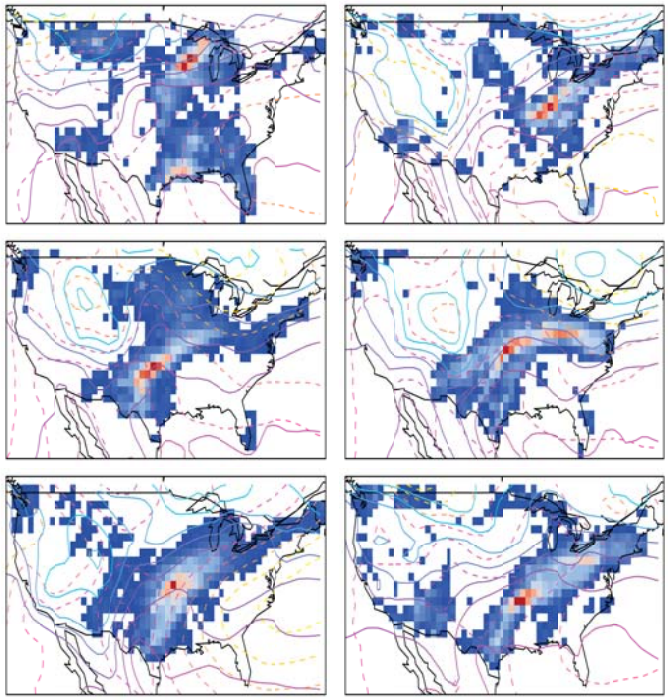


Fig. 5: Sample images of weather front correctly classified by our deep CNN model. Figure shows precipitation with daily precipitation less than 5 millimeters filtered out (color map), near surface air temperature (solid contour line) and sea level pressure (dashed contour line)

6. Future Work

In the present study, we trained deep CNNs separately for classifying tropical cyclones, atmospheric rivers and weather fronts. Ideally, we would like to train a **single** neural network for classifying all three types of events. Unlike object recognition in natural images, climate patterns detection have unique challenges. Firstly, climate events happen at vastly different spatial scales. For example, a tropical cyclone typically extends over less than 500 kilometers in radius, while an atmospheric river can be several

thousand kilometers long. Secondly, different climate events are characterized by different sets of physical variables. For example, atmospheric rivers correlate strongly with the vertical integration of water vapor, while tropical cyclones has a more complex multi-variable pattern involving sea level pressure, near surface wind and upper troposphere temperature. Future work will need to develop generative CNN architectures that are capable of discriminating between different variables based on the event type and capable of handling events at various spatial scale. Note that we have primarily addressed **detection** of extreme weather patterns, but not their **localization**. We will work on architectures for spatially localizing weather pattern in the future.

Several researchers have pointed out that deeper and larger CNNs perform better for classification and detection tasks[9], [10] compared to shallow networks. However, deep networks require huge amount of data to be effectively trained, and to prevent model over fitting. Datasets, such as ImageNet, provide millions of labeled images for training and testing deep and large CNNs. In contrast, we can only obtain a small amount of labeled training data, hence we are constrained on the class of deep CNNs that we can explore without suffering from over-fitting. This limitation also points us to the need for developing unsupervised approaches for climate pattern detection. We believe that this will be critical for the majority of scientific disciplines that typically lack labeled data.

7. Conclusion

In this study, we explored deep learning as a methodology for detecting extreme weather patterns in climate data. We developed deep CNN architecture for classifying tropical cyclones, atmospheric rivers and weather fronts. The system achieves fairly high classification accuracy, range from 89% to 99%. To the best of our knowledge, this is the first time that deep CNN has been applied to tackle climate pattern recognition problems. This successful application could be a precursor for tackling a broad class of pattern detection problem in climate science. Deep neural network learns high-level representations from data directly, therefore potentially avoiding traditional subjective thresholding based criteria of climate variables for event detection. Results from this study will be used for quantifying climate extreme events trend in current day and future climate scenarios, as well as investigating the changes in dynamics and thermodynamics of extreme events in global warming contend. This information is critical for climate change adaptation, hazard risk prediction and climate change policy making.

8. Acknowledgments

This research was conducted using "Neon", an open source library for deep learning from Nervana Systems.

This research used resources of the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. This work was supported by the Director, Office of Science, Office of Advanced Scientific Computing Research, Applied Mathematics program of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

References

[1] D. S. Nolan and M. G. McGauley, "Tropical cyclogenesis in wind shear: Climatological relationships and physical processes," in *Cyclones: Formation, Triggers, and Control*, 2012, pp. 1–36.

[2] F. Vitart, J. Anderson, and W. Stern, "Simulation of interannual variability of tropical storm frequency in an ensemble of gcm integrations," *Journal of Climate*, vol. 10, no. 4, pp. 745–760, 1997.

[3] ——, "Impact of large-scale circulation on tropical storm frequency, intensity, and location, simulated by an ensemble of gcm integrations," *Journal of Climate*, vol. 12, no. 11, pp. 3237–3254, 1999.

[4] K. Walsh and I. G. Watterson, "Tropical cyclone-like vortices in a limited area model: comparison with observed climatology," *Journal of Climate*, vol. 10, no. 9, pp. 2240–2259, 1997.

[5] K. Walsh, M. Fiorino, C. Landsea, and K. McInnes, "Objectively determined resolution-dependent threshold criteria for the detection of tropical cyclones in climate models and reanalyses," *Journal of Climate*, vol. 20, no. 10, pp. 2307–2314, 2007.

[6] Prabhat, O. Rübel, S. Byna, K. Wu, F. Li, M. Wehner, W. Bethel, *et al.*, "Teca: A parallel toolkit for extreme climate analysis," in *Third Worskhop on Data Mining in Earth System Science (DMESS) at the International Conference on Computational Science (ICCS)*, 2012.

[7] Prabhat, S. Byna, V. Vishwanath, E. Dart, M. Wehner, W. D. Collins, *et al.*, "Teca: Petascale pattern recognition for climate science," in *Computer Analysis of Images and Patterns*. Springer, 2015, pp. 426–436.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2012, pp. 1097–1105.

[9] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Internaltional Conference on Learning Representation (ICLR)*, 2015.

[10] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.

[11] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

[12] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 30–42, 2012.

[13] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6645–6649.

[14] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[15] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," in *International Conference on Learning Representations (ICLR)*, 2014.

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 580–587.

[17] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.
[18] R. Chattopadhyay, A. Vintzileos, and C. Zhang, "A description of the madden–julian oscillation based on a self-organizing map," *Journal of Climate*, vol. 26, no. 5, pp. 1716–1732, 2013.
[19] J. Gorricha, V. Lobo, and A. C. Costa, "A framework for exploratory analysis of extreme weather events using geostatistical procedures and 3d self-organizing maps," *International Journal on Advances in Intelligent Systems*, vol. 6, no. 1, 2013.
[20] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *Advances in Neural Information Processing Systems: Twenty-Ninth Annual Conference on Neural Information Processing Systems (NIPS)*, 2015.
[21] G. Iglesias, D. C. Kale, and Y. Liu, "An examination of deep learning for extreme climate pattern analysis," in *The 5th International Workshop on Climate Informatics*, 2015.
[22] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," in *International Conference on Learning Representation (ICLR)*, 2015.
[23] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin, "Exploring strategies for training deep neural networks," *The Journal of Machine Learning Research*, vol. 10, pp. 1–40, 2009.
[24] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *International conference on artificial intelligence and statistics*, 2010, pp. 249–256.
[25] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," in *Advances in neural information processing systems*, 2012, pp. 2951–2959.
[26] E. Brochu, V. M. Cora, and N. De Freitas, "A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," *arXiv preprint arXiv:1012.2599*, 2010.
[27] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML)*, 2010, pp. 807–814.
[28] K. E. Kunkel, D. R. Easterling, D. A. Kristovich, B. Gleason, L. Stoecker, and R. Smith, "Meteorological causes of the secular variations in observed extreme precipitation events for the conterminous united states," *Journal of Hydrometeorology*, vol. 13, no. 3, pp. 1131–1141, 2012.
[29] D. A. Lavers, G. Villarini, R. P. Allan, E. F. Wood, and A. J. Wade, "The detection of atmospheric rivers in atmospheric reanalyses and their links to british winter floods and the large-scale climatic circulation," *Journal of Geophysical Research: Atmospheres*, vol. 117, no. D20, 2012.

SESSION

SYSTEMS BIOLOGY AND RNA SEQUENCE DATA PROCESSING + HIGH PERFORMANCE COMPUTING

Chair(s)

TBA

ISBN: 1-60132-427-8, CSREA Press ©

Star Plot Visualization of Ultrahigh Dimensional Multivariate Data

Shabana Sangli[1], Gurminder Kaur[1] and Bijaya B. Karki[1,2,3]
[1]School of Electrical Engineering and Computer Science, [2]Department of Geology and Geophysics, [3]Center for Computation and Technology, Louisiana State University
Baton Rouge, USA

Abstract - *Visualization-based analysis of multivariate data suffers from a high degree of clutter when the number of dimensions (variables) becomes too large. Here, we extend the standard star plot technique to visualize large datasets of ultrahigh number of dimensions by a) drawing overlapped star plots with one star per data item, b) shifting the origins of radial axes away from the central point to open space in the low-value scale, and c) dynamically partitioning the dimensions into groups and mapping them to different concentric circular regions to provide multilevel star plot visualization. Our test on multivariate datasets of high dimensionality suggests that the proposed extensions with appropriate interaction options can handle large number of dimensions of potential relevance to big data analytics.*

Keywords: Multivariate data; star plot; information visualization; big data analytics

1 Introduction

Analysis of multivariate data poses tremendous challenge not only when the size of the dataset increases but also when the number of variables or attributes (also referred to as dimensions) increases. Such high-dimensional multivariate data perhaps are one of the most important contributors to today's big data problems [1,2]. Extracting useful information from large, complex datasets is too difficult by directly looking at the data presented in a tabular form and/or simply using standard query languages. Visualization aims to take maximal advantage of human perception capabilities by graphically mapping a given dataset in its entirety, irrespective of its size, to a visual form for gaining insight into the data [3]. The visualization output though it is likely cluttered may still convey some information about the overall nature and structure of the data. When used interactively, visualization can provide us with a qualitative overview of large and abstract datasets thereby helping us quickly search for interesting features such as patterns, trends, anomalies, relationships, and clusters [4,5]. These features in turn can serve as basis for further analysis, which usually involves more focused (quantitative) explorations of selected data regions or variables. The goal is thus to find a visual representation of a given big data problem with a minimum clutter possible. So, the user can actually view the data as a whole to make some sense out of the display and then further dig into the data. This process may be referred to as an exploratory visualization-based analysis.

Several visualization techniques currently exist for analyzing multivariate data, which involve three or more quantitative variables (dimensions). Popular examples include star plot [6], parallel coordinates [7,8], star coordinates [9], scatterplot matrix [10], etc. In principle, these techniques should work for dataset of arbitrary size and arbitrary number of dimensions. However, there is still a need for assessing the potential of effectively using these techniques in visualization of very large datasets and big data. This is true in the case of star plot, which is a simple, widely used method for multivariate data visualization. The star plot method maps an n-dimensional data space into a 2-dimensional display space by representing n (≥ 3) variables on axes starting from the same central point [6,11]. In this mapping, each data item is displayed as a star-shaped icon in which radial spoke represents the value for a variable. All icons, one for each data observation, are usually displayed in a rectangular arrangement on a single page or screen. This visual presentation is effective and mostly used for small- to moderate-sized datasets perhaps containing no more than one hundred observations for no more than a couple of dozens of variables [6]. As the dimensionality and size of data increase, star plot visualization becomes increasingly overwhelming and the individual stars eventually become too small to represent recognizable shapes.

In this paper, we discuss the weaknesses of the classic star plot visualization technique for large multivariate datasets and mainly deal with ultra-high dimensionality of the data. In the situations involving a large number of variables/attributes, the radial axes are very closely packed and the star plot display becomes too cluttered. We propose different ways of adopting/extending the star plot method to analyze and understand large, complex multivariate datasets. We call these approaches as *overlapped star plot*, *shifted origin star plot*, and *multilevel star plot* visualization techniques. Tested on different multivariate datasets, the proposed extensions enable us to get a better visual display of data in their entirety by effectively mapping an ultrahigh number of dimensions as radial axes and rendering all data values for these dimensions on a given display plane.

2 Related work and motivation

The star plot is a multivariate data visualization method, which represents each data item as a star icon consisting of a sequence of equi-angular spokes (called radii) with each spoke/ray representing one of the variables [6,11]. It is also called a radar chart or web chart or circular parallel coordinates [12]. The star plot for single observation can provide information about relative dominance of a variable with respect to other variables in the observation. A set of multiple star plots helps identify the similarities in the observations and form clusters. The multi-plot also helps in the detection of outliers and anomalies. With these capabilities, the star plot enables the user to comprehend the data and extract useful information in a simple way, thus serving as one of the most useful visualization techniques.

However, the star plot technique falls short when the number of variables or dimensions becomes too large, needing its further improvement as we show in this paper. The human eye can perceive and understand the information conveyed by a visual display more easily when the display is presented in a bigger size. Instead of plotting every data item as a rather small star at a separate location, it is better to map all data items together and display all stars at the same location. This approach has been previously considered [13,14], but here we further emphasize its essence.

The star plot technique treats dimensions uniformly as in other radial methods of information visualization [6,12,14]. As the number of dimensions increases, it allows more compaction to accommodate all dimensions in display plane of fixed size. When data values are plotted on a compact axial layout, the result obviously is a cluttered visualization. To minimize the clutter, we consider two approaches: First, if the numbers of data items and dimensions are excessive near the origin, we can widen space between successive axes in their lower value ends. Second, to handle a very high number of dimensions, we can divide the dimensions into multiple groups and map them in different regions (levels), which are concentric circular areas. Finally, each approach is expected to work more effectively when appropriate interaction options are supported, as is the case with any multivariate data visualization [5].

3 Test datasets and implementation

Three datasets used for illustrating and testing the proposed extensions of the star plot method were taken from the UCI Machine Learning Repository [15]. The first is the data about 21 attributes of a car. The second US census 1990 dataset involves 68 variables of both numerical and categorical type. The third is the data about hand movements for LIBRAS (the official Brazilian signal language) with 91 variables representing the coordinates of the movement. Further details about the scaling and coding of the variable values in these datasets can be obtained from the repository [15]. Here, we provide some information for the "cars" dataset in Table I, which consists of 7 categorical variables (numbered as 0, 1, 2, 3, 4, 10 and 20) and the remaining 14 numerical variables.

The proposed improvements on the star plot visualization method were implemented using C++ with OpenGL and glut libraries for graphics rendering. A few interaction options were implemented to assess how visualization can be further enhanced. They include options for highlighting one or more data items of interest, adjusting the number of dimensions to be mapped to each display ring (circular region), and changing the boundaries between the display rings.

4 Proposed star plot extensions

A major concern with star plot is its limited ability to convey information under situations when the number of data items increases, when the values lie near the low end of the rays of the star plot, and when the dimensionality is too high. Therefore, we seek to adopt and further develop the star plot technique to visualize and explore large multivariate datasets as described in this paper.

4.1 Overlapped star plot

The star plot method displays a given dataset of N items as an array of N stars, one for each data item of k dimensions (or variables). In each plot, the values of a data point of the dimensions d_1, d_2,…,d_k are linearly scaled from the fixed origin $O(O_x,O_y)$ and mapped at values v_1, v_2,…, v_k on the k radial axes. The intersection points on these axes are then joined together to form a closed polyline of star shape, which graphically depicts the corresponding data item. These individual stars drawn separately need to be shrunk to accommodate all of them in the available display space, so that it is increasingly hard to view and comprehend these small stars. Here, we consider displaying all the data items together at the same position using the same radial axes instead of mapping each data item as a separately located star icon. This *overlapped star plot* approach uses the entire space for displaying each data point. In Fig. 1 (top), all stars are drawn together in the same large space for the cars dataset so they are visible (except some clutter caused by overlapping) and easy to compare.

TABLE I
VARIABLES OF CARS DATASET

Variable Number	Values of the variable with their numerical scaled equivalent used. The appropriate values of *min* and *max* were used for scale.
0	Fuel-type: diesel=1, gas=2 ; (*min*=0, *max*= 2)
1	Aspiration: std = 1, turbo = 2
2	Number-of-doors: two = 2, four = 4
3	Body-style: convertible = 1, hatchback = 2, sedan = 3, wagon = 4, hardtop = 5
4	Engine-location: front = 1, rear = 2
5	Wheel-base: 86.6 to 108.0
6	Length: 141.1 to 192.7
7	Width: 60.3 to 71.4
8	Height: 47.8 to 59.8
9	Curb-weight: 1488 to 3296
10	Number-of-cylinders: two = 2, three = 3, four = 4, five = 5, six = 6
11	Engine-size: 61 to 181
12	Bore: 2.91 to 3.94
13	Stroke: 2.19 to 3.90
14	Compression-ratio: 7 to 21.9
15	Horsepower: 48 to 200
16	Peak-rpm: 4150 to 6000
17	City-mpg: 17 to 49
18	Highway-mpg: 20 to 54
19	Price: 5151 to 23875
20	Make: alfa-romero = 1, audi = 2, bmw = 3, chevrolet = 4, dodge = 5, honda = 6, porshe = 7, benz = 8, mitsubishi = 9, nissan = 10, peugot = 11

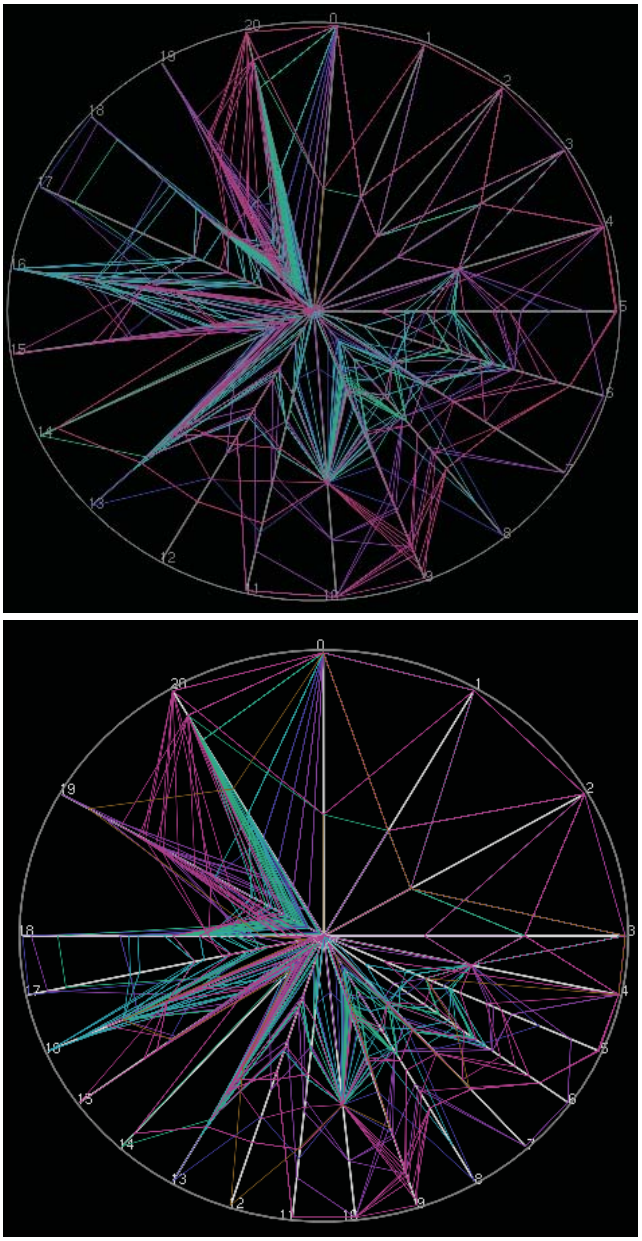


Fig. 1. Visualization of cars data set containing 21 variables using the overlapped star plot with uniform (top) and non-uniform (bottom) axial layout.

For a uniform radial layout of k dimensions, the angle between the neighboring axes is $360^o/k$, which becomes too small when the dimensionality is too high. It is desirable to have a large interaxial angle. For instance, an angle of 30^o that corresponds to one dozen variables is generally preferred. To have such wide-diverging axes, we can use one of half of the total space (i.e., 180^o angular region) to display a few selected dimensions, say 6 axes. The remaining k-6 axes are then packed in other half space. This non-uniform axial layout shown in Fig. 1 (bottom) for the cars dataset helps display data distributions and relationships for selected dimensions more clearly (by giving a zoom-in like view) while still accommodating the remaining dimensions. The total display area can be split in two halves horizontally, vertically, or perhaps at any orientation in an interactive manner.

4.2 Shifted origin star plot

In the normal star plot technique, data values are mapped onto the radii from the fixed center point (O_x, O_y). When many dimensions are represented by closely packed radial axes emanating from the same origin (Fig. 2, top), high degree of crowdedness occurs towards the lower ends of the plot thereby making the data lines hardly visible. To overcome this problem, we propose the *shifted origin star plot* technique, in which the data values $(d_1, d_2,\ldots,d_k)$ are scaled from different origin points $(O_1, O_2,\ldots,O_k)$, where each dimension d_i has its own origin O_i which is at some shifted

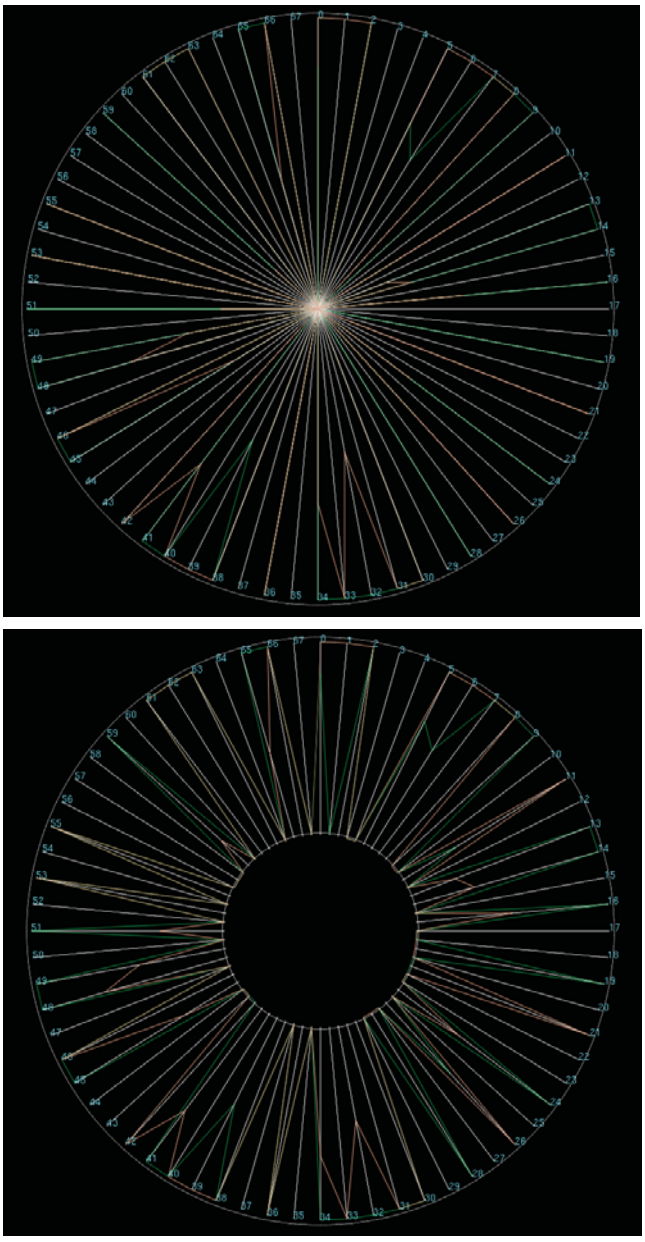


Fig. 2. Layout of 68 axes for the census dataset using single origin (top) and shifted origins (bottom).

distance from the center. This radial outward shift opens extra space between the successive axes at their lower ends thereby improving visibility even when data points are crowded toward the origin.

Before laying out the axes and finding data locations along each radial axis, we need to calculate the shift amount (l), which depends on the total number of dimensions (k) and the maximum scale (L) available for plotting:

$$l = (k/360)L \qquad (1)$$

As k increases, l increases. However, the shift distance cannot be arbitrarily large otherwise the inner void space eventually covers the entire display area. We constrain the shift distance to be the maximum of l and $0.5L$. Fig. 2 (bottom) illustrates the shifted origin star plot using a shift of $0.2L$ for the census dataset. The user can adjust the size of the circular perimeter containing all variable origins by dragging it inward or outward in an interactive manner.

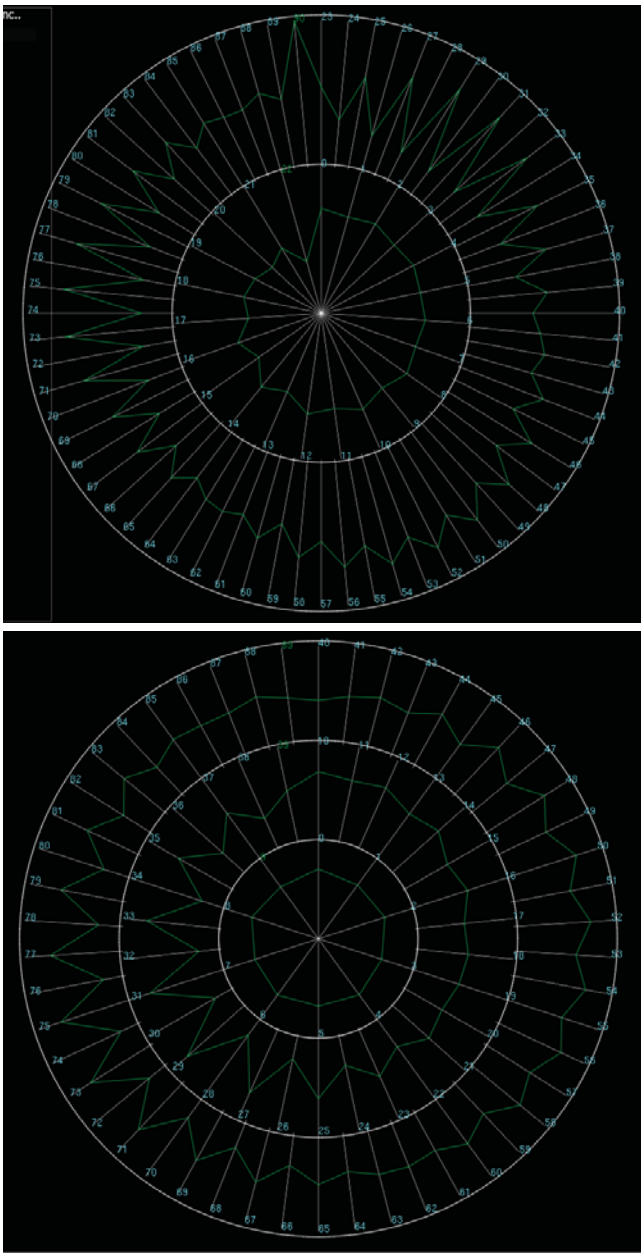


Fig. 3. Two-level (top) and three-level (bottom) layout of 91 axes for the LIBRAS data. One data item is drawn.

4.3 Multilevel Star Plot

As the number of dimensions increases, effectively using available fixed display area to accommodate all dimensions without obscuring cognitive information about the data becomes even more challenging. The shifted origin star plot method discussed above may not be good enough because of high visual clutter arising from closely packed radial axes and connecting data lines. A possible solution is to reduce mapping of radial lines (axes) in a given area as much as possible. Note that this shifted origin star plot creates a void circular space around the center (Fig. 2, bottom). It is good idea to not waste this space. We can use the space for representing a subset of the dimensions and displaying the corresponding data values. We now divide the total dimensions into two groups and map them at two levels (Fig. 3, top), with the selected (fewer) dimensions assigned to the inner region (first level) and the remaining (majority) dimensions assigned to the outer region (second level). Thus, two stars of which the larger one lies in the outer region and the smaller one lies in the inner region together display each data item. We can extend this idea to multiple levels in which different groups of dimensions are represented in different concentric rings that constrain the respective display regions out of the total space. Our *multilevel star plot* thus manages many dimensions in different subsets thereby mapping the axes and displaying the corresponding data values in multiple levels (regions) instead of plotting all together in a single region as one star icon. The result is that a star displaying each data item is now split into multiple stars of different sizes drawn in different regions.

Before laying out the radial axes and calculating data locations along each axis, a decision is to be made on how many dimensions appear at each level. Let us consider a scenario in which the original large circular display space is divided into concentric rings of equal concentric radii. Obviously, the area of the outer ring is larger than the ring below it. So, it makes sense that a higher-level (outer ring) plot be assigned more dimensions than a lower-level (inner ring) plot. The number of dimensions assigned to the i^{th} level depends on the total number of dimensions (k), the total number of plot levels (m), and the level of plotting (N_i). We require that the number of dimensions represented in a level be proportional to the area of the corresponding ring. For the i^{th} ring, the area is given by $A_i = \pi i^2 r^2 - \pi(i\text{-}1)^2 r^2 = (i + i\text{-}1)\pi r^2$, where r represents the concentric radii. The number of dimensions (D_i) can be thus calculated as:

$$D_i = \frac{k}{N_m^2}\left(N_i + N_{i-1}\right), \text{ where } i = 1,2...,m \qquad (2)$$

Here, N_i represents the i^{th} level. Fig. 3 (top) shows the star plot with 2 shift levels ($N_m = 2$) for the 91-dimensional LIBRAS dataset where $1/4^{th}$ of the total dimensions are assigned to the inner circle and the remaining $3/4^{th}$ to the outer ring. Fig. 3 (bottom) shows a similar plot with 3 shift levels ($N_m = 3$) for the same dataset with approximately $1/9^{th}$ dimensions in the inner circle (first level), $3/9^{th}$ dimensions in the middle ring (second level), and the remaining $5/9^{th}$ dimensions in the outer ring (third level).

5 Visualization-based analysis

We now present some analysis of the proposed star plot extensions on two high-dimensional multivariate datasets, namely the US census data with 68 variables and the LIBRAS movement data with 91 variables. When the number of radial axes is large, all data points mapped to the scale may not be visible. As shown in Fig. 4 (top) for the census dataset, the data values that lie in the upper end of the scale are visible. However, the dense region around the origin does not give much information about any data values that lie closer to the lower ends of the scale. This visual clutter can be attributed to two factors. First, the dimensionality is too high which would result in mapping many radial axes from the fixed origin. Second, many values lie in the lower ends of the scale further adding to the clutter. This is the suppression of data with the loss of information. Hypothesizing information that has concealed the data likely produces a hypothesis, which may not match with the information presented by the actual data. Consider the values plotted on the radial axis numbered 65, representing the variable YEARSCH in Fig. 4 (top). A careful observation of the display reveals that the range of values in the dataset for this variable are in the mid to upper range of education with considerable range of points falling in the mid-range. One can make hypothesis that the most people have an educational qualification of 9th Grade-10th Grade or more. However, the information in the lower ends of the scale is hidden due to visual clutter.

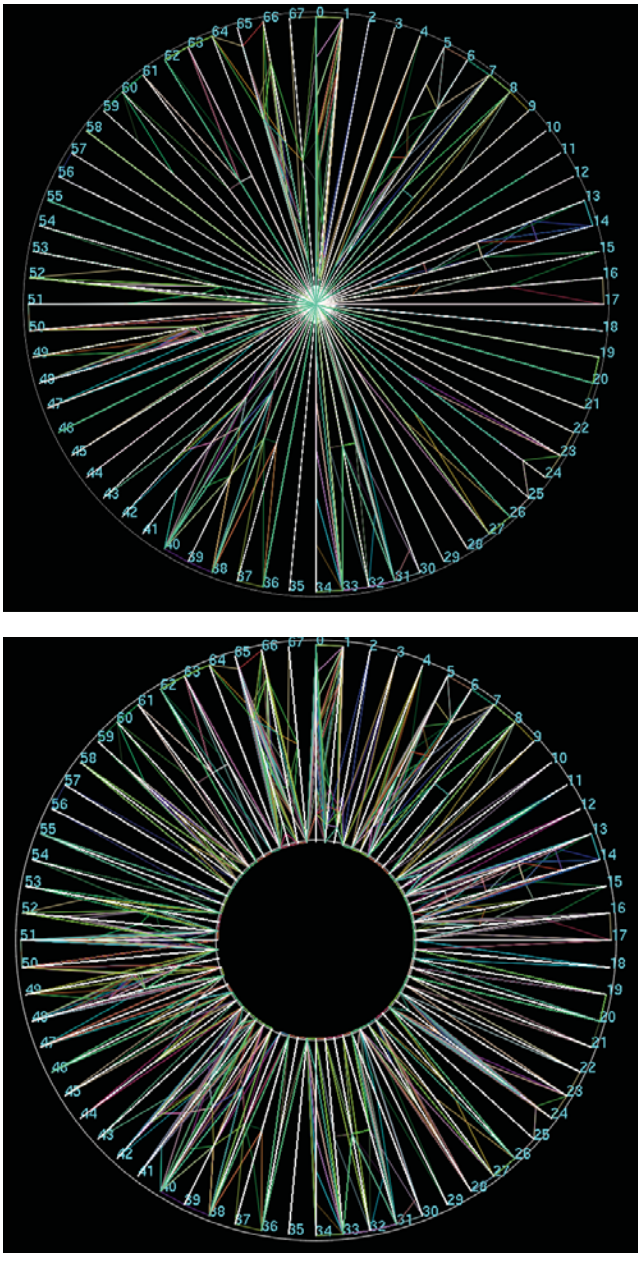


Fig. 4. Visualization of the 68-dimensional census data using the overlapped (top) and shifted origin (bottom) star plots.

As shown in Fig. 4 (bottom), the shifted origin star plot of the census dataset reduces the clutter in the low-value region and reveals information that might otherwise be hidden. For the variable YEARSCH, notable information that becomes apparent after the origin shift is that many data points do fall in the lower end scale of the axis. This means that a considerable portion of the population has an educational qualification lesser than 9th Grade, an important information that remained obscure earlier because of high clutter in the Fig. 4 (top).

The overlapped star plot and origin shifted star plot map all dimensions and all data in the same region. They may not be effective when the number of dimensions is very large because the inter-ray angles become too small. Fig. 5 (top) illustrates such a scenario for the 91-dimensional dataset. The clutter becomes prevalent over a wide region around the center of the display. Splitting the dimensions at three groups considerably reduces the clutter and makes all star-polylines visible (Fig. 5, bottom). One can see open space between the rays, which arises because fewer dimensions are drawn at each level and the outer regions have bigger origin shifts. It is remarkable that the multilevel star plot visualization can give a better presentation of multivariate dataset consisting of large number of dimensions by mapping each data star to a multiple sub-star icon, for example, a three sub-star drawn in Fig 5 (bottom).

