BACHELOR OF SCIENCE IN COMPUTER SCIENCE AND ENGINEERING

Disease Identification from Unstructured User Input

Authors

Fahim Faisal	Id: 124443
Shafkat Ahmed Bhuiyan	Id: 124433

Department of Computer Science and Engineering (CSE)
Islamic University of Technology (IUT)
Organisation of Islamic Cooperation (OIC)
Gazipur-1704, Bangladesh

November, 2016

Islamic University of Technology

Department of Computer Science and Engineering (CSE)

Disease Identification from Unstructured User Input

Authors

Fahim Faisal	Id: 124443
Shafkat Ahmed Bhuiyan	Id: 124433

Supervisor

Dr. Abu Raihan Mostofa Kamal
Associate Professor
Department of CSE
IUT

A thesis submitted to the Department of CSE

in partial fulfillment of the requiremnts for the degree of B.Sc.

Engineering in CSE

Academic Year: 2015-16

November - 2016

Declaration of Authorship

This is to certify that the work presented in this thesis is the outcome of the analysis and experiments carried out by Fahim Faisal and Shafkat Ahmed Bhuiyan under the supervision of Dr. Abu Raihan Mostofa Kamal, Associate Professor of Department of Computer Science and Engineering (CSE), Islamic University of Technology (IUT), Dhaka, Bangladesh. It is also declared that neither of this thesis nor any part of this thesis has been submitted anywhere else for any degree or diploma. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

Authors:

Fahim Faisal

Student ID - 124443

Shafkat Ahmed Bhuiyan

Student ID - 124433

Supervisor:

Dr. Abu Raihan Mostofa Kamal

Associate Professor

Department of Computer Science and Engineering

Islamic University of Technology (IUT)

Acknowledgement

We would like to express our grateful appreciation to **Dr. Abu Raihan Mostofa Kamal**, Associate Professor, Department of Computer Science & Technology, IUT for being our advisor and mentor. His motivation, suggestions and insights for this thesis have been invaluable. Without his support and proper guidance this research would not have been possible. His valuable opinion, time and input provided throughout the thesis work, from first phase of thesis topics introduction, subject selection, proposing algorithm, modification till the project implementation and finalization which helped us to do our thesis work in proper way. We are really grateful to him.

It was my pleasure to get the cooperation and coordination from Professor Dr. M.A. Mottalib, Head of CSE Department, IUT during various phases of the work. I am grateful to him for his constant and energetic guidance and valuable advice. My deep appreciation extends to all the respected jury member of my thesis committee for their insightful comments and constructive criticism of my research work. Surely they have helped me to improve this research work greatly. Lastly I would like to thank the faculty members of CSE Department, IUT who have helped to make my working environment a pleasant one, by providing a helpful set of eyes and ears when problems arose.

Abstract

In this information age the number of internet users are growing rapidly. Now a days people first search internet if they face any health hazard rather than asking a doctor for health related advice as online medical help or health care advice is easier to grasp. Sometimes, people give less importance to minor symptoms which may cause serious health hazards. In this context, online health advice can be instant beneficiary. Moreover, existing online symptom checkers give possible sense of disease but these systems are not reliable enough. Also existing systems are not interactive and time consuming. Herein, we propose an automated disease identification system that takes unstructured user input and provides a list (topmost diseases that have greater likelihood of occurrence) of probable diseases. We use Conditional Random Field and Support Vector Machine to detect the word phrases and to classify the class labels.By not considering demographic information, it gives 4.603% accuracy improvement whereas, considering demographic information we get slightly better performance with 5.783% accuracy improvement.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

Number of internet users is growing exponentially over the years. More than a third of adults in the United States regularly use the internet to self-diagnose their ailments, using it both for non-urgent symptoms and for urgent symptoms such as chest pain[2]. While there is a wealth of online resources to learn about specific conditions, self-diagnosis usually starts with search engines like Google, Bing, or Yahoo. However, internet search engines can lead users to confusing and sometimes unsubstantiated information, and people with urgent symptoms may not be directed to seek emergent care.

People also post their health related queries (such as asking about what kind of disease that they might be suffering from) on various healthcare forums. There are other group of people who leave their responses to those posts with identifications of possible diseases. However, these identifications may not be always accurate, and also there is no assurance that users will always get a reply on their post. Moreover, some posts are fabricated or made up which can drive the user in a wrong direction. According to a survey conducted by CNN[3], it is found that 25% users lie on social networking sites. Therefore, reliability is a big issue here.

As a result there has been a proliferation of more sophisticated programs called online disease identification system that attempt to more effectively provide a potential diagnosis for patients and direct them to the appropriate care setting.

In technical terms, an online disease identification system is a web-based Clinical Decision Support System (CDSS). Powerful CDSS design is an active research area and professional practitioner uses various types of CDSS which mainly rely on structured high volume clinical documents and patient health records. Most of the proposed and designed CDSS are doctor centric. Less amount of research work has been done on patient centric CDSS. An efficient patient centric CDSS can help a patient to act without direct supervision from a professional [4].

Profound research work has been done on diagnosis using clinical texts and Electronic Health Record (EHR) data. Clinical text documents are domain specific with frequent use of clinical terminologies whereas, general users express their problems using non-medical terms [5]. It is unlikely that a user is familiar with cardio(heart)-myo(muscle)-pathy(disease) related terms and use these terms to define his problem. So, an intelligent disease identification system needs to map these non-medical texts with corresponding technical terms to make identification.

During the time of conversation, a consultant lets the patient to narrate his problem. It is expected that an intelligent disease identification system will do the same too. Besides, a feasible web based system needs to be user-centric with improved usability. This can be achieved if the user can express the problem in his own word and gets an automated suggestion with preferable accuracy. So, the expected input is unguided patient's narrative in text format, and the challenging task is to extract relevant information from an emotionally biased, noisy and unstructured format where grammatical and spelling mistakes can frequently occur. State of the art online disease identification systems rely on guided user input, long question and answer session and symptom-disease relation ([6],[7],[8],[9],[10],[11],[12]).

1.2 Motivation

- Need for an automated, reliable and patient-centric web-based disease identification system.
- Need for a user interactive system where users can describe their problems in their own "living room language".
- Need for an intelligent online disease diagnosis system which takes not only symptoms, intensity and time duration into account but also many other signature attributes like taken medication and food, family history, physical and psychological state changing triggers etc.

1.3 Problem Statement

The problem statement of our thesis work is as follows:

> People frequently search online for health related advice. Typing acute symptoms into search engines to seek emergent care is a naive approach. Following online health forums is another common practice but credibility of these user generated contents are not guaranteed. Though online symptom checkers give a sense of possible diagnoses, the tools are frequently wrong and not user-interactive. So, there is a need of an intelligent online disease identification system which can identify disease from flexible user input. To solve these issue, we propose an automated, user-interactive and reliable online disease identification framework.

1.4 Our approach

In this work, we propose an online disease identification framework integrating case based reasoning and machine learning methods. Using numerous natural language processing techniques, relevant attributes are extracted from the unstructured user input. Then these attributes are used to generate a ranking of probable diseases from a symptom based clustered disease database.

Chapter 2

Related Work

A number of renowned research works have been done on disease identification system. Works in this domain can be divided into two types easily. One area is focused on online health forum and social network post analysis. Another area is Electronic Health Record (EHR) data analysis. Using EHR data a number of sophisticated Clinical Decision Support Systems (CDSS) and specific disease diagnosis systems have been designed. Information extraction from noisy data is individually a big research area which is increasingly gaining attention.

2.1 Information and Relation Extraction

Information extraction and feature selection from text using Natural Language Processing (NLP) is an active research field. The ultimate goal of information extraction is the automatic transfer of unstructured textual information into a structured form. In this context, entities are typically short phrases representing a specific object such as "pancreatic neoplasms". The second logical step is the extraction of associations or relations between recognized entities, a task that has recently found increasing interest in the information extraction (IE) community. Relation Extraction (RE) deals with the problem of finding associations between entities within a text phrase. Common approaches for relation extraction use

rule-based, co-occurrence-based and kernel-based methods.
In biomedical domain most early research focused on the mere detection of relations but the classification of the type of relation is of growing importance. A lot of work has been done on biomedical relation extraction focusing on rule-based and machine learning techniques. In the last decade, the focus has changed to hybrid approaches like CRF+SVM showing better results.

[13] is a renowned study of this field. In this work the authors perform “semantic relation extraction” (SRE) which is the combined task of detecting and characterizing a relation between two biomedical entities: disease-treatment and gene-disease.
The authors benchmark their approach on two different tasks. In the first experiment, they identify semantic relations between diseases and treatments from PubMed abstracts using the cascaded CRF model. The detected relations are then classified into seven predefined types. In the second experiment, they extract semantic relations between genes and diseases from GeneRIF sentences using both the cascaded and the one-step CRF. CRFs are probabilistic graphical models used for labeling and segmenting sequences and have been extensively applied to named entity recognition (NER). Then, the extracted semantic associations of genes and diseases are provided as a resource description framework (RDF) graph. Thus, the association network is represented in terms of RDF triplets, i.e. subject (gene), predicate (association) and object (disease).
The authors treat RE/SRE as a sequential labeling problem typically applied to NER or part-of-speech (POS) tagging. They employed a number of features: orthographic features, word shape features, ngram features, dictionary features, context features, negation feature etc. Text corpus or dataset used in this model was generated from MEDLINE 2001 abstracts. In a total of 3570 sentences, en-

tities describing diseases and treatments were extracted and disease-treatment relations were classified as cure, only disease, only treatment, prevents, side effect, vague, does not cure.

The result obtained by this model is compared with results obtained by a Support Vector Machine (SVM), a multilayer Neural Network (NN), probabilistic generative models, and with rule-based methods. The proposed model achieve higher or comparable accuracy on two evaluation data sets.

