

Extraction of Disease Relationship from Medical Records: Vector Based Approach

Ayisha Noori V. K

*Department of Computer Science and Engineering,
Govt. Engg. College, Sreekrishnapuram,
Palakkad, India-678633*

P. C. Reghu Raj

*Department of Computer Science and Engineering,
Govt. Engg. College, Sreekrishnapuram,
Palakkad, India-678633*

Abstract- Biomedical natural language processing deals with the application of text mining techniques to clinical documents and to scientific publications in the areas of biology and medicine. A crucial area of Natural Language Processing is semantic analysis, the study of the meaning of linguistic utterances. Natural language processing of biomedical text benefits from the ability to recognize broad semantic classes from different clinical notes.

This paper proposes a method that extract semantics from medical discharge summaries using vector based approach. In particular, this is concerned with the identification of relationships between different diseases from diabetic discharge summaries and enlist the list of possible diseases that the patient may encounter. Vector dot product method is used to identify the semantic relationship between diseases.

Keywords – Biomedical natural language processing, Semantic Analysis, Vector dot product method

I. INTRODUCTION

Patient Data is critical in healthcare domain and it should be secure, consistent and coded for the secure transfer from one potential user to another. Since its inception in the 1800s, the narrative patient record has been a goldmine of information for members of the medical community. For the physician, medical records chronicle previous diagnoses and allergic reactions of a patient, providing continuity of information as the patient is transferred from one doctor to another. For researchers interested in studying various clinical trends, discharge summaries provide a large archive of data. Furthermore, clinical records can be used to generate statistical models that, for example, predict patient outcome given clinical findings, provide diagnoses given symptoms, and identify epidemic outbreaks on the basis of regional data[19].

This work comes under Biomedical NLP (also known as BioNLP) which refers to natural language processing techniques applied to texts and literature of the biomedical and molecular biology domain. It is a rather recent research field on the edge of natural language processing, bioinformatics, medical informatics and computational linguistics. Often, existing NLP solutions are less successful in the biomedical domain relative to their non-biomedical domain performance (e.g., relative to newspaper text).

In this paper, we tackle the challenge of "understanding" patient records. We describe an approach to extract salient information from natural language medical text and create appropriate representations of this information. We focus on medical discharge summaries which are documents that a physician dictates after a patient's hospital stay, highlighting the patient's reasons for hospitalization, test results and findings, diagnoses, and prescribed medications. This work mainly contains two submodules : Training phase performs section identification of discharge summaries in which different segments like diagnosis, symptoms and investigations are identified. Then it extracts related diseases based on the findings from training using cosine similarity measure.

II. RELATED WORK

A supervised machine learning approach to discover relations between medical problems, treatments, and tests mentioned in electronic medical records has been introduced by Bryan Rink et al. They have developed a state of the art method that automatically extracts relations between medical concepts[3]. Kyung-Mi Park et al. made an analysis of how much each different NLP technique improves the performance of Biomedical Interaction Extraction. Their paper[8] devised a two-step method: 1) an interaction verb extraction step to find biomedically salient verbs, and 2) an argument relation identification step to generate partial predicate-argument structures between extracted interaction verbs and their NE arguments. Oana Frunza and Diana Inkpen describes their study on identifying semantic relations that exist between diseases and treatments in biomedical sentences[12]. Ivelina Nikolova and Galia Angelova present a research work enabling automatic extraction of relations between medical concepts[7]. "Automatic Identification of Treatment Relations For Medical Ontology Learning: An Exploratory Study" is a Manuscript submitted to ISKO 2004 by Chew-Hung Lee et al. This is part of a project to develop an automatic method to build ontologies, especially in a medical domain, from a document collection[4]. Dimitrios Kokkinakis worked on Medication Event Extraction Using Frame Semantics for Centre for Language Technology; the Swedish Language Bank[5]. Another work for Identifying Semantic Relations for Disease-Treatment in Medline carried out by P. Menaka and Prof.D.Thilagavathy. The main objective of this work is to show what Natural Language Processing (NLP) and Machine Learning (ML) techniques used for representation of information and what classification algorithms are suitable for identifying and classifying relevant medical information in short texts[13]. This work focuses on extracting relation between diseases based on certain feature values. Diabetic discharge summaries are used as dataset. So this work automatically identifies the possible diseases that the diabetic patient may encounter, and extracts certain feature values. Discharge summary collection is split into training set and test set. From the training set, the system extracts the relevant information which is later used to find out the related diseases for test set.