If there is simply too much data, the display becomes too cluttered because of a lot of over-plotting and closely packed axes. So, normal star plot visualization may not be of much help in analyzing large multivariate dataset of high dimensionality. We need specialized visualizations and effective modes of interaction to get to know such data [5]. Our proposed approaches for star plot visualization attempt to systematically increase inter-axial angular spacing. As such, it is possible to trace all axes individually and then encode extra information about the data along them. For example, we can draw a box plot to display distribution of numerical data values through their quartiles [12]. Similarly, we can draw circles to display the relative sizes of each value belonging to a particular categorical variable, as demonstrated in Fig. 6 for the cars dataset for seven categorical variables (Table I). Also, outliers may be displayed as individual points.

Employing interaction rather than simple renderings can help manage large-scale data [5]. Highlighting and brushing may be applied to aid interpretation of star plot visualization. A straightforward option is to highlight the lines representing selected observations or outliers with different color and/or thickness. As shown in Fig. 6, the shapes of two highlighted stars are clearly recognizable for their analysis and comparison. A brushing allows us to specify a region of interest along one axis and to focus on the corresponding

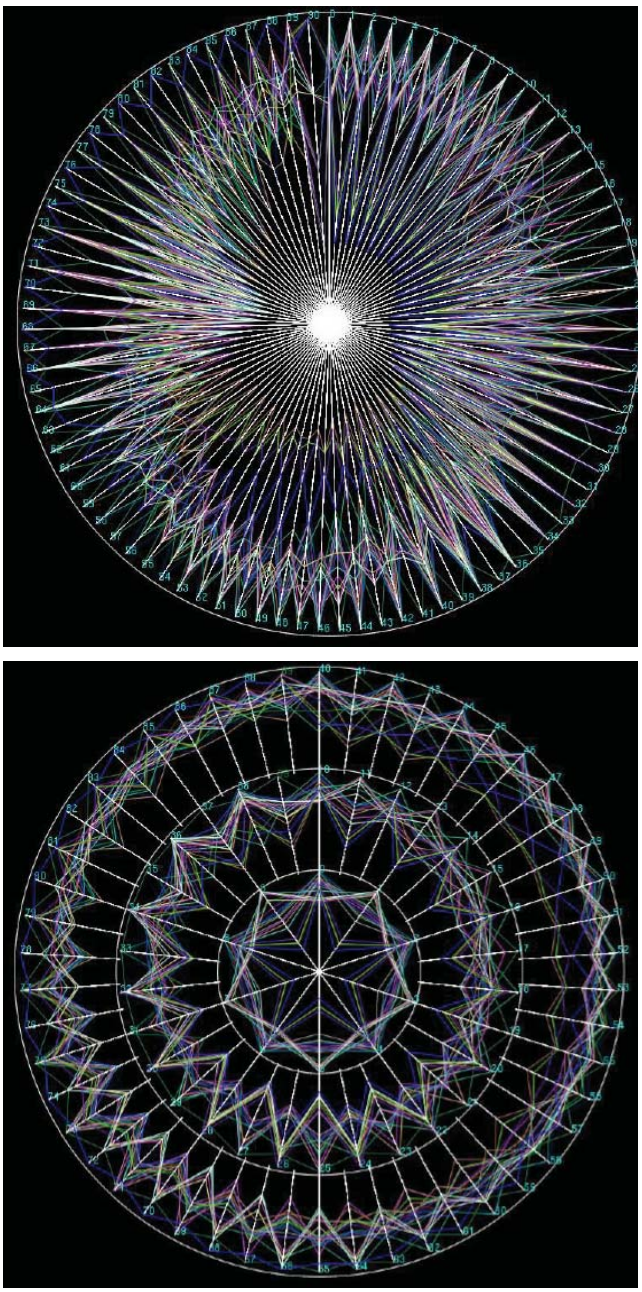


Fig. 5. Visualization of the 91-dimensional LIBRAS data using the overlapped (top) and multilevel (bottom) star plots.

stars. Brushing can be combined with other selection functionalities to make the star plot more interactive.

Moreover, it is possible to reduce visual clutter and discover interesting features in data by dimension suppression and reordering [4,16]. We can draw fewer dimensions by omitting dimensions of little or no importance. The star plot technique is generally considered to be effective only when quantitative variables are represented. The dimensions can also be ordered or clustered according to similarity [17]. Some axes may be reversed as well. While a linear scaling is often used for mapping data values on the axes, other scaling options such as a multi-scale mapping may be worth

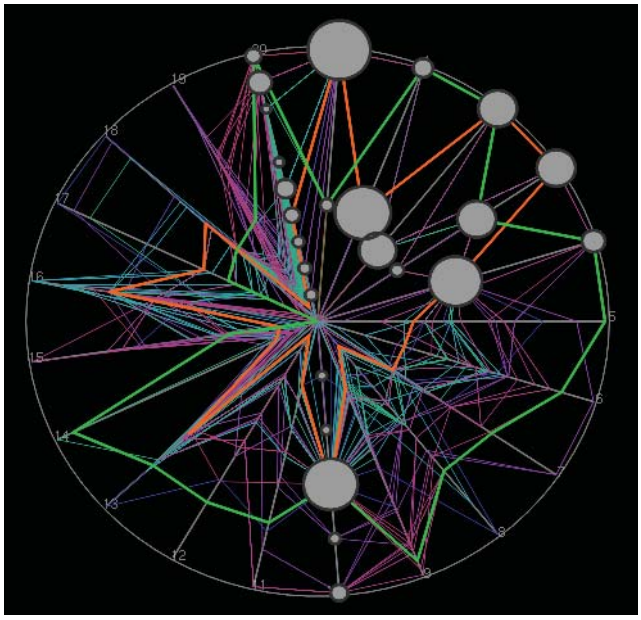


Fig. 6. Highlighting two data items with thick lines in the star plot visualization of the 21-dimensional cars dataset. Circles encode the counts of different data values on categorical dimensions numbered as 0, 1, 2, 3, 4. 10, and 20 (Table I).

considering. If multiple pairwise relationships are of interest, the corresponding axes should be drawn next to each other.

Displaying the aggregation information derived by combining many data cases and variables can further reduce visual clutter. Data items that are similar in most dimensions should be drawn together rather than individually drawing them. Several techniques exist for forming clusters of high-dimensional data with respect to all dimensions. For instance, hierarchical clustering is a popular technique, which has been previously used with other visualization techniques including parallel coordinates [8] but not yet used with star plot visualization. Thus derived clusters can be displayed at different levels of abstraction with proximity-based coloring and structure-based brushing [8].

6 Conclusions

The star plot visualization method has been used in a wide variety of data domains. However, it becomes less effective under situations when the number of data items increases in the dataset and when the dimensionality of the data becomes too high. Many rays/dimensions have to be closely packed within a small circular area for each data item and individual star icons become too small. To overcome these problems, we have proposed different ways of effectively using the star plot visualization method.

First, instead of displaying multiple star-shaped icons spatially separately (one for each data item), we plot all data items together in the same star plot setting. Such overlapping allows the use of a larger visual display for all data and also makes identification of clusters and other features easier. Second, the shifting of the origin away from the fixed center for each ray (radial axis) widens the space between the axes. This shifted origin star plot reduces the ray crowdedness in the lower-value region of the scale. Third, our multilevel star

plot divides the drawing of dimensions in different groups (levels) and maps them to different concentric circular regions (rings). This approach can handle very large number of dimensions such as those expected in big data problems. By visualizing three multivariate data sets of large dimensions (21, 68 and 91 variables or attributes), we anticipate that the proposed star plot visualization extensions can be potentially useful in analyzing ultra-high dimensional multivariate data.

As we have shown in this study, specialized visualizations and methods of interaction with the data are required to gain insight onto large multivariate data set of high dimensionality. Our focus has been mainly on effective representation of a large number of dimensions (variables) for star plot visualization. The supported user interface allows the user to choose desired star plot option, to highlight certain stars (data cases) and to dynamically adjust the dimension partitioning and the ring boundaries. Clearly, there is much to be done to make the proposed star plot extensions useful in a practical sense. Our further work will explore effective ways of managing large-scale data (dimension manipulation, data breakdown), enhancing line rendering (compositing, opacity), deriving and displaying aggregation information (e.g., clustering), and employing user interaction.

7 References

[1] E. Olshannikova, A. Ometov, Y. Koucheryavy, and T.G. Olsson, "Visualizing big data with augmented and virtual reality: challenges and research agenda," J. Big Data, vol. 2, pp. 22, 2015.

[2] J. Zhang, and M. L. Huang "5Ws model for bigdata analysis and visualization," IEEE 16th Int'l Conf. Comput. Sci. and Eng., 2013, pp. 1021-1028.

[3] C. Ware, Information visualization: Perception for design, Morgan Kaufmann Publishers, 2004.

[4] J. Yang, W. Peng, M. O. Ward, and E. A. Rundensteiner, "Interactive hierarchical dimension ordering spacing and filtering for exploration of high dimensional datasets," IEEE Symp. Info. Vis., pp. 105-112, 2003.

[5] S. Few, "Multivariate analysis using parallel coordinates," Perceptual Edge, 2006. (www.perceptualedge.com)

[6] J. Chambers, W. Cleveland, B. Kleiner, and P. Tukey, Graphical Methods for Data Analysis, Wadsworth, 1983.

[7] A. Inselberg, "Parallel coordinate: VISUAL multidimensional geometry and its applications," Springer ISBN 978-0387215075, 2009.

[8] Y. H. Fua, M. O. Ward, and E. A. Rundensteiner, "Hierarchical parallel coordinates for exploration of large datasets," VIS'99 Proc., 1999, pp. 45-50.

[9] E. Kandogan, "Star coordinates: A multi-dimensional visualization technique with uniform treatment of dimensions," Proc. IEEE Info. Vis. Symp., 2000, vol. 650, pp. 22.

[10] W. Cleveland, and M. McGill, Dynamic Graphics for Statistics, Wadsworth, 1988.

[11] M. Friendly, "Stastistical graphics for multivariate data," SAS SUGI Conf. Proc., 1991, pp. 1157-1162.

[12] R. Nancy, The Quality Toolbox, 2005.

[13] B. B. Karki, and R. Chennamsetty, "A visualization system for mineral elasticity," Visual Geosciences, 2004; DOI:10.1007/s10069-004-0020-7; pp. 1-9, 2004.

[14] G. M. Draper, H. Laie, Y. Livnat, and R. F. Riesenfeld, "A survey of radial methods for information visualization," IEEE Trans. Vis. Comp. Graph., pp. 759-776, 2009.

[15] K. Bache, and M. Lichman, "UCI Machine Learning Repository Irvine, CA: University of California, School of Information and Computer Science," 2013. (http://archive.ics.uci.edu/ml)

[16] L. Di Caro, V. Frias-martinez, and E. Frias-martinez, "Analyzing the role of dimension arrangement for data visualization in RadViz," Proc. 14th Pacific-Asia Conf. Adv. Know. Dis. and Data Mining, 2010, vol. 2, pp. 125-132.

[17] A. O. Artero, M. C. F. De Oliveira, "Viz3D: Effective exploratory visualization of large multidimensional data sets," Proc. 17th Brazilian Symp. Comp. Graph. Image Process., 2004, pp. 340-347.

Simulating Spatial Correlation for Catastrophic Events

Georg Hofmann
Validus Research*, Waterloo, Ontario, Canada

Abstract— *Catastrophe models calculate the stochastic distributions of loss originating from events like hurricanes and earthquakes. These models are typically based on a stochastic event catalog. For each event spatial correlation needs to be simulated. The standard approach is based on the evaluation of a copula. However the complexity of the corresponding algorithm is $O(n^3)$ and it becomes difficult to execute for n in the thousands. So as the number n of locations in the footprint of the event grows, this approach quickly becomes infeasible. We propose a slight modification of the well-know Kriging technique, in order to solve this problem. With our solution the creation of simulation data for catastrophe models becomes manageable with the use of Big Data techniques.*

Keywords: Simulation, Spatial Correlation, Copula, Catastrophe Model, Kriging

1. Introduction

Catastrophe models are very common tools in the insurance industry. They are used, among other things, to predict distributions for financial loss originating from events like hurricanes and earthquakes. These models are usually based on a stochastic event catalog. For each simulated event in this catalog it is important to quantify uncertainty in the hazard and the vulnerability of structures affected. For this so-called **secondary** uncertainty it is important to understand spatial correlation. Claims data for historic earthquake losses show that there is a relationship between the distance of two locations and their loss correlation.

In this paper we start from the assumption that such a spatial relationship is given and provide a methodology to simulate secondary uncertainty with the prescribed correlation. The traditional approach to this simulation would be to compute a copula directly. However, this approach does not scale well computationally as the number of locations increases. Instead we propose using a copula on a coarser subset of locations and to interpolate for the finer set using a technique similar to the well-known method of Kriging. As this is an approximation of the intended correlation, its accuracy needs to be investigated. We prove results that provide upper and lower bounds for the accuracy of the simulated correlation.

* At the time of the preliminary research, Kai Cui was a member of the Validus Research team. Thanks go to him for adding the Krigging methodology to the mix of interpolation methods initially investigated.

With the proposed approach simulating single events becomes computationally feasible. A code example is provided in the appendix. However, catastrophe models often simulate many events in order to capture the full stochastic nature of a peril. Simulating as many as 1 million events is not uncommon. Implementing our approach in Big Data framework with increased computational resources allows this to be accomplished.

2. A normalized Kriging approach

In this section we define the necessary details for the proposed normalized Kriging approach. The problem at hand can be summarized in the following way: For a set of locations (referred to in the introduction as the coarser set) random variables are defined. They follow a covariance structure. The objective is the following: For a new location define a random variable as a linear combination of the given ones, such that the covariance structure is closely approximated.

The following special case is instructive: If the locations in the coarser set have normally distributed random variables, then so does a linear combination of them. If in addition to that the covariance structure $c(\cdot,\cdot)$ satisfies $c(x^*,x^*) = 1$ for all locations x^*, then all random variables will follow the standard normal distribution. In this case the covariance structure can equivalently be expressed by a correlation structure.

This case is particularly relevant for simulations. The locations in the coarser set can be simulated using a Gaussian copula. The simulation can be extended to any location outside of the coarser set while maintaining a standard normal distribution and approximating the desired correlation. The code presented in the appendix implements this particular simulation.

We use the variable x with subscripts to refer to locations. For any pair of locations (x^*, x^{**}) we denote by $c(x^*, x^{**})$ the target covariance of the two locations. Suppose the n locations $x_1,\ x_2, \ldots,\ x_n$ in the coarser set have the random variables $f(x_1),\ f(x_2), \ldots,\ f(x_n)$. Further, suppose they satisfy

$$\mathrm{Cov}\left(f(x_i), f(x_j)\right) = c(x_i, x_j)$$

for i and $j = 1,\ 2, \ldots,\ n$.

For every location x^* the objective is to define the random variable $\overline{f}(x^*)$ as a linear combination of the

$f(x_1),\ f(x_2),\ldots,\ f(x_n)$ in a way such that

$$\mathrm{Var}(\overline{f}(x^*) = c(x^*, x^*) \qquad \text{and}$$
$$\mathrm{Cov}(\overline{f}(x^*), \overline{f}(x_i)) \approx c(x^*, x_i)$$

for all i.

Set

$$C = \begin{pmatrix} c(x_1,x_1) & \cdots & c(x_1,x_n) & \\ \vdots & \ddots & \vdots & \vdots \\ c(x_n,x_1) & \cdots & c(x_n,x_n) & \end{pmatrix}$$

We use the notation

$$f(x) = \begin{pmatrix} f(x_1) \\ \vdots \\ f(x_n) \end{pmatrix} \quad \text{and} \quad c(x^*,x) = \begin{pmatrix} c(x^*,x_1) \\ \vdots \\ c(x^*,x_n) \end{pmatrix}.$$

The well-known Kriging estimator $\hat{f}$ is defined in the following way:

$$\hat{f}(x^*) = \big(f(x)\big)^T C^{-1} c(x^*, x)$$

It immediately follows that $\hat{f}(x_i) = f(x_i)$ for each $i = 1,\ldots,n$. It also can be calculated that

$$\mathrm{Cov}(\hat{f}(x^*), \hat{f}(x_i)) = c(x^*, x_i).$$

In other words, Kriging models the prescribed covariance. On the other hand we will prove in this article that

$$\mathrm{Var}(\hat{f}(x^*)) \le c(x^*, x^*)$$

and in general, equality does not hold.

In other words, Kriging does not preserve variance in the same way that it preserves covariance. For the application that we have in mind, the preservation of variance is more important than that of covariance. That is why it is stated as an objective. This leads us to using a normalization: We define the normalized estimator:

$$\overline{f}(x^*) = \sqrt{\frac{c(x^*,x^*)}{\mathrm{Var}(\hat{f}(x^*))}}\,\hat{f}(x^*)$$

This estimator preserves variance:

$$\mathrm{Var}(\overline{f}(x^*)) = c(x^*, x^*).$$

But as a consequence, covariance will be overestimated:

$$\begin{aligned}\mathrm{Cov}(\overline{f}(x^*), \overline{f}(x_i)) &= \frac{c(x^*,x^*)}{\mathrm{Var}(\hat{f}(x^*))}\,\mathrm{Cov}(\hat{f}(x^*), \hat{f}(x_i)) \\ &\ge c(x^*, x_i)\end{aligned}$$

In this article we will derive an upper bound for this overestimation.

3. Upper bound for variance

As promised in the previous section, we prove a proposition about the lower bound of the estimator variance. The proof is necessary for the foundation of the proposed bound. But it is not indispensable for the further understanding of the applications in later sections.

Proposition 3.1:

$$\mathrm{Var}(\hat{f}(x^*)) \le c(x^*, x^*)$$

Proof: The matrix

$$C' = \begin{pmatrix} c(x_1,x_1) & \cdots & c(x_1,x_n) & c(x_1,x^*) \\ \vdots & \ddots & \vdots & \vdots \\ c(x_n,x_1) & \cdots & c(x_n,x_n) & c(x_n,x^*) \\ c(x^*,x_1) & \cdots & c(x^*,x_n) & c(x^*,x^*) \end{pmatrix}$$

is positive semi-definite, in particular its determinant is non-negative.

We will denote the minors of the matrix C by $M_{i,j}$ and the minors of the matrix C' by $M'_{i,j}$. We compute this determinant by expanding along the last column:

$$\begin{aligned}\det(C') = &\sum_{i=1}^{n} c(x_i,x^*)\det(M'_{i,n+1})(-1)^{i+n+1} \\ &+ c(x^*,x^*)\det(\underbrace{M'_{n+1,n+1}}_{=C})\underbrace{(-1)^{2n+2}}_{=1}. \end{aligned} \tag{1}$$

Now compute $\det(M'_{i,m+1})$ by expanding along the last row:

$$\det(M'_{i,m+1}) = \sum_{j=1}^{n} c(x^*,x_j)\det(M_{i,j})(-1)^{j+n} \tag{2}$$

By reinserting (2) into (1), we obtain:

$$\begin{aligned}&\det(C') \\ &= \sum_{i=1}^{n}\sum_{j=1}^{n} c(x_i,x^*)\det(M_{i,j})(-1)^{i+j+2n+1}c(x^*,x_j) \\ &\quad + c(x^*,x^*)\det(C)\end{aligned} \tag{3}$$

Now note that

$$\det(C)C^{-1} = \big(\det(M_{i,j})(-1)^{i+j}\big)_{ij}.$$

Using this in (3), we obtain

$$\begin{aligned}&\det(C') \\ &= -\det(C)c(x^*,x)C^{-1}c(x,x^*) + \det(C)c(x^*,x^*) \\ &= \det(C)(c(x^*,x^*) - \mathrm{Var}(\hat{f}(x^*)))\end{aligned}$$

Rearranging this leads to

$$\begin{aligned}\mathrm{Var}(\hat{f}(x^*)) &= c(x^*,x^*) - \frac{\det C'}{\det C} \\ &\le c(x^*,x^*)\end{aligned}$$

This completes the proof. ■

4. Lower bound for variance

Similar to the upper bound in the previous section, we provide a lower bound in this section.

Proposition 4.1: For every $i = 1, 2, \ldots, n$ the following inequality holds:

$$\mathrm{Var}(\hat{f}(x^*)) \geq \frac{\big(c(x^*, x_i)\big)^2}{c(x_i, x_i)}.$$

In other words,

$$\mathrm{Var}(\hat{f}(x^*)) \geq \max_{i=1,2,\ldots,n} \frac{\big(c(x^*, x_i)\big)^2}{c(x_i, x_i)}.$$

Proof: Define

$$\langle \cdot, \cdot \rangle \;\; \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}, \;\; \langle a, b \rangle = a^T C^{-1} b.$$

This is a scalar product. So the Cauchy-Schwartz inequality holds:

$$\langle a, b \rangle^2 \leq \langle a, a \rangle \langle b, b \rangle$$

By setting $a = c(x, x^*)$ and $b = c(x, x_i)$, we obtain

$$\Big(\big(c(x, x^*)\big)^T \underbrace{C^{-1} c(x, x_i)}_{=e_i}\Big)^2$$
$$\leq \big(c(x_i, x_i)\big)^T \underbrace{c(x, x^*) C^{-1} c(x, x^*)}_{\mathrm{Var}(\hat{f}(x^*))}$$

where e_i denotes the ith unit vector. This can be simplified to

$$\big(c(x_i, x^*)\big)^2 \leq c(x_i, x_i)\, \mathrm{Var}(\hat{f}(x^*))$$

That completes the proof. ■

5. Correlation bounds

We bring the upper and lower bounds derived in the previous two sections together in a form that can be directly applied in examples.

Consider the ratio of the achieved to the intended correlations

$$R_i(x^*) = \frac{\rho\big(\overline{f}(x^*), \overline{f}(x_i)\big)}{\rho\big(f(x^*), f(x_i)\big)} = \frac{\mathrm{Cov}\,\big(\overline{f}(x^*), \overline{f}(x_i)\big)}{\mathrm{Cov}\,\big(f(x^*), f(x_i)\big)}$$
$$= \frac{c(x^*, x^*)}{\mathrm{Var}\,\big(\hat{f}(x^*)\big)}.$$

Since we want to measure, how closely $\rho\big(f(x^*), f(x_i)\big)$ is approximated by $\rho\big(\overline{f}(x^*), \overline{f}(x_i)\big)$, we want to understand, how close $R_i(x^*)$ is to 1. We can apply the bounds found for $\mathrm{Var}(\hat{f}(x^*))$ to the ratio above to obtain:

$$\boxed{\min_{j=1,2,\ldots,n} \frac{c(x_j, x_j) c(x^*, x^*)}{\big(c(x^*, x_j)\big)^2} \geq R_i(x^*) \geq 1}$$

This implies that the technique proposed in this article generally overstates correlation. This can be understood intuitively by considering the extreme case where the coarser set consists of only one location. In that case pairs of locations from the finer set will receive full correlation, regardless of the intended correlation.

The inequality above, however, provides an upper bound to the correlation overstatement. In the next section we will show in an example that the deviation of the correlation from the desired value can be reduced by increasing the resolution of the coarser location set.

6. Controlling the approximation error in the simulation example

We return to the example of simulating standard normal variates for each location. In particular the variances of the variates is 1, so we have $c(x^*, x^*) = 1$ for ever location x^*. As an example of a target correlation assume that $c(\cdot, \cdot)$ be given by

$$c(x^*, x^{**}) = \exp(-0.002\, \mathrm{dist}(x^*, x^{**})),$$

where dist stands for the distance of two locations on the earth measured in kilometers.

We require that the ratio $R = R_i(x^*)$ is between 0.95 and 1.05, which amounts to allowing a 5% error in the simulation of correlation. In other words, we need to make sure that $R < 1.05$. This can be accomplished by assuring that

$$\min_{j=1,2,\ldots,n} \frac{1}{\big(c(x^*, x_j)\big)^2} \leq 1.05$$

This is equivalent to

$$\min_{j=1,2,\ldots,n} \exp\big(0.004\, \mathrm{dist}(x^*, x_j)\big) \leq 1.05$$
$$\min_{j=1,2,\ldots,n} \mathrm{dist}(x^*, x_j) \leq \frac{\ln(1.05)}{0.004} \approxeq 12.2$$

In other words, as long as the location x^* is less than 12.2 kilometers from the nearest location in the coarser location set, the desired accuracy is achieved.

7. Conclusion

Advanced catastrophe models simulate the correlation observed among the losses originating from a catastrophic event. The traditional copula approach is computationally infeasible due to the large number of locations to be considered in the event footprint. We provide an alternative approach to approximating the correlation and show how the accuracy of the approximation can be controlled. This allows catastrophe models with millions of simulated events to be implemented.

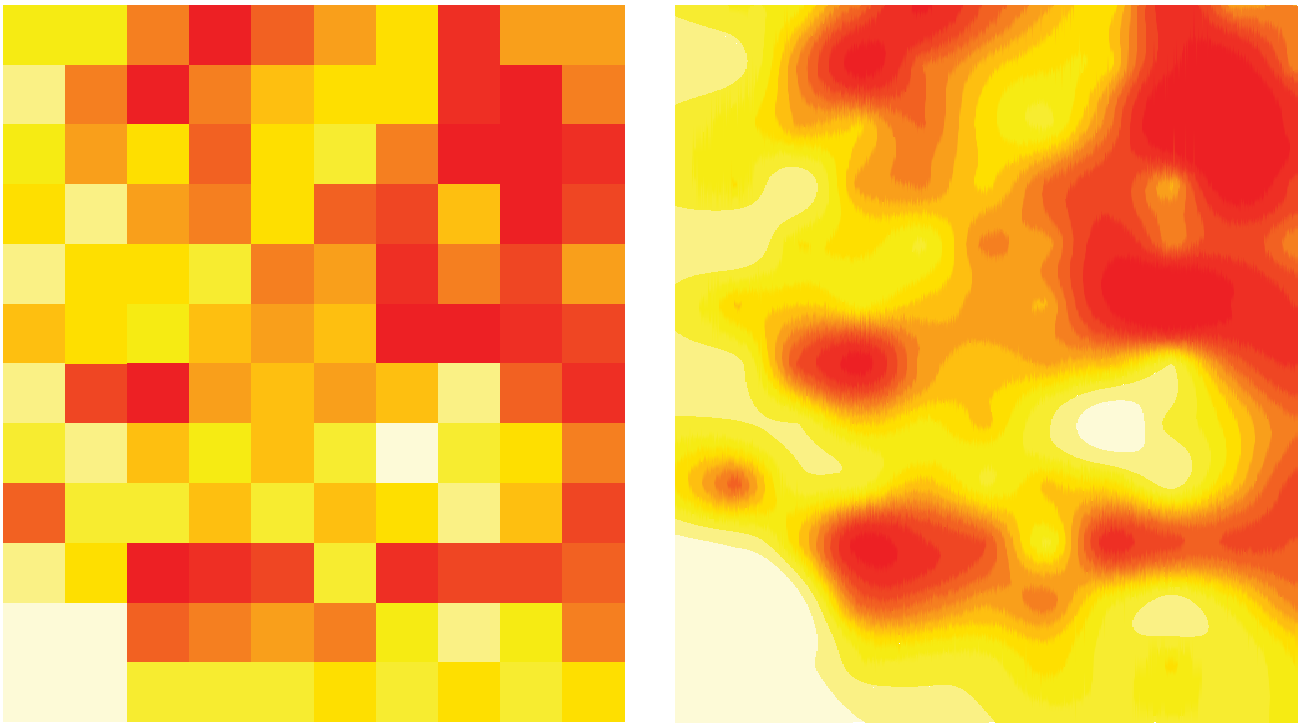


Fig. 1: Output from the R code. Visualization of the copulas with a coarse grid (left) and a finer grid (right).

Appendix

The code below runs in R version 3.1.0 once the MASS package is loaded. Its output can be viewed in Figure 1.

```
# This code is part of the article
    "Simulating Spatial Correlation
# for Catastrophic Events"

### Preset constants.
m1.nrow <- 12
m1.ncol <- 10
m1.num <- m1.nrow * m1.ncol
m2.nrow <- 720
m2.ncol <- 500
m2.num <- m2.nrow * m2.ncol

### Functions
target.cor <- function(x1, y1, x2, y2){
  dist <- sqrt((x1 - x2) ^ 2 + (y1 - y2) ^ 2)
  return(exp(- 0.01 * dist))
}

### Main Code
i1 <- rep(1:m1.nrow, times = m1.ncol)
j1 <- rep(1:m1.ncol, each = m1.nrow)

x1 <- i1 * m2.nrow / m1.nrow
y1 <- j1 * m2.ncol / m1.ncol

x2 <- rep(1:m2.nrow, times = m2.ncol)
y2 <- rep(1:m2.ncol, each = m2.nrow)

covar.mat <- outer(X = 1:m1.num, Y =
    1:m1.num, FUN = function(i, j)
  target.cor(x1 = x1[i], x2 = x1[j], y1 =
      y1[i], y2 = y1[j]))

covar.mat2 <- outer(X = 1:m1.num, Y =
    1:m2.num, FUN = function(i, j)
  target.cor(x1 = x1[i], x2 = x2[j], y1 =
      y1[i], y2 = y2[j]))

require(MASS)
set.seed(606)
sim.1 <- mvrnorm(mu = rep(0, m1.num), Sigma
    = covar.mat)
covar.mat.inv <- solve(covar.mat)
tmp.stdev <- sqrt(colSums((covar.mat.inv %*%
    covar.mat2) * covar.mat2))
sim.2 <- as.vector((t(sim.1) %*%
    covar.mat.inv) %*% covar.mat2) /
    tmp.stdev

layout(matrix(c(1,2), ncol = 2))
par(mar=c(0,1,0,1))
image(pnorm(matrix(sim.1, ncol = m1.nrow,
    byrow = TRUE)),
     xaxt="n", yaxt="n", bty = "n")
image(pnorm(matrix(sim.2, ncol = m2.nrow,
    byrow = TRUE)),
     xaxt="n", yaxt="n", bty = "n")
```

ISBN: 1-60132-427-8, CSREA Press ©

Detecting Sarcastic Tweets: A SentiStrength Modeling Approach

Samaneh Nadali*, Masrah Azrifah Azmi Murad, Nurfadhlina Mohamad Sharef
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
sm.nadeali@gmail.com*,{masrah, nurfadhlina}@upm.edu.my

Abstract— **Recently, Twitter has become a valuable source of people's opinions and sentiments. Thus, sentiment analysis for understanding the sentiment is needed. In the general area of sentiment analysis, sarcasm has an important role because it can change the polarity of a message. Sarcasm is a common phenomenon in social media, which is a nuanced form of language for expressing the opposite of what is written. Several works have been done either at the level of non-hashtag or hashtag-based sentiment analysis. In sarcasm detection; however there is no model for identifying sarcastic tweets to work on both levels. In this article, we present a new Sarcasm Detection Model (SDM) for identifying sarcastic tweets at the level of hashtag and non-hashtag sentiment, based on the strength level of the tweets. In the proposed SDM, three classifiers are used; SentiStrength Sarcasm Classifier (SSC); Sarcasm Hashtags Classifier (SHC) and Hashtag-SentiStrength Sarcasm Classifier (HSSC). SentiStrength Sarcasm Classifier works based on the strength level of tweet, whereas Sarcasm Hashtags Classifier works based on the Sarcasm Hashtags Indicator (SHI) and contrast between the orientation of the tweets and hashtag(s), and Hashtag-SentiStrength Sarcasm Classifier works based on the strength levels of tweet and hashtag(s). These classifiers work better when they are used as part of a coherent model rather than used individually. This research is still on-going where in the future we will need to test the effectiveness of the SDM. Moreover, we except to achieve good result because it is covered several types of tweets (hahstags and non-hashtags).**

Keywords- Sarcasm, Sentiment analysis, Hashtags analysis, Strength level of tweets.

I. INTRODUCTION

With the growth of the web, microblogs such as Twitter are becoming popular day by day. People use Twitter for sharing information and opinions on a variety of topics and to discuss current issues. Therefore, Twitter has become a valuable source of opinion and sentiment. Thus, understanding the opinion of the individuals is needed.

Sentiment Analysis (SA) or opinion mining (OP) is one of the areas of computational studies which deal with opinion-oriented natural language processing such as text source recognition, emotion and mood recognition and opinion oriented summarization [1].

Sentiment analysis was done in different domains such as product reviews, movies and microblogs [2]. Twitter is one of the most popular platforms of microblogs which has been used for all ordinary individuals, politics and companies [3]. Twitter allows registered users to read and post tweets (140 character messages). A Tweet is a short message mostly belonging to the sentence level sentiment classification [4].

All of the studies on sentiment analysis of Twitter messages are term-based [5, 6, 7, and 8]. Previous researchers extracted tweets based on a certain term and then analyzed the sentiment of these extracted Twitter posts. In the general area of sentiment analysis, sarcasm plays a role as an interfering factor that can flip the polarity of a message [9].

In the Oxford English Dictionary (OED) [10] "sarcasm" is defined as "a sharp, bitter, or cutting expression or remark; a bitter gibe or taunt″. Sarcasm may employ ambivalence, although it is not necessarily ironic. Sarcasm might be used to comic effect or can be used to hurt or offend. Unlike simple negation words, a sarcasm message usually expresses a negative opinion utilizing only positive words or even intensified positive words. Detection of sarcasm is important for the development of a sentiment analysis system [9]. In this paper, a sarcasm detection model for tweets is introduced. Twitter is chosen in this study, because it is one of the largest platforms where people tend to express their opinion. Twitter also provides features such as hashtags, which aid in detecting sarcasm in the tweets.

Due to the intentional ambiguity, analysis of sarcasm is a difficult task not only for machine, but also for a human. Although sarcasm detection has an important effect on sentiment, it is usually ignored in social media analysis because sarcasm analysis is too complicated. Since the goal of sentiment analysis is to automatically detect the polarity of a document, misinterpreting sarcasm represents a big challenge [11].

Different studies have been done in sarcasm detection such as; semi-supervised sarcasm recognition, investigation of the impact of lexical and pragmatic factors, identification of sarcasm based on intensifiers and exclamation, contrast between positive and negative situation verb phrases, identifying the relationship between a tweet and an author's past tweet and identifying extra-linguistic information from the context of an utterance on Twitter, such as properties of the author, the audience and the immediate communicative environment [9, 11 ,12, 13, 14, 15 and 16]. All of the mentioned works were only able to identify sarcasm *either* at the level of hashtags *or* non-hashtags. There is no work in sarcasm detection for

identifying sarcasm at the level of hashtag(s) *and* non-hashtag(s). In this paper we present a new Sarcasm Detection Model (SDM) for identifying sarcastic tweets at the level of hashtag(s) and non-hashtag(s) based on the strength level of the tweets.