In this paper, though the authors focus on extracting the relations and their types between entities and they do not take into account additional information, such as the properties under which a relation holds. For example, when extracting associations between diseases and genes, it is important to know that certain facts hold for specific populations only. Incorporating these conditions into the relation extraction task is an ongoing research.

2.2 Social media and health forum post analysis

Social media and health forum posts are unstructured data-sources containing lot of noises. Lot of work have been done on sentiment analysis, disease outbreak prediction etc using these data sources.

2.2.1 Disease outbreak prediction

A large amount of research work has been published on disease outbreak prediction using web[14] and social media data analysis [5].

In [5], the authors present a methodology for early detection and analysis of epidemics based on mining Twitter messages. One strong implication of the use of Twitter is that it provides location indicators and real-time update of disease maps. In order to reliably trace messages of patients that actually complain of a disease, first, a model is designed to learn naive medical language, second, a symptom-driven keyword analysis is adopted. According to the authors, this approach yields a very high level of correlation with flu trends derived from traditional surveillance systems. Compared with Google Flu Trends (GFT), this model performs better. This study stated the fact that people use everyday language rather than medical jargon (e.g. runny nose vs. respiratory distress) in health related conversations on social networks. As for example, consider the following striking difference in the usage of terms describing the same health conditions, the first by a clinician, the second by a patient: "*Clinicians should maintain a high index of suspicion for this diagnosis in patients presenting with influenza-like symptoms that progress quickly to respiratory distress and extensive pulmonary-involvement*"." "*For the past 3 days I have had a stuffy, runny nose, congested chest, fever, sore ears and throat and burning eyes. I've been taking cold and flu medication, and it doesn't help*". So knowledge of patients terminology is essential for the mining in this domain. In this study, the proposed algorithm first, collect from the web alternative naive and technical synonyms for each technical term reported in a disease definition (e.g. cephalea). Then, it further extends the terminology seeking additional terms on Twitter. At the end of this step, each symptom is associated with a cluster of synonyms (mostly naive). For each disease, a Boolean query is created where every symptom is searched using any of its alternative terms. Tweets reporting the combination of symptoms that match any of the five diseases are geographically and temporally traced. Finally, the ILIECDC trend in the U.S. is compared with officially available data on that disease. Empirical analysis shows that this methodology provides with data that show an impressive correlation with official U.S. trends for influenza like illness. Besides, it allows for additional, in-depth analysis of a disease outbreak, for example, the intensity and co-occurrence of specific symptoms.

The authors stated that, it is possible to apply this algorithm for pairing technical terms with everyday language to any domain, not just the medical domain.

2.2.2 Trust and Credibility management

The exchange of information online may suffer from various kinds of fallacies, including the presence of incorrect, inaccurate, incomplete, improperly emphasized, ambiguous or disputable medical advice. So the trustworthiness and credibility of user generated online contents are not guaranteed. Besides, each user has an affective state that depicts his attitude and emotions that are reflected in his posts and it is necessary to find how helpful and informative a user post is in the context of health forums.

In [15] the authors propose a method for automatically establishing the credibility of user-generated medical statements and the trustworthiness of their authors. They, introduce a joint probabilistic graphical model that learns user trustworthiness, statement credibility, and language objectivity simultaneously. Then, they apply this methodology to the task of extracting rare or unknown side-effects of medical drugs. According to the authors, online health communities are the platforms where large scale non-expert data has the potential to complement expert medical knowledge.

To assess a post's objectivity and quality the authors use two types of linguistic features: stylistic and affective features. Besides, user demographics like age, gender and location, engagement in the community reflected by the number of posts, questions, replies, or thanks received, are expected to correlate with user authority in social networks. Also, users who write long posts tend to deviate from the topic, often with highly emotional digression. On the other hand, short posts can be regarded as being objective and on topic. The authors attempt to capture these intuitive aspects as additional user features.

In this study, the authors conduct two lines of experiments with different settings. One setting aims to study the predictive power of the model which use conditional random field (CRF) in determining the common side-effects of a drug, in comparison to the baselines (SVM and SVM with distant supervision). The result shows that CRF performs better than the baselines.

The other setting aims discovering side-effects that are not covered by expert databases, and identifying the most trustworthy users that are worth following. The model reliably identifies out of knowledge base side-effects and the credibility assessment is done manually based on complete discussion thread.

Though this model achieves high accuracy in most of the test cases, it relies on a relatively simple information extraction machinery to identify candidate side-effect statements, which is prone to errors. The tool misses out on certain kinds of paraphrases (e.g. “nightmares” and “unusual dream” for Xanax) resulting in a drop in recall. So, it is likely that a more sophisticated information extraction approach can further improve this model.

2.2.3 Information gaining from specific disease community

It is well understood that accessing other patient's experiences can positively support and boost confidence, confirm treatment choices, provide new alternatives when facing troubling decisions (e.g., related to medications, dietary habits, etc.) and help alleviate loneliness while maintaining relations with others. As a result, there is a big number of specific disease communities in online. Enormous research has been going on focusing on these online disease communities.

[16] is a study where the authors uncover the role of online social networks for a growing community of chronic patients: Crohn's disease patients. Chronic Disease patients are very much conscious about how a disease should be dealt with. So they spend vast time in online social network patient communities and search for improvement of their treatment. Therefore, by analyzing the community activity in two different online social networking sites (OSNs): Facebook and Twitter the authors tries to find out the answer of the following questions. Is the mood that patients express online influenced by the use of given medications? What can be found out by characterizing the data exchange in different online social networking sites (OSNs)? How people behave online and feel about given arguments?

In this study the authors analyze relevant posts and find out sentence positivity, negativity and emotion flow in a given argument. The authors used Opinion finder which can process a corpus of text and identify subjective sentences and various aspects of subjectivity. Obtained result shows that a specific medicine Infliximab is the treatment that is predominantly influencing the CD community. Besides, the authors analyze number of members, messages and frequency of messages to present a comparison between Facebook and Twitter. Here we can see that Facebook is more reliable in term of community support as most of the twitter posts contain advertisement and fund raising campaign topic wheres Facebook has no restriction on post word count and it contains more descriptive posts.

Another part of this study is the most discussed topics. A probabilistic model based on 4 topic: Cause, Disease, Treatment and Side effects is defined. The result shows that most of the time people discuss which might be the cause focusing on different deficiency explanations.

2.3 Electronic Health Record (EHR) data analysis

Large volumes of clinical documents are generated by electronic health record (EHR) systems. On one hand, these clinical documents are unstructured or semi structured. It is a difficult task to extract information from these documents. Symptom information and medication information extraction from clinical notes need sophisticated clinical language processing methods. On the other hand, due to the individual diversity, discovering and mining relationship between symptom information and medication information from clinical texts becomes a challenge problem. These underutilized resources have a huge potential to improve health care. Besides, clinical narratives contain a lot of valuable information about patients, such as medication conditions (*diseases, injuries, medical symptoms, and etc.*) and responses (*diagnoses, procedures, and drugs*). These types of valuable information extracted from clinical narratives can be used to build profiles for individual patients, discover disease correlations and enhance patient care.

Here, a use-case scenario is indicated where an information extraction system from structured and unstructured EHR data to map symptom with related medication can result in a highly sophisticated doctor-centric clinical decision support system (CDSS).

A use case scenario:

a new patient is diagnosed with alcoholic liver disease (ALD) and type2 diabetes. A set of related symptoms are observed, so a set of medications should be prescribed to treat these symptoms. In the meantime, related clinical notes extracted *from a database with symptoms and medications highlighted will also be presented as evidences to the physician and patient. The physician can use these clinical notes to support decisions, and the patient might find the medications given by physician more convincing based on the clinical notes from other patients who had similar medical conditions.*

2.3.1 Clinical decision support system

A personalized recommender system can support practitioner's decision making in prescription.

In [17] a framework of hybrid recommender system is proposed to support general practitioners (GPs) in drug decision making. This framework relates patient's need to different drug clusters, includes meticulous patient features in free text and mines up-to-date drug trends. It integrates artificial neural network and case-based reasoning.
The authors uses EHR data to build training data and specific disease database. A patient feature space is built then from unstructured free text source and structured EHR data source. Patient feature space contains extracted symptom entity and normalized lab-test result.
Now, when a patient with morbidity comes for consultation, a GP may make inquiries and order lab tests for the patient. Information about this patient is entered into the system as a new case during the process. The system will process the new data and extract patient features. Then, a GP makes a diagnosis based on the patient's problem. The diagnosis is matched to a specific disease category in the system, to determine which symptom-drug classifier to use. Patient features in the new case are put into the classifier to predict which drug cluster/clusters to choose for this patient. Drugs in each cluster will be ranked

by the ranking module to form the final recommendation list. At last, a GP should dispense advice, prescription or other kind of treatment to patient to restore his/her health. This is where the recommender system supports GPs with drug recommendation list that is personalized for the specific patient.

This is an ongoing research work and the proposed architecture is not implemented yet. As, the system is related to health-care, it should be evaluated by experienced clinician before it comes to practical use.

2.3.2 Specific disease diagnosis system

In [18], the authors developed a multi-tasking framework for Osteoporosis (bone fracture disease) that extracts the integrated features from unstructured Electronic Health Record (EHR) data for progressive bone loss and bone fracture identification. It also selects the individual informative Risk Factors (RFs) that are valuable for both patients and medical researchers.

From ill-organized EHR data this framework finds a representation of RFs to differentiate the salient integrated features. These integrated features constructed from original RFs will become the most effective features for bone disease identification. From the original dataset, using multi-layer deep belief network (DBN) the authors built a comprehensive disease memory (CDM) of RFs to capture the characteristics for all patients to predict osteoporosis and bone loss rate simultaneously.

Another task of this study is the informative RF selection that cause the disease (osteoporosis). For this task they propose to model the diseased patients and healthy patients separately based on their unique characteristics. Two variants of disease memory are introduced for this task: Bone disease memory (BDM) and non-disease memory (NDM). BDM memorizes the characteristics of those individuals who suffer from bone diseases. Therefore, RFs reconstructed using

BDM are reflections of the diseased individuals. Now, if there is a large error between the original RF and the reconstructed one, then the RF is a noisy RF and it will not be considered as an informative RF. Similarly, NDM memorizes attributes for non-diseased individuals. In the testing stage, NDM is used to reconstruct RF and here, the more error the RF shows, the more informative it is.

