Knowledge Discovery in Medical Systems Using Differential Diagnosis, LAMSTAR, and k-NN

Rahul Isola, *Student Member, IEEE*, Rebeck Carvalho, *Student Member, IEEE*, and Amiya Kumar Tripathy, *Member, IEEE*

***Abstract*—Medical data are an ever-growing source of information generated from the hospitals in the form of patient records. When mined properly, the information hidden in these records is a huge resource bank for medical research. As of now, these data are mostly used only for clinical work. These data often contain hidden patterns and relationships, which can lead to better diagnosis, better medicines, better treatment, and overall, a platform to better understand the mechanisms governing almost all aspects of the medical domain. Unfortunately, discovery of these hidden patterns and relationships often goes unexploited. However, there is on-going research in medical diagnosis which can predict the diseases of the heart, lungs, and various tumours based on the past data collected from the patients. They are mostly limited to domain-specific systems that predict diseases restricted to their area of operation like heart, brain, and various other domains. These are not applicable to the whole medical dataset. The system proposed in this paper uses this vast storage of information so that diagnosis based on these historical data can be made. It focuses on computing the probability of occurrence of a particular ailment from the medical data by mining it using a unique algorithm which increases accuracy of such diagnosis by combining the key points of neural networks, Large Memory Storage, and Retrieval, k-NN, and differential diagnosis all integrated into one single algorithm. The system uses a service-oriented architecture wherein the system components of diagnosis, information portal, and other miscellaneous services are provided. This algorithm can be used in solving a few common problems that are encountered in automated diagnosis these days, which include diagnosis of multiple diseases showing similar symptoms, diagnosis of a person suffering from multiple diseases, receiving faster and more accurate second opinion, and faster identification of trends present in the medical records.**

***Index Terms*—Data mining, differential diagnosis, knowledge discovery, medical decision support system, neural networks, service-oriented architecture (SOA).**

I. Introduction

SINCE the advent of advanced computing, doctors have always made use of technology to help them in various possible ways, from surgical imagery to X-ray photography. Unfortunately, technology has always stayed behind when it came to diagnosis, a process that still requires a doctor's knowledge and experience to process the sheer number of variables involved, ranging from medical history to climatic conditions, blood pressure, environment, and various other factors. The number of variables counts up to the total variables that are required to understand the complete working of nature itself, which no model has successfully analyzed yet. To overcome this problem, medical decision support systems [1]–[3] are becoming more and more essential, which will assist the doctors in taking correct decisions.

Medical decision is a highly specialized and challenging job due to various factors, especially in case of diseases that show similar symptoms, or in case of rare diseases. The factors leading to misdiagnosis may vary from inexperience of the doctors, habitual and repetitive diagnosis by experienced doctors, stress, fatigue, and other occupational conditions, and also due to factors including, but not limited to misinterpretation, ambiguous symptoms, and incomplete information. Conventional algorithms completely overlook various variables involved such as prevailing conditions, the build-ups resulting in the symptoms, medical history, family history, and other factors relating to the patient, due to sheer magnitude of available unknown variables.

Experienced doctors generally classify such diseases based on the differential diagnosis [4] method. This involves doctors narrowing down the diseases to the root disease out of the list of diseases that show similar symptoms. This is done using their knowledge and experience, and it is later confirmed by performing various tests. In case of rare diseases or diseases with similar symptoms, due to the number of tests involved, it might not be always feasible. Especially in developing countries, the problem of lack of trained and experienced doctors leads to intensification of this problem [5].

This process of differential diagnosis has been emulated in the system proposed in this paper, thus making this rather tough task a lot easier. This method is further modified and enhanced to reduce the huge number of underlying variables to just one by finding the root disease, or the most probable disease, using smart pattern matching involving k-NN classification technique [6] and the next probable diseases by performing differential diagnosis, using the Hopfield neural networks theory [7] and Large Memory Storage and Retrieval (LAMSTAR) Networks [8]. Using all these, and by utilizing a database having a comprehensive list of medical history at the disposal of this system, the probability of occurrence of a disease may be calculated, regardless of the various unknown variables. The algorithm will output

Manuscript received August 15, 2011; revised November 29, 2011, January 20, 2012, June 19, 2012, July 14, 2012, and July 30, 2012; accepted August 15, 2012. Date of publication August 23, 2012; date of current version November 16, 2012. This work was supported by the Don Bosco Institute of Technology, Mumbai, India.

R. Isola and R. Carvalho are with the Don Bosco Institute of Technology, Mumbai 400070, India (e-mail: rahulisola@gmail.com; maaask3@gmail.com).

A. K. Tripathy is with the Centre of Studies in Resources Engineering, Indian Institute of Technology Bombay, Mumbai 400076, India, on deputation from the Don Bosco Institute of Technology, Mumbai 400070, India (e-mail: tripathy.a@gmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITB.2012.2215044

the disease from the symptoms entered and also gives the next highly probable disease, and thus, the most effective course of action to be performed can be determined.

This system, built on service-oriented architecture (SOA) [9], has been implemented on a web server, so that it can be accessed by anyone with an Internet connection. After successful implementation of the system, not only will this be accessible by most doctors, this can also be used by doctors in rural and remote areas, with a computer and Internet, or even a mobile phone. The doctors or any medical personnel have to enter the symptoms of the disease. The system, making use of various techniques mentioned, will in turn display the root disease along with the set of most probable diseases which have similar symptoms. This system will give the doctors the list of diseases that the patient has maximum probability of suffering from. This, in turn, will help the doctors to recommend specific tests corresponding to the diseases in the list, thus reducing the number of nonconsequential tests and resulting in saving time and money for both the doctor and the patient.