The rest of this paper is structured as follows: section II investigates related work; section III defines the proposed Sarcasm Detection Model (SDM), three classifiers and features that are used in SDM and last section is the conclusion.

II. Related Work

Automatic sarcasm detection is a relatively new research area. Few studies have been done in automatic sarcasm detection and it is supposed to be a difficult problem.

For detecting sarcastic Dutch tweets, [9] applied unigrams, bigrams and trigrams as features and used a Balanced Winnow classifier.

[11] proposed a method for sarcasm detection on Twitter and product reviews from Amazon. KNN-like classifier with 5-fold cross validation was used in their method. They obtained an F-measure of 0.55 on Twitter dataset and 0.83 on the product reviews dataset. The #sarcasm hashtag was used for acquiring the Twitter dataset. Moreover, they created a balanced evaluation set of 180 tweets. Fifteen annotators via Amazon Mechanical Turk2 were used for labelling their evaluation dataset.

[12] classified tweets into three categories; sarcastic, positive sentiment and negative sentiment. "#sarcasm" and "#sarcastic" hashtags were used for detecting sarcastic tweets. Support Vector Machine (SVM) with Sequential Minimal Optimization (SMO) and logistic regression were used in their method. Different combinations of dictionary-based and unigrams features and pragmatic features such as positive and negative emoticons and user references were used. They achieved accuracy of 0.65 for classifying tweets into sarcastic and non-sarcastic using the combination of SVM, SMO and unigrams. They employed three human for annotating 180 tweets (90 sarcastic and 90 non-sarcastic). For classifying tweets into positive and negative, 50 sarcastic and 50 non-sarcastic (25 positive and 25 negative) tweets with emoticons annotated by two judges were used. The accuracy for automatic classification and human judges was 0.71 and 0.89 respectively.

[13] proposed features for capturing properties of figurative languages such as polarity, unexpectedness, ambiguity and emotional scenarios. Five categories such as irony, humor, technology, politics and general were used in their corpus. Classification of irony and general tweets achieved the best results with F-measure of 0.65.

[14] proposed a systematic approach for identifying sarcasm effectively. They analyzed the context of the tweets and the behavioral traits of the users derived from their past activities. They built a comprehensive supervised framework for identifying sarcasm tweets by observing user behavior patterns on Twitter. Although they obtained reasonable results, other factor such as user´s social networks and their current and past interaction were not considered in their method.

[15] introduced a method for identifying sarcasm based on contextual phenomenon. A series of experiments were presented by them. For identifying the effect of extra-linguistic information on the sarcasm detection, they used features not only derived from the context of the message, but used information about the author, his relationship to his audience and the immediate communicative context they both share.

[17] examined the relevance and representativeness of conceptual features such as unexpectedness, style, signatures and emotional scenarios. These features contain quotes, capitalized words, punctuation marks, emoticons, lexicon-based features, character n-grams, and skip-grams [18]. Four categories such as humor, irony, politics and education (100,000 tweets for each category) were used in their corpus. Two distributional scenarios i.e. balanced distribution and imbalanced distribution (25% ironic tweets and 75% tweets from all three non-ironic categories) using the decision tree and Naive Bayes algorithms [19] were used in their evaluation. The classification by decision tree achieved F-measure of 0.53 on the imbalanced distribution and an F-measure of 0.72 on the balanced distribution.

[20] proposed a new approach for identifying sarcasm based on the sarcasm indicator and contrast between the orientation of the tweet and hashtags(s). Although, their work was a primary work on sarcasm detection at the level of the hashtags, the number of sarcasm indicators was small. Moreover, they could not tokenize all hashtags correctly. For example "#greatstart" is tokenized as "greats tart" which is not correct.

Sarcasm and nastiness classification in online dialogues was also explored in [21] using bootstrapping, syntactic patterns and a high precision classifier. They achieved an F-measure of 0.57 on their sarcasm dataset.

[23] proposed a bootstrapping algorithm for acquiring a list of positive sentiment phrases and negative situation phrases from tweets. Their method has been able to identify just one type of sarcasm: contrast between a positive sentiment and negative situation. Their method obtained an F-measure of 0.51 using the SVM classifier.

The proposed SDM is different from all of the related work in sarcasm detection, where SDM is able to identify sarcastic tweets based on the strength level of the tweet (at the hashtags and non-hashtags level). More detail of SDM is explained in section III.

III. Sarcasm detection model (SDM)

Figure 1 illustrates the SDM. As mentioned before, our proposed model (SDM) is able to identify sarcastic tweets whether it contain hashtag(s) or not. Based on Figure 1, if a tweet does not contain any hashtag, we apply P1. Otherwise P2 is applied on the tweet. Three classifiers are used in our proposed model (SDM); SentiStrength Sarcasm Classifier (SSC), Sarcasm Hashtags Classifier (SHC), and Hashtag-SentiStrength Sarcasm Classifier

(HSSC). Each of the classifiers will be explained in the following subsections.

A. SentiStrength Sarcasm Classifier (SSC)

SSC is applied for tweets that do not have hashtag. [9] found that people used exaggeration with sarcasm. Moreover, a previous study in sentiment analysis works on the impact analysis of the lexical and pragmatic features such as emoticons, interjection, and exclamation marks. We applied these two ideas in the proposed classifier (SSC) for identifying sarcastic tweets based on the strength level of the tweets using and used lexical and pragmatic features. Our most important contribution is to find sarcastic tweets based on the strength level of tweet using lexical and pragmatic features. More details of the features are explained below.

- Lexical Features

Two types of lexical features named as positive - negative situation phrases [24] and dictionary-based are used in SSC. The dictionary-based features were derived from i) SentiStrength dictionary [25], which is used to measure the strength of positive and negative sentiment in short informal texts. For a set of short texts, SentiStrength will allocate a positive/negative strength between 1 to 5; ii) SentiWordNet 3.0 is a resource of lexical for sentiment classification presented by [26]. SentiWordNet 3.0 is an improvement of SentiWordNet introduced by [27]; iii) AFINN dictionary [29]. Each of the English words has an emotional value in AFFIN. This value was calculated based on the psychological reaction of a person to a specific word; iv) list of positive verb phrases and negative situation phrases; v) list of interjections (e.g., ah, oh, yeah); vi) elongated words (stretched words); vii) capital words; and viii) punctuations (e.g., !, ?). In order to find the strength value of the tweet, at first we need to find the strength value for each of the mentioned features. Each of the words in SentiStrength and AFINN has a value between 0, ±5. Since, the boundary in SentiWordNet dictionary is between 0, ±1. So, we multiple each of the values in SentiWordNet to 5. Finally, we have a comprehensive dictionary which has a value between 0, ±5. Furthermore, the strength value for each of the positive verb phrases and negative situation phrases is determined based on the comprehensive dictionary.

- Pragmatic Features

We used two pragmatic features: i) positive emoticons such as smileys and ii) negative emoticons such as frowning faces.

- SentiStrength Formula

After determining the value for lexicon features, we identify SentiStrength of the tweet based on the number of Interjection, Elongate words, Capital Words and Question Mark. Before determining the SentiStrength, all of the slang words are converted to the normal word such as: "gr8" is converted to "great". Finally, we compute SentiStrength of the tweet using our proposed formula:

$$Strength\ (pos) = \sum Opw_{(pos)} \quad (1)$$

$$Strength\ (Neg) = \sum Opw_{(Neg)} \quad (2)$$

$$Strength = Max\ (Strength\ (pos), |\ Strength\ (Neg)\ |) * 2^{No.Interjection+} * 2^{No.Exclamation} * 2^{No.questionMark} * 2^{No.ElongateWord} * 2^{CapitalWords} * 2^{No.Emoticon} \quad (3)$$

Equation (1) and (2) calculate the sum of the strength of positive and negative opinions words in a tweet. After determining the strength of positive and negative opinion words, we are going to calculate the final strength value. Equation (3) demonstrates how final strength value is calculated.

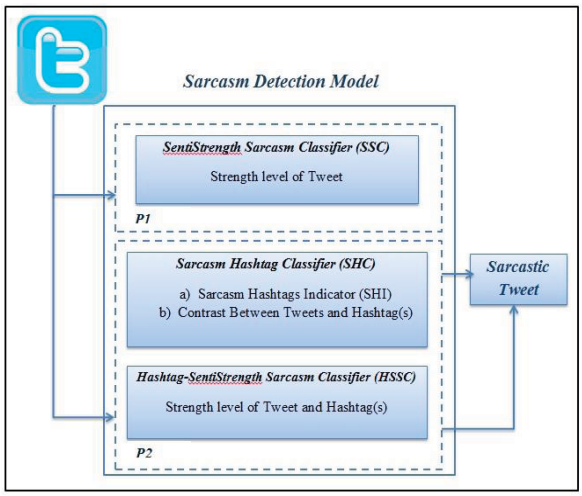


FIGURE 1. SARCASM DETECTION MODEL (SDM)

No.Interjection, No.Exclamation, No.questionMark, No.ElongateWord, No.CapitalWords and No.Emoticon respectively represent the frequency of occurrence of interjection, exclamation, question mark, elongate words, capital words and emoticons.

B. Sarcasm Hashtags Classifier (SHC)

In section A, we identify sarcastic tweets when there is no hashtag in the tweet. Based on Figure 1, if a tweet contains hashtag(s), P2 is applied on the tweet. P2 consists of two classifiers, SHC and HSSC. This section presents the details of Sarcasm Hashtags Classifier (SHC). In general, SHC works based on two main steps; Sarcasm Hashtags Indicator (SHI) and contrast between the polarity of the tweet and hashtag(s). Sarcasm Hashtags Indicator (SHI) is a list of hashtags that indicates sarcasm. To date, no systematic approach has been done to extract sarcasm indicator. In 2014, [20] just work on sarcasm detection using sarcasm indicator. In their approach, a list of sarcastic hashtags from corpus random tweets was collected manually. Then, they extended this list by automatically collecting pairs of hashtags where one hashtag contained an existing sarcasm hashtag (e.g. #sarcasm), using the GazetteerListCollector GATE plugin. For example, the following tweets contain pairs of sarcastic hashtags:

I love living with a 7 year old ***#NotReally #sarcasm***

They then added the other hashtag to their list of sarcasm indicators, e.g. "#notreally", "#sarcasm". Although, their approach was a primary work in sarcasm detection at the level of hashtags, they did not use a systematic approach for extracting sarcasm indicator. Moreover, the number of the indicators used is very small i.e. only 77 sarcasm indicators were extracted [20].

In our proposed classifier (SHC), we applied systematic approach used by [24]. Extracting SHI contains three main steps as follow:

- extract candidate and score them based on estimating the probability
- compute the number of times the candidate appear in the tweets and
- rank the candidate based on the probability and frequency.

First of all, a list of sarcasm hashtags from a corpus of random tweets is extracted. Then the list is extended by automatically collecting pairs of hashtags where one hashtag contained an existing sarcasm hashtag (either #sarcasm or #sarcastic). For example, the following tweets which contain #sarcasm and #sarcastic are used for extracting SHI candidate. From the flowing tweets, #NotReally is added into the list of Sarcasm Hashtags Indicator (SHI) candidate. If one or more of the SHIs are presented in a tweet, it is considered as a sarcastic tweet. The SHI is then used in Sarcasm Hashtags Classifier (SHC) to facilitate sarcasm identification.

Secondly, after extracting the Sarcasm Hashtags Indicator (SHI) candidates, each candidate from the SHI is scored by estimating the probability. The score of SHI is computed based on [24] approach i.e. the number of times the Sarcasm Hashtags Indicator (SHI) candidate appear in sarcastic tweets divided by the number of times the candidate appears in all tweets (Eq.4). Candidates that have a frequency < 3 in the tweets collection are discarded.

$$\frac{|\text{follows(sarcasm hashtags indicator) \& sarcastic}|}{|\text{follows(sarcasm hashtags indicator)}|} \quad (4)$$

Finally, the candidates are ranked based on this probability, using their frequency as a secondary key in case of ties. The top candidate with a probability >.90 are added to the Sarcasm Hashtags Indicator (SHI) list. In the proposed model (SDM), 23 00 Sarcasm Hashtags Indicator (SHI) are extracted. Samples of indicators are shown in Table 1.

TABLE 1. EXAMPLE OF SARCASM HASHTAGS INDICATOR (SHI)

Sarcasm Hashtags Indicator
#funny, #notreally, #FatFuck, #yeahright, #lifewasamazingtoday, #fantastic, #ugh, #iloveyouthough

Although, identifying SHI helps us to identify sarcastic tweets more easily, it is not sufficient and we need to analyze more. Based on Figure 1 the second part of SHC works based on the contrast between the orientation of the tweet and hashtag(s). Hashtags are words or un-spaced phrases that are followed by hash character (or number sign), "#", to form a label [1]. Among Twitter users, hashtags are a convention for creating and following a thread of discussion. Popular hashtags and words are used in trending topics. Twitter users also use hashtags for expressing their feelings, so most of the hashtags contains sentiment orientation such as, #bad, #sad, and #love which can flip the polarity of the tweets. For these reasons, in this study, sentiment analysis of the hashtags is considered. In order to identify the polarity of the hashtag(s), at first we need to tokenize them. In our proposed classifier (SHC) we used Norvig algorithm [23] which helps us to tokenize hashtags more accurately. After tokenizing hashtags, we identify the polarity of the tweets and hashtag(s).

Several lexical features such as positive verb phrases and negative situation phrases [24], SentiStrength [25], SentiWordNet dictionary [26] and AFINN [28] are used. Moreover, CMU tagger [29] and some linguistic rule such as negation rules are applied for finding the orientation of the tweet and hashtag(s). Table 2 illustrates negation rules. After recognizing the orientation of the tweet and hashtag(s), if there is a contrast between the orientation of the tweet and hashtag(s), we consider the tweet as sarcastic. For instance, the following tweet is sarcastic because it contains negative situation phrases (wake up) followed by positive hashtag (greatstart).

Wake up at 5 am. #greatstart.

TABLE 2. NEGATION RULES

Negation Rules
Negation +Negative → Positive //e.g., "no problem"
Negation +Positive → Negative // e.g., "not good" Negation
Neutral → Negative // e.g., "does not work",

C. Hashtag-SentiStrength Sarcasm Classifier (HSSC)

In section B, we identify sarcastic tweets based on the SHI and contrast between the polarity of the tweet and hashtag(s). People usually use hashtag(s) to emphasize and attract attention; we use these two facts for proposing a new classifier for identifying sarcastic tweets based on the strength level of the tweet and hashtag(s).

For recognizing sarcastic tweet at the level of non-hashtag; two features such as lexical and pragmatic features are used in SentiStrength Sarcastic Classifier (SSC) (see section A). The features and methodology for HSSC is the same as SSC where, Eq (1, 2 and 3) is applied for tweet and hashtag(s) separately. If the total SentiStrength value of the tweet is more than ≥ 3, we consider the tweet as sarcastic. Table 3 illustrates sample of the tweets that our proposed model (SDM) is able to identify.

TABLE 3. SAMPLE OF SARCASTIC TWEETS THAT IDENTIFIED IN SDM

Sarcastic Tweets
1) *Being ignored <3 ☺.*
2) *Wake up at 5 am. #greatstart.*
3) *waiting forever for the doctor #fantastic*

IV. Discussion and Future Work

The proposed SDM model for sarcasm detection will help us overcome the problem of misinterpreting the sarcasm. Two key difficulties of sarcasm detection namely detection at the level of hashtags and non-hashtags, and detection based on the strength level of tweet can be overcome. Three classifiers namely SSC, SHC and HSSC are used in the SDM. These classifiers work better when they are used as part of a coherent model rather than used individually. The novelty of the proposed model (SDM) is in identifying sarcastic tweets by analyzing strength level of the tweets. Moreover, our proposed model is applicable at the level of hashtags and non-hashtags whereas the previous works were done either at the level of hashtags or non-hashtags. In our model, different types of sarcasm such as positive phrase followed by negative situation phrases, and tweets contain hashtags and tweets without hashtags are analyzed which help us to achieve better results. The proposed SDM model works only on Twitter data. In future, we are going to work on other types of social media such as Facebook.

References

[1] A. Kumar, and T.M.Sebastian, Sentiment analysis on twitter. IJCSI International Journal of Computer Science Issues, 9(3):372–378, 2012.

[2] K. L.,Liu, W. J., Li, & M.Guo, Emoticon Smoothed Language Models for Twitter Sentiment Analysis. InAAAI, 2012 Jul 22.

[3] I. S. Himelboim I, S.McCreery, M.Smith, Birds of a feather tweet together: Integrating network and content analyses to examine cross‐ideology exposure on Twitter. Journal of Computer‐Mediated Communication. 1;18(2):40-60, 2013 Jan.

[4] G. Gebremeskel, Sentiment Analysis of Twitter posts about news. Sentiment Analysis. Feb. 2011.

[5] A. Go, R. Bhayani, L. Huang, Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford. 2009 Dec;1:12.

[6] A. Bermingham and A.F., Smeaton, Classifying sentiment in microblogs: is brevity an advantage?. InProceedings of the 19th ACM international conference on Information and knowledge management,pp. 1833-1836. ACM, 2010 Oct 26.

[7] A. Pak, and P. Paroubek, "Twitter as a Corpus for Sentiment Analysis and Opinion Mining." In LREC, vol. 10, pp. 1320-1326. 2010.

[8] L. Barbosa, and J. Feng, "Robust sentiment detection on twitter from biased and noisy data." In Proceedings of the 23rd International Conference on Computational Linguistics: Posters, pp. 36-44. Association for Computational Linguistics, 2010.

[9] C. C., Liebrecht, F. A. Kunneman, and A. P. J. van den Bosch, "The perfect solution for detecting sarcasm in tweets# not.",2013.

[10] O. E. Dictionary, "Oxford: Oxford university press.",1989.

[11] D. Davidov, O. Tsur,"Enhanced sentiment learning using twitter hashtags and smileys." In Proceedings of the 23rd International Conference on Computational Linguistics: Posters, pp. 241-249. Association for Computational Linguistics, 2010.

[12] R. González-Ibánez, S. Muresan. and N. Wacholder," In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2, pp. 581-586. Association for Computational Linguistics, 2011.

[13] A. Reyes, P. Rosso and D. Buscaldi, "From humor recognition to irony detection: The figurative language of social media." Data & Knowledge Engineering 74 (2012): 1-12.

[14] A. Rajadesingan, R. Zafarani and H. Liu, "Sarcasm detection on Twitter: A behavioral modeling approach." In Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, pp. 97-106. ACM, 2015.

[15] D. Bamman, and N.A. Smith, "Contextualized Sarcasm Detection on Twitter." In Ninth International AAAI Conference on Web and Social Media. 2015.

[16] O. Tsur, D. Davidov and A. Rappoport, "ICWSM-A Great Catchy Name: Semi-Supervised Recognition of Sarcastic Sentences in Online Product Reviews." In ICWSM. 2010.

[17] A. Reyes, P. Rosso and T. Veale, "A multidimensional approach for detecting irony in twitter." Language Resources and Evaluation 47, no. 1 (2013): 239-268

[18] D. Guthrie, B. Allison, W. Liu, L. Guthrie, and Y. Wilks, "A closer look at skip-gram modelling." In Proceedings of the 5th international Conference on Language Resources and Evaluation (LREC-2006), pp. 1-4. 2006

[19] I.H. Witten, and E.Frank, Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, 2005.

[20] D. Maynard, and M.A. Greenwood, "Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis." In Proceedings of LREC. 2014.

[21] S.Lukin, and M. Walker, "Really? well. apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue." In Proceedings of the Workshop on Language Analysis in Social Media, pp. 30-40. 2013.

[22] D. Guthrie, B. Allison, W. Liu, L. Guthrie and Y. Wilks, "A closer look at skip-gram modelling." In Proceedings of the 5th international Conference on Language Resources and Evaluation (LREC-2006), pp. 1-4. 2006.

[23] P.Norvig, "Natural language corpus data." Beautiful Data (2009): 219-242.

[24] E. Riloff, A. Qadir, P. Surve, L. De Silva, N. Gilbert, N. and R. Huang, "Sarcasm as Contrast between a Positive Sentiment and Negative Situation." In EMNLP, pp. 704-714. 2013.

[25] M. Thelwall, K. Buckley, and G. Paltoglou, "Sentiment strength detection for the social web." Journal of the American Society for Information Science and Technology 63, no. 1 (2012): 163-173.

[26] S. Baccianella, A. Esuli, A. and F. Sebastiani, "SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining." In LREC, vol. 10, pp. 2200-2204. 2010.

[27] A. Esuli, and F. Sebastiani, "Sentiwordnet: A publicly available lexical resource for opinion mining." In Proceedings of LREC, vol. 6, pp. 417-422. 2006.

[28] M.M., Bradley, and P.J., Lang, Affective norms for English words (ANEW): Instruction manual and affective ratings. Technical Report C-1, The Center for Research in Psychophysiology, University of Florida, 1999.

[29] O. Owoputi, B. O'Connor, C. Dyer, K. Gimpel, N. Schneider and N.A., Smith,"Improved part-of-speech tagging for online conversational text with word clusters." Association for Computational Linguistics, 2013.

Why We Need Big Data?

E. Kinoshita[1], **and T. Mizuno**[1]
[1]School of Urban Science, Meijo University, Kani, Gifu, Japan
kinoshit@meijo-u.ac.jp

Abstract – *We describe necessity of Big Data from aspects of macroeconomics. In service science capitalism, measurements of values of products need Big Data to construct large knowledge systems. Service products are classified into stock, flow, and rate-of-flow-change. Immediacy of Big Data implements and makes sense of each classification. And we provide a macroeconomic model with behavioral principles of economic agents. The principles have mathematical representation with high affinity of correlation deduced from Big Data.*

Keywords: macroeconomics, service science, Thetical economics and Antithetical economics

1 Introduction

To manage our society, we need appropriate use of Big Data. Complication of society increases issues which we have to solve. If we solve an urban problem, the solution makes new issues. It is represented ironically as Parkinson's law. Strategic solutions for the issues often need large data. Urbanization, which makes ICT social infrastructures, and appearance of Big Data change our approaches for the issues. Before appearance of Big Data, we search data for causal relationships. But after Big Data, because of its huge size, we can acquire sufficient correlation to solve the issues. It means that correlation substitutes for causation (Figure 1).

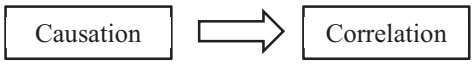


Figure 1. Relationship in the world dominated by Big Data.

ICT reduces sectionalism of governments and enables us to solve urban problems by collaboration of mutual sections. Importance of a person who plays role of control tower of each section increases in such mutual society. The person is CEO or president in corporations, and is prime minister in government. Decision makings of the person design our society. Grand design of society consists of designs of each sector of the society. Each design must be verified that the design is in accordance with the grand design. We need Big Data to construct a strategy by integrating designs and to verify accordance between the strategy and the designs.

The society is dominated by economics. Big Data from economic activities are important us for our decision makings or for the construction of the strategy. In this paper, we describe necessity of Big Data from aspects of macroeconomics.

2 Big Data for service science capitalism

In 1980's, researchers of macroeconomics recognized difference between goods products and service products, and they have tried to define what service products are. Now, service products are defined as products that have properties: intangibility, immediacy, variability, perishability, and customer's high satisfaction.

A major premise of macroeconomics is that our world is capitalism. If the world is not capitalism, then every theory of macroeconomics has non-sense. So, researchers of macroeconomics, managers of companies, or government administrators must consider whether we are in the world with capitalism.

The most important concept of capitalism is fixed price sales. Fixed price sales enable us to run our planned business and guarantee value of capitals.

To enforce fixed price sales without any contradictions on our business, we must measure values of our products precisely. In a word, precise measurements of products provide bases of every index about economics and managements in the world of capitalism; the measurement of values of products is an element forming economics and managements.

For any goods products, we can measure its values relatively easily. Because the goods have physical entities and properties, we can reduce eventually their values to their length, weight, temperature, velocity, or entropy.

On the other hand, we cannot measure values of service products easily. Service products often stand on relations between goods and goods, or between services and services. Relationship is combinations of products, and increasing the number of the combinations makes measurements of values of the products complex. As service products consist of some lower level services, they are developed in high abstraction level far from physical goods products. To overcome the complexity and the distance abstraction level, we need much knowledge of many fields.

In early 2000's, IBM researchers advocated a necessity of "service science" which is a new research filed to construct

knowledge systems for service products. We need accumulation of knowledge. It means that we must collect Big Data and extract new theories form Big Data.

We refer to a society in which almost all employees work for service industry as service science capitalism society. In the society, every price value has large amount of information in the background of the value, and the value is detected in high abstraction level far from its physical entity. To fill the gap between abstraction levels, we must learn techniques which reduce from Big Data to a value through experience.

3 Big Data and classification of services

Big Data provide us new measurements for service products, and enable us to classify service products into three services: stock service, flow service, and rate-of-flow-change service. Stock service is construction of social infrastructures or information infrastructures. Flow service is ordinary everyday service which provided by government administrators and private companies. Rate-of-flow-change service is unusual service.

There is an analogy between physics and economics. In physics, a phenomenon is described in distance, velocity, and acceleration. Establishing the three concepts makes modern physics since 17th century. While economics was made by establishing three concepts: stock, income, and growth rate. In economics, a product is described in the three concepts. Distance, velocity, and acceleration in physics correspond to stock, income, and growth rate in economics, respectively. Distance and stock are measured by some accumulations. Velocity and income are represented in time differentiations. Acceleration and growth rate are represented in twice differentiations. The classification of service products corresponds to the concepts of physics and economics. We summarize it in Table 1.

The classification presumes that we can trace changes of values of service products every times. It corresponds to time derivative in physics. Immediacy of Big Data provides us feasibility the classification.

4 A macroeconomic model on Big Data

When we use Big Data sufficiently, correlation plays important roles in any analyses of economics. So we must build macroeconomic models which we can construct by detecting parameters from correlation deduced from Big Data.

4.1 Thetical economics and Antithetical economics

Kinoshita provides a macroeconomic model which is referred to as "Thetical economics and Antithetical economics." That is a rearrangement of theories of macroeconomics into two set; a set of them is Thetical economics and another set is Antithetical economics. If Say's law is valid in an economic phase in an economic cycle, then the Thetical economics dominates the phase. We feel that we are in normal economy and economic growth in the phase. While if the Keynes's effective demand is effective in an economic phase, then the Antithetical economics dominates the phase. We feel that we are in depressed economy in the phase. Economic phases dominated by Thetical economy and economic phases dominated by Antithetical economy are illustrated in Figure 2.

Easy to say, Thetical economics represents what prosperity is, while Antithetical economics represents what recession is.

With the macroeconomic model, we can provide behavioral principles of economic agents such as corporations and governments as follows:

A principle of corporations under Thetical economics

Objective function (maximize profits)

$$\text{Max } \sum_{j=1}^{n} c_j x_j \quad (1)$$

Constraint condition

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i, \qquad i = 1, \dots, m \quad (2)$$

A principle of corporations under Antithetical economics

Objective function (minimize debts)

$$\text{Min } \sum_{i=1}^{m} u_i b_i \quad (3)$$

Constraint condition

$$\sum_{i=1}^{m} a_{ij} u_i \geq c_j, \qquad j = 1, \dots, n \quad (4)$$

Following list is correspondence of variables and its meanings.

x_j : The number of units of a product j made by the corporation.

c_j : The amount of profits of one unit of a product j; $P_j - (1 + r)h_j$, where P_j is price of the product j, r is interest rate, and h_j is cost of the product j.

a_{ij} : Costs in an account subject i to produce the product j for one unit.

b_i : The amount debts of an account subject i.

u_i : Unpaid balance rate for the accounting subject i; $u_i = 1 - \text{amortization\_rate}$.

A principle of governments under Thetical economics

Objective function (fiscal reconstruction)

$$\text{Min } \sum_{j=1}^{N} G_j K_j \quad (5)$$

Constraint condition

$$\sum_{j=1}^{N} A_{ij} K_j \geq B_i, \qquad i = 1, \dots, M \quad (6)$$

A principle of governments under Antithetical economics

Objective function (fiscal stimulus)

$$\text{Max } \sum_{i=1}^{M} Y_i B_i \quad (7)$$

Constraint condition

$$\sum_{i=1}^{M} Y_i A_{ij} \leq G_j, \qquad j = 1, \dots, N \quad (8)$$

Following list is correspondence of variables and its meanings.

K_j : A rate of the remainder of national loans for an administrative service j. Increasing the rate increases expenses of the service j.

G_j : Demand for funds as national loans for an administrative service j.

A_{ij} : Satisfaction of a resident i when the government gives the resident one unit of costs of a service j.

B_i : A desiring level of total services of the government for a resident i.

Y_i : The amount of public money to increase satisfaction by one unit for a resident i.

In usual studies of the macroeconomics, economic agents, such as customer, corporations, and governments, are modeled simply. All agents expand their profits, they are well-disciplined, they can acquire all information of markets, and their behavior is rational. The principles, which we provide, give a concrete mathematical model of the rationality.

The behavioral principle is linear equation system. Construction the principle is detecting parameters of the equations. So, the model has high affinity with correlation obtained from Big Data.

4.2 Macroeconomic explanation on the model

As an example, we provide an explanation of macroeconomic phenomena in Japan since 1980 with the model. Let us see Figure 3, which represents transition of financial net worth of corporations (non-financial enterprises) in Japan. Japan is dominated by Thetical economics before 1995, and is dominated by Antithetical economics after 1995.

Before 1995, corporations increase investments. It is an evidence of behavior of maximization of their profits; the Japanese economy was dominated by Thetical economics. In Japan, Heisei bubble collapse at February 1990. Five years later, Japanese economy was into recession in 1995. Since the year, corporations decrease their debts and increase their savings. It shows a change of behavioral principle of them; the economy is dominated by Antithetical economics.

GDP (Gross Domestic Products) is a macroeconomic index which represents business conditions of the nation. GDP (often denoted in Y) is sum of national consumption (C), national investment (I), governmental fiscal stimulus (G), and trade gap(E).

$$Y = C + I + G + E \quad (9)$$

Transition of GDP of Japan is shown in Figure 4. From the change of the index, we can confirm that Japanese corporations do not expand their profits since 1995.

5 Conclusions

We describe necessity for our society of Big Data in the macroeconomic aspect. Because our society has entered service science capitalism, the necessity becomes larger. In this paper, we explain that we need Big Data for measurements of service products, and we provide a macroeconomic model which can be constructed from correlation deduced from Big Data. As an example of use of the model, we show analyses of Japan since 1980. We enforce to analyses of other countries with the model in future works.

6 Acknowledgement

In Figure 3, financial net worth of non-financial enterprises in Japan from 1980 to 2015, we use data arranged by Dr. Takanobu Hiromiya.

7 References

[1] Eizo Kinoshita. "A Proposal of Primal and Dual Problems in Macro-Economics"; China-USA Business Review, Vol. 10, No. 2, 115-124, February 2011.

[2] Eizo Kinoshita. "Why Bubble Economy Occurs and Crashes? --Repeated History of Economic Growth and Collapse"; Chinese Business Review, Vol. 10, No. 2, 102-111, February 2011.

[3] Eizo Kinoshita, "A Proposal of Thetical Economy and Antithetical Economy-Mechanism of Occurrence and Collapse of Bubble Economy"; Journal of Business and Economics (Academic Star Publishing Company), Volume 3, No. 2, 117-130, February 2012.

[4] Eizo Kinoshita, Takafumi Mizuno. "Trap of Economics the World Has Fallen in - A Survey of Kinoshita Theory in Macro-Economics"; European Scientific Journal SPECIAL edition, 75-80, 2015.

[5] Eizo Kinoshita. "A Proposal of Thetical Economy and Antithetical Economy by Using Operations Research Techniques"; European Scientific Journal July 2015 edition Vol.11, No.19, 29-48, July 2015.

Table 1: Classification of Service Goods

Mathematics	Physics	Economics	Service goods	Service Examples	Measuring Service Values
Original variation	distance	Asset (stock)	Stock service	Government Service Social Security System Social Infrastructure Information Infrastructure	Measurement of Stock Service Values Integrating with time axis : cost-benefit analysis
Differentiate once with time	speed	Income (flow)	Flow service (common service)	Fast food Convenience stores Yoshinoya's "beef bowls,"etc.	Measurement of flow service values Differentiating with time axis: CS research
Differentiate twice with time	accelera tion	Growth rate (rate of flow variability)	Rate-of-flow-change service (uncommon service)	Kagaya Ritz-Carlton, Osaka Gion	Measuring Value of flow rate of variability service Comfortable variation : fractal measurement

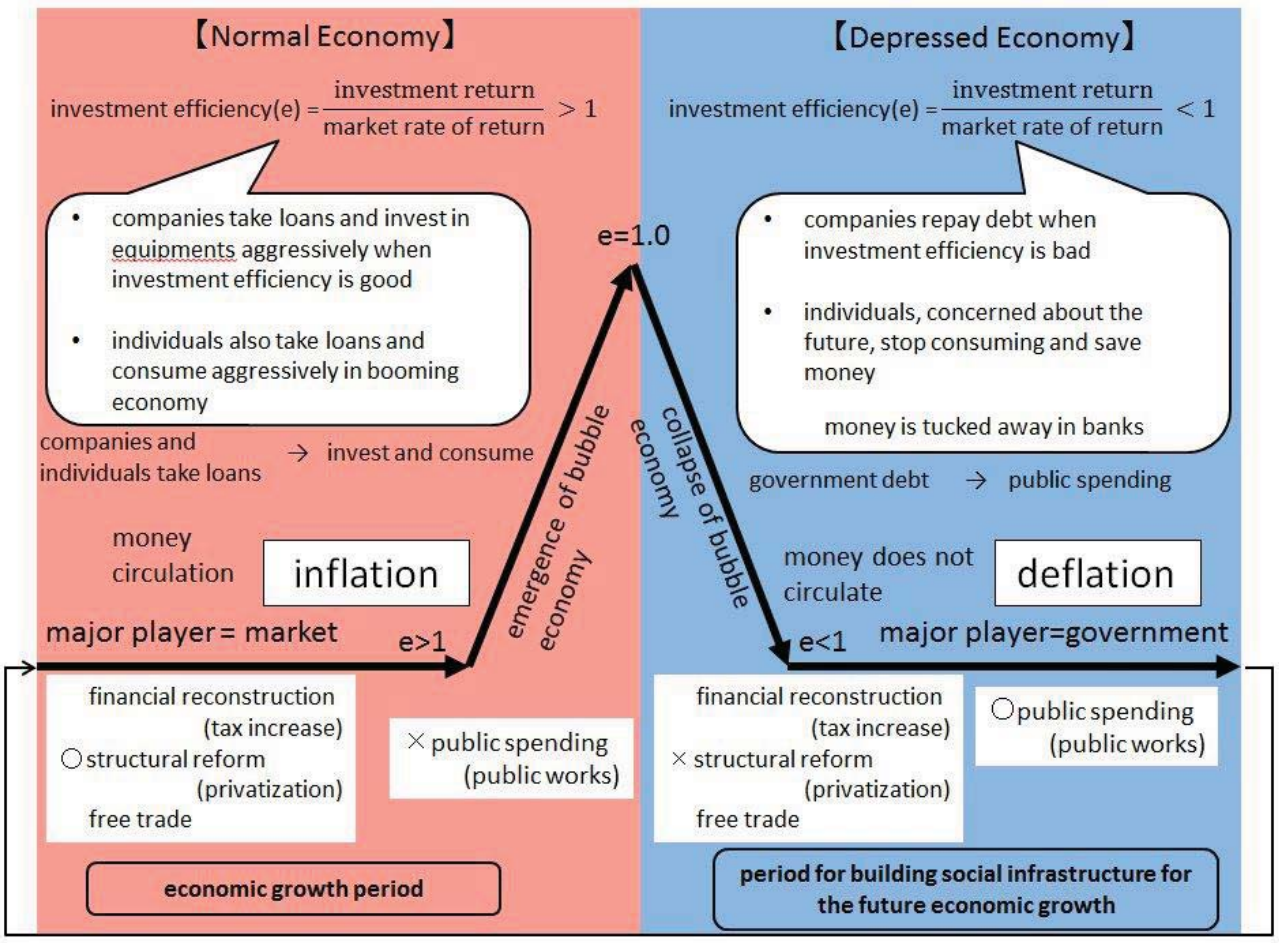


Figure 2. A macroeconomic cycle.

