Intelligent Mental Health Diagnosis Architecture using Data Mining and Machine Learning

Ghassan Azar
Lawrence Technological University
Southfield, MI, USA
gazar@ltu.edu

Naser El-Bathy
North Carolina A&T State University
Greensboro, NC, USA
nielbath@ncat.edu

Su Yu
Shanghai University of Engineering and Science
Shanghai,China
suyu_sh@hotmail.com

Rajasree Himabindu Neela
Lawrence Technological University
Southfield, MI, USA
rneela@ltu.edu

Kholoud Alfarwati
North Carolina A&T State University
kwalfarw@aggies.ncat.edu

***Abstract*—Inappropriate diagnosis of mental health illnesses leads to wrong treatment and causes irreversible deterioration in the client's mental health status including hospitalization and/or pre-mature death. About 12 million patients are misdiagnosed annually in US. In this paper, a novel study introduces an Intelligent Mental Health Diagnosis Architecture using Data Mining and Machine Learning that aids in preliminary diagnosis of the psychological disorder patient. This is accomplished based on matching description of a patient's mental health status with the mental illnesses illustrated in DSM-IV-TR, Fourth Edition Text Revision. The study constructs the semi-automated system based on an integration of the technology of genetic algorithm, classification data mining and machine learning. The goal is not to fully automate the classification process of mentally ill individuals, but to ensure that a classifier is aware of all possible mental health illnesses could match patient's symptoms. The classifier/psychological analyst will be able to make an informed, intelligent and appropriate assessment that will lead to an accurate prognosis. The analyst will be the ultimate selector of the diagnosis and treatment plan.**

I. INTRODUCTION

In societies, diagnosing mental disorders in individuals is often poor due to the lack of understanding of their behavior, symptoms and inadequate knowledge. The therapists and/or psychiatrists possess to weed through the many mental health illnesses identified in the Diagnostic Statistical Manual IV, Fourth Edition Text Revision (DSM-IV-TR) [1]. Many a times, improper diagnosis leads to wrong treatment which may cause irreversible deterioration in the client's mental health status including hospitalization and/or pre-mature death [2].

The objectives of this study include reducing biased and incomplete assessment while introducing a consistency in diagnosing mentally ill individuals and minimizing further deterioration in mentally ill individuals due to miss-diagnosis. Objectives also include enhancing prognosis by utilizing multiple expert's knowledge base and their cumulative years of training and practical experience in solving new cases.

The final objective is improving diagnosis results through intelligently using and implementing approved previous successfully-solved cases based on the historical data as well. This method can be used just rather than depending on textbook rules & only individual assessor's experience [3].

The rest of this paper is structured as follow: Section two identifies the study model and procedures. Section three presents the full text indexing, Section 4 introduces intelligent genetic algorithm, Section five present experimental results, and finally the conclusion is given.

ISBN: 1-60132-430-8, CSREA Press ©

II. Study Model and Procedures

Figure 2 illustrates the study model and procedures. The book 'Diagnosis and statistical manual of mental disorders' in text format has a softcopy. Its data (that is keywords) related to each disorder have been loaded into the database. Criteria for each disorder have been identified and formed into a question that needs to be asked to the user and loaded into the database.

An intelligent genetic algorithm has been developed to extract keywords from the user's symptoms. Finally, a graphical user interface has been developed such that the users can enter the description of their symptoms [4].

Based on description, match keyword to the keywords presented in the database to extract classification. The classification order must be from highest probability of disorder to lowest. Classification is shown in percentage basis [5]. The interface allows the user to ask questions relevant to highest classification. Based on users input, new classification is generated and reordered from highest to lowest. Based on the symptoms, 10 relevant disorders are classified along with the percentage of matching. Some disorders have specific criteria which are formed into questions.

Based on the questions answered, the percentage value will be changed. If all the questions are answered yes, then the percentage value increases. If all the questions are answered no, then the percentage value decreases. If few questions are answered yes and no, then according to number of yes and no, the percentage is calculated. The final result shows highest percentage disorder from the classification. If doctor does not accept the given the classified disorder, then doctor will be given a choice to determine his/her own perspective disorder for the patient.

