May 2014

Disease Name Extraction from Clinical Text Using Conditional Random Fields

Omid Ghiasvand
University of Wisconsin-Milwaukee

DISEASE NAME EXTRACTION FROM CLINICAL TEXT USING CONDITIONAL RANDOM FIELDS

by

Omid Ghiasvand

A Thesis Submitted in

Partial Fulfillment of the

Requirements for the Degree of

Master of Science

in Engineering

at

University of Wisconsin-Milwaukee

May 2014

ABSTRACT
DISEASE NAME EXTRACTION FROM CLINICAL TEXT USING CONDITIONAL RANDOM FIELDS

by

Omid Ghiasvand

The University of Wisconsin-Milwaukee, 2014
Under the Supervision of Rohit J. Kate, PhD

The aim of the research done in this thesis was to extract disease and disorder names from clinical texts. We utilized Conditional Random Fields (CRF) as the main method to label diseases and disorders in clinical sentences. We used some other tools such as MetaMap and Stanford Core NLP tool to extract some crucial features. MetaMap tool was used to identify names of diseases/disorders that are already in UMLS Metathesaurus. Some other important features such as lemmatized versions of words, and POS tags were extracted using the Stanford Core NLP tool. Some more features were extracted directly from UMLS Metathesaurus, including semantic types of words. We participated in the SemEval 2014 competition's Task 7 and used its provided data to train and evaluate our system. Training data contained 199 clinical texts, development data contained 99 clinical texts, and the test data contained 133 clinical texts, these included discharge summaries, echocardiogram, radiology, and ECG reports. We obtained competitive results on the disease/disorder name extraction task. We found through ablation study that while all features contributed, MetaMap matches, POS tags, and previous and next words were the most effective features.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to express the deepest appreciation to my advisor, Dr Rohit J. Kate, who patiently guided me through the project. Without his supervision and constant help, this project would not have been possible. I could not have imagined having a better advisor and mentor for this study. I would like to thank to my committee members, Professor Susan McRoy and Professor Rashmi Prasad. Also, I would like to thank Majid Rastegar-Mojarad, who as a good friend was willing to help and give his best suggestions. It would have been a lonely lab without him.

I am dedicating this thesis to my parents who were always supporting me and encouraging me with their best wishes.

1.Introduction

1.1 Background

Building an automated extraction tool is crucial to manage huge amount of clinical texts. Biomedical Named Entity Recognition (NER) system use in clinical and biomedical texts is growing fast in healthcare and biomedical systems. These systems play important roles in handling clinical and biomedical texts, and also the output of these systems can be used by other tools such as gene-gene interaction [1], protein-protein interaction [2], gene-protein interaction [2], drug-drug interaction [3], etc..

Thus, it is difficult for biomedical researchers to find information of interest from a vast database that is continuously updated. This reinforces the necessity of information extraction based on computational text processing. The task of identifying words and phrases in free text that belong to certain classes of interest is an important first step for many of information management goals. As an example, recent information extraction of protein–protein and protein–nucleotide interactions from MEDLINE abstracts has received the spotlight in bioinformatics. In such biomedical information extraction systems, recognizing named entities such as protein, DNA, RNA, and cell names is one of the most fundamental tasks [45].

In this thesis we have developed a named entity recognition tool to extract disease or disorder names from clinical texts. Clinical texts that were used in our research include discharge summaries, echocardiogram, electrocardiogram, and radiology reports. Later in this chapter, in section 2 the problem is described, and in section 3 SemEval 2014 event has been introduced. In section 3 and 4 introductions to the approach and results of the designed system are presented.

1.2 Problem

Around us there are huge amounts of texts that talk about many things. These texts have been distributed in many different sources such as books, scientific papers, websites, reports, newspapers, etc.. These sources provide information that is not covered in other knowledge sources like databases and/or thesauri.

In biomedical and clinical domain, extracting knowledge from textual sources and mapping them to knowledge sources is an ongoing research that is progressing fast via novel kinds of intelligent recognition methods. These methods are included in natural language processing and text mining approaches. These tools utilize machine learning and statistics to extract useful information from text.