II. Related Research

A simple search on any search engine gives every possible list of symptoms with even the required medication for any disease, consisting of both accurate and inaccurate remedies. However, this nonverified overinformation may also prove fatal. The authenticity of such results cannot be validated. Even if it can, the efficiency of the remedy cannot be calculated, as the diagnosis itself might be wrong. Some sites give the feature to diagnose the disease based purely on the input of symptoms [10]. However, these sites too cannot be entirely relied upon information as the symptom/diagnosis may not be relevant for the patient, for example, a symptom arising from a pre-existing condition, or family history and other possible affecters. The patient ultimately has to consult doctors for authentic and safe prescription.

Considering a small subset of medical datasets, algorithms have been formed [11]; and accurate results have been achieved by some of them [12]. Applied to a much larger generalized dataset, for every medical field, obtaining accurate results has yet been very difficult.

Iliad [13] is an expert system program that uses Bayesian reasoning to calculate the posterior probabilities of various diagnoses under consideration, given the findings present in a case. Similarly, DXplain [14] is a decision support system which acts on a set of clinical findings (signs, symptoms, laboratory data) to produce a ranked list of diagnoses which might explain (or be associated with) the clinical manifestations. DXplain provides justification for why each of these diseases might be considered, suggests what further clinical information would be useful to collect for each disease, and lists what clinical manifestations, if any, would be unusual or atypical for each of the specific diseases [15]. DXplain takes the advantage of a large database of the crude probabilities of different clinical manifestations associated with different diseases, but unfortunately, it is still limited to the research laboratory or medical training setting.

III. Data Acquisition

The system described here is based on a single fact that the only way to surely conclude that a person is suffering from a particular disease is by conducting clinical tests. Generally, a doctor diagnoses a patient on his first attempt based on his experience. If his first attempt fails, then the remaining assumptions are followed by conducting clinical tests. This system helps in this aspect by emulating the doctor at the first attempt but in the second attempt provides a differential list of diseases based on data mining and neural networks. This differential list is sorted probability wise, so the doctor can know which disease to choose next. Thus, unnecessary tests are reduced and faster diagnosis is achieved.

Existing medical systems, including hospital management systems and decision making systems, focus on collecting and mining the entire medical data. The entire patient records are loaded and all factors are considered. The medical data cannot be easily analyzed, because for generating a probabilistic rating, not only symptoms but also factors like test results, current epidemics, medical history, external climate conditions, and various other factors are required, which may or may not be present in the report. Existing systems have failed to utilize and understand the importance of misdiagnosis, a very important attribute which interconnects and addresses all these issues. Mining the misdiagnosis attribute is the key because the first diagnosis by the doctors would have already covered all the underlying variables like patient's medical history, climatic conditions, neighborhood, and various other factors, allowing the doctor to just concentrate on either missed variables like hidden symptoms, prevailing conditions, complications, etc., or diseases that are similar to the one already diagnosed. If the misdiagnosis factor was mined and stored as an attribute, it will solve the problems discussed above, because doctors misdiagnose only in the presence of ambiguities, and in similar cases, there is a high chance of other doctors also performing the same misdiagnosis.

The accuracy of the system in this paper increases as the number of entries in the database increases. This results in the system requiring an enormous repository of data for providing accurate results. Providentially, such huge databases already exist in form of Electronic Health Records or EHRs, also known as Electronic Medical Records or EMRs [15].

Retrieving relevant medical data for this system is a major task, with various issues rising about data confidentiality, data security, and ambiguity. A complete Hospital Management System with this system as an integral part was developed and implemented by us to tackle these issues. This way, not only the issue of data acquisition was addressed, but as the system will be accessible only by the hospital representatives, patient confidentiality was also maintained. The system will communicate with the various EHRs and use the common EHR repositories that have already been implemented. The system will not only retrieve data from these repositories, but also update it with every new/updated patient record, resulting in an updated and complete database. More users will result in a more complete and accurate database. There are huge number of EHRs in use and the data storage and representation format of all are not

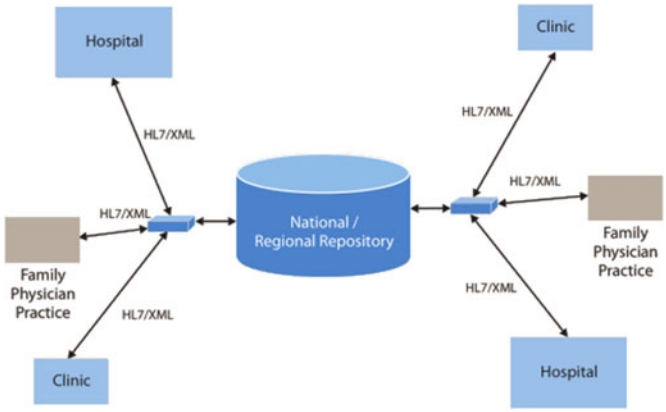


Fig. 1. Tata consultancy services EHR repository model [16].

homogenous. Using SOA, catering to these heterogeneous sources becomes easier, as the data requests and responses are in XML format, so only the middle business layer needs to be changed according to the data layer. The application layer remains constant.

Fig. 1 shows the repository model used by Tata Consultancy Services [16]. The EHR system cited in the paper uses SOA. This makes implementation of our system even easier, as it can easily be deployed as new services utilizing the existing EHR architecture.

The data are retrieved using the XML queries. The integration hubs perform validation and verification of messages sent by different applications and performs further rule-based processing. The messages are sent over secure channels of communication through a high-speed dedicated network or alternatively through secure encryption and decryption mechanisms to the repository.

The event-based summaries stored in the repository can be queried and retrieved by different users in different scenarios and by different inputs where the patient's personal data and history do not come in picture. The retrieval is done through messaging, which can be done either through synchronous or asynchronous messages depending on the urgency, complexity, and importance of the data that are being retrieved.