III. Full Text Indexing

FULLTEXT search function matches a natural language query against a text collection (which is simply the set of columns covered by a FULLTEXT index). For every row in a table it returns relevance - a similarity measure between the text in that row (in the columns that are part of the collection) and the query.

The rows returned are automatically sorted and retrieved. Relevance is a non-negative floating-point number. Zero relevance means no similarity. Relevance is computed based on the number of words in the row, the number of unique words in that row, the total number of words in the collection, and the number of documents (rows) that contain a particular word. Any "word" that is present in the stopword list is ignored. Every correct word in the collection and in the query is weighted, according to its significance in the query or collection [5]. The weights of the words are then combined to compute the relevance of the row. For calculating relevance value, frequency Ft of each term will be analyzed, normal frequency of each term is calculated using formula 1:

$$T_f = F_t / \text{Max}(F_t) \quad (1)$$

Inverse Document Frequency measure is calculated as follow:

$$Id_f = \log (N / d_f) \quad (2)$$

$N \rightarrow$ Total Number of records
$d_f \rightarrow$ Number of records where the term appears

Weight of each term is calculated as follows:

$$W = T_f X Id_f \quad (3)$$

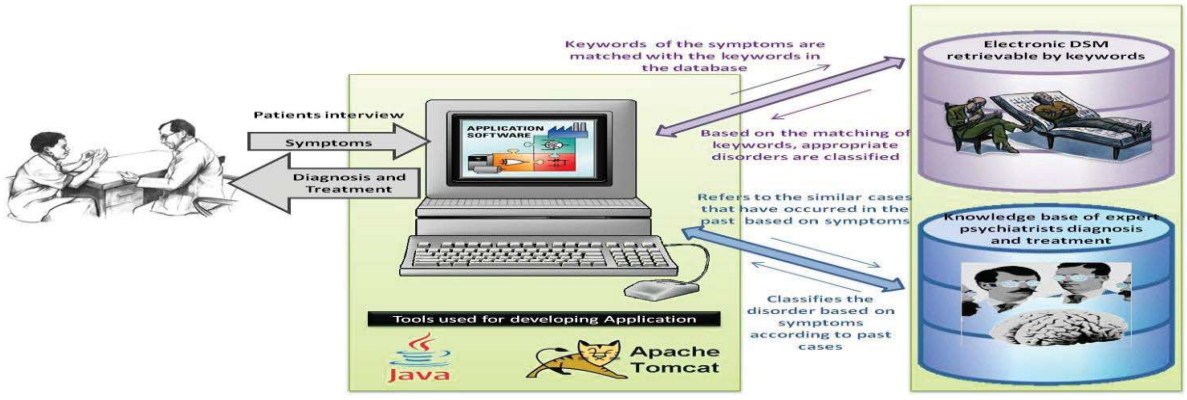

Fig. 1. The study model and procedures

Similarity of each record(relevance value) is calculated as follows:

$$Sim(Q,D) = \sum_{i=1}^{t} \frac{W_{qi} * W_{di}}{\sqrt{\sum (W_{qi})^2 * \sum (W_{di})}} \quad (4)$$

The above formula which represents the similarity is the relevance value which will be in floating numbers. By using this, percentage is calculated as figure 2 shows.

The study system has been developed via two phases. in phase 1, a database of MySQL is created and content of the DSM IV text book is loaded into the database tables. These tables include Symptoms table. This table includes symptoms for each disorder. User query is compared against the contents of this table to retrieve the output. Questions table includes questions belonging to each disorder that has to be asked to the user. Answers table stores user symptoms, the user input to the questions and questions are inserted for further reference.

In phase 1, the graphical user interface is developed as well.