One of the tasks that is widely used in information extraction is Named Entity Recognition (NER). NER deals with identification of boundaries of words, phrases, or terms in text and relations of these boundaries to related terms in knowledge sources such as UMLS (Unified Medical Language System) [5]. There has been a lot of research in this area by academics and research institutes. Some of the tools developed are MedLEE [6], MetaMap [7], and cTAKES [8]. The most recent tools of NER are based on machine learning approaches such as conditional random fields and support vector machines [9-18]. Machine learning methods use supervised or unsupervised learning algorithms [19]. Supervised approaches need annotated data that must be obtained from experts before training an NER system [20]. One of the most prevalent ways of sequence labeling is BIO format [21]. In fact, B, I, and O are three separate labels that are assigned to each word in text. B means beginning of an entity, I inside, and O outside of the entity. As an

example a short paragraph from a clinical discharge summary has been chosen. This paragraph is labeled for recognizing names of diseases/disorders.

> *The/**O** patient/**O** is/**O** a/**O** 40-year-old/**O** female/**O** with/**O** complaints/**O** of/**O** headache/**B** and dizziness/**B**. In/**O** [**2015-01-14**]/**O**, the/**O** patient/**O** had/**O** headache/**B** with/**O** neck/**B** stiffness/**I** and/**O** was/**O** unable/**B** to/**I** walk/**I** for/**O** 45/**O** minutes/**O**.*

An alternate method is very similar to BIO but it has some more labels. This method is called BIESO format method and has five labels instead of three [21]. In BIESO, B means beginning, I inside, E end of entity, S single word entity, and O means outside of an entity. Example 2 is BIESO labeled version of example 1.

> *The/**O** patient/**O** is/**O** a/**O** 40-year-old/**O** female/**O** with/**O** complaints/**O** of/**O** headache/**S** and dizziness/**S**. In/**O** [**2015-01-14**]/**O**, the/**O** patient/**O** had/**O** headache/**S** with/**O** neck/**B** stiffness/**E** and/**O** was/**O** unable/**B** to/**I** walk/**E** for/**O** 45/**O** minutes/**O**.*

The most applied tool of machine learning approaches to NER systems are Conditional Random Fields (CRF) [9-15], Support Vector Machines (SVM) [16-17], and DNorm [18] and have been described in next chapters.

1.3 SemEval 2014 Workshop

SemEval (Semantic Evaluation) is an ongoing series of computational semantic analysis evaluations. It has evolved from the SensEval word sense disambiguation evaluation series. While meaning is intuitive to humans, transferring those intuitions to computational analysis has proved elusive [22]. SemEval provides common platform to

evaluate various approaches for well-known computational semantics tasks thus helping in advancing the state-of-the-art [22].

These evaluations began with simple problems to identify word senses. They gradually have been evolved to implement and solve more complex problems of semantics in language. Moreover they have been designed to identify interrelationships among the elements in a sentence, relationships between sentences, and the actual meaning of sentences [22].

The aim of SensEval and SemEval is to measure performance of semantic analysis by new tools and approaches. The first three evaluations, SensEval 1, 1998 Sussex, SensEval 2, 2001 Toulouse, SensEval 3, 2004 Barcelona, were directed on word sense disambiguation, each time increasing in number of participants and number of different languages. Since 2007, SensEval was changed into SemEval (SemEval 1, 2007 Prague), and essence of tasks extended to cover semantic analysis task outside of word sense disambiguation. After 2012 SemEval in Montreal, SemEval community decided to hold workshops yearly in association with *SEM conferences. Also it was decided that each year tasks should be different from last years, there must not be a same task from previous years [22].

Among these ten tasks we selected task 7, Analysis of Clinical Text, to implement and to evaluate our designed system. As mentioned in website of the workshop “the purpose of this task is to enhance current research in natural language processing methods used in the clinical domain. The second aim of the task is to introduce clinical text processing to the broader NLP community. The task aims to combine supervised methods for text

analysis with unsupervised approaches. More specifically, the task aims to combine supervised methods for entity/acronym/abbreviation recognition and mapping to UMLS CUIs (Concept Unique Identifiers) with access to larger clinical corpus for utilizing unsupervised techniques" [24].

Furthermore this task includes two subtasks A and B. Subtask A, is to identify boundaries of mentions of diseases or disorders. Here there are examples of subtask A [23], the disease/disorder names to be extracted are underlined:

1. The rhythm appears to be *atrial fibrillation*.
2. The *left atrium* is moderately *dilated*.
3. 53 year old man s/p *fall from ladder*.

The interesting thing in example 2 is that there are parts of a disease that are in different positions in the sentence. The task required detecting all disjoint parts of a disease.