III. BACKGROUND

This section explains some fundamental theories used to develop this system.

A. Information Extraction (IE)

Information extraction (IE) is the process of scanning text for information relevant to some interest, including extracting entities, relations, and, most challenging events or who did what to whom, when, and where. It requires deeper analysis than keyword searches, but its aims fall short of the very hard and long-term problem of text understanding, where we seek to capture all the information in a text, along with the speaker's or writer's intention. IE typically focuses on surface linguistic phenomena that do not require deep inference, and it focuses on the phenomena that are most frequent in texts. Fig. 1 shows the overall structure of information extraction process considering a human-centered view of knowledge discovery[10].

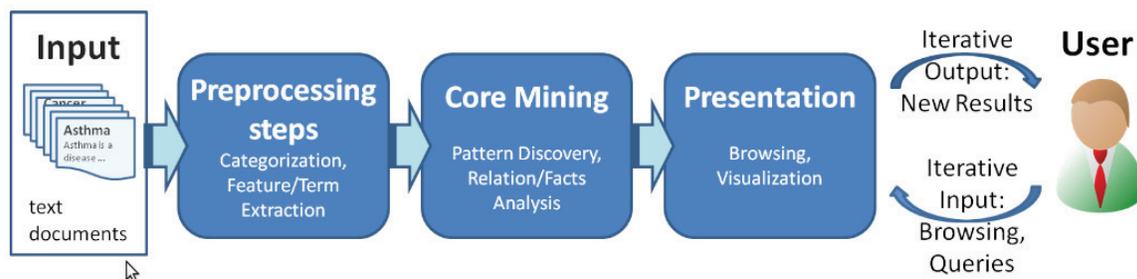


Fig 1. Information Extraction

B. BioNLP

BioNLP, also known as biomedical language processing or biomedical text mining, is the application of natural language processing techniques to biomedical data. The biomedical domain presents a number of unique data types and tasks, but simultaneously has many aspects that are of interest to the "mainstream" natural language processing community. Additionally, there are ethical issues in BioNLP that necessitate an attention to software

quality assurance beyond the normal attention (or lack thereof) that is paid to it in the mainstream academic NLP community. An embarrassing fact about BioNLP research to date is this: despite the fact that there has been a very large amount of work in the area, most deployed systems have been built not by people who self-identify as natural language processing specialists but by people who self-identify as biologists.

C. NLP Techniques

NLP techniques used in this work are explained here in detail:

1) Section Segmentation

Hospital discharge summaries serve as the primary documents communicating a patient's care plan to the post-hospital care team. It contain different sections like: Personal Information, Diagnosis, Case Summary, Investigations, Observations, Tests, Medication, Treatment, Cross Consultation etc. Section segmentation identifies the necessary sections using regular expression module and line segmentation algorithms. In this work, we are taken diagnosis, Case summary(which contains symptoms), and Investigations. In order to extract features from the discharge summary, these segments should be separated. Using this work, it is possible to extract symptoms , diagnosis and selected feature values from each discharge summary.

2) Tokenization

Tokenization is the task of cutting a string into identifiable linguistic units that constitute a piece of language data. Here data from some segments are tokenized to get necessary information. These individual tokens are examined to identify whether it is a disease, medicine or any unwanted information etc. Many bio-medical tokenizers are there for the automatic identification biological terms or diseases. But in this work, nltk tokenizer is used.

3) Line and Sentence segmentation

Sentence segmentation is the problem of dividing a string of written language into its component sentences. In English and some other languages, using punctuation, particularly the full stop character is a reasonable approximation. However even in English this problem is not trivial due to the use of the full stop character for abbreviations, which may or may not also terminate a sentence. Here sentence segmentation is used to identify symptoms from the discharge summary and line segmentation is used to identify diagnosis and also for feature vector formation.

4) Identification

Chunks are non-overlapping regions of text. Each chunk contains a head, with the possible addition of some preceding function words and modifiers. Noun phrase chunks are nouns having more than one tokens. Discharge summary contains many noun chunks like hyper glycemia, Chronic Kidney Disease etc. System should identify this as a single disease or entity. For this standard medical databases which contain disease names and details are available(SNOMED-CT, UMLS etc).