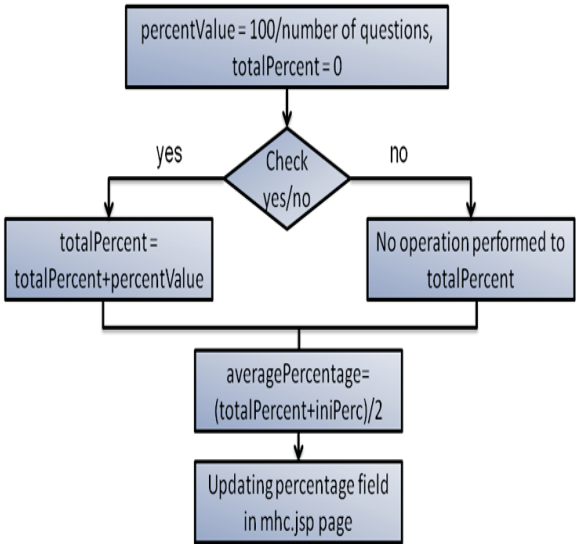

Fig. 2. Calculating Percentage

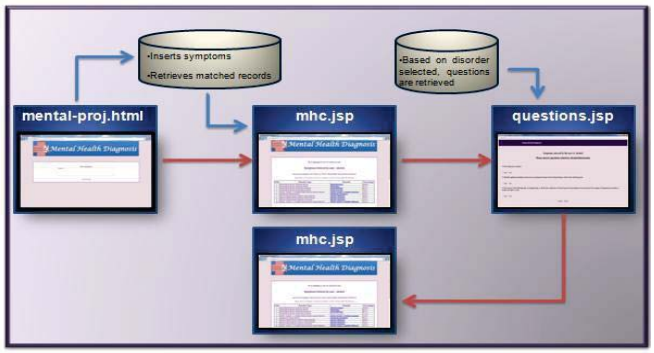

Fig. 3. Phase 1

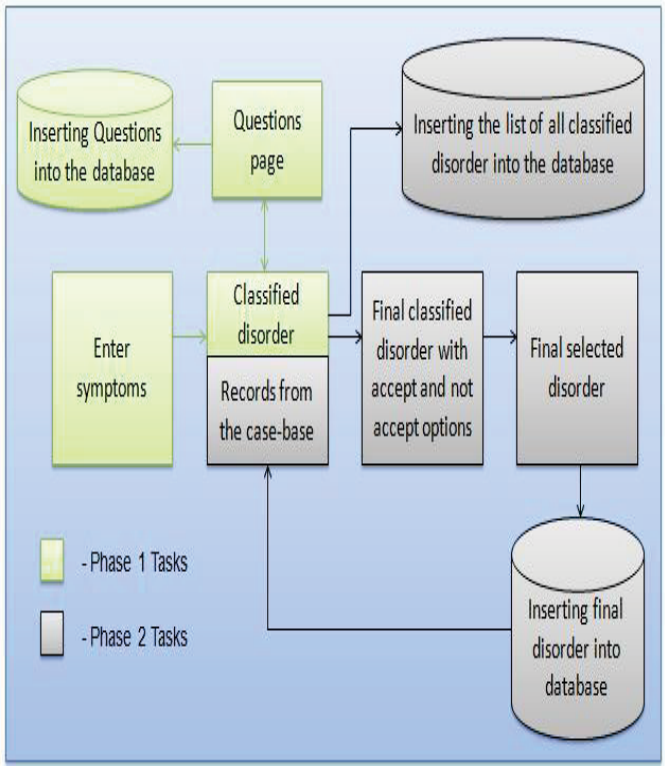

Fig. 4. System Functionality

In phase 2, the classification is done, user answers the questions, and final disorder selected by the application is displayed. (This considers the disorder having highest percentage)

IV. Intelligent Extended Genetic Algorithm

The intelligent extended genetic algorithm that has been implemented to extract keywords from the user's symptoms uses Business Process Execution Language (BPEL) to be an optimal solution for information retrieval. It improves the efficiency and performance for retrieving a proper information results that satisfy user's needs. The implemented algorithm uses several mutation operators simultaneously to produce next generation. This series of random mutation process depend on chromosome best fitness in the population and rely on high relevancy as well. The mutation operation will guarantee the success of algorithm for extracting keywords from the user's symptoms since it expands the search [6]. So the highly effective mutation operators the greater effects on the genetic process.