Using the proposed approach the authors select at most top 50 informative RFs. The best prediction performance is achieved using the proposed method when selecting the top 20 to top 25 informative RFs and feed them to the classifier for the osteoporosis prediction. For making the final prediction, a majority voting classification system is used.

According to the authors, these variety of DM models increase the flexibility for monitoring the disease for different groups of patients. Besides, the experimental results showed that the proposed method improves the identification performance and has great potential to select the informative RFs for bone diseases. Further extension of this work can be a bone disease analytic system deployed in bone disease monitoring and preventing settings which will offer much greater flexibility in tailoring the scheduling, intensity, duration and cost of the rehabilitation regimen.

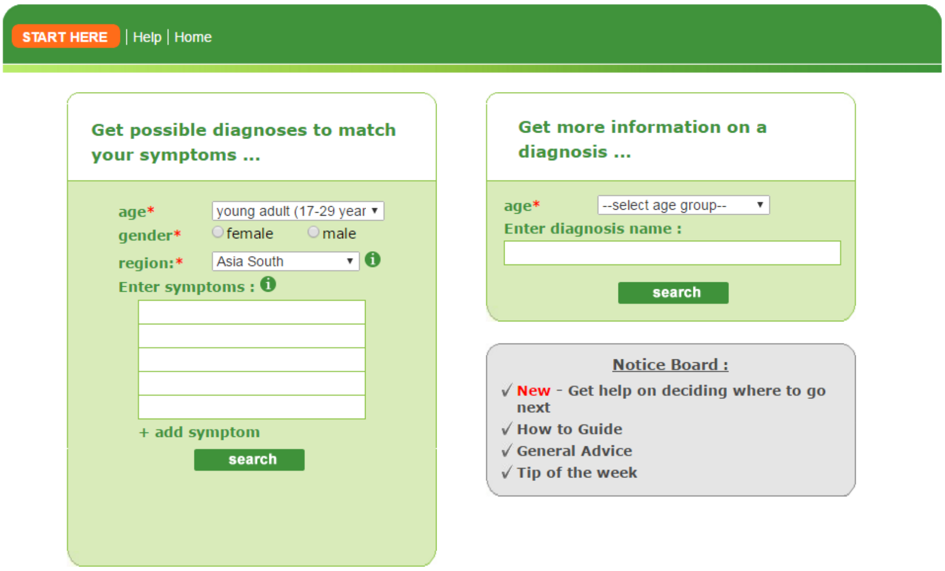
[19] is another study where both structured EHR data and unstructured textual medical data are used to predict state of Alzheimer's disease. [20] focuses on heart failure prediction. These models use different NLP and machine learning techniques to extract features from unstructured textual data.

2.4 Existing web based disease diagnosis systems

Using computerized algorithms, online symptom checkers ask users a series of questions about their symptoms([6], [11]) or require users to input details about their symptoms themselves([8], [7]). ([12], [9]) provide pictorial representation of human body for selection process. In [10], a long symptom list is given and based on the tabbed one user has to select important parameters like intensity, organ location, duration etc. The algorithms vary and may use branching logic, Bayesian inference, or other methods. Private companies and other organizations, including the National Health Service[11], the American Academy of Pediatrics, and the Mayo Clinic[10] have launched their own symptom checkers. One symptom checker, iTriage, reports 50 million uses each year. Typically, symptom checkers are accessed through websites, but some are also available as apps for smart phones or tablets[8].

Now, Symptom checkers have several potential benefits. They can encourage patients with a life threatening problem such as stroke or heart attack to seek emergency care. 21 For patients with a non-emergent problem that does not require a medical visit, these programs can reassure people and recommend they stay home. For approximately a quarter of visits for acute respiratory illness such as viral upper respiratory tract infection, patients do not receive any intervention beyond over the counter treatment, and over half of patients receive unnecessary antibiotics. Reducing the number of visits saves patient's time and money, deters over-prescribing of antibiotics, and may decrease demand on primary care providers. However, there are several key concerns. If patients with a life threatening problem are misdiagnosed and not told to seek care, their health could worsen, increasing morbidity and mortality. Alternatively, if patients with minor illnesses are told to seek care, in particular in an emergency department, such programs could increase unnecessary visits and therefore result in increased time and costs for patients and society.

2.4.1 Isabel



Get possible diagnoses to match your symptoms ...

age* young adult (17-29 year
gender* ○ female ○ male
region:* Asia South
Enter symptoms :
+ add symptom

Get more information on a diagnosis ...

age* --select age group--
Enter diagnosis name :

Notice Board :
✓ New - Get help on deciding where to go next
✓ How to Guide
✓ General Advice
✓ Tip of the week

Figure 2.1: Isabel symptom checker

The Isabel symptom checker[8] has been adapted from the same system that is used by healthcare professionals around the world. It has undergone 12 years of development and is built using the latest statistical natural language searching technologies which enables it to be much easier to use but, at the same time, also cover many more diseases and provide more accurate results. Isabel covers 6,000 diseases and allows user to enter an almost infinite number of symptoms in normal language rather than being forced to enter only the symptom that is included in a list or shown on a drawing of a human body. In addition, one can also include any other chronic conditions he may have such as diabetes or high blood pressure. The most important step in using Isabel is the symptoms one enter. One can also enter abnormal test results if he has them. However, user should enter the meaning of the result, such as high, low or whatever the medical term is, rather than the number. The system understands text but not numbers. The suggested diagnoses are not ranked in an order of likelihood but on the basis of how well what is entered matches Isabel's database of diseases.

2.4.2 HealthDirect

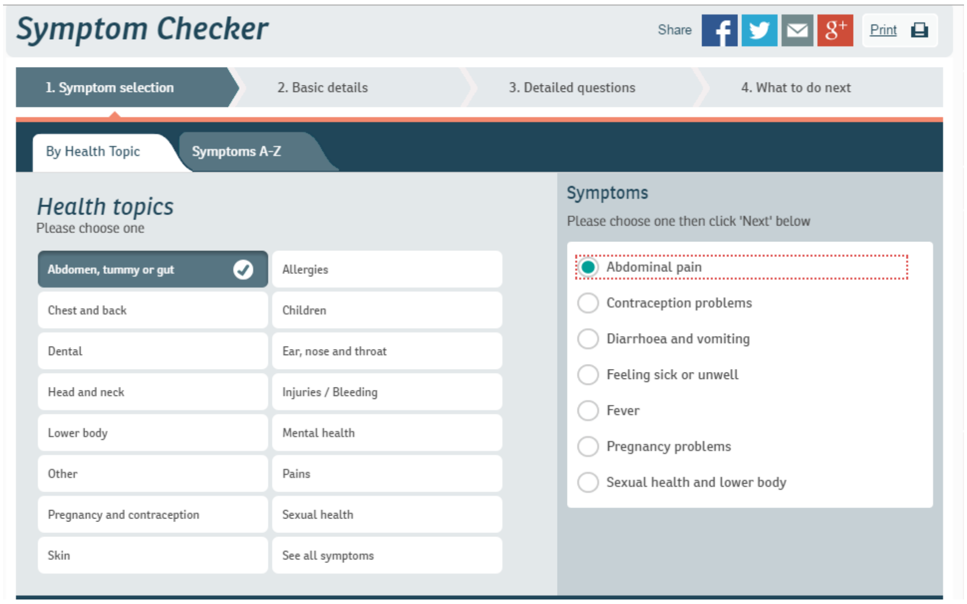


Figure 2.2: HealthDirect symptom checker

The healthdirect Symptom Checker[6] supports GPs by increasing health literacy of patients about their situation before attending their GP appointments. Evidence shows that increased health literacy for patients results in better outcomes for the GP and patient. The final output of each Symptom Checker includes a print out of the patient's answers to the questions along with evidence-based self-care advice. This can be reviewed and assist triage by GPs and practice nurses.

The self-care information acts as both a prompt and a memory aid for GPs and patients during and after a consultation. Besides, this Symptom Checker provides evidence-based information and advice which helps to triage consumers to the most appropriate entry point in the health system, based on their health issue. Patients can feel very uncertain about "what to do next", including seeking medical advice and treatment even when they need it, and this system facilitates this decision making process. This system is combined with the data from the National Health Services Directory (NHSD) to ensure that if a patient is advised to seek medical attention, the site(s) at which this help is available will be open and capable of dealing with the medical issues.

2.4.3 WebMd

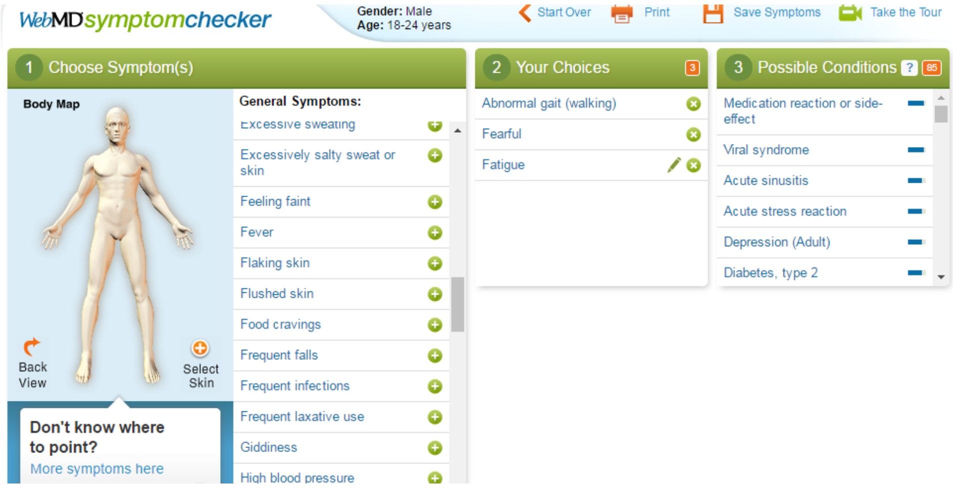


Figure 2.3: WebMd symptom checker

The WebMD Symptom Checker[12] is designed to help user understand what his medical symptoms could mean, and provide him with the trusted information. This tool does not provide medical advice. It is intended for informational purposes only. It is not a substitute for professional medical advice, diagnosis or treatment. Compared to other existing symptom checkers, this tool provides pictorial representation of human body so that users can easily select the symptom area without using explicit medical terms.