D. Similarity Measurement : Vector dot product

Vector dot product approach is used to identify the semantic relationship among diseases. The input features are collected and converted to vector form. Similarly from each discharge summary, a feature vector is formed. The dot product of input vector and each of this feature vector is calculated to find the similarity.

Cosine similarity is a measure of similarity between two vectors of an inner product space that measures the cosine of the angle between them. It is a judgement of orientation and not magnitude: two vectors with the same orientation have a Cosine similarity of 1, two vectors at 90° have a similarity of 0, and two vectors diametrically opposed have a similarity of -1, independent of their magnitude[28]. Vector dot product can be calculated using the following equation.

For two n-component vectors a and b, the angle between them can be defined by :

$$\mathbf{A} \cdot \mathbf{B} = AB \cos(\theta)$$

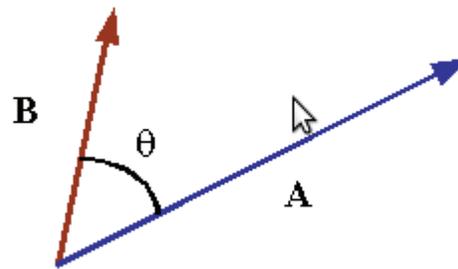


Fig 2.Vector

$$\cos\theta = \frac{a \cdot b}{\|a\| * \|b\|} \quad (1)$$

$$(a \cdot b) = a_1 * b_1 + a_2 * b_2 + \dots + a_n * b_n \quad (2)$$

$$\|a\| = \sqrt{a_1^2 + a_2^2 + \dots + a_n^2} \quad (3)$$

In this work we select the discharge summary vector with least angle values with the input feature vector and displays the diagnosis(diseases) of the corresponding vector as the related disease set.

IV. DATA PREPARATION

500 discharge summaries are collected from the Diabetic department and nearly 3000 discharge summaries of General Medicine section are collected from two different hospitals in Calicut. Among this, we selected diabetics as the working domain 500 diabetic discharge summaries are taken as data set. Among them, 150 discharge summaries are selected as test set and rest are kept as training set.

V. FEATURE SELECTION

At this step, rich feature sets are selected to encode the necessary informations from each discharge summary. In this work, features are taken from the 'Investigation' section of discharge summary, which contains values of different test results of a patient at the time of admission in the hospital. These values indicates the internal condition of the patient, which is a most important information for patient's healthcare. For this work, 10 features are selected after discussing with a Medical Practitioner. These values are typically important parameters indicating the health condition of a diabetic patient. The variations of these feature values gives indication of various diseases for a diabetic patient. The selected feature values are:

1.Gender, 2.Sugar, 3.Blood Pressure, 4.HemoglobinA1C, 5.S.Creatine, 6.ESR, 7.Pottassium, 8.Sodium, 9.BI.Urea, 10.LDL.

Since normal range of many features are different for males and females, gender is important for identifying semantic relation from discharge summary. The dataset used here is the diabetic discharge summaries. So Sugar value should be included in our feature vector. Fasting sugar is taken here as sugar value. The normal values are given below:

Value	Gender	Normal
Sugar	M & F	70-120
BP	M & F	60-80/90-120
HbA1c	M & F	4-5.6
S.creatinine	M	0.7-1.2
S.creatinine	F	0.5-1.0
Blood Urea	M	8-24
Blood Urea	F	6-21
Pottasium	M & F	3.5-5.0
Sodium	M & F	136-145
ESR	M	0-17
ESR	F	1-25
LDL	M & F	<100

Table 1. Normal Value

VI. TRAINING PHASE

E. Discharge Summary Mapping

This section explains different steps of discharge summary mapping in detail. There are mainly three different steps : Data preprocessing, Section segmentation and Feature vector generation. Fig.3 explains the discharge summary mapping.