The EHR's SOA's Enterprise Service Bus parses the XML queries and retrieves only the fields that are requested by the relevant service of the system. These fields only include the database views showing the grouped values of initial and final diagnosis and the symptoms leading to that diagnosis. Other fields, including those that include patient information, are outside the authorization scope of the service. Hence, the XML queries that retrieve the data will result in a neutral data subset as the input. This result dataset does not include any data that can be traced back to the patient in any way, hence keeping the confidentiality intact. These results are stored in the database as: symptoms, disease, disease diagnosed, and actual disease attributes. The symptoms will result in change in weightage of the symptoms for the corresponding disease, as explained in Section IV-D.

Consider a report being processed for a patient suffering from *arthritis*. The EHR will be queried to find the entries with initial diagnosis as *arthritis*. The EHR will report back the entries

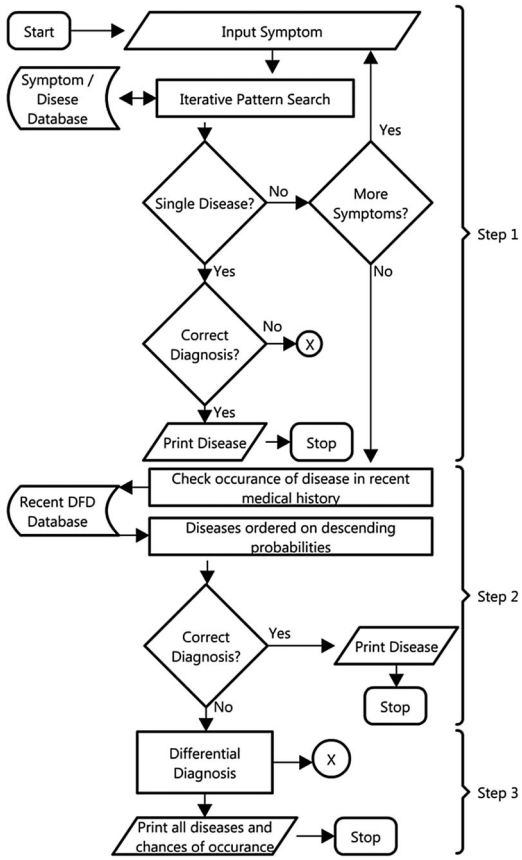


Fig. 2. Workflow pattern.

corresponding to the symptoms, initial, and final diagnosis. If the initial diagnosis was correct, this diagnosis will be the final diagnosis itself. Otherwise, the values for initial and final diagnosis will be different. So if *arthritis* was indeed the disease and was diagnosed correctly in the first attempt itself, then both initial and final diagnosis fields in the dataset, also referred as diagnosed and actual disease will be entered with *arthritis*. The dataset will also include the symptoms shown in the report, and that will change the weight of that symptom for the corresponding disease. For example, if the knee pain was the symptom here, then for *arthritis*, knee pain will gain weight, giving a better match in the next iteration.

IV. Algorithm

Various algorithms utilized in this method are explained below. Fig. 2 shows a three-tier workflow pattern of the system. The system uses a three-tier workflow method for triple precision diagnosis. Each case is explained with the help of a practical case study given below.

A. Symptom Matching Using Iterative Search

Symptom matching using iterative search utilizes data that is stored as given in Table I. The first step of the algorithm involves selecting the symptoms shown by the patient. As an output, the algorithm gives the list of all possible diseases ranked according to the number of symptoms matched in the database. The list

TABLE I
SAMPLE DATABASE FOR STEP 1: ITERATIVE PATTERN SEARCH

Diseases	Symptoms and Weight	Class Weight
Diabetes (D_1)	Headache (W_1), Increase in blood sugar (W_2), Insulin low (W_3)	Endocrine (C_1)
Pericarditis(D_2)	Chest pain (W_4), Fever (W_5), weakness (W_6), Malaria (W_7), Shortness of breath (W_8), Syncope (W_9)	Cardiovascular (C_2)
Viral Fever (D_3)	Headache (W_{10}), Cold (W_{11}), Fever (W_{12}), Running Nose (W_{13}), Weakness (W_{14})	Parasitic (C_3)
Sinusitis (D_4)	Pain in the sinuses(W_{15}), Headache (W_{16}), Heavy eyebrows (W_{17}), Blurry vision (W_{18}), Fever (W_{19})	Respiratory (C_4)

TABLE II
SAMPLE SORTED DATABASE FOR DIFFERENTIAL DIAGNOSIS

Patient	Disease Diagnosed	Actual Disease
A	Diabetes	Diabetes
B	Diabetes	Diabetes
C	Diabetes	Diabetes
D	Diabetes	Diabetes
E	Diabetes	Diabetes
F	Diabetes	Hypertension
G	Diabetes	Hypertension
H	Diabetes	Hypertension
I	Diabetes	Arthritis
J	Diabetes	Arthritis

is generated after input of every symptom. After the first iteration, for the second iteration, the next list of symptoms will be shortlisted according to the disease list that was obtained in the previous iteration. The new symptom list will contain symptoms of only those diseases that were obtained in the previous list.

These related symptoms will then be shown to the user who shortlists another symptom from the new list. The new disease list will be listed, ranked according to the number of symptoms matched. The ranking is generated according to the percentage match of the total number of symptoms entered. This procedure goes on iteratively, with diseases being placed in the ranks according to its probabilities.

After a few (usually within first three or four) initial iterations, top diseases in the list gain highly in ranking, allowing one to identify the ailment. That is, ranking of the disease varies at a much greater precision as more and more symptoms are given. On the database given in Table II, on entering *headache* as a symptom, *diabetes*, *viral fever*, and *sinusitis* will have 100% match, giving it equal ranking, whereas *pericarditis* will be excluded. On entering the next symptom, say *fever*, *viral fever* will have both the symptoms matching, giving it 100% match, while the other three will have a match of only one out of the given two symptoms, thus 50% match, resulting in a drop in their ranking. This has to search a record database of more than 20000 diseases and even more symptoms, which is very time consuming, so k-NN classification was applied to classify diseases into subgroups. If a group of symptoms match higher preference is given to that subgroup and searching in that new smaller subgroup thus reduces database access.