Subtask B involved mapping each disease/disorder, discovered by subtask A, to a UMLS CUI (Concept Unique Identifier). This subtask also is known as normalization task, and mapping is limited to UMLS CUIs of SNOMED CT (Systematized Nomenclature of Medicine-Clinical Terms). If a disease/disorder is not in SNOMED CT/UMLS or is not of the required semantic type in UMLS then it will be mapped to "CUI-less". Some examples of this subtask are as follow [23]:

1. atrial fibrillation - C0004238; UMLS preferred term *atrial fibrillation*
2. left atrium...dilated - C0344720; UMLS preferred term *left atrial dilatation*
3. fall from ladder - C0337212; UMLS preferred term is *accidental fall from ladder*

Only task A was the topic of this thesis although our team also participated in Task B.

SemEval 2014 tasks were announced officially on October 2013, and trial and training data of task 7 were realeased on November 12, and December 15 2013 respectively. Also evaluation started on March 19, 2014 and ended on March 21st. Results were available to participants on April 11th 2014 [23].

1.4 Approach

The approach that has been used in this thesis is based on conditional random fields. We have used CRF to detect clinical named entities. Also there have been some other tools used to extract features out of clinical text. To extract lexical features such as Part Of Speech tags, and lemmatized version of words Stanford NLP tool has been used. Also we have used MetaMap to get of crucial features from texts. In fact MetaMap processes text before running CRF. It automatically finds terms existent in UMLS Metathesaurus. A Boolean feature was used to represent whether MetaMap found the words as part of disease/disorder in UMLS. Furthermore some features were added to the system such as semantic group of words in UMLS (if they exist in UMLS), names and semantic groups of abbreviations, and lengths of words. The approach of developing the system is explained in chapter 3 in details.

1.5 Results

After running our system on data 199+99 reports as training and 133 reports as testing data, we got f-score 0.755 in strict and 0.884 in relaxed evaluations. In Table 1 you can also find precision and recall in strict evaluation case. Also in SemEval 2014 we got third rank in Task A out of 19 teams. Other teams ranked first and second got F-score 0.813

and 0.900, and the second one got 0.766 and 0.893 in strict and relaxed evaluation cases. In Table 1, ranking of teams participated in SemEval 2014 is presented as well based on strict evaluation.

Table 1 Ranking of teams in SemEval 2014

Team ID	Precision	Recall	F-score
UTH_CCB	0.843	0.786	0.813
UTU	0.765	0.767	0.766
UWM	0.787	0.726	0.755
IxaMed	0.681	0.786	0.730
RelAgent	0.741	0.701	0.720
ezDI	0.750	0.682	0.714
CLEAR	0.807	0.636	0.712
ULisboa	0.753	0.663	0.705
BioinformaticsUA	0.813	0.605	0.694
ThinkMiners	0.749	0.617	0.677
ECNU	0.712	0.601	0.652
UniPI	0.639	0.529	0.579
CogComp	0.524	0.576	0.549
TMU	0.561	0.534	0.547
MindLab-UNAL	0.500	0.479	0.489
SZTE-NLP	0.547	0.252	0.345
KUL	0.655	0.178	0.280
QUT_AEHRC	0.387	0.298	0.337
UG	0.114	0.234	0.153

2.Background and Related Work

2.1 Introduction

In this chapter some theory and background of conditional random fields, brief introductions to MetaMap, UMLS, and Stanford NLP tool, and related work are presented.

2.2 Conditional Random Fields

The task of predicting labels of data point sequences appears in many research areas such as bioinformatics, computational linguistics, speech recognition, and image processing. As an example, in the following there is a paragraph of a discharge summary that has been used as a part of training data. In this paragraph all words are labeled with part of speech tags, and their appropriate label.

```
The/DT/O patient/NN/O is/VBZ/O a/DT/O 40-year-old/JJ/O female/NN/O
with/IN/O complaints/NNS/O of/IN/O headache/NN/B and/CC/O
dizziness/NN/B ./O In/IN/O -LSB-/NNP/O **/NNP/O 2015-01-14/NNP/O
**/NNP/O -RSB-/NNP/O ,/, /O the/DT/O patient/NN/O had/VBD/O
headache/NN/B with/IN/O neck/NN/B stiffness/NNS/I and/CC/O was/VBD/O
unable/JJ/B to/TO/I walk/VB/I for/IN/O 45/CD/O minutes/NNS/O ./O
The/DT/O patient/NN/O also/RB/O had/VBD/O a/DT/O similar/JJ/O
```

Figure 1 Example of tagging a paragraph in our system

As it is obvious in Figure 1, the paragraph has been tagged with related POS tags, and also words have been labeled based on names that must be recognized. Here BIO approach has been used to label each word. The goal is to predict labels (B, I, or O) of each word in text.