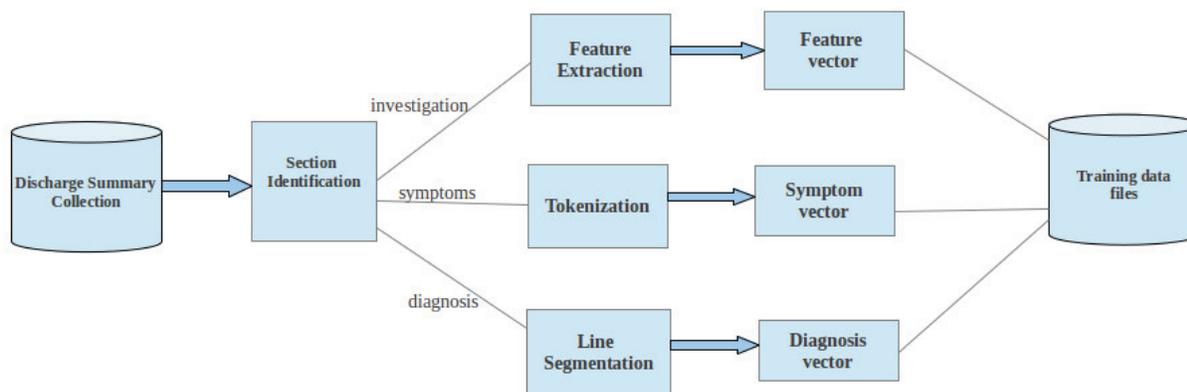


Fig 3. Training

F. Data Preprocessing

The first step of this work is the preprocessing of discharge summary collection. The discharge summaries are collected as a whole using a special software called HIS. They are split as individual files, each file contains a discharge summary. Fixed numbers are assigned to each file, such as f1, f2 etc. This stage also includes the de-identification of files, that is removal of personal information such as name, address, hospital name, doctor's name, medication details and unwanted sections like observations etc. This step is necessary to maintain the privacy of patients and the hospital authorities demanded this while providing the dataset.

G. Section Segmentation

Section segmentation extracts the headings of discharge summaries by taking section boundaries into account and identifying the category of the section. Then relevant sections (Case summary, diagnosis, Investigations) are selected for further manipulations. This work separates training set of discharge summaries into three different files *sym.lab* (contains the symptoms of each discharge summary along with file name, each line contains the symptoms of a discharge summary), *dia.lab* (similar to *sym.lab*, contains the diagnosis of each discharge summary) and *vector.lab* (contains the feature vector corresponding to each discharge summary). The first line of the three files contain symptoms, diagnosis and feature values of first file, second line refers to the second discharge summary and so on.

The gender field of the vector is extracted from the personal information section. The BP field may be at observation (O/E) section. The values of gender and BP field are extracted using regular expression and are added to the corresponding position of the vector. All other fields will be in the investigation section, which are extracted using re module and added to the vector. These values may repeated due to multiple tests conducted at the time of admission in the hospital. In this step we take the values from the first day's test results, because we need to extract the condition of the patient before starting medication.

After identifying 'Case Summary' section, symptoms of the patient are extracted. A sample case summary section is given below:

“Known case of Type 2 DM. Patient got admitted with feature of CHF- due NSAD and Pioglitazone (taken at the dose of 7.5mg once daily for 7 days). Managed in ICU with anti-failure measures. Patient also had Cellulites (Lt) calf nursed with a deep seated abscess which needs drainage. He is being referred to MIMS for further management

Also segment the Diagnosis section using regular expression. This section is much simpler than symptoms because in diagnosis, each line contains a single disease. So by splitting lines and adding each line to diagnosis vector and then this vector is added into the file *dia.lab*, which contains the diagnosis of each discharge summary files in the same order as *sym.lab*. The processing of investigation section is explained in the next section.

H. Feature Vector Generation

Selected features are converted into vector form to generate feature vector. Each discharge summary file is mapped into vector form like :

<vector> <symptoms> □ <diagnosis>

i.e : <Sex Sugar BP Hb ESR S.Creatine Pottassium Sodium LDL Bl.Urea><symptoms> □ < diagnosis(diseases)>

<F 386/536 160/90 9.9 38 7.5 4.0 126 71 113.30>

< fever hyperglycemia>□< T2DM CAD HTN CKD>

These values are extracted using regular expression and now for making a vector, we want to convert each values into fixed numbers according to their meaning. We have made some assumptions for this:

Low □ 1 : If a particular value say BP is low, then that field assigned 1.

Normal □ 3 : If a value is normal, then it assigned as 3.

High □ 5 : If value is high, then 5 is assigned.

Too high □ 7: If Too high, 7 is assigned.

Now the vector becomes :

< 5 5 3 5 5 3 3 1 3 7> <fever hyperglycemia> □ < T2DM CAD HTN CKD>

Two functions are used for this, `getrange()` and `converttorange()`.

I. Functional Modules

There are mainly four important functions for implementing this work. They are :

- ★ **medicaldatavector()** : These module is used for feature vector generation from the investigation section of the discharge summary file. This function mainly uses regular expression for extracting each values.