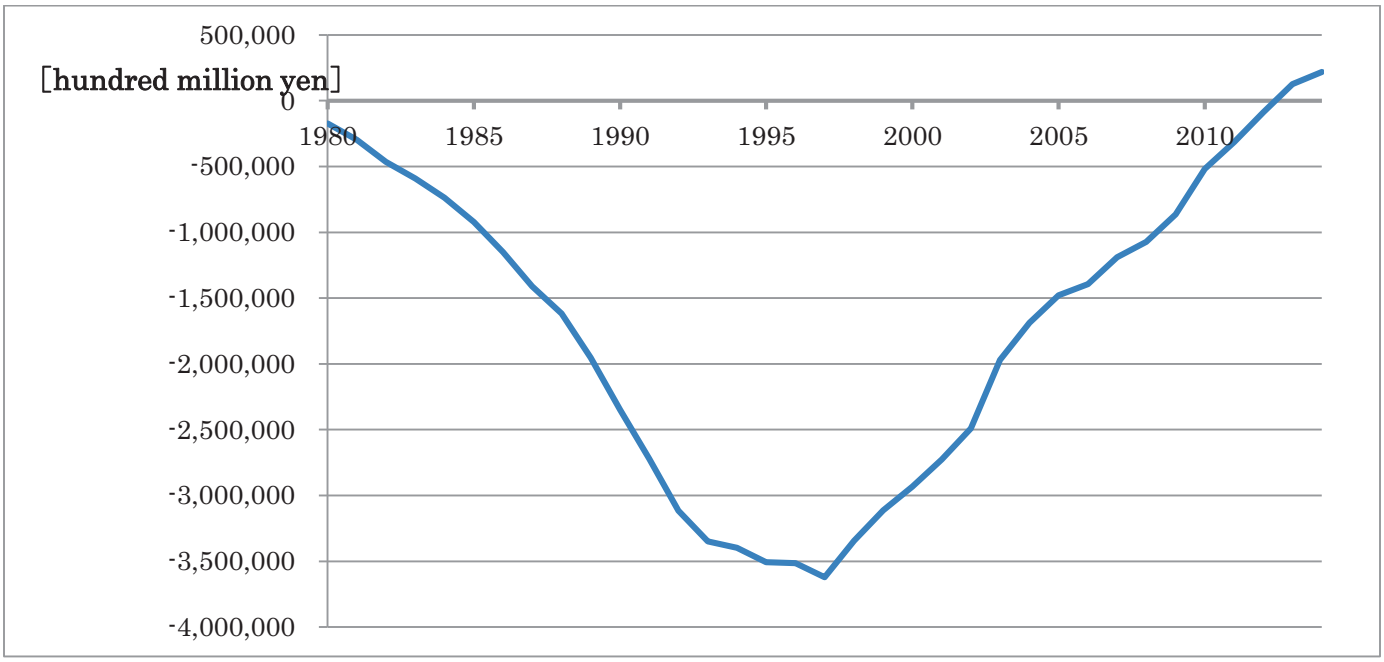


Figure 3. Financial net worth of non-financial enterprises (total) in Japan from 1980 to 2015. The data from the Bank of Japn.

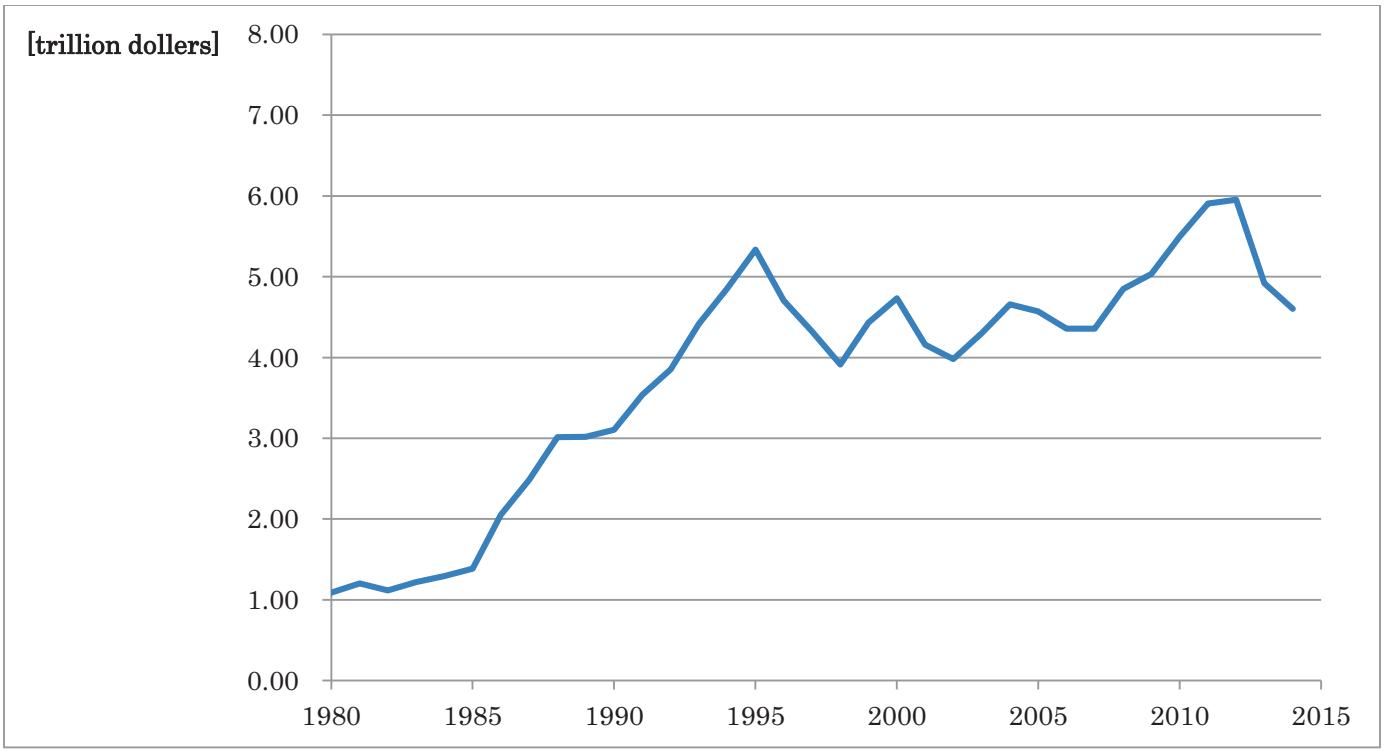


Figure 4. Nominal GDP of Japan since 1980. The data from the World Bank.

SESSION

POSTER PAPERS

Chair(s)

TBA

ISBN: 1-60132-427-8, CSREA Press ©

Data-Driven algorithms for fault detection and diagnosis in industrial process

M. EL KOUJOK, and M. AMAZOUZ
Industrial Systems Optimization Group, CanmetENERGY, Varennes, QC, Canada

Abstract— *Data-driven methods have been recognized as useful tools to extract knowledge from massive amounts of data. However, their use for process operation monitoring and fault diagnosis is still confronted by some challenges. This work aims to facilitate the development and the use of models for fault detection and diagnosis (FDD). To do so, a Matlab-based software has been developed. Latent variables methods such as principal component analysis (PCA) and projection to latent structures (PLS) are integrated into the software, along with appropriate control charts and interpretable plots of variables causing faults. The software enables loading of historical data from various sources, automatic building of models that describe the normal operation, and prediction of quality variables. The software can easily be connected online for continuous FDD of a process. Such tool serves as a decision support for process operators. The usefulness and the accuracy of the tool for FDD is demonstrated with the Tennessee Eastman testbed.*

Keywords: Data driven methods, FDD, chemical process.

1. Introduction

Quantitative data-driven methods have recently received considerable attention from chemical industries, due to the huge amounts of data and the effectiveness of its analysis and interpretation. A team of experts is required to put in place these types of methods and assist the industry in analyzing historical and real-time data. It is however cost intensive for a manufacturer to derive value from data. This work is the first step in developing a Matlab-based FDD software tool that can assist engineers to continuously monitor and rate the performance of a process and perform online FDD. The tool is a dashboard that illustrates fault detection time and the isolation of abnormal events in real time. By using this tool, the operating costs for a plant will be decreased because it will no longer be necessary to hire external consultants.

A wide range of data-driven algorithms can be found in the literature to support the design of an advanced FDD tool. Through the use of machine-learning based model, these algorithms transform data into knowledge. A process can then monitored using the constructed data driven model. Questions such as, how to deploy these methods to help the engineer must be answered. Today, one of the most challenging tasks that chemical engineers face is effective monitoring of entire chemical process complexes, with many interconnected units of operation (i.e. maintaining normal and optimal process operation, as well as ensuring component balances, product quality production rates, and environmental regulations compliance, etc.). This paper presents a developed tool based on appropriate data-driven methods for FDD. A detailed study guided the choice and implementation of two appropriate data-driven methods: (1) Principal Component Analysis (PCA), which focuses on the study of one data block for which all process variables are monitored; and (2) Projection to Latent Structures (PLS), which can serve as a powerful tool for monitoring key performance indicators (KPI). An advantage of these methods (PCA and PLS) is the projection of the original variables onto a latent subspace: latent variables will be monitored in a reduced dimensional space, thus preventing the user from having to select variables. All of the variables for a complex chemical plant can then be monitored. In addition, the use of alternative supervised classification fault diagnosis methods such as support vector machine (SVM) or Neural Network (NN), are not always adequate because it is difficult to identify the mode of abnormal operations ahead of time in real applications [1]. However, once PCA and PLS models are built on good historical data reflecting normal process operation, they can then be used to monitor and diagnose new faults, using indices and reconstruction-based multivariate contribution analysis, respectively [2].

2. Using the FDD Tool

The interface of the developed Matlab-based FDD tool is easy to navigate (Figure 1). Engineers/operators first load historic data from normal process operation. The PCA does not impose any restrictions with respect to the variables that can be employed to detect and isolate a new fault. The user clicks on a single button to build the PCA. For the PLS, a list containing the names of variables can be added to the platform, making it possible to select the input variables and output variable (KPI). The user can also construct the PLS model by a single button click. Several monitoring indices and control limits are integrated into the platform tool. In addition, the tool automatically considers the PLS output (KPI) as one of the monitoring indices and the user can establish its limits. A process is considered out-of-control or faulty if one of these monitoring indices falls outside the established control limit. Following fault detection, the

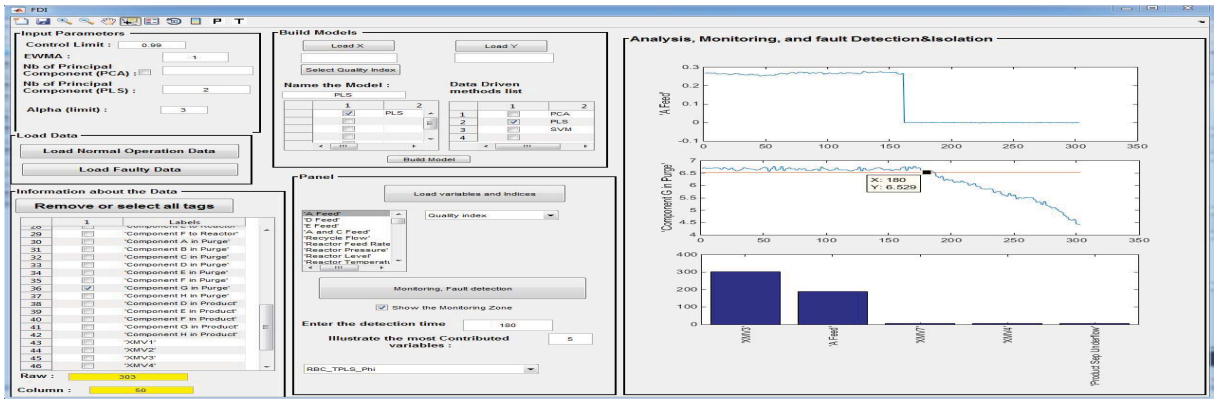


Fig. 1: Dashboard of the developed tool.

tool will display the variables that contribute the most to the detected fault and this is important in helping the engineer/operator find the sources of the detected anomaly. Measures can be taken to recover from the process fault and ensure that the operation remains normal or optimal. The tool that was developed has been applied to a sophisticated chemical process case study – the Tennessee Eastman (TE) benchmark problem [3].

3. Case study and results

The TE is a plant-wide industrial process, proposed as a benchmark for the Eastman Chemical Company. The plant consists of five main units: a two-phase reactor, a condenser, a recycle compressor, a liquid-vapor separator, and a product stripper. Two products are obtained from four reactants (A, B, C and D) in the process. The TE process consists of 11 controlled valves, and 41 measured variables that include temperature, level, pressure flow rate, and concentration. A TE simulator generated industrial data to help evaluate the two studied FDD techniques. The simulator can also generate different types of faults. To show the applicability of the proposed tool, normal operation data were generated to build the PCA and the PLS models and two faults (Table 1) were introduced separately to test each model.

Table 1: Table 1

Fault	Description	Type
Mode 1	Reactor cooling water inlet temperature	Step
Mode 2	Reactant 'A': feed loss	Step

FDD with PCA for fault mode 1: The user loads the normal operation data and the labels of each measured variable. The PCA can then be built to monitor the TE process. Process FDD was started with a click. Once fault mode 1 was seeded, the fault was detected within six minutes by one of the monitoring indices, and a bar chart presented the variables that contribute most to the fault. The result showed that the valve controlling the reactor cooling flow and the reactor temperature are most influenced by the fault. Based on this result, we could conclude that the reactor cooling water system was most likely faulty.

FDD with PLS for fault mode 2: Fault mode 2 had an impact on the production rate (a KPI), which started to decrease from its targeted value (Table 1). To detect this, a user must build a PLS by selecting the production rate as the output (KPI), and the remaining variables as input. The KPI deviation from its target can be easily detected by establishing a limit. The tool then uses the parameters of the PLS model to create a bar chart illustrating the variables that are most affected by this fault. The longest bar represents the position of the control valve for reactant 'A' and the second longest the flow rate of the same reactant. As a result, the user can identify the source of the fault to be the feed loss for reactant 'A'.

4. Conclusion and future work

This work presents a Matlab-based software tool in the early stages of development, which can be used to detect and diagnose an abnormal chemical process operation. The benchmark case study of the TE process was used to highlight the usefulness of the tool. It can help engineers and operators to detect and isolate faults when there is no prior knowledge of faults. The methods integrated in this tool are appropriate for detecting and isolating a single fault at a time. Future work will focus on developing the tool to allow for the detection and isolation of multiple faults.

ACKNOWLEDGMENTS This work was supported by Natural Resources Canadas's PERD program (NRCan).

References

[1] J. MacGregor, A. Cinar, "Monitoring, fault diagnosis, fault-tolerant control and optimization: Data driven methods," *Computers & Chemical Engineering.*, vol. 47, pp. 111–120, 2012.

[2] G. Li, S.J. Qin, J. Yin-Dong and D. Zhou,"Total PLS based contribution plots for fault diagnosis," *Acta Automatica Sinica.*, vol. 35, pp. 759–765, 2009.

[3] J.J. Downs, E.F. Vogel, "A plant-wide industrial process control problem," *Industrial Informatics, IEEE Transactions on.*, vol. 9, pp. 2226–2238, 2013.

Data and Parity block Placement Policy to enhance storage efficiency and utilization

Dayeon Kim[1], and Dongryul Shin[2]
[12]Department of Electrical and Computer Engineering, SungKyunKwan University, Su-won, Korea

Abstract - *HDFS which is known as Hadoop Storage manages fault-tolerance by data block replication. Replicas of data blocks are transmitted to other datanodes in a rack awareness manner. However prior HDFS storage management method wastes storage spaces and makes the situation of unbalanced cluster state. Proposed solution in this paper can save the storage space and maintain the balanced state of the entire cluster.*

Keywords: HDFS, Parity block, Balancer

1 Introduction

HDFS(Hadoop Distributed File System) is a typical storage of the Hadoop. Hadoop Storage is a block structured file system. It splits each file into the block of a fixed size and saves each block in the distributed storage node called "data node." Hadoop picks out the datanode to save the block at random. HDFS maintains the fault-tolerance in a way that makes a number of replica and saves them into the separate datanodes[1]. For example, in case of a file that consists of 6 data blocks and the replica parameter that is set to 3, additional storage spaces as much as 12 block sizes are needed. In other words, maintaining fault-tolerance based on replication makes storage waste. It is particularly inefficient when approachless file is saved in the storage.

To save the storage space, It is possible to replace the block replicas with parity block[2]. It is possible to save storage space using parity block. But additional policy is needed to maintain fault-tolerance characteristic of prior block replica based method and balanced state of the entire cluster. Data block and Parity block placement policy is proposed in this paper to save the HDFS storage space efficiently and maintain balanced state of the cluster.

2 Related Work

2.1 HDFS

Data file is divided into fixed size(default 64MB) of data blocks. HDFS maintains fault-tolerance by copying these data blocks. Data block and block replicas are allocated at the runtime instead of prior rule of block allocation. Hadoop selects the data nodes to save each data blocks randomly. It makes duplications of data block according to data block replica parameter. HDFS block placement uses rack awareness for fault-tolerance by placing one block replica on a different rack. To maintain balanced state of the HDFS cluster, It is essential to reassign the data blocks at the run time.

2.2 Replacement of block replica with parity block.

To maintain the fault-tolerance using parity block, stripe configuration has to be preceded. A stripe is set using the data blocks which is located in same datanode. This configuration has the advantage and disadvantage. The advantage is that low network costs are needed when encoding the data blocks to make parity block. Because parity blocks are encoded using data blocks which are located in same datanode, there is a problem that all parity blocks have to be renewed when updating or deleting the file. Hadoop is designed to process the Write-only-Read-many jobs, so stripe is configured using data blocks in the same data node. Figure 1 shows the stripe configuration of each file A, file B, and file C.

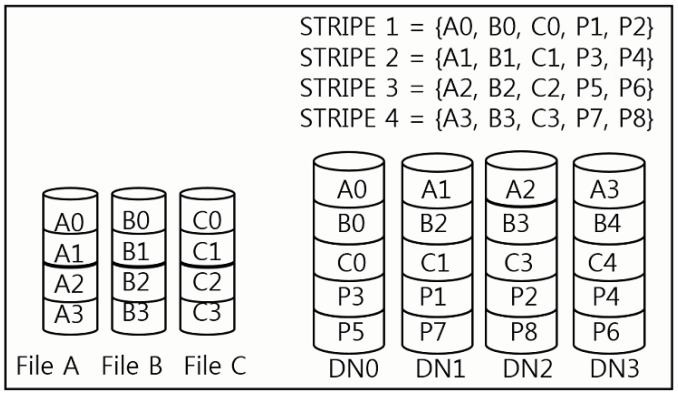


Figure 1 Stripe configuration

Stripe is set of data blocks and parity blocks. Parity block is encoded by data blcoks in same data nodes. It is possible to restore the broken data blocks as many as the number of parity blocks that consists same stripe.

3 Proposed solution

When finishing the allocation of data blocks to the HDFS, parity block is encoded using the data blocks in the same node. After encoding, namenode transmit "Block Move" instruction to datanode to distribute the blocks. To restore the data blocks when node failure occurs, there are 3 block placement policies[3]. Firstly, don't allocate the original data blocks that composes same stripe in the same data node. Secondly, don't allocate the original data block and parity block that composes same stripe in the same data node. Lastly, don't allocate the parity blocks that composes same stripe in the same data node.

When namenode gives "Move Block" command, it evaluates the utilization of each data nodes as proposed in [4] to overcome the uneven block distribution scenario. Proposed solution is represented in a mathematical standpoint as follows :

Assumptions

- U_i and T_i are used space and total capacity of a datanode respectively.
- β_i and β_c are datanode and cluster utilization respectively.
- U_c and T_c are used space and total capacity of cluster.
- Δ is Threshold value.

Datanode and cluster utilization can respectively be expressed as follows. If Data node is not the state of overutilization, it is added to the target nodes. Distinction of overutilization is decided using formula (3)

$$\beta_i = \frac{U_i}{T_i} \quad (1)$$

$$\beta_c = \frac{U_c}{T_c} \quad (2)$$

$$\beta_i = \beta_c + \Delta \quad (3)$$

Figure 2 depicts the flow of proposed solution. When namenode gets the block encode command, namenode collects the block list from each data nodes and construct the stripe. Then it transmits the encode command with stripe information to the datanodes. Datanode encodes each stripe and make a parity block. When Encoding process is done, namenode evaluates the utilization of each datanodes and select the target nodes to move the blocks. And it order "Move block" command according to block place policies.

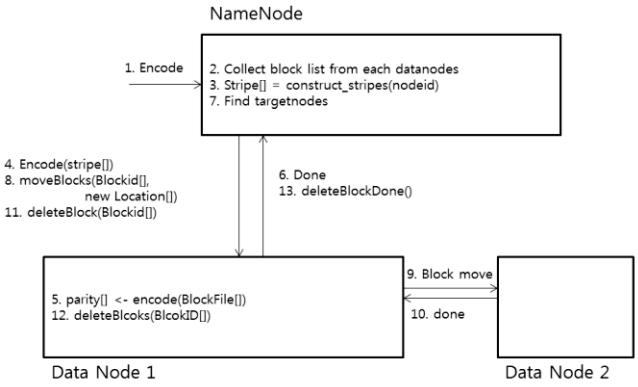


Figure 2 flow of encoding and block reallocation

f

4 Conclusion

When Parity block is used to maintain fault-tolerance, Only (P/G)*100% of Additional space to save the parity block is needed (G = the number of data blocks that composes the stripe, P = the number of created parity blocks). In case of HDFS, it needs 200% of additional space to save the duplications of original data block. So it saves the storage space efficiently. Also, proposed solution helps to keep the cluster in a balanced state when an HDFS client is trying to write data.

References

[1] Shvachko, K. et al. "The Hadoop Distributed File System," IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST), pp.1-10, 2010

[2] Park, Chan-Ik. "Efficient placement of parity and data to tolerate two disk failures in disk array systems." *Parallel and Distributed Systems, IEEE Transactions on* 6.11 (1995): 1177-1184.

[3] 안후영, 이경하, 이수호, 이윤준, 이상민, 김영균. (2013). [분산 데이타베이스] Hadoop 분산 파일 시스템의 효과적인 저장 공간 절약 기법. 정보과학회논문지 : 컴퓨팅의 실제 및 레터, 19(3), 144-148.

[4] Nchimbi Edward Pius, Liu Qin, Fion Yang, Zhu Hong Ming. "Optimizing Hadoop Block Placement Policy & Cluster Blocks Distribution." International Journal of Computer, Electrical, Automation, Control and Information Engineering Vol:6, No:10, 2012, 1282-1288

Dark Side of Social Networking and the Data It Accumulate

(Work in progress)

Muhammad Fahim Uddin
Graduate Student, School of Computer Science and Engineering, University of Bridgeport, CT. USA

Extended Abstract - *Social networking platforms such as Facebook, Twitter, Google+, LinkedIn etc have created tremendous opportunities and potential for the world of social networking and e-friendship. It has facilitated the way we exhibit our personalities, choices, likes and blogs about our causes and views. Such data accumulation and availability has helped researchers, educators, marketers and data scientist to mine and predict valuable insights from such unstructured and semi-structured data. However, beside this brighten side of big data of social networking; there is a dark side as well. This dark side has many sub faces including but not limited to security leaks, hacking, password leaks, personal information leaks, terrorists recruiting, sexual predators, time wastages, legal liabilities, etc. In recent studies and articles, such dark side has been discussed by many business and research professionals and has become a great concern in era of digital age and web where social networking usage and dependence is spreading like a wild fire. Young crowd is more susceptible to become a victim of these dark sides. Therefore, in this paper, I present ongoing research about cognitive looking at dark side of social networking through lens of data analysis, mining and intelligence and I propose a framework and data generation model to suggest a automatic data monitoring system in order to minimize if not eliminate all such faces of data accumulation that is responsible for this dark side of social networking. I conclude with future directions, real world benefits, applications, impacts and related study*

1 Data Sources

I plan to get available data from Facebook, Twitter, LinkedIn and other sources including publicly available news and data that involve crimes and abuses in result of use of such data.

.

2 Methodologies

I aim to collect unstructured data and then analyze to predict how data points can abuse trust and safety of individuals who use and create them but share with open community. I also aim to use statistical tools such as R and data analysis programming language such as Python to show our results. In essence, our methodology seeks to utilize machine learning processing of such data to predict and identify new data with likeliness of contributing to dark side.

3 Related Study

I am studying deep literature and publications of last 2 decades to understand the history and latest state of the art. I aim to put our work in context of existing work that has made progresses in recent years.

ISBN: 1-60132-427-8, CSREA Press ©

SESSION

LATE BREAKING PAPERS

Chair(s)

TBA

ISBN: 1-60132-427-8, CSREA Press ©

ISBN: 1-60132-427-8, CSREA Press ©

IDEAS: An online tool to **I**dentify **D**ifferential **E**xpression of genes for **A**pplications in genome-wide **S**tudies

William Yang
School of Computer Science
Carnegie Mellon University
5000 Forbe Ave., Pittsburgh, PA, 15213 U.S.A.
wyang1@andrew.cmu.edu

Kenji Yoshigoe, Xiaosheng Wang, Dan Li, Yifan Zhang
MidSouth Bioinformatics Center and Joint Bioinformatics Program of University of Arkansas at Little Rock and University of Arkansas for Medical Sciences, 2801 S. Univ. Ave, Little Rock, AR 72204 USA

Patrycja Krakowiak
Arkansas School for Mathematics and Sciences, University of Arkansas
Hot Springs, AR 71901 USA

Wenbing Zhao
Dept. of Electrical Engineering & Computer Science, College of Engineering, Cleveland State University, Cleveland, OH 44115 USA

Hong Zhou
School of Health and Natural Sciences, University of Saint Joseph
West Hartford, CT 06117, USA

Zuojie Luo
Office of Academic Affairs and Dept. of Endocrinology, Guangxi Medical University and the First Affiliated Hospital, Nanning, Guangxi, 530021 China

Xiang Qin
Human Genome Sequencing Center, and Department of Molecular and Human Genetics, Baylor College of Medicine, Houston, TX 77030 USA

Guo-Zheng Li
National Data Center of Traditional Chinese Medicine, China Academy of Chinese Medical Sciences, Beijing, 100700, China

Hamid R. Arabnia
Department of Computer Science
University of Georgia
Athens, GA 30602, USA

Weida Tong
Division of Bioinformatics and Biostatics, National Center for Toxicological Research, United States Food and Drug Administration (FDA), 3900 NCTR Road, Jefferson, Arkansas 72079 USA

Jun S. Liu
Department of Statistics, Faculty of Arts and Sciences and Harvard School of Public Health, Harvard University,
One Oxford Street Cambridge, MA 02138, USA

Mary Qu Yang
Department of Information Science and Joint Bioinformatics Program of University of Arkansas at Little Rock and
University of Arkansas for Medical Sciences, 2801 S. Univ. Ave, Little Rock, AR 72204 USA
mqyang@ualr.edu

ISBN: 1-60132-427-8, CSREA Press ©

I. INTRODUCTION

We developed an online tool called **IDEAS** to **I**dentify **D**ifferential **E**xpression of genes for **A**pplications in genome-wide **S**tudies. We used The Cancer Genome Atlas (TCGA) date to demonstrate the effectiveness and user friendly interface called GeneExpressor of this tool.

In particular, cancer is a disease that is not only complex, in that many genetic variations contribute to malignant transformation, but also wildly heterogeneous, in that genetic mechanisms can vary significantly between patients. Early diagnosis and effective treatment of cancer have been always remained challenging.

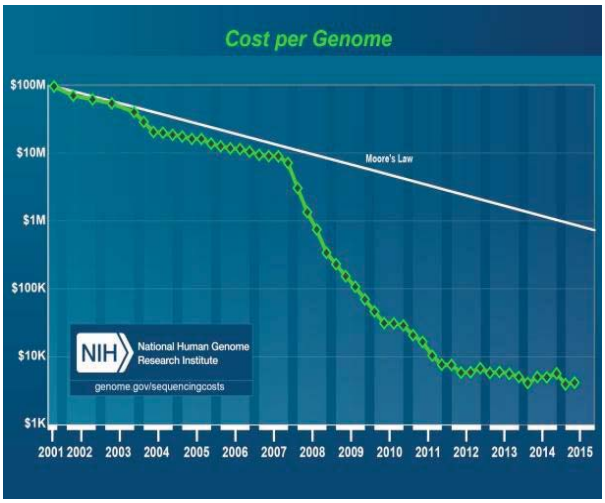


Figure 1 - Reduction in Genome Sequencing Cost. Source: NHGRI

On January of 2015, the Obama administration launched the Precision Medicine Initiative (Olson). With over 200 million dollars of funding, the Precision Medicine Initiative aims to revolutionize our methodology of patient treatment. As of right now, most medical treatments are tailored toward the average patient. Since the genetic makeup of our population is very diverse, the success of such treatments varies vastly between individuals. With the emergence of Precision Medicine, a novel term for personalized medicine, medical treatments begin to be tailored toward individual patients rather than the average patient. As an Associate Director of a Mayo Clinic's research centre puts: "don't tell me how you're going to treat kidney cancer, tell me how you're going to treat my kidney cancer!" Precision Medicine research increases clinical understanding of the complex underlying mechanisms of an individual patient's disease by accounting for the patient's lifestyle, genes, and environment. This allows clinicians to determine effective treatments with minimal adverse effects for the patient. While the research in Precision Medicine is relatively novel, advances made in Precision Medicine has already created many new powerful discoveries, and the potential of the field is just beginning to be tapped.

The invention of the newly developed next-generation sequencing (NGS) methods started a new era of personalized medicine research. NGS, also known as high-throughput sequencing, revolutionized the study of genomics and molecular biology by allowing scientists to sequence entire DNAs and RNAs along with the follow up of determining biological properties and functions at unprecedented speeds and much lower costs. In fact, the price of sequencing using NGS has been decreasing faster than the exponential decay known as the Moore's Law (Figure 1). Because of this new technology, a vast amount of genomic data has generated, allowing genome-wide identification of differentially expressed genes and systems genomics prospection at cellular or organism level to facilitate Precision Medicine research. Hence it has created tremendous demands for the development of novel computational approaches to handle the massive amount of genomic data effectively and timely. Synergistic integration of multi-layer genomic big-data at systems level can shed new light on molecular mechanisms at organism level such as disease initiation and progression, and also lead new pathway-based biomarker determination and drug deliveries. A number of genome-wide association studies (GWAS) have been lunched along with whole genome/whole exome sequencing projects. In particular, TCGA (The Cancer Genome Atlas) was launched to identify the genomic mutations that are associated with cancer and other diseases. Those studies have produced rich data sets and have successfully identified huge number of genomic

alterations, including Single Nucleotide Polymorphisms (SNP), Copy Number Variations (CNV) and Structure Variations (SV). However, most genomic mutations identified in those studies either have only a small disease risk effect, or are only present in a small fraction of the population in complex diseases such as cancer due the complexity and inter-patient heterogeneity. The Systems Genomics Laboratory of the University of Arkansas at Little Rock aims to leverage the research by combining different genomic information including eQTL mapping, differential expression of genes and protein-protein and protein–nucleotide interactions, to construct high-level gene networks for integrative genome-phoneme studies at a higher systems level. To this end, synergistic analyses of multi-layer genomic data will further progress biomedical research and will ultimately lead to the improvement of human health and the prolongation of human life.

The completion of the human genome sequence has provided a blueprint for Precision Medicine research; however, the connection of the genome to systematic gene functions relies on the development of effective computational tools and transcriptome studies. Furthermore, the advent of single cell cancer genomic sequencing research indicated that the same type of cancer can have different subtypes with different genetic mechanisms and clinical outcomes. Therefore, biomarkers derived from a single genetic type are usually not well reproducible and often vary significantly within various patient populations. The identification of the causes of aberrant gene regulatory networks is crucial for early stage diagnosis of cancer; however, this task is difficult because of the lack of effective computational methods to integrate different levels of genetic data.

We consider that by studying the transcriptome, RNA expressed from the genome along with gene networking studies, we can better understand the molecular mechanisms of diseases. Furthermore, systematic studies of differentially expressed genes can provide us a comprehensive picture of how genes actively express and interact with others. Large-scale next-generation RNA-seq data have generated unprecedented opportunities for cancer studies in the context of knowing the entire catalogue of gene expressions. Synergistic integration of next-generation sequencing data at a systems biology level can provide essential information not only about disease mechanisms and prognosis, but also pathway-based biomarker identification and drug efficacy.

Utilizing large whole genome scale re-sequencing projects such as TCGA, this project was designed in an effort to redeem the benefits of the genomic big data.

TCGA is a project that began in 2006. It included an initial investment of 50 million dollars from the National Cancer Institute (NCI) and National Human Genome Research Institute (NHGRI), both part of the National Institutes of Health (NIH) for an initially 3-year pilot period. The TCGA pilot was deemed a success and showed the importance of making genomic data publicly available for researchers around the world to make new advancements and discoveries in the field of cancer biology and the study of medicine. The success sequentially gained more attentions of NIH and as a result, NIH committed major resources to TCGA to collect more genomic data for many more different cancer types. Today, TCGA contains information for over 200 forms of cancer with many different kinds of genomic data such as DNA methylation data, copy number variations, and perhaps most useful of all at the current computing ability, the next generation RNA sequencing (RNA-seq) data.