One the most common methods for tagging sequences of data points or words is Hidden Markov Model (HMM). HMMs or probabilistic finite state automata are used to identify the most likely sequences of data points [35]. In this approach, that is a kind of generative

model, a joint probability distribution over whole data points or bunch of data points and their related labels is defined. Generative models must enumerate all feasible observation sequences in that observation elements are isolated and independent of each other. In another words, observation element at each time may only depend on the state or label of the system at that time. This is very important in labeling sequence of data. Obviously a model that can support supple deductions is needed, though a model that exhibits data without inappropriate independence is also necessary [36].

One way of satisfying both conditions is to define a conditional probability distribution over a label sequence given a particular sequence, instead of defining a joint probability over sequence elements and labels. Conditional models based on these conditional probabilities label an input sequence by choosing another label sequence that maximizes the conditional probability. This usefulness of conditional probabilities removes the inappropriate independence of data, and guaranty finding relationships between sequence elements [36].

Conditional random fields are included in statistical methods of modeling. The main purpose of them is to find the pattern of sequences in data, or even to find the structure of data. A CRF can recognize the pattern of data not only based on the features of data but also based on the sequences of them. Then it assigns labels to each data point in a data set [37]. While an ordinary classifier assigns labels to data point only based on structure of data (it does not look to the sequences of data points).

CRFs are kinds of undirected probabilistic graphical models. A CRF is a form of undirected graphical model that defines a single log-linear distribution over label

sequences given a particular observation sequence. The main advantage of CRFs against HMMs is their conditional natures that result in relaxation of the independence assumptions required by HMMs in order to ensure tractable inference [36]. They can be used to model relationships between observations and make a robust model to recognize the sequence relationships. Most of the times CRFs have been used to predict labels or to parse sequence of data points. One of the most important applications of CRFs is in natural language processing, gene prediction, and image processing. In natural language processing CRFs usage is growing fast, and they have been used for shallow parsing and named entity recognition [36].

CRFs [44] are undirected graphical models, a special case of conditionally trained probabilistic finite state automata. They can incorporate a large set of arbitrary and non-independent features while still having efficient procedures for non-greedy finite-state inference and training.

CRFs have been indicated robust and reliable in different sequence modeling tasks including named entity recognition [42].

To calculate conditional probability of desired outputs given values on inputs CRF is used. The conditional probability of state sequence *s=[s1, s2, s3, …, sN]* given input *sequence i= [i1, i2, i3, …, iN]* can be calculated by (1) [42]:

$$P_{\wedge} = \frac{1}{Z_i} exp\left(\sum_{n=1}^{N}\sum_{k=1}^{K} \lambda_k \times f_k(s_{n-1}, s_n, i, n)\right) \qquad (1)$$

Where $f_k(s_{n-1},s_n,i,n)$ is a feature function that weight λ_k is to be learned while training. The values of feature function may range between –infinity to infinity, but usually they are

between 0 and 1. Also there is a normalization factor that makes all conditional probabilities sum to 1, that is calculated by (2) [42]:

$$Z_i = \sum_{s} exp\left(\sum_{n=1}^{N}\sum_{k=1}^{K}\lambda_k \times f_k(s_{n-1}, s_n, i, n)\right) \quad 2)$$

To train a conditional random field, an objective function is needed as well. This function should be maximized to penalize log-likelihood of the state sequence given the input sequence. It is calculated by (3) [42]:

$$L_\wedge = \sum_{j=1}^{M} log\left(P_\wedge\left(s^{(j)}\middle|i^{(j)}\right)\right) - \sum_{k=1}^{K}\frac{\lambda_k^2}{2\sigma^2} \quad (3)$$

Where $\{<i^{(j)}, s^{(j)}>\}$ is the labeled training data. The second sum corresponds to a zero-mean, σ^2 –variance Gaussian prior over parameters, that facilitates optimization by making likelihood surface strictly convex. Usually parameter λ is set to maximize the penalized log-likelihood [42].

In general, to apply CRF to an NER system an input sequence is a sentence and the state sequence is its corresponding label sequence. A feature function $f_k(s_n, s_{n-1}, i, n)$ is 0 for most cases but it is 1 when s_n, s_{n-1} are certain states and inputs has certain properties [42]. CRF software that is used in this study is CRFsuite described below.