In pattern recognition, the k-nearest neighbor algorithm (k-NN) is a method for classifying objects based on closest training examples in the feature space. k-NN is a type of instance-based learning, or lazy learning where the function is only approximated locally and all computation is deferred until classification. This feature has been identified as the most suitable for the present system. The other feature that has been useful to the presented system is as follows: the k-nearest neighbor is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its k nearest neighbors (k is a positive integer, typically small). If $k = 1$, then the object is simply assigned to the class of its nearest neighbor.

k-NN has been modified to give faster processing as follows. For the data given in Table I instead of using the Euclidean distance between the neighbors, their weights will be considered. The logic of assigning weights is explained later in Section IV-D. Here, weights have been given to the individual symptoms corresponding to each disease, each individual disease, and the subclass disease category. Common symptoms like *headache* are assigned different weights in case of different diseases. This is done because if take same weight for repeated symptoms in all diseases, it will lead to improper diagnosis because each symptom has a special role to the disease and the subclass it belongs to. For example, *headache* may gain more weightage in *Viral Fever* as compared to the other diseases. k-NN will first sum the individual weights of each symptom, compare it first to the nearest subclass and then to all the diseases in that subclass resulting in faster accuracy. The choice of assigning k is given to the doctor, depending on how many comparison is desired, default is set to $k = 10$. For the data in Table I, if *headache*, *fever*, and *pain in the sinuses* are entered, then the weights W_{15}, W_{16}, and W_{19} will be considered. Next, all the weights will be added and compared to all subclasses C_1, C_2, C_3, and C_4 of which C_4 is most likely the answer depending on its weight. Finally, all the diseases in class C_4 are considered, and if *sinusitis* (D_4) weight is closer to the sum of all the input symptoms weights, then it is the possible diagnosis.

If a single disease in the given subset gains maximum weight above all other diseases, it is the interpreted by the system as the possible diagnosis. Instead, if multiple diseases are found with nearest weights or same weights, then the system proceeds to mining medical records described in Section IV-B.

B. Mining Medical Records (Based on Recent Trends)

In the previous case, if multiple diseases are found with similar ranking, it becomes difficult to pinpoint to one of them, when no more symptom is unique to any single disease affecting its ranking. This is especially the case in case of some epidemic in the area, or some rare disease, or disease arising due to localized

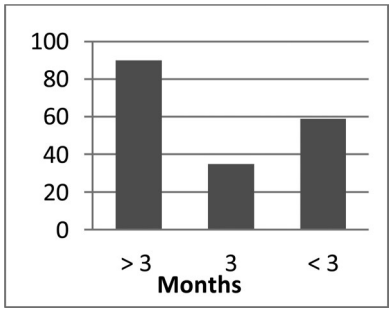


Fig. 3. Error% for varying time period.

conditions and various other factors. For example, *swine flu* epidemic initially showed the same symptoms as that of viral fever, resulting in raising the ranking of both *viral fever* and *swine flu*. In such cases, it becomes very difficult to point at one disease using the iterative pattern search method. In such cases, recent medical historical data stored in the database of the proposed system is used, with a time period of three months to rank the diseases on basis of the probability of their occurrence in the review period. Time frame of three months provided accurate diagnosis with less error (see Fig. 3). On discussion with doctors, it was concluded that variations in this duration for most diseases are geographically distributed. The duration depends mainly on the seasonal cycles in the location where the system is to be implemented. In the test location for this system, four seasons spread over a duration of 12 months resulted in three-month period.

If the interpreted diagnosis is still vague, then the system proceeds to differential diagnosis.

C. Differential Diagnosis

To perform differential diagnosis, the system uses a Hopfield network. They serve as content-addressable memory systems with binary threshold units. They are guaranteed to converge to a local minimum, but convergence to one of the stored patterns is not guaranteed. The value is determined by whether or not the units' input exceeds their threshold [17], [18]. Hopfield nets can either have units that take on values of 0, 1, or –1.

The connections in a Hopfield net typically have the following restrictions:

$$w_{ii} = 0 \quad \forall_i \text{ (no unit has a connection with itself)} \tag{1}$$

$$w_{ij} = w_{ji} \quad \forall_{i,j} \text{ (connections are symmetric).} \tag{2}$$

For the data given in Table II, if Hopfield rule (2) is applied to find $\Delta w =$ relative frequencies, the individual weights are as follows:

$$W(C_1) = 5/10, \quad W(C_2) = 3/10, \quad W(C_3) = 2/10$$

where

$C_1 =$ Diabetes, $C_2 =$ Hypertension, and $C_3 =$ Arthritis.

So the Differential Diagnosis would be

$$Diabetes \Rightarrow Hypertension \Rightarrow Arthritis.$$

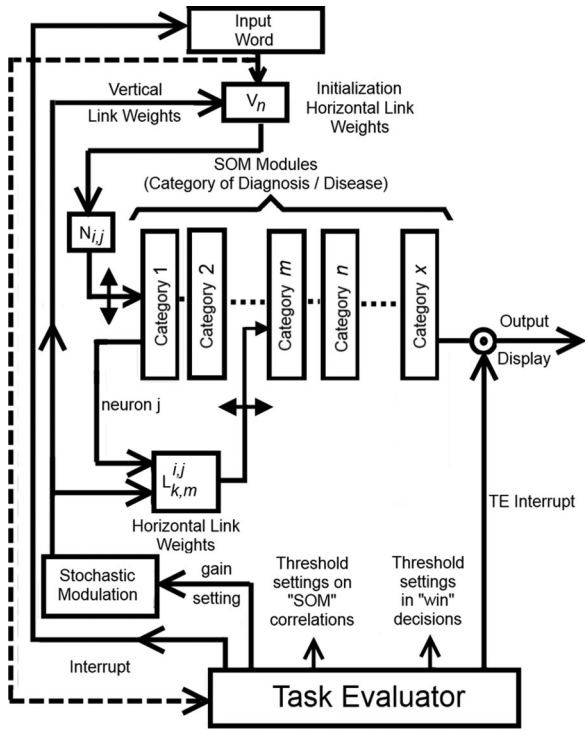


Fig. 4. General block diagram—LAMSTAR network.