Figure 2 - Bootstrap (left) vs. HTML (right)

By analysing the next generation RNA-seq data, differentially expressed genes can be identified; this data can be used synergistically with other data to detect the causes of aberrant regulatory gene networks critical to the initiation and maintenance of cancerous tumours.

Although the DNA sequences play very important roles in biological processes as they contain the templates for all genes, the RNA sequences play equally important roles as they are the expressed (used) versions of these genes. For example, since chromosomes in most cells of the human body are the same, the DNA sequencing results are nearly identical for all cell types. However, different specialized cells in the human body contain different RNAs and therefore have many different proteins and characteristics, despite having the same DNA sequences. This is because the specialized cells transcribe different parts of their DNA sequences.

Sometimes, mutations occur and change the DNA sequence of a cell by much less than 1%. However, this tiny change in the DNA can greatly change the resulting proteins of the cell. As a result, cellular functions can be compromised or completely changed, and cancer might arise through malignant transformations. Therefore, it is important to analyse the levels of different RNA sequences and subsequently locate the differentially expressed genes to identify potential biomarkers for cancer. Effectively integrating differentially expressed genes with pathway analysis approaches will enable us to comprehensively catalogue the genomic alterations and epigenetic modifications that are associated with cancer in both personalized medicine and generalized cancer studies (Jaakkota et al).

Identifying cancer-causing genetic alterations and their functional pathways remain highly challenging due to the complex biological interactions and the heterogeneity of diseases even with the power of single-cell genomics. Genetic mutations in disease causing genes can disturb signalling pathways that impact the expressions of sets of genes, each performing certain biological functions. We hypothesized that driver mutations are likely to affect disease-associated functional gene expressions, and the causal relationship between the mutations and the perturbed signals of transcription can be reconstructed from the profiles of differential gene expression pattern and disturbed gene networks. Therefore the first step to improving personalized treatment of tumours is to systematically identify the differentially expressed genes in cancer.

In order to identify the differentially expressed genes using RNA-seq data, we are required to process large amount of genomic data effectively and provided a useful tool. However, this requires a deep understanding of programming and software development. As results, our task is transformed into overcoming a major obstacle for scientists who are not familiar with computer science.

This isn't, however, the first time that researchers encountered such obstacles. There have been numerous projects implemented to overcome the need for programming in the field of biological sciences. In fact, the largest National Science Foundation (NSF) funded biological project is the iPlant collaborative, with over 100 million dollars. The iPlant collaborative is a cyberinfrastructure for plant science. The goal of iPlant collaborative is to provide computing resources and solve computational challenges in labs around the world. Our project has a similar goal, but rather than plant science, it is on human cancer.

In this project, we developed a streamlined web tool that analyses RNA sequence data from NGS to reveal the differentially expressed genes across various types of cancer. This tool helps scientists without computer science background and facilitates the advancement of Precision Medicine research.

II. Materials and Methods

The framework for the online tool was designed with emphasis on low computational resource requirements while maintaining the users' accessibility. This emphasis was not only placed because of the lack of computational resources for many wet laboratories, but also to produce a streamlined framework that could be easily maintained at a low cost.

Figure 3: User interface of IDEA: GeneExpressor Webpage

The first step in designing the online tool was to create a user interface (Figure 3). From the interface, the user is able to upload files, links, and jobs.

The user interface is part of the framework that directly interacts with the user, and because of this, the user interface has to be user-friendly, appealing, and accessible at any time regardless of the computational workload.

To make the interface user-friendly and appealing, the front end (what the user sees) of the interface or website was designed by Bootstrap, which utilizes JQuery, HTML, and CSS. Bootstrap is a free open-source front end framework designed to create a more responsive website.

In addition, the task of detecting differential expressed genes does not require much user input. It only requires that the user inputs the gene expression data for calculation and email for data retrieval. Because of this, the website contains minimal elements to keep the user interface simple, easy, convenient, and elegant to the users as complexity only increases the confusion and the learning curve.

The web page contains a relevant background image (picture of DNA retrieved from Pixelbay) to increase the page's elegance.

For the body, the page consists of a Bootstrap panel at its centre that contains a Bootstrap form, one or two Bootstrap inputs, and a Bootstrap button. The form, inputs, and button could all have been designed using regular HTML elements, but the elements in bootstrap are much more appealing than the regular HTML elements (Figure 2).

However, a regular bootstrap panel alone is not sufficient because there are two methods the user can input the gene expression data: they can either give the download link of expression data from TCGA or upload their own gene expression matrix. Several solutions were explored to fit this need (Table 1).

Of all the solutions, adding tabs to the Bootstrap panel appeared to be the best solution. In order to achieve this, Bootstrap panels with nav tabs were retrieved from bootsnipp.com (an

element, playground, and code snippets galley for Bootstrap). Using these custom panels, tabs were added to the panel, which would allow the user an easy method of selection.

To polish up the interface, a mechanism for assistance had to be created. Unlike the other aspects of the interface, the assistance aspect should only be displayed when the user requests assistance.

The Bootstrap popover element is perfect for this task as it display a pop-up box when the user clicks the element. To increase the compactness while maintaining its visibility, Bootstrap Glyhpicons (icons) of a question mark was used as a trigger for the popover element. With all these components added to the front-end of the user interface, the result was a user interface that is both user friendly and visually appealing (Figure 3).

The back end (information processing underneath the front end) of the interface was developed using the programming language PHP since PHP is a programming language designed for web development. Compared to other options like Ruby on Rails, Django, and any other back end framework, PHP is not very computationally resource intensive though it lacks scalability.

However, the task performed by this web tool does not require large scale implementation. Hence PHP was the most efficient tool for developing the back end and it fulfilled the accessibility requirement of the interface.

To maintain the accessibility, the user requests were stored in a database and were executed one at a time instead of execution on submission. If execution occurred at submission, when multiple submissions occur, the multitude of the tasks executing would overload the CPU, and leave little or no resources for the user interface and therefore prevent future job submissions.

While storing the requests in a database might slightly delay the task, it certainty ensures accessibility of the tool when multiple submissions occur simultaneously.

The database MySQL was used because it is lightweight like PHP and it pairs with PHP very well. The default PHP file upload function was used for the file upload. One drawback of this function was its limitation to upload big files, such as file sizes larger than 500 MB. Though there is no issue for uploading an expression profile of a cancer genome, which is typically smaller than 100 MB, it prevents an upload of next generation RNA sequencing data (GB) from TCGA.

Solution	Simplicity	Elegance
Adding a navigation bar on the top of the page	The main focus of the user is at the panel. Adding a navigation bar on the top of the page will distract the user from the panel. This may be an inconvenience the user.	The navigation bar is at the top of the web page, away from the centre. Users usually do not notice this, and therefore it does not hurt the elegance of the page.
Adding a tab on the top of the panel	The panel already contains a tab that describes the panel. Adding another one beside it is an effective way of telling the users their options. Therefore, this solution keeps the design simple and convenient.	The panel itself already has a tab. Adding other tabs will just occupy unused space. Therefore, this slightly increases the elegance by removing some of the empty space.
Using a radio button to select the method.	A radio button is common and very simple to use. However, this solution requires adding more inputs, which may confuse the user.	Adding more inputs will increase the size of the panel. Since some of the inputs will remain unused, this greatly decreases the elegance of the page.

Table 1 – Solution Matrix Regarding the Different Modes of Input

The usage of data from TCGA is crucial, because it facilitates individual laboratory research. However, allowing the user to upload data from TCGA project requires massive storage infrastructure, which leads to increasing costs of maintenance. A solution to this problem was to allow user to upload the download link of TCGA data directly.

In this way, the data is retrieved at job initiation and deleted at job completion. This solution removes the necessity of high storage as well as increases the convenience of the users since they are no longer required to download and upload the data from TCGA.

The last component of the backend is the email. This can be achieved using the PHP mail function or the Linux kernel send mail function. However, many modern email clients have advanced spam filtering systems. Therefore, emails sent from a personal SMTP (Simple Mail Transfer Protocol) server are usually filtered out. To bypass this filter, the Gmail SMTP server was used.

Using a Gmail account as the email sender and a python script to send the request, the email mechanism was successful in sending job notifications. With this augmentation, the user interface was complete and successful in achieving its goal: allowing user to submit jobs to the server.

A pipeline was designed to execute tasks proposed in this project. The goal of the pipeline was to execute the tasks as well as to process and format the user input.

For processing and formatting, the programming language Python was used. Python was chosen because of its reliable Linux kernel interaction. The role of the Python scripts was to fetch the RNA sequence data from TCGA, create the expression profile from the data, and generate the specific program for the expression profile. For the task execution, the programming language R was used. The package EdgeR from Bioconductor was used to identify the differentially expressed genes in the expression profile. EdgeR uses a negative binomial distribution to model the discrete RNA sequence data and detect statistically significant differences in gene expressions between the samples. The last part of the pipeline was another Python script that packages and sends the results to the user.

Lastly, an infrastructure was needed to host the tool on the World Wide Web. There are three types of resources that can achieve this: a web host, a virtual private server, and a dedicated server.

A web host is cheap but it lacks functionality. Therefore, implementing a framework on the web host is nearly impossible. A virtual private server is slightly more expensive but has few functional limitations. It only has limitation on the computational resources.

A dedicated server is very expensive but has vast amount of computational resources and no limitation on functionality. Because the detection of differentially expressed genes from Next-Generation Sequencing data is not very computationally intensive, we decided that a virtual private server would be best suited for this project.

Heat map is generated to visualize gene expression. Volcano plot shows the quantity of differentially expressed genes. The result from the automated pipeline was sent to user as an email attachment. The backup of the result is a downloadable URL link in case the user does not receive the email attachment. Overall, the pipeline was designed to utilize many artificial intelligence techniques to optimize the performance and is fully automated (Figure 4).

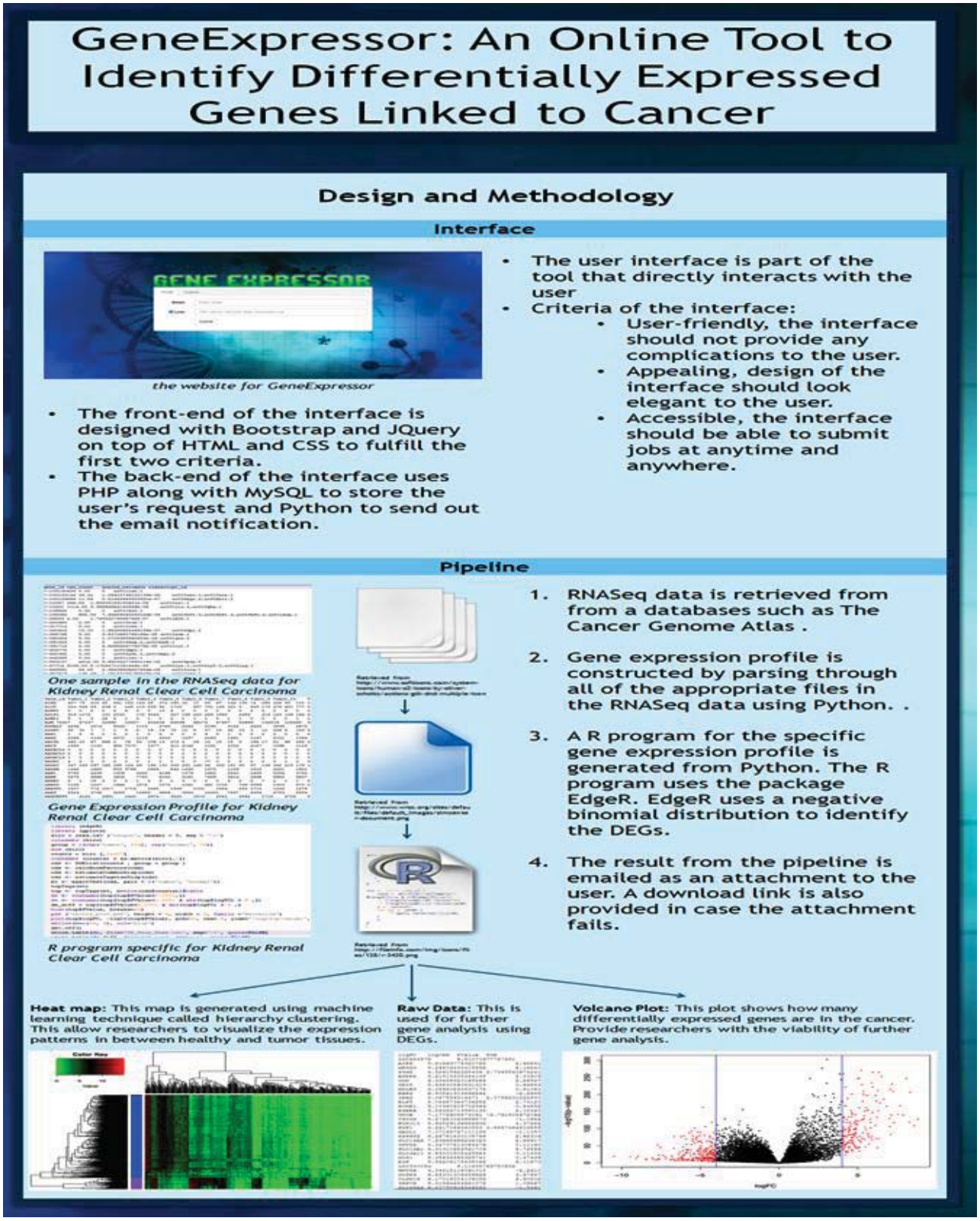
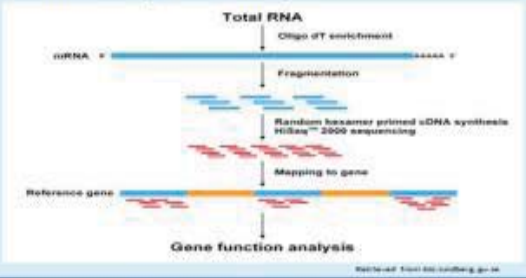


Figure 4: Flowchart and architecture of ***GeneExpressor***

Introduction

- Cancer is a widely disturbed lethal disease. The disease is hard to diagnosis and treat earlier due to its complex and heterogeneous natures and lack of accurate early biomarker.
- Single-cell cancer genomic sequencing research indicates that even pathologically same cancer can have several subtypes with different genetic mechanisms.
- The identification of aberrant gene regulatory networks is crucial for early cancer diagnosis and can be developed from the information of differentially expressed genes using RNA sequence (RNASeq) data.
- Next-generation sequencing (NGS) allows research and medical center to produce RNASeq data at an unprecedented rate.
- The emergence of Precision Medicine created enormous demand for genome-wide study and analysis.

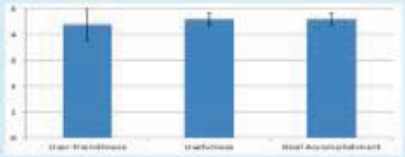


Problem Statement

- RNA Sequencing only maps RNA short reads to genes.
- RNA sequencing does not do any gene analysis.
- Gene analysis requires comparison of gene expression using a statistical model, typically a Poisson distribution or a binomial distribution.
- Genome-wide identification of differentially expressed genes (DEG) requires large amounts of computations, making computer programming a necessity for researchers.
- Computer programming is a major obstacle for many biological researchers.
- The purpose of this project is to remove this obstacle by developing an online tool called GeneExpressor for systematic genome-wide identification of DEG.
- The project has the follow components:
  - **Acquisition** of RNASeq data
  - **Analysis** of RNASeq data
  - **Integration** of RNASeq data
  - **Utilization** of RNASeq data
- This will further elucidate the regulatory mechanisms, identify effective biomarkers, disrupted gene networks, and drug targets of cancer.

Discussion

- The project has the follow constraints:
  - **CPU cycles:** overloading the CPU will starve the resource necessary for the interface, removing the accessibility element of the interface.
  - **Memory:** The infrastructure supporting the tool only has 6 GB of RAM. A single task uses about 2.5 GB of RAM. More than 2 tasks executing will starve system resources, removing accessibility element of the interface.
  - **Storage:** RNASeq data usually are around 10 GB. The infrastructure supporting the tool only has 30 GB of disk. 20 GB is needed for extraction. There isn't enough storage for multiple extractions.
- All of the constraints for the tool are physical limitations.
- The constraints are satisfied by executing tasks sequentially, rather than in parallel.
- The tool is tested extensively using RNASeq data for many cancers and custom gene expression profiles.
- An evaluation is also given out to Arkansas School for Mathematics, Sciences, and the Arts' Molecular Biology class and Web Application class.
- 5 students evaluated the online tool and the results is shown below:



- The most important part of the evaluation is their qualitative feedback.
- The evaluators had an error-free experience.
- 100% of the participants said that the most difficult part of the process is retrieving the expression data from The Cancer Genome Atlas.

Conclusion

- GeneExpressor uses machine learning technique to enhance accurate whole genome-wide identification of GEG and user-friendly interface. Testing shows it is easy to operate for users without computational skills.
- GeneExpressor also generates DEG profile, clusters and visualization of patterns between DEG and non-significant genes. The results can facilitate the advancement of both cancer research and Precision Medicine.
- GeneExpressor provides a foundation to facilitate identification of aberrant gene networks &drug targets
- GeneExpersssor runs faster for small to medium size samples, however it experiences heavier duty in performance when sample size becomes larger.
- Further development of GeneExpressor includes enhancing its speed and performance for large-size samples, as well as the identification of disturbed gene networks and pathways utilizing DEG.

Figure 5: Problem statement and summary

III. Data Processing, Analysis and Discussion

The problem statement and flowchart of the automated tool are illustrated in Figures 4 and 5. We have used artificial intelligence techniques in designing a pipeline and processing the RNA-seq data. The success of the project was determined by the application's ability to provide user the differential expressed genes from expression profiles without any complications.

To measure such criteria, students and professors from the Molecular Biology and Web Application classes in the Spring Semester of 2016 at ASMSA of University of Arkansas tested the website for its functionality and simplicity. This was done so that the Molecular Biology class would provide effective feedback on the functions while the Web Application class would provide effective feedback on the design.

Five of the students from the classes tested and evaluated the application and the results thoroughly as shown in Figure 6.

The most important part of the evaluation was the students' qualitative feedback. All participated evaluators experienced an error-free experience. Interestingly, 100% of the participants said that the most difficult part of the process was retrieving the expression data from TCGA. This showed that the simplicity of this tool has been maximized and that it is only the bottle-necked by TCGA interface. The evaluation showed that the application is successful in providing the user with differential expressed genes analysis.

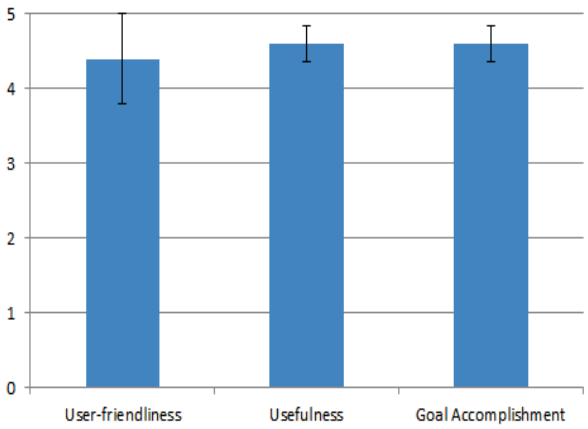


Figure 6 – Evaluation of GeneExpressor

IV. Conclusion

With the invention of next generation sequencing and the emergence of Precision Medicine, the demand to identify genome-wide differentially expressed genes has been increasing. This online tool is successful in providing users with the ability to identify genome-wide differentially expressed genes without using any computational skills.

We have used this tool in our further gene signalling network studies and made a number of discoveries. By developing synergistic computational, biological, and statistical techniques, this project may not only lead to ground-breaking discoveries and new insights into the molecular mechanisms underlying cancers, but also accelerate the advancements of both cancer studies and Precision Medicine research.

It should be noted that the tool developed in this project can handle other RNA-seq data beyond TCGA. While project was tested with a small evaluation size, further development of this tool includes larger evaluations and the identification of disturbed gene networks and pathways using the genome-wide differentially expressed genes identified by this tool. Further utilization of this tool to facilitate synergistic knowledge discovery from multi-layer genomic big data will be reported in the follow up articles.

IV. Acknowledgements

Mary Yang was supported by NIH 1R15GM114739, FDA BAA-15-00121 - HHSF223201510172C and ASTA 15-B-38. The Systems Genomics Laboratory of University of Arkansas at Little Rock provided computing resources and supports of graduate and undergraduate students. The computational resources of William Yang and Kenji Yoshigoe were also supported by NSF MRI Award #1429160. Elizabeth Pierce, Chair of Information Science Department of University of Arkansas at Little Rock is acknowledged for providing generous academic supports to faculty and students.

V. References

Yang, W., Yoshigoe, K. et al. (2014). Identification of genes and pathways involved in kidney renal clear cell carcinoma. BMC Bioinformatics, Vol. 15 (Suppl 17), S2. http://doi.org/10.1186/1471-2105-15-S17-S2

Wetterstrand, KA. DNA Sequencing Costs: Data from the NHGRI Genome Sequencing Program (GSP) Available at: www.genome.gov/sequencingcosts. Accessed August 25 2015

Parpia, S., Thabane, L., Julian, J. A., Whelan, T. J., & Levine, M. N. (2013). Empirical comparison of methods for analyzing multiple time-to-event outcomes in a non-inferiority trial: a breast cancer study. BMC Medical Research Methodology, 13, 44. http://doi.org/10.1186/1471-2288-13-44Ewelina

Pośpiech, E., Ligęza, J. et al. (2015). Variants of *SCARB1* and *VDR* Involved in Complex Genetic Interactions May Be Implicated in the Genetic Susceptibility to Clear Cell Renal Cell Carcinoma. *BioMed Research International*, *2015*, 860405. http://doi.org/10.1155/2015/860405

FACT SHEET: President Obama's Precision Medicine Initiative. (n.d.). Retrieved September 21, 2015, from https://www.whitehouse.gov/the-press-office/2015/01/30/fact-sheet-president-obama-s-precision-medicine-initiative

Olson, S. (2015). The Next Step In Human Genomics: Precision Medicine. Retrieved August 21, 2015, from http://www.medicaldaily.com/pres-obamas-precision-medicine-initiative-human-genome-project-and-your-352678

EdgeR: a Bioconductor package for differential expression analysis of digital gene expression data *Bioinformatics (2010)* 26 *(1):* 139-*140 first published online November 11, 2009 doi:10.1093/bioinformatics/btp616*

Dimensionality Reduction via the Johnson-Lindenstrauss Lemma

J. Fedoruk[1], B. Schmuland[1], J. Johnson[3], and G. Heo[1,2]
[1]Department of Mathematical and Statistical Sciences, University of Alberta, Edmonton, Canada.
[2]Department of Dentistry, Faculty of Medicince and Dentistry, University of Alberta, Edmonton, Canada.
[3]Department of Mathematics and Computer Sciences, Laurentian University, Sudbury, Canada.

Abstract— *The Johnson-Lindenstrauss lemma is a famous result that has lead to the development of tools that may be used when dealing with datasets of immense dimensionality. The lemma asserts that a set of high dimensional points can be projected into lower dimensions, while approximately preserving the pairwise distance structure. Significant improvements of the JL-lemma are summarized, followed by a detailed treatment of the more recent approach taken by Matoušek [13]. Particular focus is placed on reproving Matoušek's versions of the lemma first using subgassian projection coefficients and then using sparse projection matrices. The results of the lemma are then tested using simulated data. The simulation suggests a projection that is more effective in terms of dimensionality reduction than that which is born out by the theory.*

Keywords: Johnson-Lindenstrauss, dimensionality reduction

1. Introduction

Statistics is a branch of mathematics focused largely on *data*. To the applied statistician, a dataset is an $n \times d$ matrix X, consisting of n observations, where each observation is characterized by d covariates. From a geometric standpoint, one can view X as a collection of n points, $x \in \mathbb{R}^d$. The *dimensionality* of x refers to the number of dimensions to which x belongs; in this case, x is said to be d-dimensional, and we can express x as $x = (x_1, x_2, \cdots, x_d)$, where $x_i \in \mathbb{R}$ is said to be the i^{th} *coordinate* of x, for $i = 1, 2, \cdots, d$. From a statistical standpoint, one can view x as an observation consisting of d measurements, with each coordinate x_i of x corresponding to measurement for the i^{th} variable.

The main objective of statistics is to collect sample data in order to develop models that may be used to make claims about a population of interest. However, methods of data collection and model development have evolved over the years. David Donoho [5] argues that traditional statistical analyses relied upon the collection of a large number of observations, each characterized by a few carefully chosen variables. Accordingly, the observations themselves correspond to points in relatively low-dimensional space. However, as Donoho goes on to claim, modern data are often represented by a number of dimensions that is too large for classical statistical approaches to be feasible. Indeed, thanks to advancement in computer power, our capacity to sense and record information has grown immensely; so much so that the dimensionality of modern datasets can be in the thousands or even in the millions. This has created a new challenge for statisticians: how does one begin to fit a model to a dataset consisting of significantly more variables than observations (when d is much larger than n)?

The difficulty in analyzing high-dimensional data is known as *The Curse of Dimensionality*. Issues revolving around the Curse of Dimensionality have become commonplace in data analysis, and this has lead us to an exciting area of research known as dimensionality reduction.

1.1 Dimensionality Reduction

The first step in the analysis of a high dimensional data set is to reduce its dimensionality. That is, given some dataset $X_{n\times d}$, where $d >> n$, we wish to find a lower dimensional representation $Y_{n\times k}$ of X, with $k < d$, so that much of the information contained in X can be obtained from Y. Techniques in dimensionality reduction are being used in a variety of fields, including research in dentistry and orthodontics. For example Heo et al. explore the use of dimensionality reduction techniques to landmark-based data [7], [8], [9]. In particular, they apply dimensionality reduction techniques to orthodontic data sets in order to compare two types of rapid maxillary expansion treatments. Their initial dataset consisted of high-dimensional landmark configuration data that were obtained from cone beam CT scans. Techniques in dimensionality reduction were applied to these data in order to allow for computation of between-subject variation. The next question to address is this: what are the different methods of dimensionality reduction, and when should one method be used instead of another?

There are a number of statistical approaches that may be used to reduce the dimensionality of a dataset, and such approaches can be classified as either *feature selection* or *feature extraction* techniques. Some of the well-known methods of feature selection include model selection methods in regression and classification, as well as regularization methods such as Lasso and support vector machines. Some of the well-known methods of feature extraction include clustering, principal component analysis, multidimensional scaling, and ISO maps. Most of the statistical approaches to dimensionality reduction are based on uncovering the *intrinsic dimensionality* of a data set, which is the number of dimensions (variables) that contribute to the majority of

the observed structure in the data; on the other hand, the *extrinsic dimensionality* of a data set gives the number of dimensions in which the data are observed [14].

Although it is of interest for us to uncover the intrinsic dimensionality of a dataset, it is not always possible to do so. In particular, many of the above approaches rely matrix operations that are computationally expensive for high dimensional data. For example, regression requires matrix inversion, while MDS and PCA rely on eigendecomposition and such matrix operations require a great deal of memory when acting on high-dimensional matrices. As such, there is a growing need for methods of dimensionality reduction that enable us to significantly decrease the extrinsic dimensionality of the data while preserving its structure. Accordingly, new methods in dimensionality reduction are emerging, and such methods effectively reduce the extrinsic dimensionality of the data, without any consideration of the true intrinsic dimensionality. As a result, these new methods do not provide a clear picture of the intrinsic dimensionality of a dataset, nor do they provide us with the variables responsible for much of the structure in the data. Nevertheless, the new approaches to dimensionality reduction are becoming an integral part of various algorithms designed to deal with high-dimensional data. The following gives a brief summary of the Lemma that started this movement, and some of the key improvements it has seen since its inception.

1.2 Johnson-Lindstrauss Lemma

The Johnson-Lindenstrauss Lemma is a famous result that has lead to the creation of a new class of techniques in dimensionality reduction. The approach is much more general than some of the classical, statistical methods in that it may be applied to *any* set of points in high dimensions (unlike statistical methods of dimensionality reduction, in which it is assumed that the intrinsic dimensionality is very small relative to the extrinsic dimensionality).

The Johnson Lindenstrauss Lemma asserts that a set of high dimensional points can be projected into lower dimensions, while approximately preserving the pairwise distance structure between points. More formally, the JL Lemma states the following:

Given a set P of n points in $\mathbb{R}^d$, for some $n, d \in \mathbb{N}$, there exists $k_0 = O(\epsilon^{-2} \log n)$ such that, if $k \geq \lceil k_0 \rceil$, there exists a linear mapping $T : \mathbb{R}^d \to \mathbb{R}^k$ such that for any two points $u, v \in P$,

$$(1-\epsilon)\|u-v\| \leq \|T(u)-T(v)\| \leq (1+\epsilon)\|u-v\|.$$

Since T is a linear mapping, we can, without loss of generality, replace the quantities $u-v$ and $T(u)-T(v)$ with x and $T(x)$, for a unit vector $x \in \mathbb{R}^d$. That is, x represents the distance between two points in P, and $T(x)$ represents the distance between the two mapped points. The mapping T, is referred to as a *JL-embedding*.

The result of this theorem ensures that any set of points can be projected into $O(\epsilon^{-2} \log n)$ dimensions while maintaining ϵ-distortion of pairwise distances between points. Here, ϵ-distortion implies the ratio of distance after projection over that before projection is within $(1-\epsilon, 1+\epsilon)$.

2. Evolution of the JL Lemma

Over the years, the JL-Lemma has been reproved many times, with new proofs providing a sharpening and/or simplification of the result. However, there is one particular feature that is common to all JL-embeddings: the mapping T projects a vector into lower dimension, and the length of this projection is sharply concentrated around its expectation. Moreover, the existence of such mappings are typically established through the probabilistic method, i.e. one shows that the random mapping T has nonzero probability of being sufficiently concentrated about its expectation.

In the original paper that introduced the JL-lemma, Johnson and Lindenstrauss [12] assert the existence of a mapping T that gives an orthogonal projection of n points from $\mathbb{R}^d$ onto a random k-dimensional subspace with dimensionality $O(\log(n/\epsilon^2))$, such that pairwise distances are maintained to within a factor of $1 \pm \epsilon$. Johnson and Lindenstrauss provide a lengthy, technical proof using geometric approximation, and reading through every detail of their proof is a challenging endeavor, even for an experienced mathematician.

The first significant improvement to the JL-lemma came from Frankl and Meahara [6], who replace the random k-dimensional subspace with a collection of k random, orthonormal vectors; this approach requires a much simpler proof that attains a sharper bound on the reduced dimensionality of $T(x)$. In particular, Frankl and Meahara show that n points from $\mathbb{R}^d$ can be projected into $k \geq \lceil 9(\epsilon^2 - \epsilon^3/3)^{-1} \log(n) \rceil$ dimensions while maintaining ϵ-distortion of pairwise distances. Moreover, Frankl and Meahara establish that the mapping is of the form $T = \sqrt{\frac{d}{k}} XR$, where X=$X_{n \times d}$ is the data structure corresponding to the points in P, and R=$R_{d \times k}$ is the projection matrix consisting of random orthonormal column vectors.

Indyk and Motwani [11] then provide the next improvement by relaxing the condition of orthogonality in the projection matrix. Instead, they show that a projection matrix need only consist of independent, Gaussian random vectors, with each coordinate following $\mathcal{N}(0, 1/d)$. This result greatly simplifies the proof of the JL-lemma since independent vectors are easier to deal with than orthogonal vectors and in high dimensions, independent Gaussian vectors are almost orthogonal.

Dasgupta and Gupta [4] then provide an alternative, much simpler proof of the result of Indyk and Motwani using moment generating functions. Moreover, they provide a tighter bound than all previous versions of the JL-lemma, wherein n points from $\mathbb{R}^d$ can be projected into $k \geq$

$\lceil 4(\epsilon^2/2 - \epsilon^3/3)^{-1}\log(n)\rceil$ dimensions while maintaining ϵ-distortion. The results of both Indyk and Motwani, and Dasgupta and Gupta rely on projection coefficients that are spherically symmetric.

Achlioptas [1] then shows that spherical symmetry of the projection coefficients is not necessary in order to obtain a JL-embedding that maintains ϵ-distortion. Instead, he shows that concentration of the projected points is sufficient. In particular, he chooses projection coefficients that are independent, identically distributed (i.i.d.) random variables, uniformly distributed over $\{-1, 1\}$ or, alternatively, distributed over $1/\sqrt{3}\{-1, 0, 1\}$, where ± 1 occur with probability 1/6 and 0 occurs with probability 2/3; he then shows that the even moments of such random projections are dominated by those of the spherically symmetric case, so that a JL-embedding can be found with probability at least as large as that in the spherical case (that is, when spherically symmetric projection coefficients are used).

Finally, Matoušek [13] improves upon the above results in two ways. First, he proves a generalized version of the JL-lemma using the language of subgaussian tails, and this approach contains many of the previously mentioned approaches, which involve spherical symmetry of the projection coefficients. In particular, Matoušek shows that a JL-embedding can be found by using i.i.d. projection coefficients that follow a distribution with a mean of 0, variance of 1, and with tails that are tighter than those of the standard normal distribution. Matoušek's next contribution is an extension of Achlioptas' result mentioned above. More specifically, Matoušek proves that highly sparse projection matrices can be used, but the sparsity of the projection matrix depends on the density of the input vectors: denser input vectors allow for sparser projection matrices which is desirable since sparse projection matrices lead to faster embeddings.

3. Two Approaches to the JL-Lemma: Subgaussian Projection Coefficients and Sparse Projection Matrices

The following three theorems are based largely on Matoušek's rendition of the JL Lemma [13].

Theorem 1: Consider a set P of n points in $\mathbb{R}^d$, for some $n, d \in \mathbb{N}$. Given $\epsilon \in (0, 1/2)$, let $k = O(\epsilon^{-2}\log n)$. Then there is a mapping $T : \mathbb{R}^d \to \mathbb{R}^k$ such that

$$\mathbb{P}\big((1-\epsilon)\|u-v\| \le \|T(u)-T(v)\| \le (1+\epsilon)\|u-v\|, \forall u, v \in P\big) \ge 1/2.$$

The proof of Theorem 1[1] relies on the existence of a random linear map, $T : \mathbb{R}^d \to \mathbb{R}^k$ that satisfies the following condition: if $x \in \mathbb{R}^d$, then

$$\mathbb{P}\big((1-\epsilon)\|x\| \le \|T(x)\| \le (1+\epsilon)\|x\|\big) \ge 1 - \frac{1}{n^2}. \quad (1)$$

The proof then follows by choosing $\delta = 1/n^2$, and applying the result of either of the next two together with the union bound. The next two theorems provide two particular families of mappings T, that can be used in Theorem 1. In both theorems, the mapping T is of the form $T(x) = XR^T$, where R=$R_{k\times d}$ is the projection matrix, and X=$X_{n\times d}$ is the data structure. Theorem 2 requires that elements of R are i.i.d. random variables, with mean 0, unit variance, and uniform a subgaussian tail, while Theorem 3 uses a sparse projection matrix.