- **CRF Software**

We used CRFsuite software that is available at [41]. This software allows us to run conditional random fields on our data. The data format for CRFsuite is like columns that show features and they should be separated by a tab. The first column is the actual label of the line with the features. An example of data format for CRFsuite is as follows:

```
B    word=abdomen    prev_word=.    next_word=is
O    word=is    prev_word=abdomen    next_word=soft
```

```
O   word=soft   prev_word=is   next_word=,
O   word=,   prev_word=soft   next_word=nontender
I   word=nontender   prev_word=,   next_word=,
O   word=,   prev_word=nontender   next_word=nondistended
B   word=nondistended   prev_word=,   next_word=,
O   word=,   prev_word=nondistended   next_word=negative
O   word=negative   prev_word=,   next_word=bruits
I   word=bruits   prev_word=negative   next_word=.
```

Figure 2 Example of inputs into CRFsuite

As shown in Figure 2, each line stands up for a word in text. The first label shows the actual label that system is going to be trained on, and other columns are features. In this example each word, also counted as a feature, is on the first column and two other features that are previous and next words are on columns 3 and 4. We gathered all clinical text together and made a huge training file like example in Figure 2.

2.3 MetaMap

Lister Hill National Center for Biomedical Communications at the National Library of Medicine (NLM) developed a highly configurable application, known as MetaMap, for mapping biomedical text to UMLS Metathesaurus [6]. Also it can be used to identify Metathesaurus concepts. MetaMap utilizes a knowledge-intensive approach and natural language processing. It is widely used in different academic and research institutes in the world. Also MetaMap have been used to index biomedical literature semi-automatically and automatically at NLM [6].

Also MetaMap has some important features that are listed below [6]:

- Downloadable binary and full sources available
- Downloadable UMLS-based datasets for various UMLS releases
- DataFileBuilder suite, which allows users to create their own data sets

- MetaMap Java API to a local MetaMap installation
- Web API to our Batch and Interactive scheduling facility (currently 120 ~3GHz processors)
- MetaMap UIMA Annotator, which encodes MetaMap named entities in a format utilizable by UIMA components
- MedPost/SKR part-of-speech tagger server and Word Sense Disambiguation (WSD) server

Furthermore one of the most important features of MetaMap is MetaMap Java API. This Java-based API is used to the Indexing Initiative Scheduler facility. This facility was developed to enable programmers to use MetaMap in programming jobs. These jobs must be submitted to *Scheduler Batch* and *Interactive* facilities [6]. In our research, the Java based API of MetaMap has been used to extract particular names that are considered in the data set.

2.4 UMLS Metathesaurus

Unified Medical Language System is a repository of biomedical controlled vocabularies developed by the National Library of Medicine (NLM). UMLS includes a set of files and software applications. These files contain biomedical vocabularies, and modulus to utilize interoperability between computer systems. It is feasible to use the UMLS to develop or improve applications, like electronic health records, classification tools, dictionaries and language translators [39]. Finding relationships between health information, medical terms, drug names, and billing codes among a set of different computer systems is one the most common and useful application if the UMLS. Moreover the UMLS is widely used in search engine retrieval, data mining, public health statistics reporting, and

terminology research.

UMLS has three components which are as follows [39]:

- Metathesaurus: Terms and codes from many vocabularies, including CPT, ICD-10-CM, LOINC, MeSH, RxNorm, and SNOMED CT.
- Semantic Network: Broad categories (semantic types) and their relationships (semantic relations).
- SPECIALIST Lexicon and Lexical Tools: Natural language processing tools.

Generating of Metathesaurus includes these steps [39]:

- Processing the terms and codes using the Lexical Tools
- Grouping synonymous terms into concepts
- Categorizing concepts by semantic types from the Semantic Network
- Incorporating relationships and attributes provided by vocabularies
- Releasing the data in a common format

2.5 Stanford NLP tool

The Stanford NLP Group developed a natural language processing software and made it available to everyone. The software contains statistical NLP toolkits for many. These toolkits can be embedded into different applications by developers of NLP software [40].

All the software is written Java, and all distributions need Oracle Java 6+ or OpenJDK 7+. These distribution packages contain components command-line invocation, jar files, Java API, and source codes. Recently some people have expanded these packages with

binding or translations for other languages such as Chinese, German, etc.. Moreover majority of this software can be used from Python, Ruby, Perl, Javascript, and F# or other .NET languages [40].

Stanford NLP can do many NLP tasks such as tokenization, lemmatization, chunking, POS tagging, named entity recognition, relation extraction (dependency extraction), etc. in our research we have used this software for:

- Tokenization
- Part of speech tagging
- Sentence Splitting
- Relation Extraction
- Lemmatization

In next section we will explain how we used Stanford NLP tool, MetaMap, and UMLS Metathesaurus to extract features to feed CRF.