2.4.4 Strength and weakness of existing web based solutions

[21] presents an elaborated insight on existing symptom checkers and suggests that in many cases symptom checkers can give the user a sense of possible diagnoses but also provide a note of caution, as the tools are frequently wrong. Based on the result, it is stated that on average, symptom checkers provides the correct diagnosis within the first 20 listed in 58% of standardized patient evaluations, with the best performing symptom checker listing the correct diagnosis in 84% of standardized patient evaluations. Symptom checkers advise the appropriate level of care about half the time, but this varies by clinical severity. According to the authors, there are several potential advances that may improve the performance of existing symptom checkers in future versions. Incorporating local epidemiological data may help inform diagnoses. For instance, addition of real time information about the local incidence of illness in the community greatly improved the performance of a diagnostic tool for group -A streptococcal pharyngitis. Diagnosis rates could also be improved if symptom checkers incorporated individual clinical data from medical claims or the electronic medical record. Demographic information is critical for diagnostic decisions for programs such as symptom checkers. One surprising finding of this study is that symptom checkers that ask for demographic background information do not perform better.

So, if symptom checkers are seen as a replacement for seeing a physician, they are likely an inferior alternative. However, in-person physician visits might be the wrong comparison because patients are likely not using symptom checkers to obtain a definitive diagnosis but for quick and accessible guidance. Besides, symptom checkers are likely a superior alternative if these systems are seen as an alternative for simply entering symptoms into an online search engine such as Google, A recent study found that when typing acute symptoms that would require urgent medical attention into search engines to identify symptom-related web sites, advice to seek emergent care was present only 64% of the time. So it can be said that symptom checkers may be of value if the alternative is simply using an internet search engine for seeking health related advice.

However, analyzing the working process and success rates of symptom checkers we can state some point of facts describing the weakness of existing web based solutions.

- These systems do not provide linguistic diversity so that users can feel comfort in time of giving input.
- Surfing over the long list and Q&A session is time consuming and tedious task for user.
- During selection process these systems use explicit medical terms which are hardly appreciable by most of the users and thus, limits the scope of user interaction.
- Database matching method used in these systems are inefficient[22] . Example: Input “red eye” identifys diseases which have the words “red” and “yellow” in the text Database. In such cases, it identifies diseases which has “red rash” or “yellow eye” as symptoms but not having “red eye”.
- The demographic information is not effectively incorporated into the symptom checker's algorithms[21].

2.5 Other Works

[22] is a recent study that proposes an automated disease prediction system based on online user input. The authors stated that existing systems mostly deal with symptom checking and database matching which are not enough because many signature attributes like time, duration, intensity are not taken into consideration and there is less scope of linguistic diversity and user flexibility.

Therefore, the authors propose a machine learning and text mining based solution for online disease prediction domain. Five important attributes: symptoms, time, intensity, organ, duration are retrieved from guided user input using Natural Language Processing (NLP) techniques and the feature space is represented as a matrix. Then using similarity measurement method these feature matrix is matched with existed database disease matrix. Based on the probabilistic outcome, topmost matched ones are shown as results. Decision tree, synonym parent tree and reference tag method are used to entity recognition and mapping.

The proposed system is evaluated by comparing their result with an existing popular symptom checker site and achieved 14.35% higher accuracy than the existing system. Input dataset is generated from online health forum posts from sites like patient.co.uk.

Though this system takes important attributes into consideration, some crucial tasks of NLP like noise reduction, misspelling correction and inaccurate form of input structure are not taken into consideration. Besides, to ensure highest form of user interaction, input form needs to be unstructured.

Chapter 3

Proposed Disease identification Framework

This part describes the input-output information flow, components of the framework and how the feature vector is generated from unstructured user input to identify disease.

3.1 Framework overview

Our proposed disease identification system provides scope of user interaction in natural language text form. Then this text input is used to build the feature space for disease identification. As shown in fig. 3.1 there are two main components of our system: (1)Information extraction module (2)Identification module. Module 1 deals with text processing and attribute extraction whereas the identification part is done in module 2.

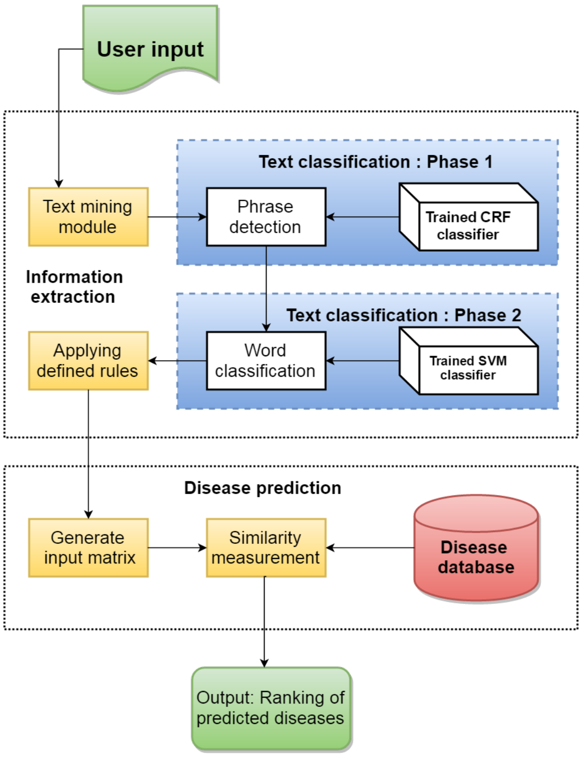


Figure 3.1: System architecture of proposed disease identification framework.

3.1.1 Information extraction module

Attribute selection based on Health forum data analysis

We are expecting an input format which is similar to the heath forum posts. So we analysed the behaviour and format of forum posts of heath domain.[1] is this kind of medical support forum with millions of registered users and over 4.5 million posted messages. Here users normally post about their health related problems and as replies, get community support and expert advice. We crawled 100 most active support group posts from [1] forum. These posts were written from July 2005 to June 2016. Previously, This data-set was used in [23] to predict age and gender from forum posts. We observed that most of the user threads describe symptoms, time period, intensity and past and present medical history. The use of affecting and emotion expressive words are also high. The most frequently used words by the daily strength forum users to describe disease, symptom, organ and drug are plotted in fig. 3.2 based on their frequencies. We designed our system in a way to make use of information similar to these forum posts.

Table 3.1: Feature attributes and class labels

Category	Attribute name	Classification label
Demographic information	Age	Modifier
	Gender	Modifier
Symptom related information	Symptom name	Symptom
	Time	Time
	Intensity	Modifier
	Organ	Organ
Medical history	Past medication	Medication
	Disease history	Disease
	Lab test result	Medication
	Hospitalization and surgery history	Medication

Existed web based disease identification frameworks mainly work on disease – symptom relation. Whereas, from the posts of health forum, we can see that user also writes about his or her past medical history, surgery and lab test result, food habits etc.. These information can give meaningful insights to identify the disease. So a feasible system needs to identify these type of attributes. Based on these analysis we selected three types of attributes which can be classified in 6 labels: Symptom, organ, disease, time, medication and modifiers as shown in table 4.1. All other words labeled as others are not considered relevant for our task.

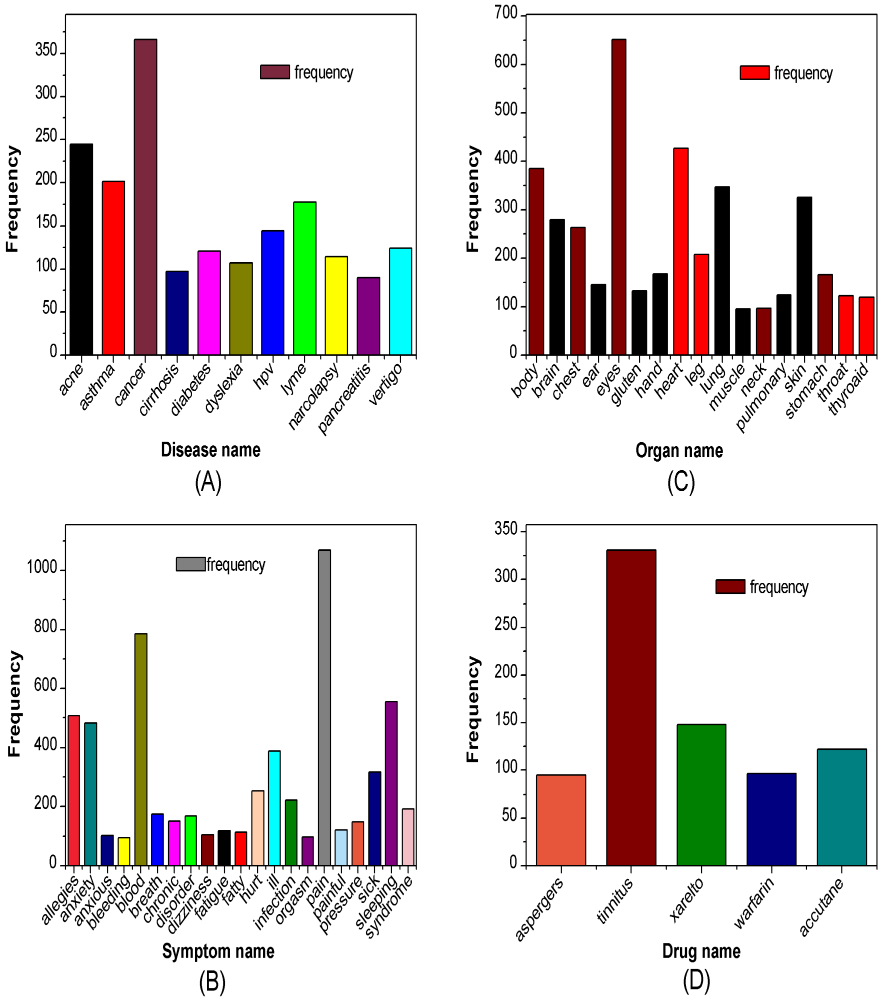


Figure 3.2: Word frequency distribution of disease (A), symptom (B), organ(C) and drug(D) from dailystrength [1] health forum