Definition 1: **Subgaussian Tails**
Let X be a real-valued random variable, with $\mathbb{E}(X) = 0$. X is said to have a *subgaussian upper tail* if $\exists\ a > 0$ so that

$$\mathbb{P}(X > \lambda) \le \exp(-a\lambda^2), \quad (2)$$

for every $\lambda > 0$. If there is some λ_0 such that equation (2) holds only when $\lambda \in (0, \lambda_0)$, then we say that X has a subgaussian upper tail *up to* λ_0. Furthermore, we say that X has a *subgaussian tail* if both X and $-X$ have subgaussian upper tails. Lastly, suppose that $X_1, X_2, \cdots$ is a sequence of random variables, each with subgaussian tail. If the constant a in the subgaussian tail inequality is the same for each X_i, then we say that the X_is have a *uniform subgaussian tail*.

Theorem 2: Consider a collection $\{R_{ij}\}_{i,j}$ of independent random variables, where $\mathbb{E}(R_{ij}) = 0$ and $\mathbb{V}(R_{ij}) = 1$ for each R_{ij} and also, suppose that $\{R_{ij}\}_{i,j}$ has a uniform subgaussian tail. Next, for fixed $d \in \mathbb{N}$, $\epsilon \in (0, 1/2]$, $\delta \in (0, 1)$, let us set $k = \frac{C\log(2/\delta)}{\epsilon^2}$, for $C \ge 384(1+8/a_R)^2$, where a_R is the constant in the subgaussian upper tail of the R_{ij}s. Finally, let us define the random linear map $T : \mathbb{R}^d \to \mathbb{R}^k$ as follows:

$$T(x)_i = \frac{1}{\sqrt{k}}\sum_{j=1}^{d} R_{ij}x_j, \text{ for } i = 1, 2, \cdots, k,$$

where $T(x)_i$ is the i^{th} coordinate of $T(x) \in \mathbb{R}^k$, and x_j is the j^{th} coordinate of $x \in \mathbb{R}^d$. For every $x \in \mathbb{R}^d$, it turns out that

$$\mathbb{P}\big((1-\epsilon)\|x\| \le \|T(x)\| \le (1+\epsilon)\|x\|\big) \ge 1 - \delta.$$

Theorem 2, can be improved upon by further requiring that the projection matrix is sparse. That is, define the mapping $T = XS^T$, where elements of S are i.i.d. according to the following distribution

$$S_{ij} = \begin{cases} q^{-1/2} & \text{with probability } q/2, \\ -q^{-1/2} & \text{with probability } q/2, \\ 0 & \text{with probability } 1-q. \end{cases}$$

[1] In fact, all known proofs of the JL-Lemma rely on statements akin to (1).

In this case, the mapping T can be used to find a JL embedding provided the data points in X are sufficiently well-spread[2].

This idea was first introduced by Achlioptas [1], who considers the two specific cases where $q = 1$ and $q = 1/3$, and shows that $q = 1/3$ is nearly optimal. Ailon and Chazelle [2], then extend this idea by considering highly sparse matrices with $q \to 0$; they show that, as the sparsity of our projection matrix increases, so too does the need for our data points to be well-spread across the dimensions in which they are observed. That is, if our projection matrix consists largely of 0s, then each coordinate of a data point x should hold about the same mass as each other coordinate.

One advantage to using projection coefficients that are i.i.d uniform over $\{-1, 1\}$ is that each coordinate $T(x)_i$ of our projection involves only addition and subtraction of the original coordinates x_j. More specifically, $T(x)_i$ is calculated as follows: partition the coordinates of x randomly into two groups, compute the sum of each group, and set $T(x)_i$ to be the difference of these two sums. This greatly improves runtime when searching for a JL-embedding, since we need not perform repeated matrix multiplication (as is the case when our projection coordinates are i.i.d. gaussian random variables).

If we use i.i.d. projection coefficients with distribution equal to that of S, then we can obtain a JL-embedding about q times faster than when using projection coefficients that are uniform over $\{-1, 1\}$. This is because, in both cases, computation of each coordinate $T(x)_i$ involves addition and subtraction of the original coordinates, but when the projection coefficients are distributed as S, only about q of the original coordinates are considered, with the remaining coordinates sent to 0.

Before moving on, it is useful to note that Theorem 2 can be applied when projection coefficients are i.i.d. according to S, since S is mean 0, unit variance, and S has a subgaussian tail with coefficient $a_S = q^2/2$ (a simple exercise) . However, recall that the reduced space has dimension $k = \frac{C \log(2/\delta)}{\epsilon^2}$, where $C \geq 384(1 + 8/a_S)^2$, so that $q \to 0$ implies $a_S \to 0$, which further implies $k \to \infty$. Therefore, Theorem 2 is not practical when dealing with highly sparse projection matrices distributed according to S.

The following provides a formal discussion of JL-embeddings using sparse projection matrices, following closely the work present in [13]. The key difference between this theorem and Theorem 2 is that the reduced dimensionality k no longer depends on the constant a_S, so long as x is sufficiently well spread.

Theorem 3: Let each of $d \in \mathbb{N}^+$, $\epsilon \in (0, 1/2)$, $\delta \in (0, 1)$, and $\alpha \in [d^{-1/2}, 1]$ be parameters, and define the *sparsity parameter*

$$q = C_0 \alpha^2 \log(d/\epsilon\delta),$$

where $C_0 \geq 1$ and all parameters are chosen in such a way that $q \in [0, 1]$. Next, define the i.i.d. random variables

$$S_{ij} = \begin{cases} q^{-1/2} & \text{with probability } q/2, \\ -q^{-1/2} & \text{with probability } q/2, \\ 0 & \text{with probability } 1 - q, \end{cases}$$

for $i = 1, \cdots, k$, $j = 1, \cdots, d$. Next, set $k = C\epsilon^{-2} \log(4/\delta)$, where $C \geq 768$, and define the random linear mapping $T : \mathbb{R}^d \to \mathbb{R}^k$ as follows:

$$T(x)_i = \frac{1}{\sqrt{k}} \sum_{j=1}^{d} S_{ij} x_j,$$

for $i = 1, \cdots, k$. Then if $x \in \mathbb{R}^d$ such that $\|x\|_\infty \leq \alpha \|x\|$, it follows that

$$\mathbb{P}\big((1 - \epsilon)\|x\| \leq \|T(x)\| \leq (1 + \epsilon)\|x\|\big) \geq 1 - \delta.$$

3.1 Methodology

Our goal is provide a more specific bound on k than that given by Matoušek. Matoušek gives the same bounds as those given in Theorems 2 and 3, only in both cases, he does not give a specific bound on the constant C but rather, he simply asserts that C is "a sufficiently large constant". First, we reprove Matoušek's results in a more detailed manner in order to obtain specific lower bound on the constant C. Next, we perform a variety of simulations in order to empirically estimate the bound on C.

4. Theoretical Results

Through mathematical analyses similar to those used by Matoušek, we obtain the bounds on k given in Theorems 2 and 3. That is, when using subgaussian projection coefficients we obtain $k = C \log(2/\delta)/(\epsilon^2)$, where $C > 384(1 + 8/a_R)^2$, and where a_R is the coefficient in the subgaussian tail inequality of the projection coefficients R. On the other hand, when using sparse projection matrices we obtain $k = C \log(4/\delta)/(\epsilon^2)$, where $C > 768$.

5. Simulation Results and Discussion

The simulations were performed using Matlab R2013b and using the default random number generator, i.e. the random seed automatically generated by Matlab. To simulate the result of Theorem 2, the projection coefficients were chosen to be standard normal random variables (scaled so that the expected length of each row is equal to 1) using the built-in function normrnd. To simulate the result of Theorem 3, the projection coefficients were chosen to be multinomial distributed over $\{-q^{-1/2}, 0, q^{-1/2}\}$, where 0 has probability $1-q$, $\pm q^{-1/2}$ each have probability $q/2$, and where q is proportional to the L^∞ norm of the simulated

[2]A unit vector is well-spread if it is close to $\frac{1}{\sqrt{d}}(\pm 1, \pm 1, \cdots, \pm 1)$, while something close to $(1, 0, \cdots, 0)$ is not well-spread since most of its mass lies in its first dimension.

data points in accordance with Theorem 3. In an attempt to compare the results for different types of data, datasets were simulated using four different probability distributions: Uniform, Cauchy, Mixed Non-Central Cauchy, and Mixed Beta. These distributions are available through the built-in Matlab functions: unifrnd, trnd, nctrnd, and betarnd; in order to construct each of the mixed distributions, points were randomly selected from two different distributions, which further required use of the built-in function rand.

Each of the simulated datasets consist of $n = 1000$ 10000-dimensional points which are projected into lower dimensions using several different choices of the parameters ϵ and δ, and the values for C suggested by Theorems 2 and 3. The theory is then tested by using the relative frequency approach in order to estimate the probability that each JL embedding maintains ϵ-distortion. That is, for each simulated data set, and for each choice of ϵ and δ, we construct the mapping T using projection matrices outlined in Theorems 2 and 3. Then, for each simulated point x, we compute the ratio $\|T(x)\|/\|x\|$; if this ratio is within $(1-\epsilon, 1+\epsilon)$, then this particular embedding is considered to be a success. Finally, for each simulated data set, and each choice of ϵ and δ, the probability of success is estimated by the number of successful embeddings, over the number of points, $n = 1000$.

Now, according to Theorems 2 and 3, for each fixed ϵ and δ, each point should preserve ϵ-distortion with probability of at least $1-\delta$. However, for each simulated data set, 100% of the projected points preserve ϵ-distortion. This very high frequency of success seems unusual, especially for situations when δ is chosen to be rather large. This discrepancy between the theoretical and empirical results is likely due to an inflated bound on the constant C in each of Theorems 2 and 3. For this reason, the above simulations are repeated using smaller and smaller values of C until the probability bound appears to fall closer to the expected bound of $1-\delta$. Repeating the simulations in this way seems to suggest a significantly lower bound on the reduced dimensionality k than that suggested by the theory. In particular, the simulations consistently suggest that the constant C is between 0.5 and 2.

Concrete Example: The following example illustrates the above discrepancy between the theorized value for C and that which is suggested by simulations. Using $\delta = 0.2$, $\epsilon = 0.5$. and the sparse projection matrix given in Theorem 3, we project $n = 1000$ uniformly random, 10000-dimensional datapoints into k dimensions, where

$$k = C\log(4/\delta)/(\epsilon^2).$$

Thus, our choices of $\delta = 0.2$ and $\epsilon = 0.5$, together with the bound $C > 768$ imply

$$K > 768\log(4/0.2)/(0.5^2) = 9202.$$

Thus, the random mapping T sends a 10000-dimensional point x to the 9202-dimensional point $T(x)$ such that

$$P(1-\epsilon < ||T(x)||/||x|| < 1+\epsilon) > 1-\delta.$$

Due to our choices of $\delta = 0.2$ and $\epsilon = 0.5$, we should therefore expect

$$P(0.5 < ||T(x)||/||x|| < 1.5) > 0.8. \quad (3)$$

Now, in order to check the validity of (3), we simply compute the ratio of norms $||T(x)||/||x||$ for each projected point and count the number of projections that are not distorted by more than 0.5. Finally, we estimate the probability of success with the relative frequency of such successful projections.

Using the value of $C = 768$, we obtain a success rate of 100%, which is quite large compared to the probability bound of 0.8 suggested by Theorem 3. Accordingly, the above was repeated using smaller and smaller values of C until a value was found that seems to have roughly 80% success rate. It turns out that for C as low as $C = 10$, we still have 100% success rate. Choosing $C = 1$, leads to 93.9% success; choosing $C = 0.75$ leads to 89.5% success, choosing $C = 0.5$ leads to 79.6% success probability. Thus, using the value $C = 768$, given in Theorem 3, leads to a reduced dimensionality of $k = 9202$, whereas the simulations suggested instead that we can use $C = 0.5$ which leads to $k = 6$.

There are a few questions that should follow from the result of this example:

1) Do these results change significantly if we use different data points? (In this particular example, the points were simulated by generating uniformly random 10000-dimensional vectors). The answer is that after generating various random data sets and repeating the above approach, it seems that the type of data point is not a major factor contributing to the huge discrepancy between the reduced dimensionality k obtained by the math vs that obtained by simulations (different data results in slightly different reduced dimensionality, maybe as high as 20 dimensions, but never anything close to 9202).
2) Do these results change significantly if we try different values for the parameters ϵ and δ? The answer is that it does not seem to matter. Changing the values of ϵ and δ leads to different values of k and different probabilities of success (according to the math) but once again, the probabilities are consistently far too high for any fixed k, and in order to make the simulated probability (relative frequency of successful projections) match with the theoretical probability of $1-\delta$ we need to make the constant C much smaller than the value of 768 given in the theorem.
3) Do these results change significantly if we use Theorem 2 instead of Theorem 3? Once again, it seems

that the observed discrepancy is not due to the choice of theorem, but agian due to an inflated value of C.

In summary, the mathematical bounds are far too large and not of much practical use. However, the simulated results seem to suggest that the value C can simply be estimated and tweaked to the particular dataset. Moreover, the simulated results suggest a much more practical result. In the above, for example, the math says that we can go from 10000 dimensions to 9202 (not very helpful), while the simulated results suggest that we can go from 10000 dimensions into only 6 dimensions (very useful indeed).

6. Conclusions

We have discussed a non-statistical method of dimensionality reduction, where any given set of points can be embedded into lower dimensions, although such embeddings are typically subject to some form of distortion. Regardless of the initial dimensionality, the JL-lemma guarantees the existence of a lower dimensional representation, the dimensionality of which depends on the number of points as well as the level of distortion one is willing to accept.

Mathematics gives a weaker bound on k than do our simulations. In particular, the simulations seem to suggest that C is generally around $C = 1$. This means that our mathematical result (in particular, the bound on C) is hundreds of times larger than the simulations suggest (or even thousands when using Theorem 2 , depending on the choice of subgaussin projection coefficients) and as such, our bound on k is hundreds (to thousands) times larger than that which is suggested by simulation.

References

[1] D. Achlioptas, *Database-friendly random projections: Johnson-Lindenstrauss with binary coins*. Journal of Computer and System Sciences 66: 671–687, 2003.

[2] N. Ailon and B. Chazelle, *Approximate nearest neighbors and the fast Johnson-Lindenstrauss transform*. Proceedings of the 38th ACM Symposium on the Theory of Computing, 2006, pp. 557-563.

[3] K.M. Carter , *Dimensionality reduction on statistical manifolds*, PhD thesis, The University of Michigan, Department of Engineering, 2009.

[4] S. Dasgupta and A. Gupta, *An elementary proof of the Johnson-Lindenstrauss lemma*. Technical Report 99-006, UC Berkely,March 1999.

[5] D. Donoho, *Aide-memoire. High-dimensional data analysis: the curses and blessings of dimensionality*, (2000), available at http://statweb.stanford.edu/ donoho/Lectures/AMS2000/Curses.pdf.

[6] P. Frankl and H. Maehara, *The Johnson-Lindenstrauss lemmas and the sphericity of some graphs*, Journal of Combinatorial Theory Series B 44(3):355-362, 1988.

[7] G. Heo, J. Gamble, P.T. Kim. *Topological analysis of variance and the maxillary complex*, J Am Stat Assoc. 2011

[8] H. Gao, W. Hong, J. Cui, Y. Zhao and H. Meng, *Pattern recognition of multivariate information based on non-statistical techniques*, International Conference on Information and Automation, 2008, pp.697–702.

[9] J. Gamble, H. Geo, *Exploring uses of persistent homology for statistical analysis of landmark-based shape data*, J Multivar Anal. 101:2184-2199, 2010.

[10] A.C. Gyllensten and M. Sahlgren , *Navigating the semantic horizon using relative neighborhood graphs*, 2015, CoRR, abs/1501.02670.

[11] P. Indyk and R. Motwani, *Approximate nearest neighbors: towards removing the curse of dimensionality*, 30th Annual ACM Symposium on Theory of Computing, Dallas, TX, ACM, New York, 1998, pp.604-613.

[12] W.B. Johnson and J. Lindenstrauss, *Extensions of Lipshitz mappings into a Hilbert space*, Conference in modern analysis and probability, New Haven, CI, 1982, American Mathematical Society, Providence, RI, 1984, pp.189-206

[13] J. Matoušek, *On variants of the Johnson-Lindenstrauss lemma*, Random structures and algorithms 33(2):142–156, 2008.

[14] J. Wang, *Classical multidimensional scaling. Geometric Structure of High-Dimensional Data and Dimensionality Reduction*, (2011) Springer Heidelberg Dordrecht, London New York, pp. 115-129.

Correlating Algorithm and Model for Personality Features with Academic Relevance in Big Data

Muhammad Fahim Uddin[1], **Jeongkyu Lee**[2]
[1'2]School of Computer Science and Engineering, University of Bridgeport, Bridgeport, CT, USA

Abstract – *In last decades, social networking such as Twitter, Google blogs and Facebook has provided great potential and opportunity to capture huge dataset about individuals and their behaviors, as they exhibit in the posts, blogs and tweets. Big Five Personality Traits have widely been used in research to categorize individuals. In this paper, we propose a Correlating Algorithm and Model (CAM). This algorithm efficiently correlates personality features with relative academic attributes that shows better identification of various individual types for various academics. This is part of our research work in progress that uses big and unstructured data to do the data mining and analytics to correlate the relevant features for predicting good fit students and good fit job candidates. We provide related study and conclude with future works.*

Keywords: Social Networking data, Unstructured Data, Big Data, Personality Prediction, Personality Traits.

1 Introduction

According to IBM (2012), we have created about 90 % of all the data in last 2-3 years. This growth has been promoted and facilitated by Internet of things, and today's web technologies and platform such as Facebook, Twitter, Google blogs, Flicker, etc. These big data companies store and process huge data set every minute. Many software companies such as Oracle, IBM, Microsoft, SAP and others have provided great tools and research to exploit the potential of Big data for predictions, pattern recognitions, data mining, artificial intelligence, cognitive computing and other real world challenges[1][2][3][4][5][6][7]. Individuals spend significant time on these portals daily and reveal great data about their personalities, behaviors and type in real world. This fact has encouraged lot of research to predict personality utilizing many algorithms and famous model known as Big Five Personality Traits.[8][9][10]. Facebook and Twitter's data has been used in many researches to predict personality using machine learning algorithms and other great data mining and analytics techniques.[11][12][13][14][15]. We propose CAM to identify a particular personality trait that has implication in academic choices. In section 2, we provide related work and discuss some great opportunities and challenges. In Section 3, we discuss our model and algorithm and show some promising results in context of our main research (PAE) in section 4. In Section 5, we conclude with future work.

1.1 Motivation

Our motivation is based on avoiding poor academic choices and reducing drop outs, poor academic performances, costly re-admissions, change of study majors and colleges, etc. We focus on features and structures that can contribute to Predicting Educational Relevance For an Efficient Classification of Talent (PERFECT) Algorithm Engine (PAE) that is our main research in progress. This engine includes various algorithms and model that work in conjunction to produce Good Fit Students (GFS) and Good Fit job Candidates (GFC). This paper presents an important milestone of PAF. We use Python libraries, Microsoft SQL Server and Excel Data mining and analysis tools.

2 Related Work

Authors in[16] have extended the previous research about profile pictures and personality impression. They looked at relation between profile picture selection and message user is intending to the world through their profile on Facebook. Their survey and feedback from sample users as chosen concluded that users are aware of the importance of selecting the type of profile picture they keep for short or long term. They found User's personality traits that had an influence on the picture choice they make as profile picture. For example, extraverted users select more self-representative photos and narcissistic users select more physically attractive pictures. Work done by[17] reveals that users wall and newsfeed are important segments to investigate and research to further understand personality patterns and self presentation. They categorized self presentational information on wall and self presentational behaviors at news feed, to research personality traits and study their inter-relation. Study in [18] supports the potential of big social data to predict a five factor model of personality. They cited the relevant study done to indicate the accuracy of personality prediction is in moderate range with typical correlation between the prediction and personality is in the range of r = 0.2 and r = 0.4, where r = reliability.(good references in the paper, use it if needed). Authors in [19] shows significance work done towards distinguishing personality types (out of Big five) for both popular users and influential's user's posts. Their study supports that popularity is linked to imaginativeness and influential users show also organized behavior. They use three counts or parameters in twitter data set as i) following, ii) followers, iii) listed counts. They show that root mean squired error below 88 % for prediction of user's five personality traits in active user status.

They argue that privacy data access is still a hurdle in improving accuracy and still remains an open problem. They propose for future research three important directions, Marketing, User Interface design and Recommender system, based on personality traits that are revealed by studying user's data.

Study in [20] supports predicting personality with social behavior only in light of Big Five traits. Authors outline features of user behavior with the following groups. Network Bandwidth(NET), Message Content(MSG), Pair Behavior(PAIR), and Reciprocity of actions (REC), Informativeness(INF) and Homophily(HOM). Their results verify that personality can equivalently be predicted using behavior features as with text features. A detailed survey on Personality Computing in [21], elaborates on Automatic Personality Perception, Automatic Personality Synthesis and Automatic Personality Recognition. A work done in [22] shows that two important traits of personality as conscientiousness and agreeable predicted less dishonesty in academics. Though their meta-analytic results were limited to small number of studies, did contribute to a better understanding of factors that influenced such behaviors in academics. Their study opens a wide door to pursue further research to better understand more personality features to find out potential in students in academic world towards non-ethics, such as cheating, dishonesty, etc. A survey in[23] discussed the new research area coined s SSP (Social Signal Processing) , between social animal, human and unsocial machine in three main problems. i) Modeling, ii) Analysis, iii) Synthesis of nonverbal behavior in social interactions.

2.1 Big Five Personality Traits/Five Factor Model

Big Five Model/Five Factor Model[8][9][24] known as OCEAN Traits has been used for many researches to predict the personality type and categorization.

Figure 1 shows the traits of this model.

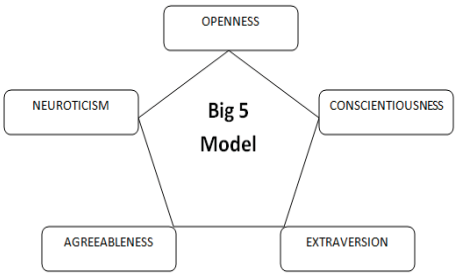


Figure 1 – OCEAN Traits

There are few weaknesses reported in literature[25] that motivates our research to produce ISPF. The weaknesses are briefly summarized as lack of independence variables, based on questionnaire, inability to consider current health issues that can impact one's attitude, identify built in talent of individuals that can contribute to academics, inaccuracy to predict to specific behavior, inability to predict based on specific culture and circumstances and lacks some other attributes of human like humor, sincerity, honesty, etc.

3 Proposed Algorithm and Mathematical Model

3.1 CAM System Flow

Personality Prediction[26][27][28] and Feature Extraction[29] is focus of various researchers for developing great recommendation and prediction models and engines, such as Amazon, Netflix, eBay and other great big data companies use. In the below visual, we show a framework to utilize the data from sources such as Facebook profiles, blogs and tweets to collect in a data repository (structured and unstructured data). Then we run CAM on data to identify and correlate features through lens of academic features and measure predictive performance for individuals and we compare this with existing research using Big Five Personality traits as a reference. We show improved results. We provide details in the following sections.

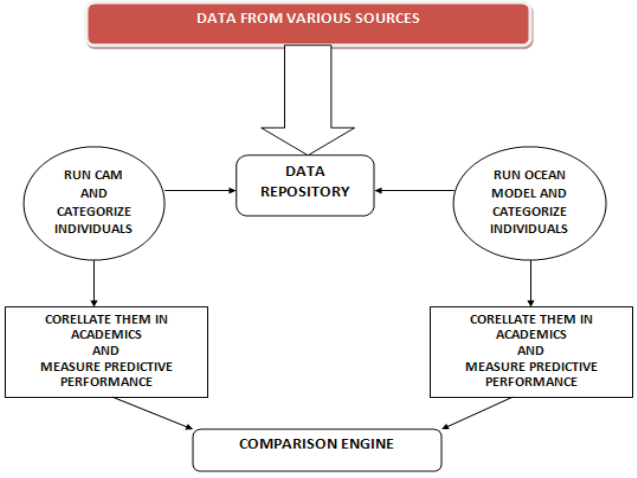


Figure 2 – CAM based Personality traits Correlation

3.2 Academic Attributes

We use the following attributes.

Demographic, Pre-college (GPA, SAT scores, SAT scores, ACT scores, etc), Field of Study (Pre-admission), Field of Study (At time of admission), Parents field of study, Marital Status at time of Admission, Marital Status at time of Graduation, Degree Level, English language Proficiency, Age, Type of Semester Projects and Final Project, If Orphan, Father Education, Mother Education, Family income, No. of Siblings, Religion, Is International, No. Of Absences.

3.3 Proposed CAM

For our proposed algorithm and model, we build on idea of personality talent of an individual prediction that can be applied to several industries, such as Academia and Real world careers. We present the CAM Algorithm as below, due to limit of paper, we omit details of algorithms in this paper. However, we use the language that is self explanatory.

Algorithm to Use CAM Model

1. BEGIN
2. LOAD [Data Repository]
3. WHILE DATASET is UNIQUE
4. {
   a. LOAD Test data
   b. EXTRACT OCEAN Features
   c. CORELLATE with Academics
   d. MEASURE Probability of Prediction for Performance
      i. FOR EACH TRAIT in OCEAN
         1. {
            Calculate_Efficiency();
            }
   e. LOAD Test data
      i. RUN CAM
      ii. CORRELATE with Academics
      iii. MEASURE Probability of Prediction for Performance
         1. FOR EACH TRAIT in CAM
            {
            Calculate Efficiency ();
            }
5. COMPARE Results
6. }
7. END

Due to the page limit of this paper, the detailed mathematical model is omitted. To develop CAM, we must first identify structures and noise (irrelevant unstructured data) from the testing sample; we run our algorithm to correlate personality features with academic records. In Equation (1), we show the Probability of Noisy data (₦ = Noisy Data) and ₴ = Signal or Structured data.

$$\mathbb{P} = P(₦ * ₴) \quad (1)$$

$P(₦ * ₴)$ is associated with probability score and obtained from various sources, such as Facebook, twitter, etc.
In Equation (2), we separate signal/structured data:

$$₴ = \frac{I}{N} + \sum_{i=0}^{\infty} (i(\mathbb{P} - P(₦)) \quad (2)$$

In Equation 3, we estimate the size of data under test and then we calculate the time that it takes to process over a test period.

$$S(₴) = \left\{ \prod_{i=1}^{k} \sum_{t=0}^{l} [i(₴1, ₴2, ₴3, \ldots . ₴n)] * \Delta l \right\} \quad (3)$$

Finally, we present equation 4, for efficiency that utilizes the acceptable size as per equation 3 and correlate the relevant features from data set.

$$Ĕ = \mathbb{P}\left(S(₴)\frac{\omega}{1+\omega}\right. \quad (4)$$

Where Ĕ = Efficiency that CAM calculates based on data size and ω = a correlation function that is based on best fitness match for the relevant features from academic data and social networking dat.

4 Results and Discussions

In Figure 3 – We show the time taken in seconds to process data in MB. As data size increases, it takes more time to search data. We show this as a reference to support the speed improvement of our work in results given below.

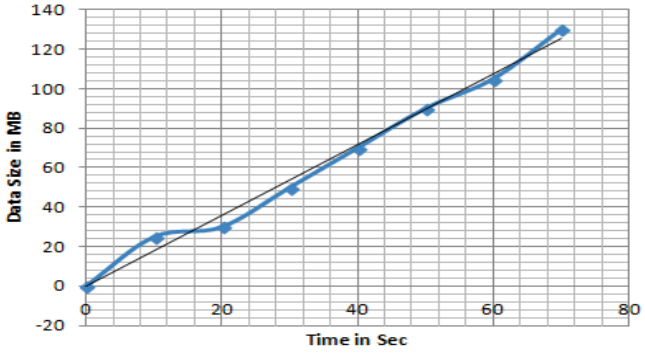


Figure 3 – Time taken to process certain size of data

Figure 4 shows, the processing efficiency of Feature search using Big 5. It can be seen that for low data size, the efficiency was much higher as 70 % but as data size increases, it drops and finally saturate at 300 MB sample size. This saturation behavior needs more study as future work.

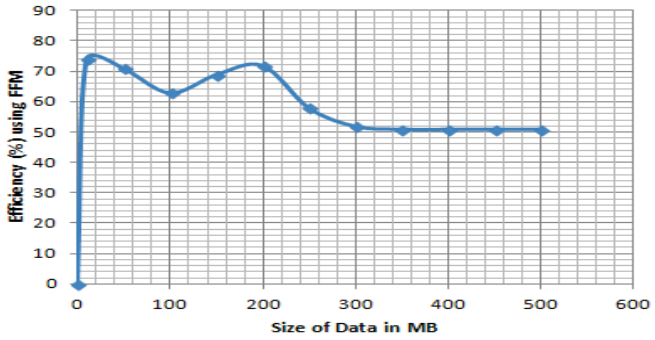


Figure 4 – Show efficiency using Five Factor Model

Figure 5 shows the results given by using CAM based feature correlation. Though, it shows little improvement as compared to Big Five traits model, but we can see better speed of prediction in Figure 6 due to the fact that CAM based correlation and prediction performance better for specific domain such as academics.

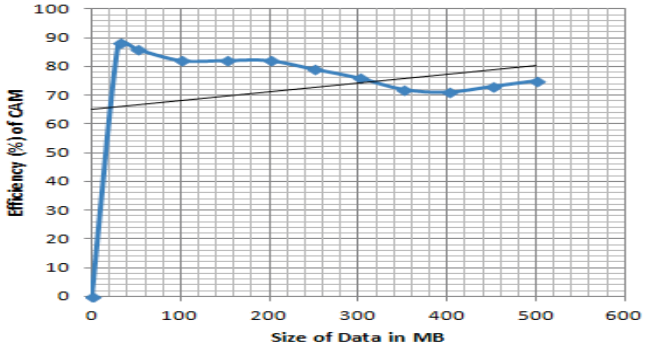


Figure 5 – Shows improved efficiency using proposed model.

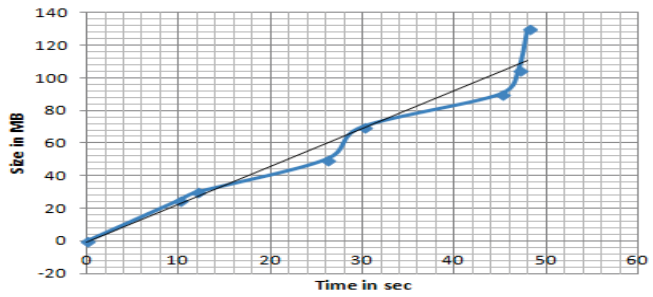


Figure 6 – Shows little improvement in time efficiency.

5 Conclusions and Future Work

In paper, we show a subset of our research in regard to *Prediction Educational Relevance For an Efficient Classification of Talent*, PERFECT Algorithm Engine. We propose an algorithm of it, as CAM (Correlating Algorithm and Model) for academic relevance for individual personality. We show improved results as compared to Big Five traits that are too general and lack flexibility. Our work shows better efficiency to correlate relevant features. We observe saturation at 300 MB that requires further research in using Big Five traits. In CAM, we find higher efficiency of about 74 % for 500 MB data set. We seek to improve these results even further as future work and propose few more algorithms.

6 References

[1]J. Dijcks, "Oracle: Big data for the enterprise," *Oracle White Pap.*, no. June, p. 16, 2012.

[2] M. Gualtieri and R. Curran, "The Forrester Wave™: Big Data Predictive Analytics Solutions, Q2 2015," *Forrester Res.*, pp. 1–18, 2015.

[3] C. M. Bishop, *Pattern Recognition and Machine Learning*, vol. 4, no. 4. 2006.

[4] C. Perez, "Big Data Analytics with Oracle," 2012.

[5] D. Sheet, "IBM accelerators for big data."

[6] Oracle, "Big Data & Analytics Reference Model," no. September, pp. 1–44, 2013.

[7] Oracle, "Big Data Analytics," no. March, 2013.

[8] J. A. N. Ciecіuch, "the Big Five and Belbin," 2014.

[9] D. P. Schmitt, J. Allik, R. R. McCrae, and V. Benet-Martinez, "The Geographic Distribution of Big Five Personality Traits: Patterns and Profiles of Human Self-Description Across 56 Nations," *J. Cross. Cult. Psychol.*, vol. 38, no. 2, pp. 173–212, 2007.

[10] S. Bai, B. Hao, A. Li, S. Yuan, R. Gao, and T. Zhu, "Predicting big five personality traits of microblog users," *Proc. - 2013 IEEE/WIC/ACM Int. Conf. Web Intell. WI 2013*, vol. 1, pp. 501–508, 2013.

[11] M. Kosinski and D. Stillwell, "Personality and Patterns of Facebook Usage," 2012.

[12] D. Chapsky, "Leveraging online social networks and external data sources to predict personality," *Proc. - 2011 Int. Conf. Adv. Soc. Networks Anal. Mining, ASONAM 2011*, pp. 428–433, 2011.

[13] J. Golbeck, C. Robles, M. Edmondson, and K. Turner, "Predicting personality from twitter," *Proc. - 2011 IEEE Int. Conf. Privacy, Secur. Risk Trust IEEE Int. Conf. Soc. Comput. PASSAT/SocialCom 2011*, pp. 149–156, 2011.

[14] R. Wald, T. Khoshgoftaar, and C. Sumner, "Machine prediction of personality from Facebook profiles," *Proc. 2012 IEEE 13th Int. Conf. Inf. Reuse Integr. IRI 2012*, pp. 109–115, 2012.

[15] D. Stillwell and M. Kosinski, "The personality of popular facebook users," *Proc. ACM 2012 Conf. Comput. Support. Coop. Work*, pp. 955–964, 2012.

[16] Y.-C. J. Wu, W.-H. Chang, and C.-H. Yuan, "Do Facebook profile pictures reflect user's personality?," *Comput. Human Behav.*, Dec. 2014.

[17] E. Lee, J. Ahn, and Y. J. Kim, "Personality traits and self-presentation at Facebook," *Pers. Individ. Dif.*, vol. 69, pp. 162–167, Oct. 2014.

[18] B. R. Lambiotte and M. Kosinski, "Tracking the Digital Footprints of Personality," vol. 102, no. 12, pp. 1934–1939, 2014.