2.6 Related Work

Bondari et al [9], presented a method based on supervised CRF model to identify of disorder named entities from Electronic Medical Records (EMR). The CRF system in the research uses external knowledge from specialized biomedical terminologies and Wikipedia. The system performance was evaluated at F-measure score 0.598 in strict evaluation case and 0.711 in relaxed evaluation.

In [10] a named entity recognition system is developed based on Structural Support Vector Machines (SSVM). Colgey et al used SSVM with an array of feature types

including lexical, semantic, and cluster based knowledge. The F-measure for their designed system is 0.656 in strict and 0.832 in relaxed evaluation cases.

In [11] the authors participated in ShARe/CLEF 2013 NLP challenge. Fan et al used an existing NLP system developed at Kaiser Permanente and modified that to explore concepts of disorders in clinical texts. The main parts of their system are section detection, tokenization, sentence chunking, probabilistic POS tagging, rule-based phrase chunking, terminology look-up (using UMLS), rule-based concept disambiguation. Finally they got F-score 0.503 in strict and 0.684 in relaxed cases.

In [12] for identifying names of disease or disorders Gung took a supervised learning, chunking-based approach, to identify disorder spans. In particular his system introduced a method for diagnosing sequences of disjoint and overlapping disorder entities using relation extraction and Semantic Rule Labeling (SRL). By using a CRF, he found initial disorder spans. Using these spans, he applied a locational relation extractor and SRL system to locate pairs of spans belonging to the same disorder mention. Performance of the system was evaluated at F-score 0.687 and 0.836 in strict and relaxed evaluations respectively.

In [13] Hervas et al developed a system to participate in ShARE/CLEF 2013 NLP challenge. They took these steps: automatic orthographic correction, acronyms and abbreviation detection, negation and speculation phrase detection and medical concepts detection. The main tool in their system is MetaMap that has been used to detect disease/disorder names. They got F-score 0.504 and 0.660 in strict and relaxed evaluations.

A tool based DNorm is introduced in [14]. Leaman et al used application of DNorm -a mathematically principled and high performing methodology for disease recognition and normalization, even in the presence of term variation- to clinical notes. The main part of NER of DNorm is base BANNER NER system. They got F-measure 0.707 and 0.849 as strict and relaxed evaluation runs respectively.

Another team that participated in ShARE/CLEF 2013 NLP challenge, used integrated cTAKES for concept mention detection [15]. Liu et al from Mayo Clinic used MedTagger implemented in integrated cTAKES (icTAKES), and principle of concept detection in that is based on conditional random fields. F-score of their designed system is 0.668 in strict evaluation, and in relaxed evaluation F-score is 0.844.

Osborne et al is another team that used MetaMap and YTEX as the basis of their designed tool to identify names of disorders in clinical texts [16]. They did not modify the system basis but they filtered results based on stop words and UMLS semantic type. F-scores of 0.505 and 0.734 are the best performance of their system.

Patrick et al [17] used conditional random field to recognize clinical concepts, and also a support vector machine based method was used to capture more complex named entities. First a CRF was used to detect names of disorders. After finding them, discovered names were passed through an SVM to find any relation among the identified disorder mentions to decide whether they are a part of a complex disorder. F-scores of performance of their systems in strict and relaxed evaluations are 0.604 and 0.793.

The last paper that offered a model based on Cocoa, an existing dictionary/rule based entity tagger that tags multiple semantic types in biomedical domain including diseases,

on disease/sign/symptom detection in clinical records [18]. Ramanan et al also added a small module for event-based detection of annotated sentence fragments containing verbs/gerunds. 0.562 and 0.779 are F-measures of the team in the challenge in strict and relaxed evaluations.

In our proposed approach, we used not only common features used in other researches but also some novel features such as abbreviations, MetaMap matches, and lemmatized versions of words as separate features to train a CRF.

3.Approach

3.1 Introduction

Named entity recognition (NER) systems like other tasks in natural language processing can be implemented by machine learning approaches. In named entity recognition (NER) systems most of the methods that have been applied are those from supervised learning. As shown in chapter two (Related Work), different approaches were used.

In previous chapter we saw that most of named entity recognition systems were developed based on sequence labeling tools. Among all sequence labeling tools, conditional random fields or CRFs are widely used to detect named entities, not only for single spans but also for disjoint spans of entities. CRFs are very robust and reliable in these kinds of systems and research in this area is still active. In this study, we have proposed a CRF based named entity recognizer presented in the next sections.

3.2 Proposed Approach

Our approach uses CRF. The features that have been used here are structural and semantic features. Structural features are surrounding words, POS tags of surrounding words, length, and lemmatized version. All of these features are extracted by Stanford NLP tool that was described in previous chapter.