Also use nltk and string modules of python. First the label of each feature(eg: HbA1C, BP, S.Creatinine etc.) is extracted and then take the corresponding value using regular expression. Assignment of feature position in the feature vector is also performed.

- ▲ **extractingDiagnosis()** : The diagnosis field contains the diseases identified by the doctor after examining the patient. This section is identified and then each of these diseases are collected and stored in a file. The input to this function is a file which contains a single discharge summary. Diagnosis of specific files can also be obtained.
- ▲ **extractingSymptoms()** : Symptoms of each file is extracted from the Case summary section. This function also uses nltk, string and re modules of python. The argument is a file and output is a set of symptoms. Symptoms of specific files can be taken by extractingSymptoms(f).
- ▲ **dotpd()**: Calculates the vector dot product of input vector and all feature vectors from the file vector.lab.

Other functions are **getrange()** for converting to range values, **getvalues** for receiving the input values from GUI and generating input vector, and **getdiafromleastvalueFiles()** for getting the diagnosis of least angle value files. These results are given as the output through GUI.

VI. TESTING PHASE}

J. Disease Extraction

In this phase, user can enter his feature values which can be obtained from blood/urine test results. The values are entered through a graphical user interface(GUI). These values are transformed into feature vector similar to the training phase vectorization and then converted into corresponding range values. Assigning these range values into predefined position will form the input vector. The size of the input vector is also 9, which is same as that of training feature vector. If any feature value is missing or not entered by user, then that feature is considered as normal and value of 3 is assigned in the corresponding position of input vector. The following figure shows the overall data flow in testing phase.

Usecase:

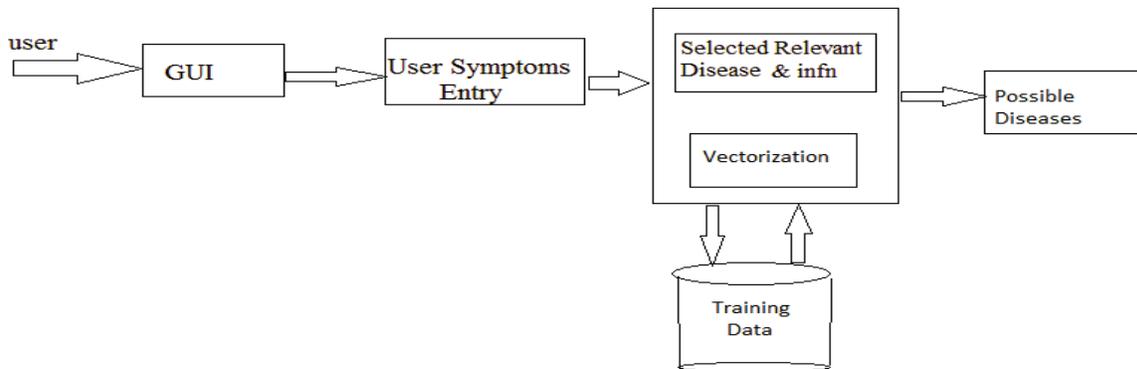


Fig 4. Usecase

K. GUI Implementation

A simple GUI is created for interacting with the user. The code for GUI is written in HTML and it is connected with python code using Eclipse platform. Eclipse is a multi language Integrating Development Environment(IDE). Compressing a base 'workspace' and an extensible plug-in system for customizing the environment. It is written mostly in java. From eclipse environment, we can simply write html codes, and python codes can be imported to its workspace. In this work, input template and output template are created and combined it with python code by using this eclipse platform. Python flask module is used here for providing client-server architecture for python. The screenshot of the input template is given below:

WELCOME !!

Please Enter the Following Field

Gender :	Male
Enter the Lab values:	
Sugar:	<input type="text"/>
BP:	<input type="text"/> / <input type="text"/>
HbA1c:	<input type="text"/>
S.Creatinine:	<input type="text"/>
ESR:	<input type="text"/>
Pottasium:	<input type="text"/>
Sodium:	<input type="text"/>
Blood Urea:	<input type="text"/>
LDL:	<input type="text"/>
Evaluate	

The gender value is selected first because according to the gender we select, range of other values are determined. After entering all the values, just click the evaluate button. Then it calculates the angle between input vector and all training vectors and displays the output. This template only allows numbers as the entry. Fixed port number is provided and we can access the code anywhere in the network by providing server's IP address and this port number.