By implementing the algorithm, data can evolve into information in a way that produces robust flexibility [6]. While other algorithms have made numerous advancements, there is still a lack of the ability to evolve. Nowadays, various applications of genetic algorithms are in the early stages of being used to actively and efficiently extract data based on relevancy. Such applications aim to produce information that has adapted over time based on user requests.

The structure of genetic algorithm is extended to hold multiple populations in the population space. The Algorithm is designed using artificial intelligence methodologies, not geometric approaches, to the information retrieval problem [6].

Our method uses an extended genetic algorithm to find an ideal solution instead of a more mathematical methods such as the k-means algorithm. This key difference allows for more adaptive behavior within our algorithm. Also, web services can induce very large amounts of data.

As it is important to manipulate data accurately and efficiently, Business Process Execution Language approach has been proposed. It implements dynamic service capabilities with genetic algorithms to apply reasoning and flexible service workflows [7], [8], and [9].

This paper builds a utility-based intelligent agent that implements a faster genetic algorithm with greater efficiency than the original algorithm. The genetic algorithm supports a flexible service composition mechanism while having the ability to improve efficiency over time, all while reusing previously tested efficiency. While semi-system can be made bigger, modern paradigm breakthroughs are evolving to make semi-system smarter. The orchestrations of genetic algorithm provide flexible service workflows that can quickly adapt to changes. The orchestration of web services is supported by Business Process Execution Language [9]. BPEL composes, or orchestrates, the services into business flows.

Web service is a technology that enables programs to communicate through Hypertext Transfer Protocol (HTTP) on the Internet [8]. Service standards are effective platforms for publishing services. These standards are Web Services Description Language (WSDL), Extensible Markup Language (XML), and Simple Object Access Protocol (SOAP). WSDL provides a model for describing services. XML adds an intelligent level to distribute information on the Internet. SOAP exchanges structured information in the implementation of the service [7].

Chromosomes are encoded to represent a genetic algorithm and to be parsed into tree structures, which prevents syntax crossovers and allows for mutation stages. Once proper genetic algorithms are put in place, the desired service item from the web part can be requested. Upon this initial request, the first generation of information retrieval is randomly generated, which can lead to a slight decrease of efficiency. What makes up for this initial sacrifice in performance is that as the workflow processes information, the algorithm creates a new generation of logic and the results are assessed based on goodness of fit to results. As new logic workflows are developed, they can be selected and mutated to produce better results. As this process continues, eventually the service matchmaking with user requirements can be provided in such a way to enable increased efficiencies over time. Upon delivery of the user request, the generation cycle is terminated.

The fitness of an individual is computed based on the "distances" between the keywords appearing within the user's symptoms. The keywords are compared by their weights, meaning the ratio of their appearances to the total sum of words in the user's symptoms. These weights are then treated as if they were coordinated for the user's symptoms point on an n-dimensional grid, where n is the number of different keywords appearing within the set of the user's symptoms being retrieved by algorithm.

In the algorithm, an individual with a lower fitness value actually represents a solution of greater quality than one with a greater fitness value. This is because the quality of the retrieval solution is the closeness of the keywords being extracted. Only the most individual fit is passed on to the next generation. The fitness for a chromosome is found through repetition of the math used for finding the similarity of the keywords in the user's symptoms. For each chromosome in the generation, the fitness is computed by finding the average of the similarities for each keyword. By using this method, the fitness is also the average distance between any two keywords in any one user's symptoms in the solution.

Mutation is a way that changes the population to produce the best solution [10]. The process involves a series of mutations that will evolve over time taking only the mutations with a high relevancy, and mutating those further. The algorithm used one type of mutation. This type is known as a one-point mutation. Either a single keyword's position is moved through the chromosome, switching its place in the user's symptoms with another symptom, or the point at which a symptom is organized is moved.