[19] D. Quercia, M. Kosinski, D. Stillwell, and J. Crowcroft, "Our Twitter Profiles, Our Selves: Predicting Personality with Twitter," *2011 IEEE Third Int'l Conf. Privacy, Secur. Risk Trust 2011 IEEE Third Int'l Conf. Soc. Comput.*, pp. 180–185, Oct. 2011.

[20] S. Adali and J. Golbeck, "Predicting Personality with Social Behavior," *2012 IEEE/ACM Int. Conf. Adv. Soc. Networks Anal. Min.*, pp. 302–309, 2012.

[21] A. Vinciarelli and G. Mohammadi, "A survey of personality computing," *IEEE Trans. Affect. Comput.*, vol. 5, no. 3, pp. 273–291, 2014.

[22] T. L. Giluk and B. E. Postlethwaite, "Big Five personality and academic dishonesty : A meta-analytic review," *Pers. Individ. Dif.*, vol. 72, pp. 59–67, 2015.

[23] A. Vinciarelli, M. Pantic, D. Heylen, C. Pelachaud, I. Poggi, F. D'Errico, and M. Schröder, "Bridging the gap between social animal and unsocial machine: A survey of social signal processing," *IEEE Trans. Affect. Comput.*, vol. 3, no. 1, pp. 69–87, 2012.

[24] G. Chittaranjan, B. Jan, and D. Gatica-Perez, "Who's who with big-five: Analyzing and classifying personality traits with smartphones," *Proc. - Int. Symp. Wearable Comput. ISWC*, pp. 29–36, 2011.

[25] G. J. Boyle, "Critique of the five-factor model of personality," 2008.

[26] O. Celiktutan, E. Sariyanidi, and H. Gunes, "Let me tell you about your personality!†: Real-time personality prediction from nonverbal behavioural cues," *2015 11th IEEE Int. Conf. Work. Autom. Face Gesture Recognition, FG 2015*, p. 6026, 2015.

[27] D. Nie, Z. Guan, B. Hao, S. Bai, and T. Zhu, "Predicting Personality on Social Media with Semi-supervised Learning," *2014 IEEE/WIC/ACM Int. Jt. Conf. Web Intell. Intell. Agent Technol.*, pp. 158–165, 2014.

[28] O. Celiktutan and H. Gunes, "Automatic Prediction of Impressions in Time and across Varying Context: Personality, Attractiveness and Likeability," *IEEE Trans. Affect. Comput.*, vol. 3045, no. January 2016, pp. 1–1, 2016.

[29] A. Arnold, J. E. Beck, and R. Scheines, "Feature discovery in the context of educational data mining: An inductive approach," *AAAI Work. - Tech. Rep.*, vol. WS-06–05, pp. 7–13, 2006.

[30] M. A. U. D. Khan, M. F. Uddin, and N. Gupta, "Seven V's of Big Data understanding Big Data to extract value," in *Proceedings of the 2014 Zone 1 Conference of the American Society for Engineering Education - "Engineering Education: Industry Involvement and Interdisciplinary Trends", ASEE Zone 1 2014*, 2014.

Gremlin By Example

Kornelije Rabuzin [1], **Mirko Maleković**[1], **Martina Šestak**[1]
[1]Faculty of Organization and Informatics, University of Zagreb, Varaždin, Varaždin, Croatia

Abstract— *Graph databases, which are presently classified as one of the most interesting database types, are being used on a daily basis. However, different systems are available and different languages are used to operate using graph databases. To summarize, a large number of languages hinders graph databases' wider use. There also have been attempts to create universal SQL for graph databases in order to make things more standardized and intuitive to use. In this paper, we propose Gremlin By Example. This is a new language for graph databases that resembles Query By Example, which is a query language for relational databases. Because people are familiar with Query By Example, and Gremlin By Example looks almost the same, by using the similar interface people should be able to use graph databases without knowing the details of different graph query languages.*

Keywords: NoSQL, SQL, graph databases, Gremlin

1 Introduction

NoSQL databases are becoming increasingly important and are being used in different domains in order to store and manage large amounts of semi- and unstructured data. Among the different available NoSQL database types (document-oriented, column-oriented, etc.), graph databases are quite important.

Graph theory is not new. In the eighteenth century, Euler solved the problem of the bridges of Königsberg by using the graph theory. The problem was whether it was possible to cross all seven bridges only once in order to visit all parts of the city. Euler managed to prove that this was not possible. By representing each land mass by means of a node (vertex), and the nodes were connected by means of lines used to represent the bridges (edges). Such a structure is called a graph. Thus, the graph theory is not new, but the use of graph theory as a concept for storing data could be called new—at first. However, the idea of network databases is somehow similar, as we will demonstrate.

Graph databases store information in nodes and in relationships between nodes. Such a way of representing and handling information is intuitive and easy to understand. Nodes can have a different number of attributes, and they can be labeled as well. A small graph database is given below; it contains cities in Australia as well as distances between the cities. Each city, as well as each relationship, could have different properties. The CREATE statement, which creates nodes and relationships, is given below (Cypher Query Language is used as well as Neo4j, which is the most popular graph database system).

CREATE (c1:City {name: "Sydney", population: 4840600}),

(c2:City {name: "Melbourne", population: 4442919}),

(c3:City {name: "Brisbane", population: 2274600}),

(c4:City {name: "Canberra", population: 381488}),

(c1)-[:distance {km: 893}]->(c2),

(c1)-[:distance {km: 1027}]->(c3),

(c1)-[:distance {km: 304}]->(c4),

(c2)-[:distance {km: 1736}]->(c3),

(c2)-[:distance {km: 655}]->(c4),

(c3)-[:distance {km: 1295}]->(c4)

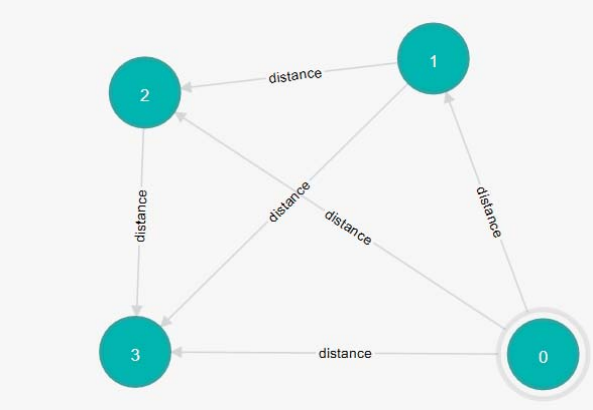


Figure 1. Graph database

As previously mentioned, the problem with graph databases was, and still is, that several languages could be used, including Cypher Query Language (CQL), Gremlin, and Datalog. However, they are all different, which hinders the wider use of graph databases. Because each language is specific, these graphs could cause problems for end users. For example, CQL has its own set of statements, and Gremlin has its own set of statements. Thus, inevitably, one has to learn (again) new languages to use graph databases. In [7] the authors showed that existing query languages (primarily CQL) do not support recursion or views and that logic programming languages could be used as well. The term "deductive graph database" was coined in the paper, which shows how Datalog could be used to query graph databases.

In relational databases, the number-one used language is SQL. SQL is a standardized language, which is supported by all database management system (DBMS) vendors. SQL statements are usually classified as follows:

- Data definition language contains several statements that are used to create, alter, and delete tables, functions, users, sequences, etc. The most important statement is the CREATE statement, but ALTER and DROP belong here as well.
- Data manipulation language contains the following set of statements: INSERT, UPDATE, and DELETE. They are used to insert new rows, to update existing rows, etc.
- Query Language contains the SELECT statement that is used to retrieve the data.

The number of statements has increased over the years, and sometimes people have problems with complex SQL statements. Fortunately, other graphically oriented alternative languages have occurred as well, including Query By Example (QBE). QBE is supported in Microsoft Access, for example (we show an example later on). Basically, one does not need to write complex SQL statements manually; on the contrary, one can build queries visually by filling in the visible table structure. In that way, queries are produced (built) visually, which is much simpler for end users as well.

In this paper, we propose a new language that should help users to use graph databases. It is especially intended for users who are familiar with relational databases and QBE. Namely, the language we propose is used to query graph databases, but it looks like QBE, and we know that users typically do know how to use QBE. Thus, basically, the language is new, but it looks similar; at the same time, it can be used to query graph databases.

The rest of this paper is organized as follows: the first graph databases are explained as well as languages that are used for graph databases. Then a new language is proposed (Gremlin By Example). Afterward, we show how GBE could be used in several examples. In the end the conclusion is presented.

2 Graph databases

As previously stated, the graph theory is not new, but the use of graph theory as a concept for storing data could be called new—at first. If we look back, one can tell that network databases were quite similar concepts in the past, and graph databases do resemble network databases. Namely, network databases occurred to solve the problems of hierarchical databases. One of the problems was that each child record could have had only one parent record (parent records could have had many child records). In network databases, this was changed, and hierarchy was no longer obligatory. Because of that, the schema of a network database also could be viewed as a graph (it contains records that are connected by means of relationships). But the network model never became popular for two main reasons: IBM supported hierarchical model, and, almost at the same time, the relational data model occurred as well. Thus, graph databases are not entirely new, but perhaps now is a good time for them because technology is mature, and all the preconditions are fulfilled at this point of time.

In the next section, we briefly present two graph query languages, including Cypher and Gremlin. Graph databases are explained in [1] and [5]. More information on graph query languages could be found in [2].

2.1 Cypher

Cypher is a popular query language for graph databases. We have already created a small graph database (above), and now we show how some queries could be implemented in Cypher. In order to find all the cities in the database, the following statement has to be written (MATCH and RETURN clauses are important for data retrieval):

```
MATCH (n: City)
RETURN n.name, n.population
```

n.name	n.population
Sydney	4840600
Melbourne	4442919
Brisbane	2274600
Canberra	381488

Returned 4 rows in 59 ms

In order to list distances between cities, the following statement should be used:

```
MATCH (m: City)-[v: distance]-(n: City)
WHERE m.name < n.name
RETURN m.name, n.name, v.km
```

m.name	n.name	v.km
Melbourne	Sydney	893
Brisbane	Canberra	1295
Brisbane	Sydney	1027
Brisbane	Melbourne	1736
Canberra	Sydney	304
Canberra	Melbourne	655

If the WHERE clause was omitted, then Brisbane–Canberra as well as Canberra–Brisbane would be included in the result (which is basically the same information).

But an interesting query is the one that finds the shortest path between the cities. Such queries are ideal. This query starts in Brisbane and ends in Brisbane, but passes through all the other cities (this is known as the traveling salesman problem).

```
MATCH p=(a)-[distance*4]-(b)
WHERE a.name='Brisbane' and b.name='Brisbane'
RETURN p
```

Now when we sum the distances on the path, and we order the result by the sum of distances in ascending order. In that way we get the shortest path:

```
MATCH p=(a)-[distance*4]-(b)
WHERE a.name='Brisbane' and b.name='Brisbane'
RETURN p, reduce(total_km = 0, n IN relationships(p)| total_km + n.km) AS TotalDistance
ORDER BY TotalDistance
```

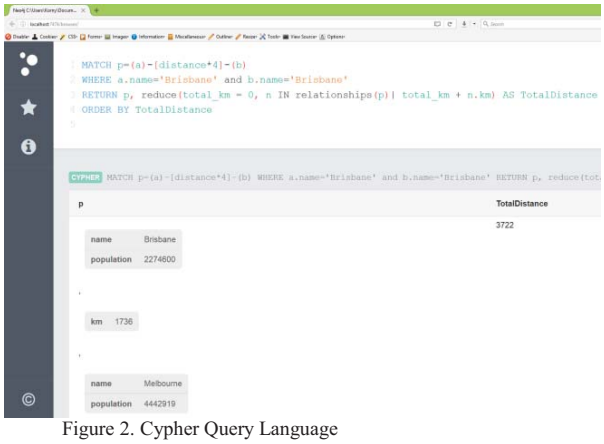


Figure 2. Cypher Query Language

Actually there are two answers with the same distance:

Table 1. Traveling Salesman problem – two solutions

Solution 1		
From	To	Distance
Brisbane	Melbourne	1736
Melbourne	Canbera	655
Canbera	Sydney	304
Sydney	Brisbane	1027
	Total	3722
Solution 2		
From	To	Distance
Brisbane	Sydney	1027
Sydney	Canbera	304
Canbera	Melbourne	655
Melbourne	Brisbane	1736
	Total	3722

Such queries as well as some other (path oriented) queries, would be much more complicated in relational databases. However, note that Cypher statements have to be entered manually (Figure 2). Syntax is not too complex, but for beginners it could be a bit confusing, especially the last example.

2.2 Gremlin

Gremlin is another language for graph databases. As with any language, it has its syntax and requires time to become a proficient user. However, there is an interesting site http://sql2gremlin.com/. This site uses the well-known Northwind database, which shows how SQL queries are related to Gremlin queries (the same query is expressed in both languages). Because we implement a few Gremlin examples later on, at this point in time we present a few examples that we borrowed from the same site. First, the SQL query is presented, and then the appropriate Gremlin query is given.

Table 2: SQL vs. Gremlin (source: http://sql2gremlin.com/)

SQL	Gremlin
SELECT * FROM Categories	gremlin> g.V().hasLabel("category").valueMap()
SELECT Products.ProductName FROM Products INNER JOIN Categories ON Categories.CategoryID = Products.CategoryID WHERE Categories.CategoryName = 'Beverages'	gremlin> g.V().has("name","Beverages").in("inCategory"). values("name")
SELECT TOP(1) UnitPrice FROM (SELECT Products.UnitPrice, COUNT(*) AS [Count] FROM Products GROUP BY Products.UnitPrice) AS T ORDER BY [Count] DESC	gremlin> g.V().hasLabel("product").groupCount(). by("unitPrice").order(local).by(valueDecr). mapKeys().limit(1)

We see that Gremlin queries are not similar to Cypher queries, and there is a debate over which of them should be used and when. Thus, different graph query languages do exist, and one common standardized language represents a challenge. It also could solve many problems faced by users of graph databases. There is even an idea of a common graph query language http://neo4j.com/blog/open-cypher-sql-for-graphs/. Because of this, we propose Gremlin By Example.

3 Gremlin By Example

We have shown that both languages (Cypher and Gremlin) can be used, but statements look quite different. Thus, it is likely that, in the future, some standardization will occur with a unified graph query language in mind. Until that happens, we propose a new language for graph databases called Gremlin By Example. Gremlin By Example is a visual query language for graph databases and, hence, more intuitive and easier to use than Cypher or Gremlin itself. Namely, in both mentioned languages, statements have to be entered manually, which can cause unnecessary problems because syntax could be confusing. By using a visual interface, these problems should be resolved as the new language will resemble a well-known language that is called Query By Example (QBE). QBE is a language that can be used to pose queries visually; at the same time, people are familiar with QBE (even less-experienced users can pose moderately complex queries in QBE without knowing SQL). The query below (Figure 3) finds distances between the cities (Query Design in Microsoft Access).

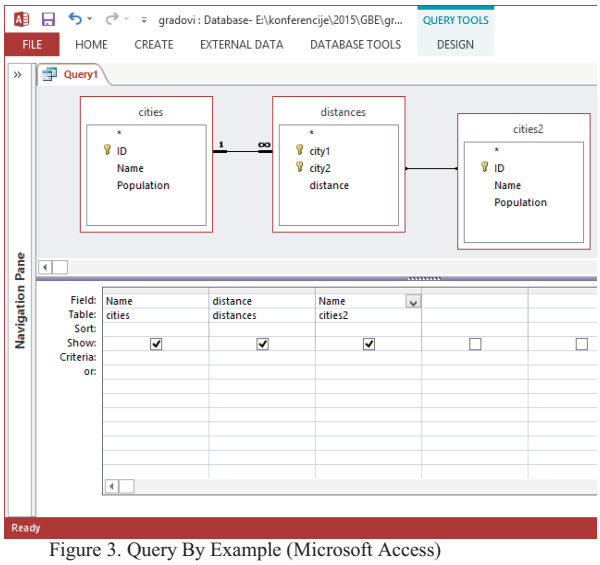


Figure 3. Query By Example (Microsoft Access)

The query in Figure 3 is built visually and it lists distances between the cities. Since QBE is used, even less experienced users can build such queries. Joining the table to itself (in SQL) is something that could cause problems for sure, but in QBE such problems seem to be solvable. Based on the QBE that is supported in Microsoft Access, we propose Gremlin By Example. GBE should have a similar interface, and queries should be posed against the graph database.

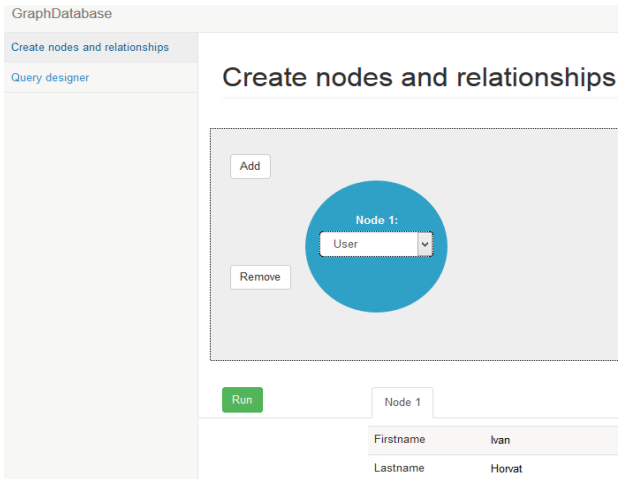


Figure 4. Creating nodes

For implementation purposes, a web application has been built, which uses Django, Python web framework, and Bulbs, Python persistence framework for working with graph databases. The purpose of this application is to provide GUI for executing queries in a Rexster graph database without using the Gremlin console. For now, one can create nodes and relationships; one also can query nodes and relationships.

3.1 Creating nodes

Hereby we create one new node in the database. After selecting the "User" label from the dropdown list of labels in the database, it is necessary to enter values for properties of the User label, which are displayed in a table, as shown in Figure 4.

After running this query (Run button, Figure 4), a new user, Ivan Horvat, is added to the database. There are no ingoing or outgoing edges, and the *create_user(<User object>)* method is called. All vertices from the database are retrieved so that we can check if that User already exists:

```
@staticmethod
def create_user(self, user):
    client = RexsterClient()
    script =
client.scripts.get("get_vertices")
    self.gremlin = client.gremlin(script,
params=None)
    result = self.gremlin.results
    if result:
        for v in result:
            if v:
                if v.get("Label") == "User":
                    if v.get("Firstname") ==
user.Firstname and v.get("Lastname") ==
user.Lastname:
                        return v
            else:
                d =
dict(Label=user.element_type,
Firstname=user.Firstname,
Lastname=user.Lastname)
                u =
client.create_vertex(data=d).results
                result_node = u
                return result_node
        d = dict(Label=user.element_type,
Firstname=user.Firstname,
Lastname=user.Lastname)
        u =
client.create_vertex(data=d).results
        result_node = u
    else:
        d = dict(Label=user.element_type,
Firstname=user.Firstname,
Lastname=user.Lastname)
        u =
client.create_vertex(data=d).results
        result_node = u
        return result_node
    return result_node
```

The query, which is executed in the method above, would be equivalent to the following Gremlin query:

g.addVertex(null, [Firstname : 'Ana', Lastname : 'Anić', Label : 'User'])

3.2 Creating relationships between nodes

To add a new WROTE relationship between the Author and the Book nodes, it is necessary to select the two node labels and Wrote relationship label, as well as to enter values for their properties, as we can see in Figure 5.

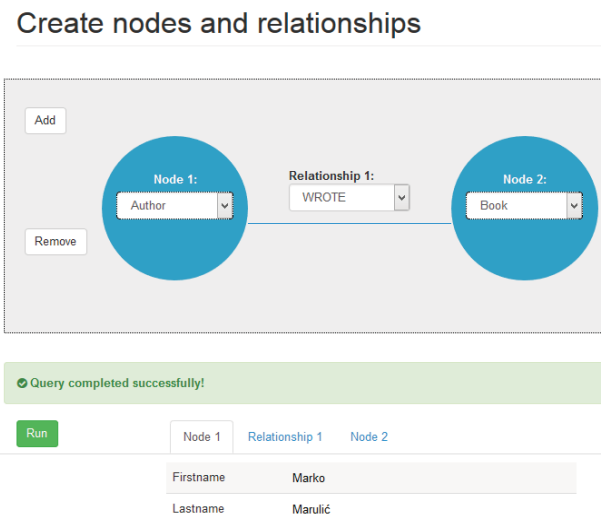


Figure 5. Creating two nodes and a relationship

After pressing the Run button, if Author or Book nodes with entered properties do not exist in the database, they are created along with the WROTE relationship that connects them. In this example, to create a WROTE relationship between Author "Marko Marulić" and Book "Judita" - it is necessary to call methods for adding a new Author and Book. After that, all vertices are retrieved from the database in order to ensure that these two vertices exist, so that a new relationship (edge) could be created between them as well. Thus, for now we see that one can add new nodes as well as relationships. But, we also can query the graph database, as we show below (Figure 6).

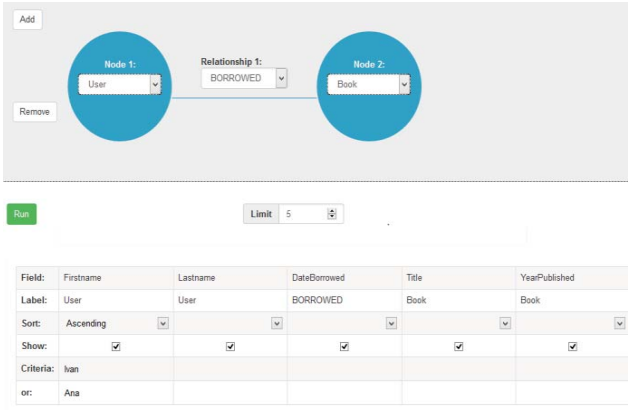


Figure 6. Querying nodes

3.3 Querying nodes

To get a list of five users who borrowed a book sorted ascending by their first name, but whose first name is Ivan or Ana, the query needs to be defined, as shown in Figure 6. In this example, the "copySplit" clause is used to return a list of objects of different types (User, Book, BORROWED), and the "T.in" clause is used to set multiple criteria. The query, which is executed by using Gremlin clauses, looks like this:

```
result = client.gremlin("g.V('Label', '" +
node1 + "').has('" + prop1Node1 + "', T.in,
['" + criteria1Prop1Node1 + "', '" +
criteria2Prop1Node1 + "']).order{it.a." +
prop1Node1 + " <=> it.b." + prop1Node1 +
"}copySplit(_().outE.inV.has('Label', '" +
node2 + "')).fairMerge.path._()[0.." +
str(limit-1) + "]").content
```

This is an equivalent to the following Gremlin query:

g.V('Label', 'User').has('Firstname', T.in, ['Ana', 'Ivan']).order{it.a.Firstname <=> it.b.Firstname} copySplit(_().outE.inV.has('Label','Book')).fairMerge.path._()[0..4]

For now we see that the implemented solution is fully functional. Nodes and relationships can be created and queries can be posed as expected. We demonstrated how to create a single node, two nodes, and a relationship as well as how to query two nodes and a relationship. Some minor interface corrections have been identified, and they will be implemented in future versions.

At this point of time, we are working on a similar solution for another graph query language. Further on, we see possible improvements for the future such as, for example, the implementation of additional advanced features.

4 Conclusion

Graph databases have been gaining the most attraction in recent years. The ability to store data in nodes and relationships that do not need to have the same structure, i.e., the same attributes, are suitable for different problem domains. However, existing graph query languages are not standardized and, because of that, graph databases are not as used as they could be.

In this paper, Gremlin By Example has been proposed. This is a new visual query language for graph databases. It relies on the idea of Query By Example language, which is used for relational databases. The prototype has been implemented, and things do operate as expected. Nodes, as well as relationships, can be implemented visually (Gremlin queries are executed in the background). Moreover, queries can be posed visually, which represents a huge benefit. For now we can say that the prototype operates as expected and that more advanced features should be supported and implemented in the future.

5 References

[1] E. Redmond and J. R. Wilson, Seven Databases In Seven Weeks. Dallas, USA: Pragmatic Programmers, 2012.

[2] F. Holzschuher and R. Peinl, Performance of Graph Query Languages, Proceedings of the Joint EDBT/ICDT, 2013, pp. 195-204.

[3] G. Butler, L. Chen, X. Chen, And L. Xu, Diagrammatic Queries and Graph Databases, Workshop on Managing and Integrating Biochemical

Data, Retrieved March 15, 2015, from http://users.encs.concordia.ca/~gregb/home/PDF/eml-digrammatic.pdf

[4] H. He, and A. K. Singh, Graphs-at-a-time: Query Language and Access Methods for Graph Databases, SIGMOD'08, 2008, pp. 405–418.

[5] Robinson, J. Webber, E. Eifrem, Graph Databases. Sebastopol, USA: O'Reilly Media, 2013.

[6] J. Cheng, Y. Ke, & W. Ng, Efficient Query Processing on Graph Databases, ACM Transactions on Database Systems, 2008, V, pp. 1–44.

[7] K. Rabuzin, Deductive graph database – Datalog in Action, The 2015 International Conference on Computational Science and Computational Intelligence, 2015, pp. 114 – 118.

[8] K. T. Yar and K. M. L. Tun, Predictive Analysis of Personnel Relationship in Graph Database, International Journal of Engineering Research & Technology, vol. 3(9), 2014. Retrieved March 12, 2015, from http://www.ijert.org/view-pdf/11117/predictive-analysis-of-personnel-relationship-in-graph-database

[9] P. T. Wood, Graph Views and Recursive Query Languages, BNCOD, 1990, pp. 124-141.

[10] P. T. Wood, Query Languages for Graph Databases, Third Alberto Mendelzon International Workshop on Foundations of Data Management. Retrieved February 4, 2015, from http://www.dcs.bbk.ac.uk/~ptw/tutorial.pdf

Proposing Good Fit Student Algorithm (GFS-A) to Utilize Big Data and Academic Data

Muhammad Fahim Uddin[1], **Jeongkyu Lee**[2]
[1,2]School of Computer Science and Engineering, University of Bridgeport, Bridgeport, CT, USA

Abstract – *Education is the backbone sector for our economy and every industry. Common goal is to improve it for all times. Our students choose academics based on traditional process and very little to no data mining or analytics are used to suggest right academics that can otherwise dramatically improve student performance. In this paper, we discuss issues of drop outs, poor performances in school, scholarship wastages and costly readmissions. To address these problems, we use personality features from social networking and correlate with academic data and propose Good Fit Student Algorithm (GFS-A) and stochastic probability based mathematical construct to predict the success of a student and promote early decision. Our preliminary results show positive impact of matching features to reduce such issues we mentioned. This work is a part of our ongoing research that we briefly mention. We provide brief related work and conclude with future work.*

Keywords: Good Fit Students, Personality Features, Social Networking data, Big Unstructured data, Academic Data Mining, Stochastic Probability Modeling.

1 Introduction

Every industry in the world depends upon education industry. Universities, Colleges and schools play a vital role in shaping individuals for real world challenges and applications. Education at early stages of life is consistent and applicable in every society and type of individuals. However, as we grow, talent, natural gift and motivation become more applicable and useful in academic life at advance stages, starting with college degree. Mostly, admission choices are governed by family trends, affordability, basic motivation, market trends and natural instinct. However, natural gift and talent are minimally used to select such directions in academics. It is a challenge to predict the likeliness of success for an individual with certain set of talent, in academic world. For this reason, we have seen increased drop outs, poor performances, surprised college and degree changes, costly re-admissions, etc, that further impact roles in job and real world industry. As per U.S Dept of Education[1], only 20 % of young individuals finish their degree on time. This trend is consistent for two and four year college degree. With today's data mining and analysis, we can study and find out many causes for such incidents including, lack of interest, unmatched skill with education, poverty, wrong student in wrong academics, poor teaching methods, inefficient testing methods and poor curriculum etc.

In this paper, we show our research progress and propose usage of related data to predict success of a particular student with particular set of skills and personality talent for a particular line of academics. One of the greatest challenges is the reliability of such prediction because, there are numerous factors that can impact decision and likeliness of success for a student in future and factors itself depends on other factors. In short, such factors are interdependent. Second challenge is the difficulty of collecting academic data due to privacy issues. These factors pose great complexity to build prediction models. College drop outs are increasing that clearly indicates the bad fit students for academics in our system. Numerous studies support prediction of students to reduce drop outs and find the triggers that can help to improve educational system through data mining [2][3][4][2][5][6]. Therefore, we build our model using Stochastic Probability based modeling and test it using empirical data, to advance the research in educational data mining and improving decision making for better academic achievements and performances. We use Microsoft SQL Server[1], Python[2] and Excel data mining[3] and data analysis tools to test and produce our results at preliminary stages.

1.1 Motivation

We are developing *Predicting Educational Relevance For an Efficient Classification of Talent (PERFECT Algorithm Engine – PAE)* that contains many algorithms including the one we present in this paper. With PAE, we are committed to build an efficient model and framework to utilize and correlate personality features with academic data to draw prediction that help educational system. This helps decision making process and avoids costly re-admissions, dropouts, poor performance and lack of motivation during academic years, all by using the data set that we produce anyways.

In section 2, we provide related work for student predictions for retentions and performances. In section 3, we introduce GFS-A and its working principle. In section 4, we show results and discuss the implications of it. In section 5, we conclude with future work.

2 Related Work

Personality study has been a topic for decades for research in psychology and social sciences. However, with recent era of internet and social networking platform such as Facebook and Twitter has revolutionized the researcher's ability to study personalities and envision the applicable usage in real world. A survey done by authors in [7] stresses the personality computing applications with discussing related technologies. The conclude with focus on improving machine learning algorithms to create better models and integration of human sciences and computing to better utilize the personality features for better predictions for various applications. Study

[1] https://msdn.microsoft.com/en-us/library/bb510516.aspx
[2] https://www.python.org/
[3] https://msdn.microsoft.com/en-us/library/dn282385.aspx

done by authors in [8] study measures from social behavior to predict personality. They use Gaussian Processes and Zero R with Five Factor Model (FMM). Other numerous studies [9][10] utilizes behaviors and smart hubs to predict personality, mainly form social networking data. Twitter and Facebook researches, such as in [11] authors addresses personality that is result of people interacting and communicating in Facebook. Machine Prediction in study done on Facebook profiles to rank individuals using Big Five Model[12]. A study done in [13] finds the relationship of Faceboook popular users contacts in real world to their electronic world. Twitter data has widely used to predict personality and traits based on what they tweet and other relevant posts. Various studies [14][15][16][17] shows improved work in data mining and machine learning algorithms, including K-means, Bayesian networks, Support Vector Machines, etc to do personality prediction for many applications. Similar to data on social networking that contributes to Personality Prediction Opportunities and research, EDM and Big Education[18] contributes to prediction and classification of factors in huge data set for success and failures of students and educational system in context. A stream of research work[19] [20] has supported and advanced the techniques and data mining[21][22] algorithm to improve prediction accuracies. Various case studies[23][2] support to predict drop outs and improve student performance. Student behavior[24][25] prediction plays role in such type of data mining and conclude predictive metrics and measures. Student Recommender system[26] has matured through various researches in student data mining[27][6].

3 GFS-A and Working Principle

In table 1, we define some of academic attributes with description and values. We select 15 attributes for our model and algorithm to predict the good fit student for a particular study area based on particular personality and academic background. Though there are many other features, we select these 15 to be more influential for data mining and analytics. These attribute specifically helps to predict success of an individual in academic world and many educators are also looking for such features. We believe that using more features may not improve results because any more extra features may not add to the prediction fitness factor. However, more data sets (sample) can help to improve the prediction accuracy, on the other hand.

Table 1 – Academic Attributes

No.	Description	Feature/VAR	Values
1	GPA	GPA	{0:4}
2	Score	SCORE	{40:100}
3	Previous Grade	P_GRADE	{A,A-,B+,B,B-,C+,C,C-,D+,D,D-,F}
4	Language	LANG	{English, non-Eng}
5	Age	AGE	{18:50}
6	Previous degree Level	P_DEG_LVL	{HS, College, Masters}
7	Previous degree Area	P_DEG_AREA	{Science, Arts}
8	Sex	Sex	{Male, Female}
9	Previous Housing	P_HOUSE	{Family, Boarding}
10	Future Housing	F_HOUSE	{Family, Boarding, Off campus}
11	Family Size	S_FAMILY	{1:N}
12	Financial Status	F_STATUS	{Poor, Rich, Medium}
13	Scholarship	IS_FUNDED_SCH	{Yes, No}
14	Full Time Status	IS_FULL_TIME	{Yes, No}
15	Major Changing	SAME_MAJOR	{Yes, No}

In table 2, we define some of social networking data attributes and their descriptions with values.

Table 2 – Social Networking Sites Attributes

No.	Description	Feature/VAR	Values
1	Marital Status	MARITAL	{YES, NO}
2	Five Factor Model	FFM	{O,C,E,A,N}
3	Facebook	FB	{YES, NO}
4	Twitter	TWTR	{YES,NO}
5	No of FB Friends	FB_FRD	{0:N}
6	No of FB Posts Daily	FB_POSTS	{0:N}
7	No of Tweets daily	TWEETS	{0:N}
8	Type of Posts Involvement	TYPE_FB_POSTS	{Biased, Unbiased}
9	Consistency	IS_CONSIST	{0:1}
10	Friendliness	IS_FRIENDLY	{0:1}
11	Writing Activity	WRITE	{0:1}
12	Liking Activity	LIKES	{0:1}
13	Re-tweeting	RETWEETS	{0:N}
14	Types of Tweets	TYPE_TWEETS	{Biased, Unbiased}
15	Blog writer	IS_BLOG_WRITER	{YES, NO}

3.1 – Algorithm Definition

We show high level code of our algorithm to use academic attributes in correlation with personality features, and see how fitness factor is improved. We use our FF in our GFS model equation to get the reliability of it. More the reliability is, better the prediction of such data, for future great fit student is.

GOOD FIT STUDENT – ALGORITHM

BEGIN

STORE Academic Attributes (AA) //This imports the 15 attributes that we are considering for our test.

RAND_SET ← *RANDOMIZE* values for AA//This function randomly select the values to create a random attributes set.

IDEAL_STD = CREATE (IDEAL_SET); // this function creates an ideal student as a reference

STORE_INIT ← *INITIALIZE:* Pcrt //Performance Criteria, :Dout//dropout trigger,:Rad://Re-admission trigger, :Trs //Transfer School, :Trdm //Transfer degree major, to values between 0 and 1.

STORE_PF ← *EXTRACT_PF* // Here we use our database from social networking site to extract relevant features.