D. Weight Assigning Using LAMSTAR Network

Using algorithm described in Section IV-C, the correct disease shortlisted by the doctor is obtained who confirms it by taking the necessary tests. The final report is then mined to obtain the correct symptoms. The correct symptom thus obtained is then compared with the original symptoms entered. This information is now fed to the LAMSTAR [19] network for assigning weights. Information in the LAMSTAR network is encoded via correlation links (see Fig. 4) between individual neurons in different self-organizing maps (SOM) [20] modules. The LAMSTAR network does not create neurons for an entire input word. Instead, only individual subwords are stored in SOM modules (weights), and correlations between subwords are stored in terms of creating/adjusting links that connect neurons in different SOM modules. When the new input word is presented to the system during the training phase, the LAMSTAR network inspects all weight vectors in SOM module that corresponds to an input subword to be store.

If any stored pattern matches the input subword within a preset tolerance, the system updates weights according to the following procedure:

$$W_{i,m}(t+1) = W_{i,m}(t) + \alpha_i (X_i(t) - W_{i,m}(t)), \quad \text{for } m: \ \epsilon_{\min} < \epsilon_i \text{ (const)} \tag{3}$$

where

$W_{i,m}(t+1)$ modified weights in module i for neuron m;

α_i learning coefficient for module I;

$\epsilon_{\min}$ minimum error of all weights vectors W_i in module.

If no match was found, the system creates new pattern in the SOM module. It stores input subword x_i as a new pattern $\boldsymbol{W}_{\text{in}}$, where index n is the first unused neuron in the ith SOM module. The above storage procedure is repeated for every input subword x_i to be stored. Link weight values are determined by evaluating distance minimization to determine winning neurons, where each win (successful fit) is counted by a count-up element associated with each neuron and its respective input-side links, the values of the links are modified according to

$$L_{i,j}^{k,m}(t+1) = L_{i,j}^{k,m}(t) - \propto (L_{i,j}^{k,m}(t) - L_{\max}), \quad \text{for } L_{i,j}^{k,m}(t) > T_H \tag{4}$$

$$L_{i,j}^{k,m}(t+1) = 0, \quad \text{for } L_{i,j}^{k,m}(t) < T_H \tag{5}$$

where

T_H	forgetting threshold (applies to L weights);
$L_{i,j}^{k,m}$	links between neuron i in the kth module and neuron j in the mth module;
$\propto$	learning coefficient;
$L_{\max}$	maximal links value.

The count up (as the subsequent weight delay due to forgetting) set mean weight values to be stochastically modulated. Link weights L_i, j decay over time, as a result of the learning formula of (4) and (5). Hence, if not chosen successfully, the appropriate L_i, j will drop toward zero. This helps to avoid the need to consider a very large number of links, thus contributing to the network efficiency.

Initially, all weights are assigned the value zero. For each correct matched symptom the weight increases $+1$ and for each unmatched symptom the weights are kept constant. The reason for keeping weights constant for unmatched symptoms is that if the unmatched symptoms were assigned negative weights, then certain symptoms would be repeatedly degraded and when they would actually surface in some diagnosis, because of too much negative weight, the change in the ratio of weight of the symptom for that particular disease to the total weight of all symptoms for the disease will be more significant. This will lead small changes in trend to result in bigger change in the ratio as compared to not subtracting negative weight. To keep the weight ratio to be as stable and precise as possible, the fluctuations should not be much. Hence, only positive weights are considered.

The weights assigned here have been found out from the rigorous three tier process. It is very important in pattern matching in the sense that they incorporate the misdiagnosis factor and the doctor may get 100% result in the first step itself**.** During trials, LAMSTAR was found to be faster in assigning of weights as compared to back propagation. LAMSTAR also gave more priority to recent weight due to its link forgetting capability.

V. Implementation

The rapid rise of technology and its adoption into the healthcare field has caused healthcare organizations to collect an accumulation of non-interoperable systems that not only need to work together within the organization, but are also accessed from outside. The burden of integration usually falls on the users of the system, who are forced often to access many different systems to complete one task. The use of SOA, however, can improve the delivery of important information and make the sharing of data across the community of healthcare professionals more practical in cost, security, and risk of deployment [21].

For implementation on SOA, four web services are used in the system, namely, pattern matching, recent trends, differential diagnosis, and recent differential diagnosis. Two main databases used are: Disease/Symptoms database and the Records Database. Disease/Symptom database is a centralized database and it contains the list of the existing known diseases and their corresponding symptoms along with their weights. The Records Database maintains records of the patients from all the hospitals in the network. These databases are replicated across various servers to achieve fault tolerance with the standard concurrency protocols to achieve atomic transactions.

The doctor first queries the system by entering a set of symptoms. Pattern matching service is then activated which fires the query and presents the result to the doctor. If the doctor is not satisfied with the results, he/she can invoke the Recent Trend Service. Recent Trend Service makes use of the Disease/Symptoms database and the Records database and the result obtained from pattern matching service to get results. Finally, to avoid ambiguity in decisions, the doctor make use of differential diagnosis which invokes differential diagnosis and recent diagnosis feature which make use of the Disease/Symptoms database and the Records database and result obtained from Recent Trend service to get results. Since the medical data are huge, using simple client–server architecture would not suffice in the effective discovery of aforementioned services and would increase the response time of the system.