To further increase the genetic diversity present in each generation of the algorithm, the algorithm includes a step where a new individual is added to the population. This individual is randomly generated with each generation iterated, to create additional diversity, even without the crossover step's inclusion in the algorithm.

This algorithm removed crossover step although it is a key part of numerous genetic algorithms. The reason is that crossover decreases the efficiency of our algorithm. It would build new chromosomes out of sections from two different chromosomes, creating new generations with greater diversity. The lesser number of generations required comes with a cost in the form of a drop in efficiency.

Currently, extended genetic algorithm stores each chromosome as a sequence of characters representing the user's symptoms. The order of the characters in our chromosomes is of great importance and no repeats are allowed. Traditional genetic algorithms use a series of bits which represent in turn a series of operations and values. Using crossovers in the source code of our genetic algorithm negatively affects the efficiency of the algorithm more than it would lower the amount of generations required. With just our current generation loop utilizing only varying degrees of mutations, we are likely creating the same chromosomes which would result from crossovers. The proposed genetic algorithm is simply a way to go through a vast number of possible solutions with greater speed and efficiency than other strategies. With or without crossovers, our genetic algorithm should arrive at the same value.

1. Create initial random population P of N individuals
2. i ← 0
3. If i is equal to the number of desired generations,
   return the best individual of the most recent generatic
4. P_{i+1} ← empty set
5. B ← the most fit individual of the previous generatic
6. Add B to P_{i+1}
7. Insert into P_{i+1} mutate(B)
8. Repeat 7 until P_{i+1} has N individuals
9. Evaluate the fitness for all individuals from P
10. i ← i + 1
11. Goto 3

Fig. 5. The Algorithm

V. Experimental Results

The algorithm is tested on set of sample data. The data is based on 50 generations/iterations of the IECGA or K-means respectively, using the same random sample set of 15 symptoms with 600 words each.

The results are listed in Table 1 were collected over 15 test runs of both clustering methods on the same data set. The table shows the statistics collected from our genetic algorithm and K-Means algorithm to demonstrate their relative performance capabilities. The values given are the fitness of the final solution generated by each run, which means that the lower fitness are from better solutions, while higher fitness values are worse solutions. As each method uses a random starting point, there is room for variation in solutions.

From this data, we can observe that on average, implementing IECGA algorithm excels k-means algorithm. The test runs did not find as good a solution with k-means as the best solution from the genetic algorithm, and even the worst solution from the genetic algorithm is of better fitness than the average solution from k-means.

While the data collected does not represent all possible input cases, and cannot claim to represent all of them, it shows a trend of the genetic algorithm exceeding the performance shown the clustering process we had used previously.

As a result, The classifier/psychological analyst is able to make an informed an accurate prognosis. The analyst will be the ultimate selector of the diagnosis and treatment plan. Figure 6 shows that doctor now has option of selection accepting or not accepting the final classified disorder. If not accepted, doctor can select another disorder.

Figure 7 presents the description of all disorders entered into the database.

TABLE 1
IECGA AND K-MEANS PERFORMANCE

	IECGA	***K-Means***
Maximum	1.66384	1.86476
Average	1.56938	1.67881
Minimum	1.35574	1.40269