WHILE {(FF IMPROVES} REPEAT
{
TEST_STD = CREATE (RAND_SET); //This function returns a test student.
FF = EVALUATE (TEST_STD, IDEAL_STD, STORE_PF) // This function returns a FF value and if value is less than 5, we have to keep searching for it.
If (FF > 5)
{
FF_Measured = RETURN (FF, TEST_STD = IDEL_STD)
} //End If
} //End While

//Once we have FF_Measured, we can use our features set to see how it improve the reliability with increased number of feature set.
FOR (I = 1 to N) DO // N for our example is up to 15
{
CALC_RELI =+ GFS (N, FF_Measured)
}
END

3.2 - Mathematical Model

Due to the page limit of this paper, the detailed mathematical model is omitted. In this model, we correlate data that we obtain from social networking sites such as Facebook, Twitter, etc and students past academic records so we can predict their success and make them Good Fit Student to finally improve their performances. Let U_d = unstructured data, A_{rec} = Academic records, C_d = Correlated data, and the equation can be formulated as in (1)

$$\mathrm{Cd}(Ud, A\_rec) = \frac{|Cd \cap \mathrm{A\_rec}|}{|Ud \cup A\_rec|} \quad (1)$$

Next, we sample the data, as S(Cd) so we reduce noisy (irrelevant) data from each sample and finally get Noise-free data set as ₣ = Noise Free Sample of unstructured data, N_d = Noisy data, S_d = Signal data given in equation (2)

$$₣ = \sum_{i=0}^{N} (A\_reci) - Prob\left(\frac{Nd}{Sd}\right) \quad (2)$$

Next, we develop equation to finally get matched sample that creates GFS, as in equation 3, and notations are, $\mathrm{Cd}_{\_m}$ = correlated matched samples, ₣′ = *Noise free sample of structured data.*, # = number or samples, Ud = Unstructured data.
Equation 3 is as follows:

$$\mathrm{Cd\_m}(₣, ₣') = \frac{1}{\#} \sum_{i=0}^{\#-1} \left\{(A\_reci) - \left(Prob\left(\frac{Nd}{Sd}\right)\right)\right\} = \sum_{i=0}^{N} \{(Udi) - \infty\} \quad (3)$$

Finally we develop equation for Reliability of GFC prediction. :R_{coeff} = Reliability Coefficient, that is etween -1 and $+1$. , Nt = true negative, Pt = true positive, Nf = false negative, Pf = false positive,

$$R_{coeff} = \frac{\{(Nt \times Pt) - (Nf \times Pf)\}}{\sqrt{\{(Pf + Pt)(Nf + Pt)(Pf + Nt)(Nf + Nt)\}}} \quad (4)$$

4 Results and Discussions

In Figure 1, we show the results for two cases. Red curve shows the results that were produced by correlating personality features from social networking in relevance with academic features to predict the success and measure the reliability. It can be seen that Red outperforms the blue curve that did not use any attributes from social networking data. We use 15 features for both cases, as listed in above tables.

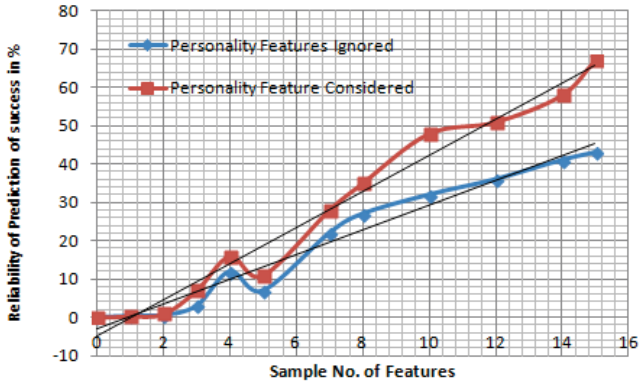


Figure 1 – Prediction with and without Personality features

In Figure 2, we show the results for matching academic attributes with social networking and calculate the probability of unnecessary admissions. It can be observed that when match was at maxi (90 %), the Probability of re-admission was at minimum, zero. But as match gets reduced, the probability gets higher and at match value of 20 %, the probability was higher according to our model.

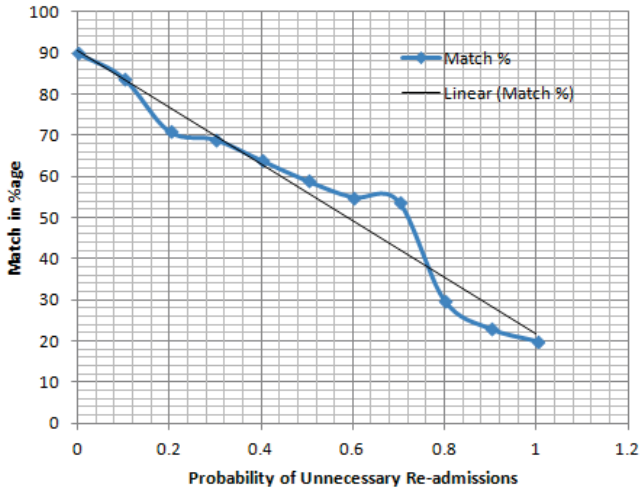


Figure 2 – Attributes Matching effect on Re-admissions

In Figure 3 – We run test on Fitness factor between 0 and 1 and see the corresponding probability of success for particular academic course. Even when Fitness was that course was Zero. There about 30 % Probability of success. For our model, it is low but for general purposes, this score should be much high. Then as fitness function was increased, we notice higher

probability of success and at maximum Fitness function (ideal case), the probability max out to 80 % prediction rate.

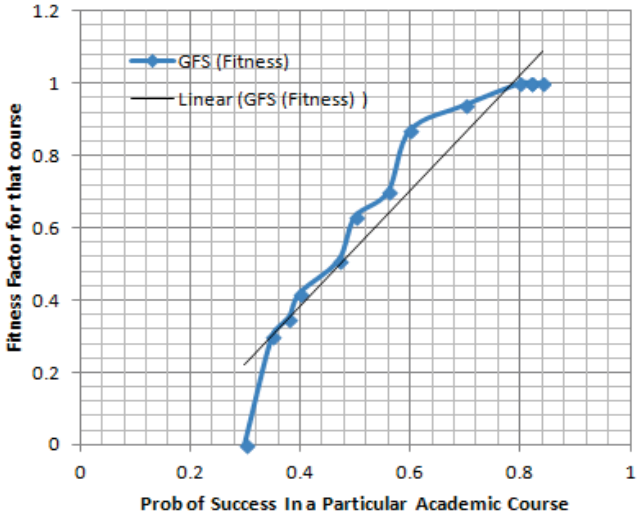


Figure 3 – Performance of Good Fit Student

5 Conclusions and Future Work

In this work, we present an important mile stone towards developing Family of algorithms known as PERFECT Algorithm Engine (PAE). Predicting Educational Relevance For an Efficient Classification of Talent, utilize unstructured data from social networking and correlate it with academic data to predict the success through lens of talents of individuals. In this paper, we show the algorithm and model to implement Good Fit Student and show its fitness function based on available data. We show results for three cases: i) we show prediction reliability of success in % for sample taken using personality features and without using personality features. We show high reliabilities when we use personality features in predicting success, ii) we evaluate the re-admission probabilities based on higher and lower match of features, iii) Finally, we run our model for fitness factor for a particular course and show the probability of success accordingly between 0 and 1 (1 being ideal and higher), as good fit student. We also show how linear are our curves and with more data and features incorporation, linearity can further be improved. For future work, we aim to develop few more algorithms including Retention of scholarship prediction.

6 References

[1] J. Johnson, J. Rochkind, a Ott, and S. DuPont, "With their whole lives ahead of them," *Public Agenda*, p. 52, 2009.

[2] G. W. . Dekker, M. . Pechenizkiy, and J. M. . Vleeshouwers, "Predicting students drop out: A case study," *EDM'09 - Educ. Data Min. 2009 2nd Int. Conf. Educ. Data Min.*, pp. 41–50, 2009.

[3] S. Jones, "Freedom to fail? The Board's role in reducing college dropout rates," *Trusteeship*, p. 5, 2011.

[4] S. Pal, "Mining Educational Data Using Classification to Decrease Dropout Rate of Students," *Int. J. Multidiscip. Sci. Eng.*, vol. 3, pp. 35–39, 2012.

[5] F. Washington, "Graduation and Dropout Statistics," no. September, 2006.

[6] A. A. Al-shargabi and A. N. Nusari, "Discovering vital patterns from UST students data by applying data mining techniques," *2010 2nd Int. Conf. Comput. Autom. Eng. ICCAE 2010*, vol. 2, no. 2, pp. 547–551, 2010.

[7] A. Vinciarelli and G. Mohammadi, "A survey of personality computing," *IEEE Trans. Affect. Comput.*, vol. 5, no. 3, pp. 273–291, 2014.

[8] S. Adali and J. Golbeck, "Predicting Personality with Social Behavior," *2012 IEEE/ACM Int. Conf. Adv. Soc. Networks Anal. Min.*, pp. 302–309, 2012.

[9] G. Mohammadi and A. Vinciarelli, "Automatic personality perception: Prediction of trait attribution based on prosodic features," *IEEE Trans. Affect. Comput.*, vol. 3, no. 3, pp. 273–284, 2012.

[10] O. Celiktutan, E. Sariyanidi, and H. Gunes, "Let me tell you about your personality!†: Real-time personality prediction from nonverbal behavioural cues," *2015 11th IEEE Int. Conf. Work. Autom. Face Gesture Recognition, FG 2015*, p. 6026, 2015.

[11] F. Celli and L. Polonio, "Relationships between personality and interactions in facebook," *Soc. Netw. Recent Trends, Emerg. Issues Futur. Outlook*, pp. 41–53, 2013.

[12] R. Wald, T. Khoshgoftaar, and C. Sumner, "Machine prediction of personality from Facebook profiles," *Proc. 2012 IEEE 13th Int. Conf. Inf. Reuse Integr. IRI 2012*, pp. 109–115, 2012.

[13] D. Stillwell and M. Kosinski, "The personality of popular facebook users," *Proc. ACM 2012 Conf. Comput. Support. Coop. Work*, pp. 955–964, 2012.

[14] L. Qiu, H. Lin, J. Ramsay, and F. Yang, "You are what you tweet: Personality expression and perception on Twitter," *J. Res. Pers.*, vol. 46, no. 6, pp. 710–718, 2012.

[15] E. Kafeza, A. Kanavos, C. Makris, and P. Vikatos, "T-PICE: Twitter personality based influential communities extraction system," *Proc. - 2014 IEEE Int. Congr. Big Data, BigData Congr. 2014*, pp. 212–219, 2014.

[16] J. Golbeck, C. Robles, M. Edmondson, and K. Turner, "Predicting personality from twitter," *Proc. - 2011 IEEE Int. Conf. Privacy, Secur. Risk Trust IEEE Int. Conf. Soc. Comput. PASSAT/SocialCom 2011*, pp. 149–156, 2011.

[17] C. Sumner, A. Byers, R. Boochever, and G. J. Park, "Predicting dark triad personality traits from twitter usage and a linguistic analysis of tweets," *Proc. - 2012 11th Int. Conf. Mach. Learn. Appl. ICMLA 2012*, vol. 2, pp. 386–393, 2012.

[18] L. Cen, D. Ruta, and J. Ng, "Big Education : Opportunities for Big Data Analytics," pp. 502–506, 2015.

[19] R. Shaun, J. De Baker, and P. S. Inventado, "Chapter 4: Educational Data Mining and Learning Analytics," *Springer*, vol. Chapter 4, pp. 61–75, 2014.

[20] N. T. N. Hien and P. Haddawy, "A decision support system for evaluating international student applications," *Proc. - Front. Educ. Conf. FIE*, pp. 1–6, 2007.

[21] R. Jindal and M. D. Borah, "A Survey on Educational Data Mining and Research Trends," *Int. J. Database Manag. Syst.*, vol. 5, no. 3, pp. 53–73, 2013.

[22] R. S. J. D. Baker, "Data mining for education," *Int. Encycl. Educ.*, vol. 7, pp. 112–118, 2010.

[23] A. Merceron and K. Yacef, "Educational data mining: A case study," *Artif. Intell. Educ. Support. Learn. through Intell. Soc. Inf. Technol.*, pp. 467–474, 2005.

[24] J. Sheard, J. Ceddia, J. Hurst, and J. Tuovinen, "Inferring student learning behaviour from website interactions: A usage analysis," *Educ. Inf. Technol.*, vol. 8, no. 2002, pp. 245–266, 2003.

[25] A. El-Halees, "Mining Students Data To Analyze Learning Behavior : a Case Study Educational Systems," *Work*, 2008.

[26] M. Goga, S. Kuyoro, and N. Goga, "A Recommender for Improving the Student Academic Performance," *Procedia - Soc. Behav. Sci.*, vol. 180, no. November 2014, pp. 1481–1488, 2015.

[27] J. K. J. Kalpana and K. Venkatalakshmi, "Intellectual Performance Analysis of Students' by using Data Mining Techniques," vol. 3, no. 3, pp. 1922–1929, 2014.

The 2016 World Congress in Computer Science, Computer Engineering and Applied Computing (WORLDCOMP)

Tutorial Lecture
Resonance of big-data analytics and precision medicine research is producing a profound impact on optimized individual healthcare

William Yang
School of Computer Science, Carnegie Mellon University
Pittsburgh, Pennsylvania 15213, U.S.A. Email:wyang1@andrew.cmu.edu

Rapid advancement of high-throughput next-generation sequencing technologies has generated sheer volumes of multidimensional big-data that will ultimately transform the current healthcare based on average patient to individualized precision diagnosis and accurate treatment. This tutorial presents our newly developed synergistic high-performance computing and big-data analytics approaches to facilitate the advancement of precision medicine research.

In particular, identifying cancer-causing genetic alterations and their disrupted pathways remain highly challenging due to the complex biological interactions and the heterogeneity of the disease even with the power of single-cell genomics. Genetic mutations in disease causing genes can disturb signaling pathways that impact the expressions of sets of genes, each performing certain biological functions. We consider that driver mutations are likely to affect disease-associated functional gene expressions, and the causal relationship between the mutations and the perturbed signals of transcription can be reconstructed from the profiles of differential gene expression pattern and disturbed gene networks. Therefore the first step to improving personalized treatment of tumors is to systematically identify the differentially expressed genes in cancer. We will present a novel online tool called **IDEAS** to **I**dentify **D**ifferential **E**xpression of genes for **A**pplications in genome-wide **S**tudies, and further utilize this tool for the integrative acquisition and analysis of multi-layer genomic big-data. The utilization of IDEAS along with pathway analysis facilitates the construction of high-level gene signaling networks that will eventually lead to find disease-casing genomic alterations and effective drug targets.

Developing synergistic high-performance computing and big-data analytics approaches has been efficiently used in multidimensional genomic big-data integration, hence we will demonstrate our newly developed computational framework to automate data quality assessment, mapping to reference genome, variant identification and annotation, single nucleotide polymorphism and differentially expressed gene identification. We combine multiplatform genomic big-data to enhance detection power of genomic alterations and drug targets. Synergistic development of high- performance computing and big-data analytics

ISBN: 1-60132-427-8, CSREA Press ©

methods utilizing high-dimensional data has provided computational solutions for important precision medicine research that can ultimately lead to the improvement of human health and prolongation of human life.

Biography of Speaker

William Yang is an American software developer, researcher, educator, advocator and writer in synergistic computer science and big-data genomics research. He completed United States National Science Foundation (NSF) Research Scholarship from University of Texas at Austin for the NSF iPlant collaborative project. He holds computer science degree with honors. He has contributed significantly to both computer science and biomedicine and published extensively in both computer and biomedical journals including *Journal of Supercomputing*, *Human Genomics*, *BMC Bioinformatics* and *BMC Genomics*. He also frequently presented his research at IEEE, ACM, international conferences and academic events to promote the synergies of high-performance computing and big-data analytics in precision medicine research. He has accomplishments in interdisciplinary fields, and one of his award-winning highly accessed scientific research articles (https://www.researchgate.net/publication/282747684) was selected by Harvard University in Cambridge, Massachusetts, U.S.A. for open digital access (https://dash.harvard.edu/handle/1/14065393). William has received numerous recognitions including ACM SIGHPC (Special Interest Group on High Performance Computing) Travel Fellowship, Academic All Star Award, Best Hack Award in hackathon competition, Best Programmer Award in non COBOL competition, Highly Accessed Distinction in journal and Best Paper Award in research conference. William is the founder of LearnCTF (Capture The Flag), an online codecademy for cybersecurity research and education. He considers his cybersecurity research as a hobby to hone his brain and problem-solving skills, but his key interest is to develop powerful computer science methods to solve difficult and challenging biomedical problems. William is currently conducting cutting-edge supercomputing and artificial intelligence research at School of Computer Science of Carnegie Mellon University in Pittsburgh, Pennsylvania, U.S.A.

An Approach to Developing and Maintaining the Information Security Systems in Design of Big data Systems

Venkateswarlu Sunkari [1], Amareswarapu V Surendra[2], S. Kranthi Kumar [3],S.Ranga Chaitanya[4]

[1] Asst.Prof, Addis Ababa Institute of Technology, Addis Ababa, Ethiopia.

[2] Asst.Prof., Priyadarshini Institute of Pharmaceutical Education and Research,AP ,India.

[3] Assoc.Prof, Dept. of CSE, JSPM NTC College of Engineering , Nahre,Pune, Maharastra, India.

[4] Asst.Prof, Dept. of EEE, JSPM NTC College of Engineering , Nahre,Pune, Maharastra, India.

Abstract - *Big Data is a collection of data sets. It is so large and complex that it becomes difficult to process using on-hand database management tools or traditional data processing applications. Data is accumulating from almost all aspects of our everyday lives that it becomes huge and multi-structured and has hidden useful information. The challenges with Big Data include capture, curation, storage, search, sharing, transfer, analysis, and visualization. Big Data provides materials for mining hidden patterns to support innovation mostly by data mining. The interaction research with Big Data support methods for innovation is rare at present. Knowledge discovered by data mining is novel and quantitative. However, it still lacks a uniform knowledge management model to support the innovation process effectively. The fact shows that since there emergence, big data techniques are changing very fast. In this paper, developing and maintaining the information security system to illustrate how those big data systems evolve and also describes some key design factors and challenges for future big data systems. Proposed design generic systems that can provide near real-time analytic services for many netease applications, such as spam detection, game log analysis and social community mining. No solutions can address all big data problems, especially when data size keeps increasing, more complex user requirements need to handle, the emergence of new hardware violates the old design and the old system becomes too complicated for maintenance. We face a series of technical challenges that have not been well addressed by both academic community and industry.*

Keywords: Map Reduce, Real time analysis, Information Security.

1. Introduction

Nowadays the Internet represents a big space where great amounts of information are added every day. The IBM Big Data Flood Info graphic shows that 2.7 Zettabytes of data exist in the digital universe today. Also according to this study there are 100 Terabytes updated daily through Face book, and a lot of activity on social networks this leading to an estimate of 35 Zettabytes of data generated annually by 2020. Just to have an idea of the amount of data being generated, one zettabyte (ZB) equals 10^{21} bytes, meaning 10^{12} GB. We can associate the importance of Big Data and Big Data Analysis with the society that we live in. Today we are living in an Informational Society and we are moving towards a Knowledge Based Society. In order to extract better knowledge we need a bigger amount of data. The Society of Information is a society where information plays a major role in the economical, cultural and political stage. In the Knowledge society the competitive advantage is gained through understanding the information and predicting the evolution of facts based on data. The same happens with Big Data. Every organization needs to collect a large set of data in order to support its decision and extract correlations through data analysis as a basis for decisions. The main importance of Big Data consists in the potential to improve efficiency in the context of use a large volume of data, of different type. If Big Data is defined properly and used accordingly, organizations can get a better view on their business therefore leading to efficiency in different areas like sales, improving the manufactured product and so forth.

Big Data can be used effectively in the following areas:

- In information technology in order to improve security and troubleshooting by analyzing the patterns in the existing logs;
- In customer service by using information from call centers in order to get the customer pattern and thus enhance customer satisfaction by customizing services;
- In improving services and products through the use of social media content. By knowing the potential customers preferences the company can modify its product in order to address a larger area of people;
- In the detection of fraud in the online transactions for any industry;
- In risk assessment by analyzing information from the transactions on the financial market.

- In the future propose to analyze the potential of Big Data and the power that can be enabled through Big Data Analysis.

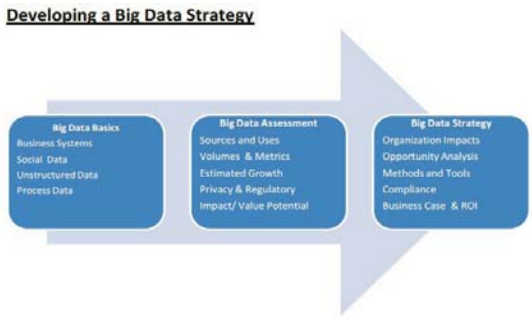


Fig 1. Developing a Big Data Strategy

The understanding of Big Data is mainly very important. In order to determine the best strategy for a company it is essential that the data that you are counting on must be properly analyzed. Also the time span of this analysis is important because some of them need to be performed very frequent in order to determine fast any change in the business environment. Another aspect is represented by the new technologies that are developed every day. Considering the fact that Big Data is new to the organizations nowadays, it is necessary for these organizations to learn how to use the new developed technologies as soon as they are on the market. This is an important aspect that is going to bring competitive advantage to a business. Privacy and Security are also important challenges for Big Data. Because Big Data consists in a large amount of complex data, it is very difficult for a company to sort this data on privacy levels and apply the according security. In addition many of the companies nowadays are doing business cross countries and continents and the differences in privacy laws are considerable and have to be taken into consideration when starting the Big Data initiative. Therefore, the first step to process big data is to collect data from source and pre-process, in order to provide uniform high quality data set to the subsequent process. As a result, due to the inundation of data acquisition, large data become more likely to be "discovered" as a sensitive target, and be more and more attention. Due to the openness of big data, in the process of network transmission, information would be damaged, such as hackers intercepted, interruption, tampering and forgery. Encryption technology has solved the data confidentiality requirements as well as protecting data integrity. But encryption cannot solve all of the safety problems.

2.3 Storage of Data

The formation of network society creates the platform and channel of resource sharing and data exchange for the big data in the field of various industries. In recent years, from the chain reaction of user account information being stolen on the Internet, it can be seen that big data is more likely to attract hackers, and once being attacked, the volume of stolen data is huge. Before big data, data storage is divided into relational database and file server. And in current big data, diversity of data type makes us unprepared. For more than 80% of the unstructured data, NoSQL has the advantages of scalability and availability and provides a preliminary solution for big data storage. But NoSQL still exist the following problems: one is that relative to the strict access control and privacy management of SQL technology; secondly, although NoSQL software gain experience from the traditional data storage, NoSQL still exist all kinds of leak.

2.4 Data Mining

With the development of computer network technology and artificial intelligence, network equipment and data mining application system is more and more widely used, to provide convenient for big data automatic efficient collecting and intelligent dynamic analysis. On the one hand, big data itself exits leak. Big data itself can be a carrier of sustainable attack. Viruses and malicious software code hidden in large data is hard to find. On the other hand, the technique of attack improves. At the same time of the big data technology such as data mining and data analysis gaining value information, the attacker using these big data technology either, just as the two following aspects. A general view about big data Is: data itself can tell everything, the data itself is a fact. In fact, if not carefully screened, the data can deceive people, just as people can sometimes be deceived by their eyes. One of the threats of big data credibility is counterfeit or deliberately manufacturing data, and the wrong data often lead to wrong conclusions. If data application scenarios is clearly, someone could deliberately manufacturing data, and create a "false scent", to induced analysts come to the conclusion that was on their side. Because of false information often hidden in a lot of information, it make impossible to identify authenticity of information, so as to make wrong judgment.

3. Data Security Protection Technique

Key technologies in Security protection fields are in great demands to face the security challenges. In this section, we introduce important relevant fields are **Individual User, Internet Enterprise and Cloud Service Provider.**

4. In-Stream Big Data Processing

The shortcomings and drawbacks of batch-oriented data processing were widely recognized by the Big Data community quite a long time ago. It became clear that real-time query processing and in-stream processing is the immediate need in many practical applications. In recent years, this idea got a lot of traction and a whole bunch of solutions like Twitter's Storm, Yahoo's S4, Cloudera's Impala, Apache Spark, and Apache Tez appeared and joined the army of Big Data and NoSQL systems. One can see that this environment is a typical Big Data installation: there is a set of applications that produce the raw data in multiple data enters, the data is shipped by means of Data Collection subsystem to HDFS located in the central facility, then the raw data is aggregated and analyzed using the standard Hadoop stack (MapReduce, Pig, Hive) and the aggregated results are stored in HDFS and NoSQL, imported to the OLAP database and accessed by custom user applications. The design of the in-stream processing engine itself was driven by the following requirements:

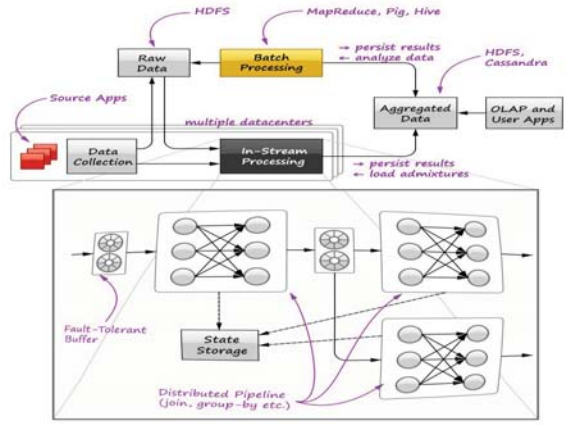


Fig.3. High level over view of Big Data Environment

SQL-like functionality, Modularity and flexibility, Fault-tolerance, Interoperability with Hadoop
High performance and mobility.

- First, we explore relations between in-stream data processing systems, massive batch processing systems, and relational query engines to understand how in-stream processing can leverage a huge number of techniques that were devised for other classes of systems.
- Second, we describe a number of patterns and techniques that are frequently used in building of in-stream processing frameworks and systems. In addition, we survey the current and emerging technologies and provide a few implementation tips.

4.1 Stream Replay

Ability to rewind data stream back in time and replay the data is very important for in-stream processing systems. This is the only way to guarantee correct data processing. Even if data processing pipeline is fault-tolerant, it is very problematic to guarantee that the deployed processing logic is defect-free. One can always face a necessity to fix and redeploy the system and replay the data on a new version of the pipeline. Issue investigation could require ad hoc queries. If something goes wrong, one could need to rerun the system on the problematic data with better logging or with code alternations. Although it is not always the case, the in-stream processing system can be designed in such a way that it re-reads individual messages from the source in case of processing errors and local failures, even if the system in general is fault-tolerant. As a result, the input data typically goes from the data source to the in-stream pipeline via a persistent buffer that allows clients to move their reading
The system is able to revoke a part of the produced results, replay the corresponding input data and produce a new version of the results. The system should work fast enough to rewind the data back in time, replay them, and then catch up with the constantly arriving data stream.

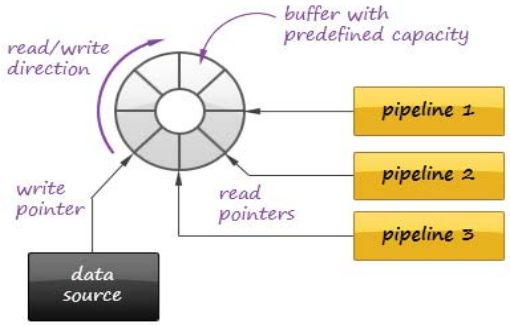


Fig 4. Big Data Stream Replay

4.2 Towards Unified Big Data Processing

It is great that the existing technologies like Hive, Storm, and Impala enable us to crunch Big Data using both batch processing for complex analytics and machine learning, and real-time query processing for online analytics, and in-stream processing for continuous querying. The key observation is that relational query processing, Map Reduce, and in-stream processing could be implemented using exactly the same concepts and techniques like shuffling and pipelining. At the same time In-stream processing could require strict data delivery guarantees and persistence of the intermediate state. Among the emerging technologies, the following two are especially notable

in the context of this discussion:

1. Apache Tez , a part of the Stinger Initiative . Apache Tez is designed to succeed the MapReduce framework introducing a set of fine-grained query processing primitives. The goal is to enable frameworks like Apache Pig and Apache Hive to decompose their queries and scripts into efficient query processing pipelines instead of sequences of MapReduce jobs that are generally slow due to materialization of intermediate results.

2. Apache Spark. This project is probably the most advanced and promising technology for unified Big Data processing that already includes a batch processing framework, SQL query engine, and a stream processing framework.

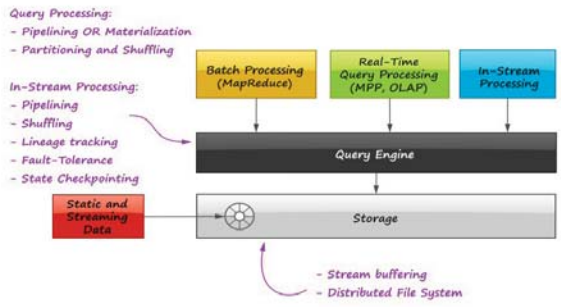


Fig 5. Unified Big Data processing

5. A generic real-time analytic system

One major problem with streaming system is lack of flexibility. The architecture of *probability node*, *classification node* and *clustering node* is tailored for the Naive Bayes model. If we want to adopt a more comprehensive model to improve the accuracy, we need to completely change the design. Moreover, it is impossible to extend the system to support other analytic jobs such as game log analysis and social community detection. In summary, the architecture is limited to the spam detection system using Naive Bayes model. Another issue is the scalability and load imbalance problem. Clustering nodes are the bottleneck of the system which may slow down the whole system, The size of data will continuously increase (e.g., the number of email per second is expected to increase to more than ten thousands). The analytic system should support the deployment on multiple data centers. Update The analytic system is expected to handle both batch processing and real-time processing. Model complexity the system needs to provide a flexible programming interface, so that different applications and models can be efficiently developed on top of the system. The interface is required to be compatible with Hadoop to reduce the efforts of migration, as most existing analytic jobs of Netease are processed in Hadoop.

5.1 In-memory processing

Users are no longer satisfied with the performance of offline analysis. For example, Netease game designers want to have a real- time interactive tool to analyze the game log. They may start from a very general query, such as retrieving all gamers who have not logged in for three days. After examining the list, they can submit a more specific query, e.g., grouping gamers who have not logged in for three days by their characters' levels. A distributed memory array which follows the same design philosophy of RDD to support scalable and fault tolerant data storage in memory. The memory array provides an interface for epic's units to access its data. In this way, we extend epic to an in-memory processing engine. Simply storing the data in memory cannot fully exploit the benefit of new architecture. We need some specific optimizations. For example, new index structures should be designed to maximize the hit ratio of L2 cache, instead of reducing I/O costs. Radix sort works better than quick sort and merge sort. data are frequently swapped in or out from memory.

5.2 Processing updates

As a real-time system, we cannot apply the batch update scheme which is widely adopted in the Map Reduce systems. Up- dates and queries must be processed concurrently. To guarantee the consistence of analytic results, two typical strategies are employed, locking and multi-versions. In Google's Spanner, the conventional two-phase locking is used. Although locking affects the system's throughput, Google argues that the programmers and users should be responsible for that. If they want a better performance, they should avoid using the locks. In our system, we run analytic queries and updates together. After a few updates, current dataset is newer than V. To handle such cases, we replicate a tuple t be- fore applying any update to it. Let T be the tuples that have not been updated during the query processing. We use T^{r} to denote the set of replicated tuples. V *is* then materialized as $T \cup T^{r}$. In other words, by using replication, we avoid locking tables for the analytic query. The storage overhead, how- ever, is low, because we will discard

the old versions of data after those queries have been processed.

5.3 Deployment on multiple data centers

Designing a system that be deployed on multiple data centers is much more challenging. There are a few issues affecting the performance or even correctness of the data processing. First, the network latency between data centers is much higher and unstable. When partitioning data, we should always store the data that are frequently accessed together in the same data center. An efficient analytic algorithm is partition-aware which intention ally avoids showing data between nodes in different data centers. Second, the clocks of different data centers are not synchronized. This makes timestamp based approach does not work anymore. Two consecutive messages may be handled in different or- ders at each data center. To address the problem, we select some nodes as time servers from each center and synchronize their clocks. Other nodes will ask the time servers to get the correct clock. Finally, it is difficult to keep the CAP property. If one data center fails or disconnects from the network, all its data are not accessible. If we keep a replica for data in that center, we will have the consistent issue. It is too costly to keep the replica (normally in other data centers) consistent with the master copy. Besides, if we use the replica to process updates and failures, when the data center recovers, A more sophisticated approach is being developed to support more complex multi-center architecture.

6. Conclusions

In this paper, we use the information security system in big data system evolves when users' requirements keep changing. How to design a generic system that can provide near real-time analytic services for many related applications, such as spam detection, game log analysis and social community mining. Based on our experiences, no solution can address all big data problems, especially when 1) data size keeps increasing; 2) more complex user requirements need to be handled; 3) the emergence of new hard- ware violates the old design; and 4) the old system becomes too complicated for maintenance. New applications will emerge when we combine big data techniques with other conventional industries while in the combination process those applications will pose new requirements for big data systems, pushing us to search and propose new solutions.

7. References

[1] A Navint Partners White Paper, "Why is BIG Data Important?" May 2012, http://www.navint.com/images/Big.Data.pdf.

[2] http://www.informatioweek.com/software/business-intelligence/sas-gets.hip-to-hadoop-forbigdata/ 240009035? / pgno.2.

[3] Chen Mingqi, Jiang He. USA Information Network Security New Strategy Analysis in Big Data [J]. Information Network Security. 2012(8):32—35

[4] Narayanan A, Shmatikov V. How to break anonymity of the Netflix prize dataset. ArXiv Computer Science e-prints, 2006, arXiv:cs/0610105: 1-10

[5] Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik-Lun Lee. Exploiting geographical influence for collaborative point-of-interest recommendation.//Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval (SIGIR'11), Beijing, China, 2011: 325-334

[6] Goel S., Hofman J.M., Lahaie S., Pennock D.M. and Watts D.J... Predicting consumer behavior with Web search. National Academy of Sciences, 2010, 7 (41): 17486– 17490

[7] http://www.wired.com/science/discoveries/magazine/ 16-07/pb_theory

[8] Study Finds Web Sites Prying Less: Shift May Reflect Consumer Concerns [EB/OL]. http://www CNN.com, 2002-03-18

[9] G. Caruana, Maozhen Li, Man Qi, A mapreduce based parallel svm for large scale spam filtering, in: Fuzzy Systems and Knowledge Discovery, FSKD, 2011, pp. 2659–2662.

[10] Alfons Kemper, Thomas Neumann, Hyper: a hybrid oltp&olap main memory database system based on virtual memory snapshots, in: ICDE, 2011,pp. 195–206.