Text Mining Module

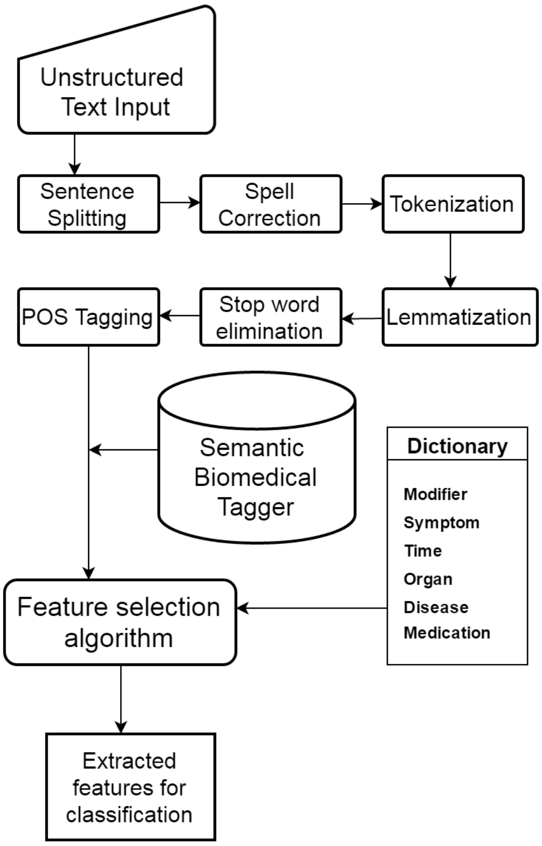


Figure 3.3: Text mining module

Our input is in free text format. The system tags the informative words from this input according to their class attributes through the classification pipeline. To build the feature space for word classification a big scale of text processing task is performed. We used python NLTK package for these natural language processing tasks. As shown in fig. 3.3, first the system does some basic text pre-processing like spell correction, parts of speech tagging, tokenization and lemmatization. Then our feature generation algorithm prepare a feature space for word classification task. To build effective feature space, we use three types of semantic and syntactic features, which are as follows:

1. Semantic bio-medical word tagging

2. Dictionary feature

3. Word feature

Semantic bio-medical word tagging We use Ontotext[24] as one of the tagging features. This is an online API service which provides bio-medical text tagging based on UML[25] database. As for example, headache will be tagged as *Sign or Symptom* by this service.

Dictionary feature As normal user does not use extensive medical terms, there is a need of a dictionary of non-medical terms which are frequently used to describe health problems by users. So we built a dictionary based on the characteristics and language of health forum posts. This dictionary mostly contains of non-medical terms with their corresponding technical mapping, class, id, semantic similarity rating and synonym score. In the identification module, the system generates a numerical mapping of extracted information based on this dictionary feature. To generate dictionary feature, we do not just consider dictionary searching. Beside, we implemented two additional data-structures to enhance overall classification accuracy, which are as follows:

1. Semantic word similarity

2. Synonym mapping

Semantic word similarity: Dictionary tagging is not enough to identify all forms of a word. As for example, a system needs to identify pain, pains, painful as a same word meaning. Word steaming and lemmatization might fail to find these type of variations. To deal with these cases, we use [26] to measure the difference between two words. The [26] between two strings a, b (of length $|a|$and $|b|$ respectively) is given by $lev_{a,b}(|a|, |b|)$ where,

$$lev_{a,b}(i,j) = \begin{cases} max(i,j) & \text{if } min(i,j) = 0 \\ min\begin{cases} lev_{a,b}(i-1,j)+1 \\ lev_{a,b}(i,j-1)+1 \\ lev_{a,b}(i-1,j-1)+1_{a_i \neq b_j} \end{cases} & \text{otherwise} \end{cases} \tag{3.1}$$

where $1_{a_i \neq b_j}$ is equal to 0 when $a_i = b_j$ and equal to 1 otherwise, and $lev_{a,b}(i,j)$ is the distance between the first i characters of a and the first j characters of b.

Besides, [27] based semantic similarity is used to identify the dictionary word root. [27] is a lexical database which provides word similarity measurement based on semantic features.

Synonym mapping: There might exist words which might not be identified by dictionary but have relevant UML tagging. These type of words need to be identified. So for these cases, we assign synonym scores to these words. Then, compare the score with the scores of dictionary words. These synonym scores are defined based on [27] synonym score. Then this unidentified word is inserted into dictionary as the synonym of the most similar dictionary word.

Word feature Additional word features like parts of speech tag, bi-gram, tri-gram and regular expression based replacers, word shape features are used as these features are relevant to natural language processing tasks. The algorithm used for word classification feature generation is given in algorithm 1

Algorithm 1 Feature selection algorithm for word classification

```
function classification_feature($input$)
    $pre\_text \leftarrow$ text_preprocess($input$);
    Let $feature[1 \ldots number\ of\ words\ in\ pre\_text\ ]$ be new array;
    for all sentence in pre_text do
        for all word in sentence do
            $feature[word] \leftarrow$ feature_selection($word$);
        end for
    end for
    return $feature$
end function

function feature_selection($word$)
    Let $feature[1 \ldots number\ of\ feature]$ be new array
    $feature[word\_feature] \leftarrow$ wordfeature($word$);
    $feature[uml] \leftarrow$ semantic_biomedical_tagger($word$);
    $dict\_tagged \leftarrow$ dictionary_tagging($word$) ;
    if $dict\_tagged \leftarrow null$ then
        $synonym \leftarrow$ wordnet.semantic_similarity($word$);
        $dict\_tagged \leftarrow$ dictionary_tagging($synonym$);
    end if
    $feature[dictionary] \leftarrow dict\_tagged$;
    return $feature$
end function

function dictionary_tagging($word$)
    $tagged\_result \leftarrow$ dictionary.lookup($word$);
    if dictionary.lookup($word$)$\leftarrow null$ then
        $tagged\_result \leftarrow$ dictionary.lavenstain_similarity($word$);
    end if
    return $tagged\_result$
end function
```

Text classification

Using prepared feature set, we performing a two phase machine learning based classification for sequential word tagging. In the first phase we use conditional random field(CRF) to detect word phrases by identifying word boundary. Then, we make sequential tagging of all words according to our class attributes. In this phase we use support vector machine(SVM).

CRF CRF is discriminative undirected probabilistic graphical model which takes the neighbouring samples into account to predict the labels.CRF is a structured prediction method where a huge number of variables are dependent on each other.[28] define a CRF on observations X and random variables Y as follows:
Let $G = (V, E)$ be a graph such that

$Y = (Y_\vartheta)_{\vartheta \in V}$ so that Y is indexed by the vertices of G .Then (X, Y) is a conditional random field when the random variables Y_ϑ ,conditioned on X , obey the Markov property with respect to the graph:$p(Y_\vartheta | X, Y_w, w \neq \vartheta) = p(Y_\vartheta | X, Y_w, w \sim \vartheta)$, where $w \sim v$ means that w and v are neighbours in G.

SVM SVM [29] is a widely used supervised machine learning technique for creating feature vector based classifier. Each instance to be classified is represented by a vector of real numbered feature. Training data is used to generate a high-dimensional space that can be divided by a hyperplane between positive and negative instances. New instances are classified by finding their position in the space with respect to the hyperplane.In SVM adata point is considered as a n dimensional vector and we want to whether it can be separated in n-1 dimensional hyperplane or not.If we want to find a separating straight line for a linearly separable set of 2D-points which belong to one of two classes.

According to [30].Let's introduce the notation used to define formally a hyperplane:

$$f(x) = \beta_0 + \beta^T x,$$

where β is known as the weight vector and β_0 as the bias.
The optimal hyperplane can be represented in an infinite number of different ways by scaling of β and β_0. As a matter of convention, among all the possible representations of the hyperplane, the one chosen is

$$|\beta_0 + \beta^T x| = 1$$

where x symbolizes the training examples closest to the hyperplane. In general, the training examples that are closest to the hyperplane are called support vectors. This representation is known as the canonical hyperplane.

Now, we use the result of geometry that gives the distance between a point x and a hyperplane (β, β_0):

$$\mathrm{distance} = \frac{|\beta_0 + \beta^T x|}{||\beta||}$$

In particular, for the canonical hyperplane, the numerator is equal to one and the distance to the support vectors is

$$\mathrm{distance}_{\text{ support vectors}} = \frac{|\beta_0 + \beta^T x|}{||\beta||} = \frac{1}{||\beta||}$$

Recall that the margin introduced in the previous section, here denoted as M, is twice the distance to the closest examples:

$$M = \frac{2}{||\beta||}$$

Finally, the problem of maximizing M is equivalent to the problem of minimizing a function $L(\beta)$ subject to some constraints. The constraints model the requirement for the hyperplane to classify correctly all the training examples x_i. Formally,

$$\min_{\beta,\beta_0} L(\beta) = \frac{1}{2}||\beta||^2 \text{ subject to } y_i(\beta^T x_i + \beta_0) \geq 1 \; \forall i$$

where y_i represents each of the labels of the training examples.