Fig. 6. Semi-system Options

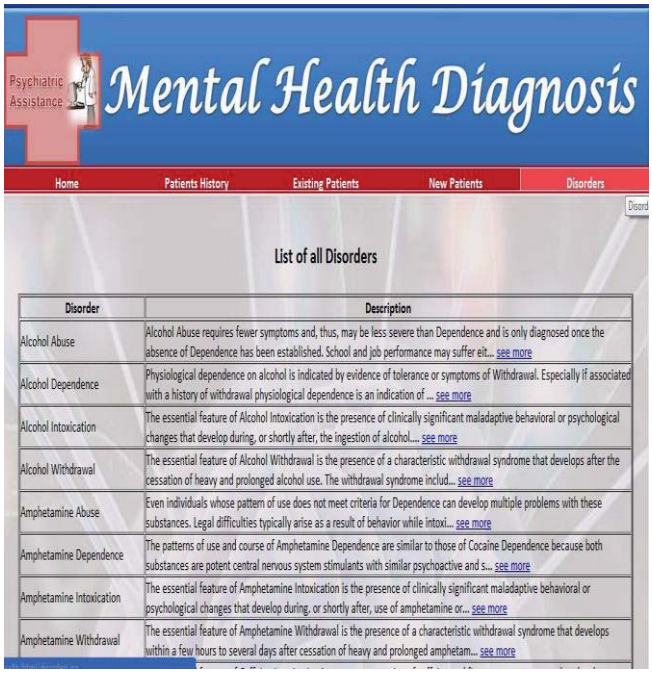

Fig. 7. Disorder Description

VI. CONCLUSION

The main objective of this paper is to ensure that classifier/psychological analyst is capable of making an informed, intelligent, appropriate assessment, and an accurate prognosis. This study proved the applicability of potential Extended Clustering Genetic Algorithm to solve the efficiency and limitation problems in data extraction. The algorithm solution has markedly increased the success of information retrieval and relevancy between keywords-matching and relevant user's symptoms as shown.

The intelligent data extraction is a challenging research problem that arises in many applications. This Genetic Algorithm can be used for many different applications requiring data mining, information retrieval, computational biology, text categorization and image annotation. It enhances an organization's ability to collect information faster at lower cost and to make accurate decisions. Implementing this algorithm provides acceptable benefits in terms of agility and integrity. The orchestrations of genetic algorithms by implementing Business Process Execution Language allow flexible service workflows to be immediately adjusted to modifications and make systems smarter.

Our future work will concentrate on the implementation of the algorithm to large data sets, generalization of the proposed approach to general graph structures, and investigation of the possibility of integrating multiple sources of data for improving the data extraction quality.

REFERENCES

[1] American Psychiatric Association. [DSM-IV-TR], (2000). Diagnostic and statistical manual of mental disorders (Revised 4th ed.). Washington, DC: Author

[2] S. Fernando, "Mental Health in a Multi-Ethnic Society," A Multi-Disciplinary Handbook, Routledge

[3] J. Shedler, "The illusion of Mental Health," American Psychologist, Vol 48 No 11 1117-1131, 1993.

[4] N. El-Bathy, G. Azar, M. El-Bathy, and G. Stein, "Intelligent Extended Clustering Genetic Algorithm," IEEE, Electro/Information Technology, Mankato, MN, United States, pp. 1-5, 2011.

[5] R. Akerkar and P. Lingras, Building an Intelligent Web – Theory and Practice. Sudbury, Massachusetts: Jones and Bartlett Publishers, 2008.

[6] B. Coppin, Artificial intelligence illuminated. Sudbury, Massachusetts: John and Bartlett Publishers, 2004.

[7] N. El-Bathy, P. Chang, G. Azar, and R. Abrahiem , "An Intelligent Search of Lifecycle Architecture for Modern Publishing and Newspaper Industries Using SOA," IEEE, Electro/Information Technology, Normal, IL, United States, pp. 1-7, 2010.

[8] N. El-Bathy and G. Azar, "Intelligent Information Retrieval and Web Mining Architecture Applying Service-Oriented Architecture," KG. Saarbrücken, Germany: LAP LAMBERT Academic Publishing AG & Co. 2010.

[9] N. El-Bathy, G. Azar, M. El-Bathy, and G. Stein, "Intelligent Information Retrieval Lifecycle Architecture Based Clustering Genetic Algorithm using SOA for Modern Medical Industries," IEEE, Electro/Information Technology, Mankato, MN, United States, pp. 1-7, 2011.

[10] H. P. Pfeifer, "An exhaustive Analysis of Recombination and Mutation variances for Genetic Algorithms," Protocol Labs, Munich, 2010.