1) Input Vector Generation

After entering the feature values, user clicks 'the evaluate' button. Then all these values are collected and converted to vector form using `patient_medicine()` function defined in the python flask module. The input to this function is the feature values and it returns the input vector.

eg : <5 5 3 3 1 7 5 3 5>

2) Finding Similarity

`getvalue()` function calculates Vector dot product of input vector with all training vectors. This function will returns the least angle values along with the corresponding file name(f15, f6 etc.)as a list, based on the vector similarity measurement. This list is then passed to `getdiafromleastvalueFiles()` function. Comparing the diagnosis of above files with that of input, this function will return those diagnosis set as the set of possible diseases. Window size is set as 5 so that it will display maximum 5 set of possible diseases. This output implies that the result set is the set of diseases possessed by the patients with similar health condition or parameter values as the user who gives input feature values. Sample output is given in Fig.6.

POSSIBLE DISEASES

Type 2DM Liver Cirrhosis with oesophagal varices Hepatic encephalopathy
 Type 2DM Painful Peripheral Neuropathy
 Type 2DM HTN Leg Eczema with seconday infection
 Type 2DM with uncontrolled hyperglycaemia
 Type 2DM Sever Hyperglycaemia Essential tremor &BA

Fig 6.Output Template

VII. RESULTS AND EVALUATION

To evaluate this system, Precision and Recall are used as evaluation metrics. The total discharge summary collection is divided into two sets, test set and training set. 150 documents are selected as test set and remaining is kept as training set. As an initial set, 20 discharge summaries are taken as test set. The training set contains 50 documents. Feature values are entered without giving diagnosis. 14 of them got the diseases they possess. The evaluation is performed many times by increasing the size of the training set and test set.

Training Set Size	Test Set Size	Precision	Recall
50	20	78%	69%
80	20	87%	80%
100	30	85%	79%

Table 2. Result

The results are shown in Table II. The system gives around 80% accuracy and a moderate precision and recall measures. It can be improved by adding more training files. Also these outputs are checked manually by a doctor for ensuring the authenticity of the result.

VIII. CONCLUSION AND FUTURESCOPE

L. Future Enhancements

This work explores the potential of relation extraction in medical domain. The system can be extended to different domains other than diabetic, provided discharge summary has a unique format. Some possible enhancements are:

- ⤴ Post Processing can be done by adding NLG module on output side. Then the disease abbreviations can be expanded to simple disease names that can be understood by naive user.
- ⤴ More semantic informations (Tests and medication) can be extracted. Discharge summaries contains enormous informations and many relationships, the system can be expanded to extract some more semantics from the file.
- ⤴ Fuzzy modules can be incorporated while converting the range of different features.

M. Conclusion

This paper presents a novel vector based method for extracting disease relationships from medical discharge summaries. This area is relatively new and very interesting since the domain is realistic and the data used is from the healthcare domain. It triggers clinical care guidelines and care protocols and also can be used to improve healthcare. If the discharge summary is having a unique format, then this can be easily extended to any other diseases. This work is not intended for self identification of diseases, rather it can be used to get an awareness of our own health condition.

In the emergency room, every wasted second compromises lives: doctors often have insufficient time to review past discharge summaries, even though these records may contain crucial information. One of the overarching goals of our studies is to produce automated systems that doctors can use to rapidly query a patient's discharge summaries for critical information. In this thesis, we have taken some of the first steps in making such a vision a reality.