Semantic features that have been extracted and used in this project are semantic type of surrounding words, MetaMap match of word and surrounding words, abbreviations' extensions, semantic groups, and UMLS match. Metamap match of the words are provided by Metamap tool. Semantic types, UMLS match, and abbreviation's extensions are directly obtained from UMLS.

3.3 Features Used

In our research we have selected some common features and some novel ones. As a summary the features that we have used are listed in Table 2.

Table 2 List of features used in the research

ID	Feature Name	Tool used to extract
1	Word	Programming Language
2	Next word	Programming Language
3	Previous word	Programming Language
4	POS tag of Word	Stanford NLP
5	POS tag of next word	Stanford NLP
6	POS tag of previous word	Stanford NLP
7	Next two words	Programming Language
8	Previous two words	Programming Language
9	Length of the word	Programming Language
10	Semantic group of the word	UMLS
11	Semantic group of next word	UMLS
12	Semantic group of previous word	UMLS
13	Exact match of bigram	UMLS
14	Exact match of trigram	UMLS
15	Exact match of reverse bigram	UMLS
16	CUI of the word	UMLS
17	MetaMap match of the word	MetaMap
18	MetaMap match of next word	MetaMap

19	MetaMap match of previous word	MetaMap
20	Lemmatized version of the word	Stanford NLP
21	Parent of the word in dependency tree	Stanford NLP
22	Abbreviation full name	List of Abbreviations
23	Abbreviation full name exact match into UMLS	UMLS
24	Abbreviation full name semantic group	UMLS

As shown in Table 2 lexical features are:

- Word
- Next word
- Previous word
- POS tag of Word
- POS tag of next word
- POS tag of previous word
- Next two words
- Previous two words
- Length of the word

Next group of features are those extracted from UMLS Metathesaurus. Firstly we get the word then we look for semantic group of it. Next we get CUI of each word that we find in

UMLS, and next and previous words' semantic groups. These features are listed as follows:

- Semantic group of the word
- Semantic group of next word
- Semantic group of previous word
- Exact match of bigram
- Exact match of trigram
- Exact match of reverse bigram
- CUI of the word

Here exact matches are Boolean values, and if the words, bigrams, reverse bigrams, or trigrams have been found in UMLS Metathesaurus, the value of each feature would be "true" else it is "false."

Also semantic group in UMLS can be one of these categories:

- Congenital Abnormality
- Acquired Abnormality
- Injury or Poisoning
- Pathologic Function
- Disease or Syndrome
- Mental or Behavioral Dysfunction
- Cell or Molecular Dysfunction
- Experimental Model of Disease
- Anatomical Abnormality

- Neoplastic Process
- Signs and Symptoms
- Finding

Next group of features that are novel features are extracted by MetaMap tool. In fact MetaMap extracts those names that map into UMLS Metathesaurus. By this, we added three more features:

- MetaMap match of the word
- MetaMap match of next word
- MetaMap match of previous word

These values are all Boolean and “true” means that the token is found in UMLS by MetaMap software.

Other important features are lemmatized version of the word and parent of the word in dependency tree of the sentence in that the word is. These two features have been extracted by Stanford NLP tool. Moreover we have extracted full name of abbreviations in text. In fact we have created a list of all biomedical abbreviations. Next we map the word to our list, and if we find equivalent of that word in our list of abbreviation extracted from [46], then we add full name of that. It should be mentioned that it is only based on the list that we have, and we are not using any algorithm to find abbreviations. Based on that finding, full name is mapped to UMLS Metathesaurus, and exact match and semantic group of that also are added to our training database. In total, these novel features can be listed as below:

- Lemmatized version of the word
- Parent of the word in dependency tree
- Abbreviation full name
- Abbreviation full name exact match into UMLS
- Abbreviation full name semantic group

These features in a form of a CRF file feed into CRFsuite software to be trained. After training we test our test data set and make output files containing positions of diseases or disorders in clinical reports. Finally by using an evaluation tool, that is a program written in Perl provided by SemEval committee, we evaluate our results. In next chapter, results achieved by our proposed approach are presented.

4.Results

4.1 Introduction

In this chapter results of the designed system are presented. In section 2, results of the system on training and development data sets are presented. Also a comparison between different sets of features has been done that shows performance of each group of features. Section 3 includes results achieved in SemEval 2014 NLP challenge in which training and development data sets were used to train the system, and test data set was used to get the final results. In the last section conclusion and future work will be described.