After considering various models, it was concluded that SOA was best suited to implement this system. This was due to the fact that various existing EHRs, which are essential in their as the data providers to this system, are already using this very successful and efficient architecture. The system, implemented as various services in the existing SOA, will result in easy implementation, integration, and scalability with existing EHRs. SOA also handles the issues related to data security and patient confidentiality.

VI. Results

A prototype system using web tools and technologies like PHP, MySQL, and AJAX was developed. The system was implemented on simple client–server architecture with limited database to test the above theory. The data have been obtained from limited scope clinical trials. The system was run for a sample dataset of malaria. Specific test cases were run, and the following results were obtained. On initial comparisons with existing sources providing differential diagnosis, this system gave similar results, with the added information of the list of other probable diseases. This list of diseases obtained by performing differential diagnosis was compared with the existing differential diagnosis relationships already established, and the list was also found to be accurate.

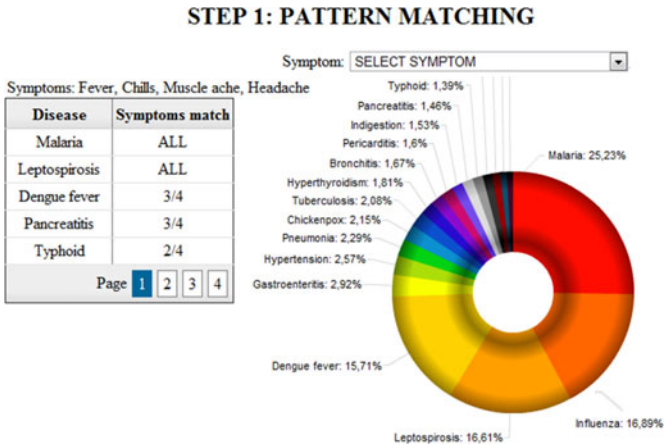


Fig. 5. Pattern matching.

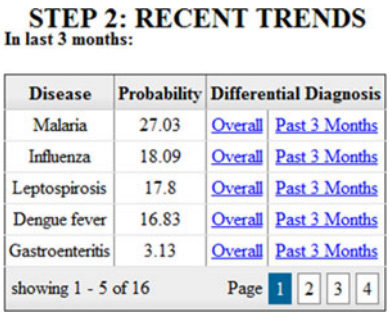


Fig. 6. Symptom-based recent trends.

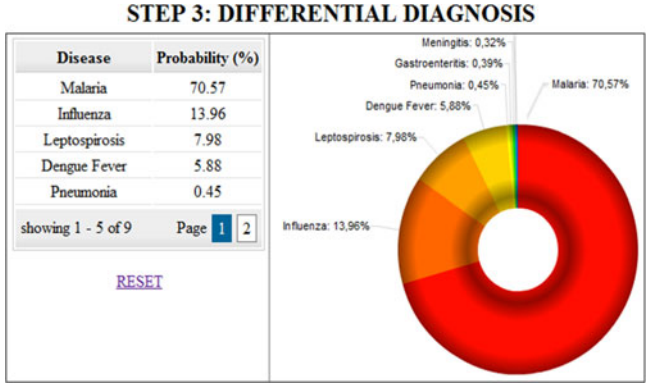


Fig. 7. Differential diagnosis.

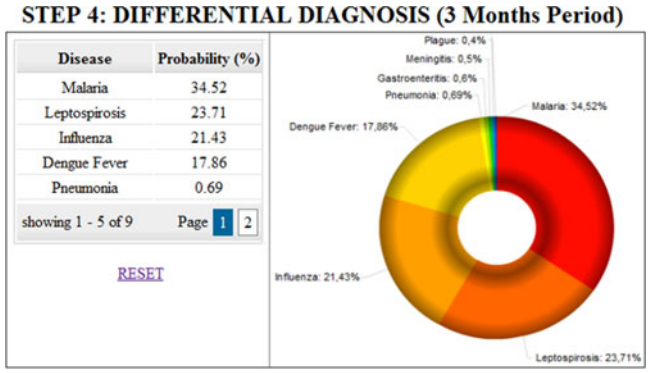


Fig. 8. Differential diagnosis in previous three-month period.

The table on the left of Fig. 5 depicts the number of symptoms matching and the graph shows the probability considering the ranking adjusted according to the weights. The first step resulted in accurate prediction of diseases based on the symptoms entered.

The system also shows the list of diseases found by matching symptoms and its probabilities of occurring calculated based on its occurrence in the past three months. The second step resulted in very accurate prediction of diseases based on the recent trends. It accurately caught *malaria* and *leptospirosis* for the given symptoms, during the monsoon season, where mosquitoes were a menace in the test locality and there was an outbreak of *leptospirosis*. This diagnosis matched very accurately with the patient's actual ailment. Fig. 6 shows the same result, but with more priority given to diseases occurring within a three-month filter.

The table on the left in Fig. 7 shows the list of diseases and their probabilities of occurring calculated on the basis of differential diagnosis technique. The graph shows a graphical representation of the same.

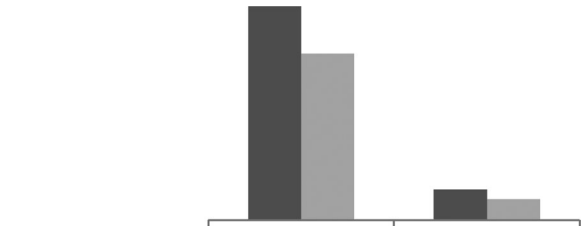
A pilot study was conducted for testing where manual data entry was performed, as electronic records were not available. Doctors' reports were typed into text files, which were then mined. Total 5152 records were obtained from the data gathering phase using the reports. This was aggregated and verified by the panel of doctors before it could actually have been used in real time. During the test period of three months, a total of 52 appropriate cases of patients were taken for observation/learning. These were diagnosed with *malaria* after eliminating possibilities of suffering from *common cold*, *viral fever*, *influenza* and other common ailments. Thirty-five patients were diagnosed correctly with *malaria*. Out of the remaining 17, 12 were diagnosed with *leptospirosis*, two with *dengue*, two with *typhoid*, and one, after due consideration of tests results, was diagnosed as *meningitis*. After *malaria* and *influenza* (since it was already eliminated by the doctors after due consideration), *leptospirosis* was accurately caught by the system.