REFERENCES

- [1] Barbara Rosario, "Extraction of semantic relations from bioscience text", PhD Thesis, University of California, Berkeley, 2005
- [2] Barbara Rosario and Marti A. Hearst, "Classifying Semantic Relations in Bioscience Texts", University of California, Berkeley, 2004.
- [3] Bryan Rink, Sanda Harabagiu, Kirk Roberts, "Automatic extraction of relations between medical concepts in clinical texts", Journal of American Medical Inform Assoc 2011, Vol.18:594-600.
- [4] Chew-Hung Lee, Christopher Khoo, Jin-Cheon Na, "Automatic Identification of Treatment Relations For Medical Ontology Learning: An Exploratory Study" In I.C. McIlwaine (Ed.), Knowledge Organization and the Global Information Society: Proceedings of the Eighth International ISKO Conference (pp. 245-250), 2004.
- [5] Dimitrios Kokkinakis, "Initial Experiments of Medication Event Extraction Using Frame Semantics", Scandinavian Conference on Health Informatics (SHI), Linköping Electronic Conference Proceedings s. 41-47, 2012.
- [6] Guergana Savova et al., BiomedNLP 2011. Proceedings of the Workshop on Biomedical Natural Language Processing, Sep 2011. 8th International Conference on Recent Advances in Natural Language Processing
- [7] Ivelina Nikolova and Galia Angelova, "Identifying Relations between Medical Concepts by Parsing UMLS Definitions", In proceeding of: Conceptual Structures for Discovering Knowledge-19th International Conference on Conceptual Structures, ICCS 2011, Derby, UK, July 25-29, 2011.
- [8] Kyung-Mi Park, Han-Cheol Cho and Hae-Chang Rim, "Utilizing Various Natural Language Processing Techniques for Biomedical Interaction Extraction", Journal of Information Processing Systems, Vol.7, No.3, September 2011. <http://dx.doi.org/10.3745/JIPS.2011.7.3.459>.
- [9] Michael Tepper, Daniel Capurro et al., Statistical Section Segmentation in Free-Text Clinical Records". Proceedings of the Eight International Conference on Language Resources and Evaluation} (LREC'12), May 2012
- [10] Min Ye, "Text Mining for Building a Biomedical Knowledge Base on Diseases, Risk Factors, and Symptoms", Master's Thesis, Center for Bioinformatics, Saarland University, 2011
- [11] Neil Barrett, "Natural language processing techniques for the purpose of sentinel event information extraction" PhD Thesis, Dept. of CS, University of Victoria
- [12] Oana Frunza and Diana Inkpen, Extraction of Disease-Treatment Semantic Relations from Biomedical Sentences", School of Information Technology and Engineering University of Ottawa, 2010
- [13] P. Menaka, Prof.D.Thilagavathy, "Identifying Semantic Relations for Disease-Treatment in Medline", International Journal of Electronics and Computer Science Engineering, ISSN-2277-1956/V1N2-566-571, 2009
- [14] Rune Saetre, "Natural Language Understanding (NLU): Automatic Information Extraction (IE) from Biomedical Texts", Diploma Thesis, IDI, NTNU, Norway, 2003
- [15] Russell J. Ryan, Groundtruth Budgeting: A technique for Weakly-Supervised Relation Extraction of Medical Language", Master Thesis, Massachusetts Institute of Technology, Jan 2011
- [16] Saman Hina, Eric Atwell, Owen Johnson, "Secure Information Extraction from Clinical Documents Using SNOMED CT Gazetteer and Natural Language Processing" School of Computing, University of Leeds, ICITST May 2011 .
- [17] Son Doan, Lisa Bastarache, et al., "Integrating existing natural language processing tools for medication extraction from discharge summaries". J Am Med Inform Assoc 2010;17:528e 531. <http://jamia.bmj.com>.
- [18] Sophia Ananiadou, "Advances of biomedical text mining for semantic search" University of Manchester, National Centre for Text Mining, 2012
- [19] Tawanda Carleton Sibanda, "Was the Patient Cured? Understanding Semantic Categories and Their Relationships in Patient Records", Master Thesis, Center for Bioinformatics, Saarland University, 2011
- [20] Thomas C. Rindfleisch, Integrating Natural Language Processing and Biomedical Domain Knowledge for Increased Information Retrieval Effectiveness", National Library of Medicine, Bethesda, 1995
- [21] Ying Liu, Robert Bill et al., "Using SemRep to Label Semantic Relations Extracted from Clinical Text", University of Minnesota, Minneapolis, MN, 2013.
- [22] Course: SS3096, "DISEASE CLUSTERS: AN OVERVIEW", Case Studies in Environmental Medicine, U.S. DEPARTMENT OF HEALTH AND HUMAN SERVICES, 2002.
- [23] <http://care.diabetesjournals.org>, Last visited on May 2013.
- [24] www.medicinenet.com Last visited on June 2013.}
- [25] www.diabetescare.net, Last visited on June 2013.
- [26] <http://diabetes.webmed.com>. Last visited on June 2013.}
- [27] <http://ask4healthcare.com> last visited on May 2013.
- [28] <http://emweb.unl.edu/math/mathweb/vectors/vectors.htm> Last visited on July 2013.