Phase one This phase is used to identify the word phrases by tagging each word using BIO format. We used conditional random field (CRF) method to train a classification model based on [1] health forum dataset. Conditional random field is a popular algorithm mostly used for named entity recognition and word boundary detection. It uses sequential tagging techniques to make the prediction.

Phase two In the second phase of our classification module we used linear support vector machine to tag all words according to the seven attribute class labels: symptom, organ, disease, time, modifier, medication and others. We used the dataset trained by the phase one to build the classifier.

Applying defined rules

After performing two phase word classification, we have all words tagged with relevant attribute names. The next step is to apply some predefined rules on these tagged entities to match against our disease database. This is done in two steps:

1. Numerical mapping of modifiers

2. Matching predefined patterns

3. Symptom mapping

Numerical mapping of modifiers: We know, high, sever and extreme, all these three words are modifier values and express high intensity. According to our system, intensity is measured in three range values: high(3), medium(2) and low(1). So high, sever and extreme these words will be mapped as high(3). In these way, age number and time values will be mapped in a age-group and time-range.

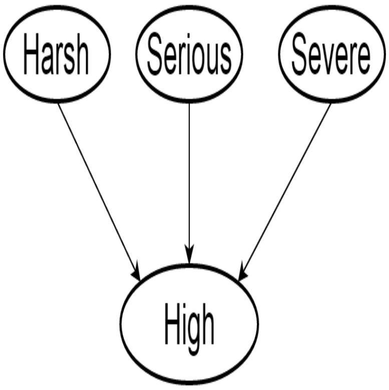


Figure 3.4: Numerical mapping of modifier.

Table 3.2: Defined rules and examples

Defined rules	Examples
Modifier(color) + organ = symptom	Red skin
Modifier(direction) + organ = organ	Left shoulder
Modifier(direction) + symptom = symptom	Back pain
Lab test + modifier = test result	Blood test report is fine
Modifier(number) + ’years’ + ’old’ = age	15 years old (age = 15)
Modifier(negation + normal activity) = symptom	Unable to walk, painful breathing

Matching predefined patterns: In this step, tagged attributes are mapped following some predefined patterns and . Some of the patterns with example are shown in table 3.2.

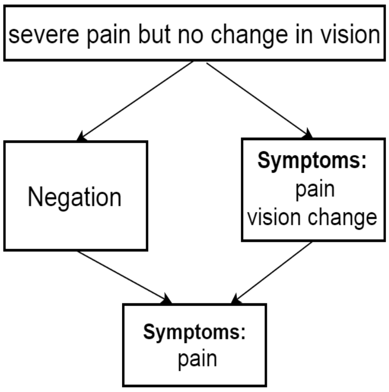


Figure 3.5: Symptom mapping.

Symptom mapping: The last step is symptom mapping. A synonym can be in a phrase and this phrase can express different meaning from the individual words meaning of the phrase. Synonym mapping is needed in this case. As for example, If there is a symptom name vision change with negation word not before that, then vision change will not be considered as a symptom.

A complete breakdown example of a sample input using our information extraction module is shown in fig. 3.6.

After performing entity tagging, now we have a formatted input text where each entity is associated with relevant tag. The next step is identification part which is done in identification module.

Free text format input:
Hello, I'm 24 years old. For sme days I have fever with moderate shaking chills. Sometimes the fever goes up to 106 degree.. I have also diarrhoea and muscle pain.
Lower case conversion, replacing short form, spell correction:
hello, i am 24 years old. for some days i have fever with moderate shaking chills. sometimes the fever goes up to 106 degree.I have also diarrhoea and muscle pain.
Text preprocessing:
(be, 24, years, old) (some, days, have, fever, moderate, shake, chill) (sometimes, have, fever, goes up, 106, degree) (have, diarrhoea, muscle, pain)
BIO tagging:
(be-B, 24 -B, years -I, old-I) (some-B, days-I, have-B, fever-B, moderate-B, shake-I, chill-B) (sometimes-B, have-B, fever-B, goes-B up-I, 106-B, degree-I) (have-B, diarrhoea-B, muscle-B, pain-I)
Word tagging:
(be-B others, 24 –B modifier, years –I time , old-I modifier) (some-B modifier, days-I time, have-B others, fever-B symptom, moderate-B modifier, shake-I symptom, chill-B symptom) (sometimes-B time, have-B others, fever-B symptom, goes-B others, up-I modifier, 106-B modifier, degree-I modifier) (have-B others, diarrhoea-B disease, muscle-B organ, pain-I symptom)
Applying defined rules:
Sentence1 - Age: 24 [rule: modifier (Intensity: medium(2) [intensity mapping]
Sentence3 - Disease: diarrhoea Organ: muscle Symptom: pain

Figure 3.6: A sample input processing performed in information extraction module

3.1.2 Identification module

Disease database

In this module we match the output of information extraction module against a disease database. This database is prepared based on recorded disease-symptom associations from expert sources [10],[31],[12]. In the database, each disease case is described based on its symptom and other relevant attributes described in table 4.1. Here, each disease case is a standard template which integrates two components: disease-document matrix and disease-data matrix.

Disease-document matrix Disease-document matrix reflects the behaviour of unstructured user input. As we are identifying diseases from queries which are similar to the health forum posts, we need to consider the behaviour of these forum posts. So each disease object in our database has some predefined sample input cases. These input cases are in a text representation where each word is associated with entity type which is defined based on our classification model. From these formatted text cases, our identification module prepares $tfidf$ [32] disease-document matrices. $tfidf$ is a descriptive method based on database vocabulary feature space. Here the product of two factors $term\ frequency(tf)$ and $inverse\ term\ frequency(idf)$ is calculated. Whereas tf considers the frequency of a term in a document, idf assigns greater weight value for terms which are rare with respect to all documents. If in a disease database D, a disease object d has formatted text representation, then for each $d \in D$ we can calculate $tfidf(d)$ as follows:

$$
\begin{aligned}
tfidf(d) &= \sum_{t \in q \cap d} tfidf_{t,d} \\
&= \sum_{t \in q \cap d} tf_{t,d} \times idf_{t,d} \\
&= \sum_{t \in q \cap d} tf_{t,d} \times log_{10} \frac{N}{df_t}
\end{aligned}
\tag{3.2}
$$

$$
\begin{aligned}
&here, \\
tf_{t,d} &= number\ of\ occurrences\ of\ term\ t\ in\ disease\ case\ d \\
df_t &= number\ of\ disease\ case\ containing\ term\ t \\
N &= total\ number\ of\ disease\ cases
\end{aligned}
$$

Disease-data matrix Besides user input behaviour, another important fact is entity-entity association, for which identification module needs to consider word sequence ordering which is not performed in the bag-of-word representation of *tf–idf* matrix. As for example, two text "pain in muscle and weak feeling", "weak muscle and pain feeling" are same text in *tf–idf* representation. So association between entities: (pain, muscle),(weak,muscle) is lost. In our disease database, each disease object has a data-matrix component where related entities are associated with each other. In [22], the authors used this type of measurement based on five attributes which are: symptom(s), time(t), duration(d), intensity(i) and organ(o). In our implemented data-matrix, each row can define a symptom and it's related class attributes described in table 4.1. If a row[0] element defines a symptom, then other elements of that row are time, related organ, intensity and color which are related to that symptom. As for example, [33] and [34] are two diseases and their corresponding data-matrix representation $dm_{eye-redness}$ as

$dm_{conjunctivitis}$ are as follows:

$$\begin{pmatrix}
S[0] = redcolor & T[0] = * & I[0] = * & O[0] = eye \\
S[1] = headache & T[1] = * & I[1] = * & O[1] = * \\
S[2] = visionchange & T[2] = * & I[2] = * & O[2] = eye \\
S[3] = pain & T[3] = * & I[3] = high & O[3] = eye \\
S[4] = nausa & T[4] = * & I[4] = * & O[4] = * \\
S[5] = vomitimg & T[5] = * & I[5] = * & O[5] = * \\
Agegroupe = * & Medication = * & Relateddisease = * & Food = * \\
Gender = * & \times & \times & \times
\end{pmatrix}$$

Figure 3.7: Data-matrix $dm_{eye-redness}$

$$\begin{pmatrix}
S[0] = grittyfeeling & T[0] = * & I[0] = * & O[0] = eye \\
S[1] = itchiness & T[1] = * & I[1] = * & O[1] = eye \\
S[2] = tear & T[2] = * & I[2] = high & O[2] = eye \\
S[3] = thickdischarge & T[3] = night & I[3] = * & O[3] = eye \\
Agegroupe = * & Medication = * & Relateddisease = * & Food = * \\
Gender = * & \times & \times & \times
\end{pmatrix}$$

Figure 3.8: Data-matrix $dm_{conjunctivitis}$

Similarity measurement

The main work of identification module is to generate a list of probable diseases based on user input. To do this, identification module create query-document matrix and query-data matrix representation of input text which are used to measure similarity with disease database objects. SO, if a user query input is u, then it's corresponding document matrix and data-matrix can be written as $tfidf(u)$ and d_{query}. Next, our proposed identification module calculates similarity between these user input matrices and disease object components from database to identify probable diseases.