The strong point, where the system could be very helpful, was in case of the patient suffering from *meningitis*, which otherwise would not have been considered at an initial stage. Due to the disease been shown, and after the tests for *malaria*, *leptospirosis,* and *dengue* showed negative results, the patient was advised to undergo tests to check for other diseases in the list, including *Meningitis*. The blood test result indicated that he was suffering from *Meningitis*, which was then confirmed in the CSF analysis [22] report.

On running the system, differential diagnosis was

Malaria → *Influenza* → *Dengue Fever* → *Typhoid Fever* → *Hepatitis* → *Urinary Tract Infection* → *Leptospirosis* → *Liver abscess* → *Meningitis* → *Yellow Fever* → *Babesiosis.*

In this system, an additional step, where differential diagnosis has been combined with recent medical history (see Fig. 8), to provide differential diagnosis for the past three months, has been implemented. This step has given accurate results to catch recent trends. In the locality where the test system was implemented,



	Training (hrs)	Test (sec)
Bayesian, Hebian & Back propagation	16	2.3
k-NN, Hopfield & LAMSTAR	12.5	1.6

Fig. 9. Speed test: Comparison of k-NN, Hopfield networks, and LAMSTAR versus Bayesian classification, Hebbian learning, and Backpropagation.

probability of a patient suffering from *leptospirosis* increases during monsoon. Though *malaria* still remained on top of the list, *leptospirosis* gained significantly during this period. On back-testing, this also caught swine flu with good precision.

On running the system, recent differential diagnosis was

Malaria → Influenza → Leptospirosis (Regional Outbreak during monsoon) → Dengue Fever → Typhoid Fever → Hepatitis → Urinary Tract Infection → Liver abscess → Meningitis → Yellow Fever → Babesiosis.

This system has been aimed to provide essential medical services with clinical precision to everyone. To do that, high accuracy was required (above 90% for verified dataset). Even though the system is to be used by doctors only, and the doctors have the final say, the accuracy of the system should not be compromised. To verify this, the results obtained by this system were compared with the differential diagnosis provided by various other medical systems, including the information that is available at various online medical portals, and these were also verified by a panel of experts, consisting five reputed doctors at local level. The results obtained matched up to the doctors' expectations, and since the system is self-learning, with time, as the database grows, the accuracy of the system improves.

Initially, this system was implemented using Bayesian Classification, Hebbian Learning and Backpropagation. But by using k-NN, Hopfield networks, and LAMSTAR techniques, the overall speed and accuracy of the algorithm increased considerably. Especially in case of larger datasets, LAMSTAR gave faster and better results in differential diagnosis. Speed tests (see Fig. 9) were performed by calculating the total execution time for the script, using the microtime() function of PHP. The improvement in speed was contributed due to faster calculation of weights by k-NN. Similarly, the list generated by LAMSTAR caught trends better on performing diagnosis based on recent medical history. For example, according to the local trend, *leptospirosis* accurately gained a higher place with LAMSTAR compared to others. This better accuracy in results can be attributed to LAMSTAR networks' link forgetting property, which gives priority to more recent weights.

VII. Conclusion

In this system, by using Hopfield networks, LAMSTAR, and k-NN, an attempt has been made to assist the doctors to perform differential diagnosis. A pilot study was performed and the results obtained were very promising. The system proposes an innovative utilization of the misdiagnosis factor for differential diagnosis along with a possible method of implementation using the SOA technique. The possibility of usage of vastly available EHR data for the purpose allows latest and continuously updated medical data available to the system. The approaches mentioned in this paper can be used to supplement and improve existing systems that provide differential diagnosis.

In the field of medical diagnosis, there is always the scope for uncertainty. This system has been built on the experience of doctors only, so there will always be a scope for ambiguous or uncertain diagnosis. The current system does not give 100% accurate results as not even the doctors can claim to do so; however, its results are promising. It cannot be used as a substitute or a shortcut to diagnosis, but it can definitely complement the doctors' knowledge and could assist them to reach a conclusion. The doctor always has the upper hand to decide whether to use the diagnosis given by the algorithm or not. By using this system, many essential results can be obtained, reducing the dependence on human intuitions, thus reducing the effects of misdiagnosis to a great extent. With the support of various medicinal practitioners and hospitals, higher probability of getting the diagnosis right can be obtained, compared to what individual doctors can do alone. After sufficient self-learning, with an extensive database of medical records to mine from, this can be used to build formidable medical assistance software that can be of great use to all doctors and specially the new practitioners and students. It will also help the medical fraternity in the long run by helping them in getting accurate diagnosis and sharing of medical practices which will facilitate faster research and save many lives.

Acknowledgment

The authors would like to thank Dr. R. Ranade, Dr. H. Thakker, Dr. R. R. Sharma, and Dr. A. Nanda for their expertise and medical domain guidance. They would also like to convey special thanks to Dr. R. Sundararajan and M. Iyer for their constant support in this work.

References

[1] R. Carvalho, R. Isola, and A. Tripathy, "MediQuery—An automated decision support system," in *Proc. 24th Int. Symp. Comput.-Based Med. Syst.*, Jun. 27–30, 2011, pp. 1–6.

[2] K. Kawamoto, C. A. Houlihan, E. A. Balas, and D. F. Lobach, "Improving clinical practice using clinical decision support systems: A systematic review of trials to identify features critical to success," *Br. Med. J.*, vol. 330, p. 765, 2005.

[3] R. A. Miller, "Medical diagnostic decision support systems—Past, present, and future—A threaded bibliography and brief commentary," *J. Amer. Med. Inf. Assoc.*, vol. 1, pp. 8–27, 1994.