At first, we can consider the case of similarity measurement between $tfidf(u)$ and $tfidf(d)$ for all $d \in D$. In this part, our identification module calculate *cosinesimilarity* [35] to find out most similar disease-document matrices from database. To do this, from $tfidf(u)$ and $tfidf(d)$, we have to generate vector space model where each component of a vector is the $tfidf$ value of a specific term in the database term dictionary. For user input u and disease object d corresponding vector representations are $\vec{V}(u)$and $\vec{V}(d)$. The cosine similarity between these vectors is calculated as follows:

$$Similarity_{cosine}(\vec{V}(u), \vec{V}(d)) = \frac{\vec{V}(u).\vec{V}(d)}{|\vec{V}(u)|.|\vec{V}(d)|} \ where \ d \in D \tag{3.3}$$

Besides document matrices, similarity between data-matrices needs to be calculated. After extracting relevant attributes from user input u, we get the feature space f for query-data matrix dm_{query} generation. Here, $f = [s \cup t \cup i \cup o \cup \ldots agegroup]$ where $s, t, i, o, \ldots agegroup$ represents extracted symptom, numerically mapped and normalized values of time and intensity, organ and other attributes described in table 4.1. This numerical mapping is done based on dictionary id, similarity score, synonym score and intensity rating of extracted attributes. Then min-max normalization scale the values as follows:

$$i = \frac{i' - min(i)}{max(i) - min(i)} \tag{3.4}$$

here i' is the calculated score of a symptom intensity, $max(i)$ is the maximum intensity rating and $min(i)$ is the minimum score. Then these values are mapped in the query-data matrix dm_{query}. $Jaccard\ coefficient$ [35] is a similarity measurement method which is used to calculate similarity between dm_{query} and dm_d for all $d \in D$. $Jaccard\ coefficient$ between dm_{query} and dm_d is calculated as follows:

$$Coefficient_{jaccard}(dm_{query}, dm_d) = \frac{a}{a+b+c} \tag{3.5}$$

Where,

$d \in D$

$a =$ number of attributes $\in$ both objects.

$b =$ number of attributes $\in dm_{query}$ but not in dm_d.

$c =$ number of attributes $\in dm_d$ but not in dm_{query}.

For better understanding of this part, a user input can be considered – *"I'm 23 years old male. For three days I'm facing eye problem. A lot of tears with itchiness feeling. I have headache also though there is no change in vision."* The query-data matrix of this input dm_{query} can be visualized as follows:

$$\begin{pmatrix} S[0] = tear & T[0] =< 1week & I[0] = high & O[0] = eye \\ S[1] = itchiness & T[1] = * & I[1] = * & O[1] = * \\ S[2] = headache & T[2] = * & I[2] = * & O[2] = * \\ Agegroupe = 2 & Medication = * & Relateddisease = * & Food = * \\ Gender = male & \times & \times & \times \end{pmatrix}$$

Figure 3.9: Data-matrix representation of user query

Now if we calculate $jaccard\ coefficient$ of dm_{query} and disease-data matrices: $dm_{conjunctivitis}$ and $dm_{eye-redness}$, we get as follows:

Table 3.3: Jaccard coefficient measurement

Entity name	dm_{query}	$dm_{eye-redness}$	$dm_{conjunctivitis}$
RedColor		1	
Eye(red color)		1	
Headache	1	1	
Vision change		1	
Eye (vision change)		1	
Pain		1	
High (pain)		1	
Eye (pain)		1	
Nausa		1	
Vomiting		1	
Gritty feeling			1
Eye (gritty feeling)			1
Itchiness			1
Eye (itchiness)			1
Tear	1		1
Eye (tear)	1		1
High (tear)	1		1
Thick discharge			1
High (thick discharge)			1
0-10 days (tear)	1		
Age Groupe = 2	1		
Gender = male	1		

$$\begin{aligned} Coefficient_{jaccard}(dm_{query}, dm_{eye-redness}) &= \frac{1}{1+6+9} = 0.0625 \\ Coefficient_{jaccard}(dm_{query}, dm_{conjunctivitis}) &= \frac{3}{3+4+5} = 0.2307 \end{aligned} \tag{3.6}$$

From the calculated result, it is clear that $dm_{conjunctivitis}$ is more similar with dm_{query} than $dm_{eye-redness}$. So, according to data matrix similarity measurement, the user has more chance of having *conjunctivitis* than *eye-redness*.

Finally, identification module calculates a weighted sum of previously calculated *jaccard coefficient* and *cosine similarity* where *jaccard coefficient* has greater weight value.

$$\begin{aligned} Similarity(u, d) = {} & w_{cs} \times Coefficient_{jaccard}(u, d) \\ & + (1 - w_{cs}) \times Similarity_{cosine}(\vec{V}(u), \vec{V}(d)) \\ & \qquad where,\ w_{cs} < 0.5\ and\ d \in D \end{aligned} \tag{3.7}$$

Based on this similarity measurement, identification module suggests a list of most probable diseases articulating probability group for each output which can be high or low.

Chapter 4

Experimental evaluation

4.1 Data

For evaluation of information extraction module, we use data from dailystrength heath forum. We use 196 forum posts from 10 different disease related groups. These posts contain 4915 relevant informative words which we manually tagged into seven classes: medication, organ, disease, time, symptom, modifier and others. Besides, for phrase detection this dataset is also tagged according to BIO format.To evaluate identification module of our proposed model, we collected identification data for our experimental test cases from online symptom checker site [8] .

4.2 Baselines

We use two baseline methods to evaluate the performance of svm classifier used in information extraction module, in our experimental settings. At first, we use a naive baseline which classifies simply relaying on the class distribution of training data. Then,we perform our experiment using svm classifier with varying components. In identification module, we use disease-symptom associations recorded in [31], [12], [10] as ground truth values. In this setting, we split our data-set as

70-30 as training and testing data. Then again, we perform cross-validation on the same data set.

4.3 Experiments

4.3.1 Setting I: Information extraction

Our information extraction is a two-phase module. Phase one performs phrase detection for which, conditional random field(CRF) is used. Then we perform phase two experiment using our baseline methods and SVM classifier. We use sqlite database for storing dictionary values and python NLTK library package to perform natural language processing tasks in feature selection stage.

4.3.2 Setting II: Disease identification

Based on the extracted information in setting I, we perform experiment to evaluate the performance of disease identification. Extracted information in phase one is used as input for identification module. Also, we use these extracted information as a base to give input for online symptom checker [8].Then compare all outputs in respect to ground truth values recorded from [31], [12], [10]. In our model, output is presented as a ranking of probable diseases in two clusters where one cluster presents diseases with high probability (H) and the other one is of low probability (L) disease. This is because, there can be many common symptoms result in different diseases. For example, we can consider for input case I, predicted output is a ranking of disease $D1 - D4$ and their cluster ids are as follows:

Table 4.1: Comparison between truth value and disease identification system

Disease	Truth value	identification system
D1	H	L
D2	H	H
D3	L	L

We perform total 10 experiments, each time based on different type of disease where significance of demographic information is not considered. Then, we perform same experiments considering demographic information.

4.4 Evaluation Metrices

The standard quality measurement of a machine learning based classifier is $accuracy : (tp + tn)/(tp + fn + tn + fp)$. Besides, classification record is generated which contains $precision : tp/(tp + fp)$, $recall : tp/(tp + fn)$, $f - score$ and *support value*. $f - score$ can be seen as a weighted harmonic mean of $precision$ and $recall$.

For evaluation of identification module, we compare obtained result with recorded ground truth. If for disease $D - n$, truth value and predicted value are same then it is 1, otherwise calculate it as 0. In this way, we can find out accuracy by computing the ratio of cumulative match factor and total number of disease.

$$\begin{aligned} Accuracy &= cm/N \\ Where,\ cm &= cumulative\ match\ factor \\ N &= total\ number\ of\ diseases \end{aligned} \tag{4.1}$$

For the example shown in table 4.1,

identification system accuracy = 2/3 = 66.67%

4.5 Results and discussions

4.5.1 Setting I

Table 4.2: Results of phase one classification

Class	precision	recall	F1-score	support
B	.920	.979	.949	877
I	.778	.457	.575	138
Avg/total	.900	.908	.898	1015

Table 4.3: Results of phase two classification

SVM Classifier

Iteration	Precision	Recall	F-Score	Support
1	0.926	0.927	0.926	986
2	0.935	0.936	0.935	984
3	0.914	0.915	0.913	984
4	0.938	0.939	0.938	982
5	0.935	0.934	0.934	979

At first, we present the result of our implemented information extraction part. Classification report of phrase detection part is shown in table 4.2. We obtained moderate precision and low recall in this phase.

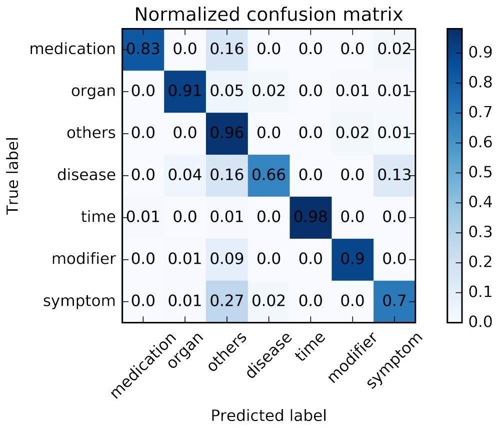


Figure 4.1: Confusion Matrix for phase two classification

In table 4.3, report of word classification using SVM classifier is presented. For this part, we perform 5-fold cross validation results in average precision of .9296. From the confusion matrix as shown in fig. 4.1, we can see that class time results in highest .98 predicted:true label ratio.

Table 4.4: Accuracy comparison in setting I for varying components

Iteration	**Naive baseline**	**SVM**		
		Without dictionary feature	**Without biomedical tagger**	**With all components**
1	56.5	75.1	79.9	92.7
2	58.6	76.5	81.1	92.4
3	56.6	76.7	80.8	93.4
4	58.5	741	82.1	95.0
5	58.1	76.2	80.3	92.9

A detailed report of accuracy comparison in setting I is presented in table 4.4. We perform 5-fold cross validation experiments and compare our result with baseline methods. At first, we use a simple naive baseline results in 57.66% average accuracy. Then we perform same experiments with varying components using SVM classifier with linear kernel. Including bio medical tagger feature results in 75.22% accuracy whereas, including dictionary feature shows 5.12% improvement. After incorporating all these features, we achieve an average accuracy improvement of 12.44%.