[4] W. Siegenthaler, *Differential Diagnosis in Internal Medicine: From Symptom to Diagnosis*. New York: Thieme Medical Publishers, 2011.

[5] S. F. Murray and S. C. Pearson, "Maternity referral systems in developing countries: Current knowledge and future research needs," *Social Sci. Med.*, vol. 62, no. 9, pp. 2205–2215, May 2006.

[6] J. Han and M. Kamber, *Data Mining Concepts and Techniques*. San Mateo, CA: Morgan Kaufmann, 2011.
[7] R. Rojas, *Neural networks: A Systematic Introduction*. Berlin, Germany: Springer-Verlag, 1996, pp. 337–370.
[8] D. Graupe, H. Kordylewsky, and N. Schneider, *Principles of Artificial Neural Networks*, 2nd ed. Singapore: World Scientific, 2010.
[9] M. Bell, *SOA Modeling Patterns for Service-Oriented Discovery and Analysis*. New York: Wiley, 2010, p. 390.
[10] (Jun. 10, 2012). WebMD: Better Information, Better Health. [Online]. Available at http://symptoms.webmd.com/symptomchecker
[11] L. Li, L. Jing, and D. Huang, "Protein-protein interaction extraction from biomedical literatures based on modified SVM-KNN," in *Nat. Lang. Process. Knowl. Engineer.*, 2009, pp. 1–7.
[12] M. Berlingerio, F. B. F. Giannotti, and F. Turini, "Mining clinical data with a temporal dimension: A case study," in *Proc. IEEE Int. Conf. Bioinf. Biomed.*, Nov. 2–4, 2007, pp. 429–436.
[13] H. R. Warner and O. Bouhaddou, "Innovation review: Iliad—A medical diagnostic support program," *Top Health Inf. Manage.*, vol. 14, no. 4, pp. 51–58, 1994.
[14] Department of Medicine Massachusetts Hospital, Boston, DXplain System. (2011). Available at :http://dxplain.org/dxpdemopp/dxpdemobrief_files/frame.htm
[15] E. Coiera, *The Guide to Health Informatics*, 2nd ed. London, U.K.: Arnold, Oct. 2003, pp. 101–123.
[16] K. A. Mohan. (Jun. 2012). National Electronic Health Record Models. Tata Consultancy Services (TCS) Whitepaper. [Online]. Available at: http://www.tcs.com/resources/white_papers/Pages/NationalElectronicHealthRecordModels.aspx
[17] D. S. Rizzuto and M. J. Kahana, "An autoassociative neural network model of paired-associate learning," *Neural Comput.*, vol. 13, pp. 2075–2092, 2001.
[18] S. M. Polyn and M. J. Kahana, "Memory search and the neural representation of context," *Trends Cognitive Sci.*, vol. 12, pp. 24–30, 2008.
[19] Hubert Kordylewski and Daniel Graupe, "A novel large-memory neural network as an aidin medical diagnosis applications," *IEEE Trans. Inf. Technol. Biomed.*, vol. 5, no. 3, pp. 202–209, Sep. 2001.
[20] T. Kohonen and T. Honkela. (2007). Kohonen network. [Online]. Available: Scholarpedia at: http://www.scholarpedia.org/article/Kohonen_network (Accessed on 4th Feb 2011,7.30 pm IST).
[21] M. H. Valipour, B. A. Zafari, K. N. Maleki, and N. Daneshpour, "A brief survey of software architecture concepts and service-oriented architecture," in *Proc. 2nd IEEE Int. Conf. Comput. Sci. Inf. Technol.*, Aug. 8–11 2009, pp. 34–38.
[22] A. R. Tunkel, B. J. Hartman, S. L. Kaplan, B. A. Kaufman, K. L. Roos, W. M. Scheld, and R. J. Whitley, "Practice guidelines for the management of bacterial meningitis," *Clin. Infectious Dis.*, vol. 39, no. 9, pp. 1267–1284, Nov. 2004.

Rahul Isola (S'11) received the B.E. degree in mechanical engineering from Don Bosco Institute of Technology, University of Mumbai, India, in 2012. He is currently working toward the M.S. degree in computer science at the University of North Carolina at Charlotte.

He has been working on techniques that utilize misdiagnosis, and other indicators to perform differential diagnosis. He is also working on projects like real-time financial advising systems, intelligent physiotherapeutic rehabilitation machines, etc. He is a freelance web developer. His research interests include data mining, knowledge discovery, neural networks, forecasting/prediction, trend projection, web designing, intelligent robotics, bioinformatics, and biomechanics.

Rebeck Carvalho (S'11) received the B.E. degree in computer engineering from Don Bosco Institute of Technology, University of Mumbai, India, in 2011.

He is currently working as Software Engineer in Infosys Limited. He has been working with R. Isola on developing Medical Decision Support Systems. He is also involved in the development of real-time financial systems, which makes the comparison of various market technical indicators and uses data mining techniques to identify the latest market trends. His research interests include data mining, artificial intelligence, and image processing.

Amiya Kumar Tripathy (M'12) received the B.E. and M.Tech. degrees in computer science and engineering in 1996 and 2004, respectively. He is currently working towards the Ph.D. degree in computer science at the Indian Institute of Technology Bombay (IITB), Mumbai, India, since July 2008.

He has been an Assistant Professor in the Department of Computer Engineering, Don Bosco Institute of Technology, Mumbai, India, since 2006 and is deputed to IITB fulltime in July 2008. He is currently working on an IndoJapan initiative research project called GeoSense: GeoICT and Sensor-Network-based Decision Support System in Agriculture and Environment Assessment. His research has been sponsored by the Department of Science and Technology, Government of India. His research interests include machine learning, data mining, wireless sensor networks, body area sensor networks for health care, and information and communication technology for rural developments. He is a member of ACM since 2006.