4.5.2 Setting II

Table 4.5: Accuracy comparison of disease identification module

Experiment no	Experiment type based on disease category [36],[37]	Symptom checker [8]	Our model	
			Without demographic information	With demographic information
1	Raspiratory Tract diseases: Nose and respiration disorder	64.34	71.54	71.54
2	Chronic diseases: Chrones's disease Alzehimer disease	59.27	61.63	64.32
3	Virus: Fatigue Sexually transmitted diseases	70.58	75.52	75.52
4	Nervous system: Restless legs Anxiety Sleep wake disorder	72.33	77.62	77.62
5	Bone diseases Osteoporosis Arthritis	70.32	74.56	78.56
6	Female Urogenital Diseases Pregnancy Complications	74.11	75.6	77.6
7	Male Urogenital Diseases	65.3	68.12	71.23
8	Cancer	61.89	63.67	63.67
9	Heart diseases	65.63	70.32	70.32
10	Occupational diseases: Asthma Pneumoconiosis	71.23	82.45	82.45

In table 4.5, we presented the accuracy comparison of our identification module with online symptom checker. Here, we compared our result with the result obtained from existed web symptom checker site. In all 10 experiments, our system results in better identification. When we do not consider demographic information, it is 4.603% accuracy improvement whereas, considering demographic information results in slightly better performance with 5.783% accuracy improvement.

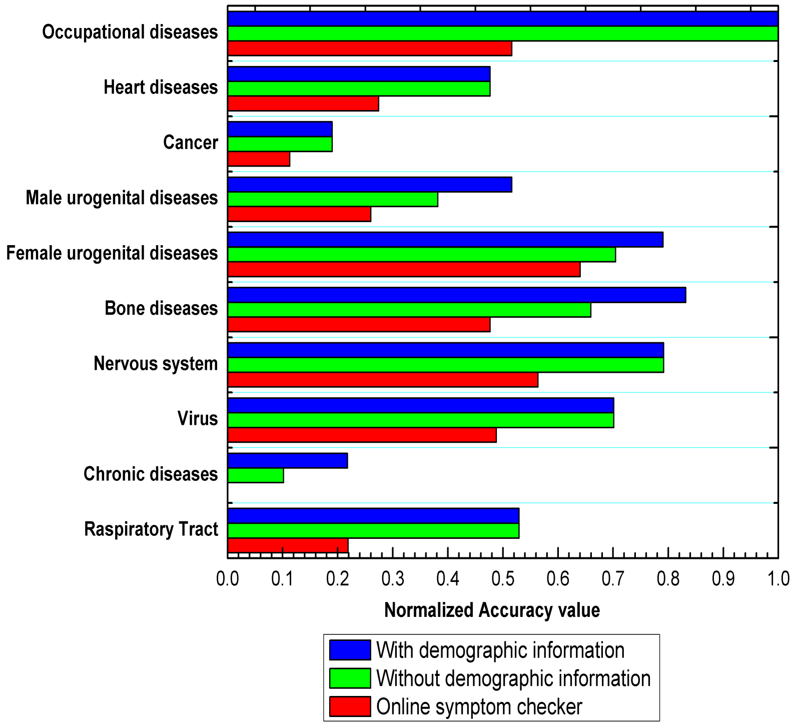


Figure 4.2: Accuracy comparison of disease identification models

In fig. 4.2, a normalized graph of comparison result is presented. We can see that, highest accuracy is achieved in case of identifying occupational diseases: asthma, pneumoconiosis etc. Whereas, identification of chronic diseases and cancer results in lower identification accuracy. Significance of demographic information is present in cases of age related bone diseases and urogenital diseases.

From this discussion, it is clear that the accuracy of our proposed disease identification system is significantly better than existed symptom checker. Besides, incorporating demographic information is useful in specific cases.

Chapter 5

Conclusion and future works

As the accuracy of the work is not far beyond the previous work so our future work includes the implementation of Bayesian network for identification module to achieve higher accuracy and improvement of database to correctly identify the word features. Besides, information extraction accuracy is another aspect of our future work.

Bibliography

[1] www.dailystrength.org.

[2] J. A. Barnett, Michael L.Linder, "Antibiotic prescribing to adults with sore throat in the united states, 1997-2010," *JAMA Internal Medicine*, vol. 174, no. 1, p. 138, 2014.

[3] http://edition.cnn.com/2012/05/04/tech/social-media/facebook-lies-privacy/.

[4] M. R. N. d. K. Elske Ammenwerth, Pirkko Nykänen, "Clinical decision support systems: Need for evidence, need for evaluation," *Artificial Intelligence in Medicine*, vol. 59, pp. 1–3, sep 2013.

[5] Velardi and et al, "Twitter mining for fine-grained syndromic surveillance," *Artificial Intelligence in Medicine*, 2014. http://dx.doi.org/10.1016/j.artmed.2014.01.002.

[6] www.healthdirect.gov.au/symptom-checker.

[7] www.patient.info/symptom-checker.

[8] www.isabelhealthcare.com.

[9] www.everydayhealth.com/symptom-checker.

[10] www.mayoclinic.org/symptom-checker.

[11] www.nhs.uk/symptom-checker.

[12] www.symptomchecker.webmd.com.

[13] M. Bundschus, M. Dejori, M. Stetter, V. Tresp, and H.-P. Kriegel, "Extraction of semantic biomedical relations from text using conditional random fields," *BMC Bioinformatics*, vol. 9, no. 1, p. 207, 2008.

[14] P. R. B. L. S. M. Ginsberg J, Mohebbi MH and et al, "Detecting influenza epidemics using search engine query data," *Nature*, 2009. 457(7232):1012-4.

[15] C. D.-N.-M. Subhabrata Mukherjee, Gerhard Weikum, "People on drugs: credibility of user statements in health communities," *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 65–74, 2014.

[16] M. Roccetti, A. Casari, and G. Marfia, "Inside chronic autoimmune disease communities," in *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015 - ASONAM '15*, Association for Computing Machinery (ACM), 2015.

[17] S. N. A. . G. Z. . J. L. . D. W. Q. Zhang ; Centre for Quantum Comput. & Intell. Syst., Univ. of Technol. Sydney, "A framework of hybrid recommender system for personalized clinical prescription," in *Intelligent Systems and Knowledge Engineering (ISKE), 2015 10th International Conference on 24-27 Nov*, pp. 189 – 195, IEEE, 2015.

[18] H. Li, X. Li, M. Ramanathan, and A. Zhang, "Prediction and informative risk factor selection of bone diseases," *IEEE/ACM Trans. Comput. Biol. and Bioinf.*, vol. 12, pp. 79–91, jan 2015.

[19] R. Y. Q. Bullard, Joseph; Murde and C. O. Alm, "Inference from structred and unstructured electronic medical data for early dementia detection," 2015. [Accessed from http://scholarworks.rit.edu/other/830].

[20] N. U. . K. N. . R. J. B. . J. H. Y. Wang ; IBM T. J. Watson Res. Center, Yorktown Heights, "Early detection of heart failure with varying prediction windows by structured and unstructured data in electronic health records," in *Proceeding of Engineering in Medicine and Biology Society (EMBC), 37th Annual International Conference of the IEEE*, pp. 2530 – 2533, IEEE, aug 2015.

[21] H. L. Semigran, J. A. Linder, C. Gidengil, and A. Mehrotra, "Evaluation of symptom checkers for self diagnosis and triage: audit study," *BMJ*, p. h3480, jul 2015.

[22] A. R. M. K. N. R. Md. Tahmid Rahman Laskar, Md. Tahmid Hossain, "Automated disease prediction system (adps): A user input-based reliable architecture for disease prediction," *International Journal of Computer Applications*, vol. 133, no. 15, 2016.

[23] P. Shrestha, N. Rey-Villamizar, F. Sadeque, T. Pedersen, S. Bethard, and T. Solorio, "Age and gender prediction on health forum data," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)* (N. C. C. Chair), K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, eds.), (Paris, France), European Language Resources Association (ELRA), may 2016.

[24] www.ontotext.com.

[25] www.nlm.nih.gov/research/umls/licensedcontent/umlsknowledgesources.html.

[26] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," in *Soviet physics doklady*, vol. 10, p. 707, 1966.

[27] wordnet.princeton.edu.

[28] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the eighteenth international conference on machine learning, ICML*, vol. 1, pp. 282–289, 2001.

[29] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[30] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*, vol. 1. Springer series in statistics Springer, Berlin, 2001.

[31] P. Ernst, C. Meng, A. Siu, and G. Weikum, "Knowlife: a knowledge graph for health and life sciences," in *2014 IEEE 30th International Conference on Data Engineering*, pp. 1254–1257, IEEE, 2014.

[32] J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of massive datasets.* Cambridge University Press, 2014.

[33] www.healthline.com/symptom/eye-redness.

[34] www.healthline.com/symptom/conjunctivitis.

[35] J. Han, J. Pei, and M. Kamber, *Data mining: concepts and techniques.* Elsevier, 2011.

[36] www.ncbi.nlm.nih.gov/mesh/1000067.

[37] www.rightdiagnosis.com/a/all/subtypes.html.

[38] P. P. U. . Y. A. . X. H. Y. Ling ; Coll. of Comput. & Inf., Drexel Univ., "A matching framework for modeling symptom and medication relationships from clinical notes," in *Bioinformatics and Biomedicine (BIBM), 2014 IEEE International Conference on 2-5 Nov*, pp. 515 – 520, IEEE, 2014.

[39] E. I. . P. T. . B. M. Thangamani ; Dept. of Comput. Sci. & Eng., Kongu Eng. Coll., "Automatic medical disease treatment system using datamining," in *Information Communication and Embedded Systems (ICICES), 2013 International Conference on 21-22 Feb*, pp. 120 – 125, IEEE, 2013.

[40] D. S. K. . S. H. M. M. Ko ; Div. of Web Sci. & Technol., Korea Adv. Inst. of Sci. & Technol., "Identifying disease definitions with a correlation kernel for symptom extractions from text," in *Healthcare Informatics (ICHI), 2014 IEEE International Conference on 15-17 Sept*, pp. 320 – 327, IEEE, 2014.

[41] patient.info/forums/discuss/gastritis-constant-severe-stomach-pain-41570.