4.2 Results on Training and Development Data Sets

In this section, the results obtained by the system using training and development data sets are presented. In this project we have used three different data sets. The first one is training data set that contains 199 clinical reports. The other data set is development set that includes 99 clinical reports. These clinical reports contain discharge summaries, echocardiogram, electrocardiogram, and radiology reports. The only difference between previous data sets and another data set, that is our test set, is that the test data set only includes discharge summaries, and there are no other clinical reports. The distribution of reports in training and development data set are shown in Tables 3 and 4. These datasets were fixed and provided by SemEval 2014 Task 7 organizers.

Table 3 Distribution of reports in training data set

Type of Report	Count (%)
Discharge Summary	61 (30.7%)
Echocardiogram	54 (27.1%)
Electrocardiograph	42 (21.1%)

Radiology	42(21.1%)

Table 4 Distribution of reports in development data set

Type of Report	**Count (%)**
Discharge Summary	75 (76%)
Echocardiogram	12 (12%)
Electrocardiograph	0 (0%)
Radiology	12(12%)

To evaluate the performance of the system there are three measures, precision, recall, and F-score. In information retrieval and pattern recognition, precision means "the ratio of the number of retrieved relevant records to the total number of relevant and irrelevant records," or "number of true positive over number of true and false positive [41]." And recall or sensitivity means "the ratio of the number of retrieved relevant records to the total number of relevant records," or "number of true positive over number of true positive and number of false negative [41]." We can define precision and recall by equations (1) and (2).

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (1)$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (2)$$

We use two scoring schemes: strict and relaxed. The strict scoring scheme only counts exact matches as success. For example, if the key is OVERLAP and the response is BEFORE-OR-OVERLAP then this is counted as failure. To find strict scoring we can use (1) and (2) [43].

In relaxed version if there is any overlap (left or right), it will be counted as success or one.

Based on definitions of precision and recall F-score can be obtained by (3):

$$F - score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (3)$$

In this study we have created different sets of features to compare effects of each group. For this reason, six groups of features have been created. These features are selected based the novelty and the tool that extracted them. First group of features are lexical features containing surrounding words, POS tags, and length of the word. Second group of features are those that extracted directly from UMLS Metathesaurus, semantic group of the word and surrounding words, exact match of bigrams, exact match of reverse bigrams, exact match of trigrams, and CUIs of words. The third group of features includes those that are extracted via MetaMap containing MetaMap match of the word, next, and previous words. Lemmatized version and parent of the word in dependency tree are other features that we are going to put them in separate groups. The last group is related to abbreviations containing abbreviation full name, abbreviation full name exact match into UMLS, and abbreviation full name semantic group in UMLS Metathesaurus. All of these grouped features are summarized in Table 5.

Table 5 Groups of selected features

Group	Feature Name	Tool used to extract
G1	Word	Programming Language
	Next word	Programming Language
	Previous word	Programming Language
	POS tag of Word	Stanford NLP
	POS tag of next word	Stanford NLP
	POS tag of previous word	Stanford NLP
	Next two words	Programming Language
	Previous two words	Programming Language
	Length of the word	Programming Language
G2	Semantic group of the word	UMLS
	Semantic group of next word	UMLS
	Semantic group of previous word	UMLS
	Exact match of bigram	UMLS

	Exact match of trigram	UMLS
	Exact match of reverse bigram	UMLS
	CUI of the word	UMLS
G3	MetaMap match of the word	MetaMap
	MetaMap match of next word	MetaMap
	MetaMap match of previous word	MetaMap
G4	Lemmatized version of the word	Stanford NLP
G5	Parent of the word in dependency tree	Stanford NLP
G6	Abbreviation full name	List of Abbreviations
	Abbreviation full name exact match into UMLS	UMLS
	Abbreviation full name semantic group	UMLS

These features are applied to the system to train CRF, and results of performance of the system based on each group are presented in Table 6. The system was trained on the training data and tested on the development data.

Table 6 Evaluations of groups of features

Feature Group	S/R	Precision	Recall	F-score
G1	Strict	0.774	0.470	0.585
G1+G2		0.807	0.630	0.708
G1+G2+G3		0.830	0.658	0.734
G1+G2+G3+G4		0.828	0.668	0.740
G1+G2+G3+G4+G5		0.826	0.665	0.737
All		0.829	0.673	0.743
G1	Relaxed	0.937	0.581	0.717
G1+G2		0.950	0.758	0.843
G1+G2+G3		0.957	0.775	0.856
G1+G2+G3+G4		0.959	0.790	0.866
G1+G2+G3+G4+G5		0.957	0.787	0.864
All		0.958	0.795	0.869

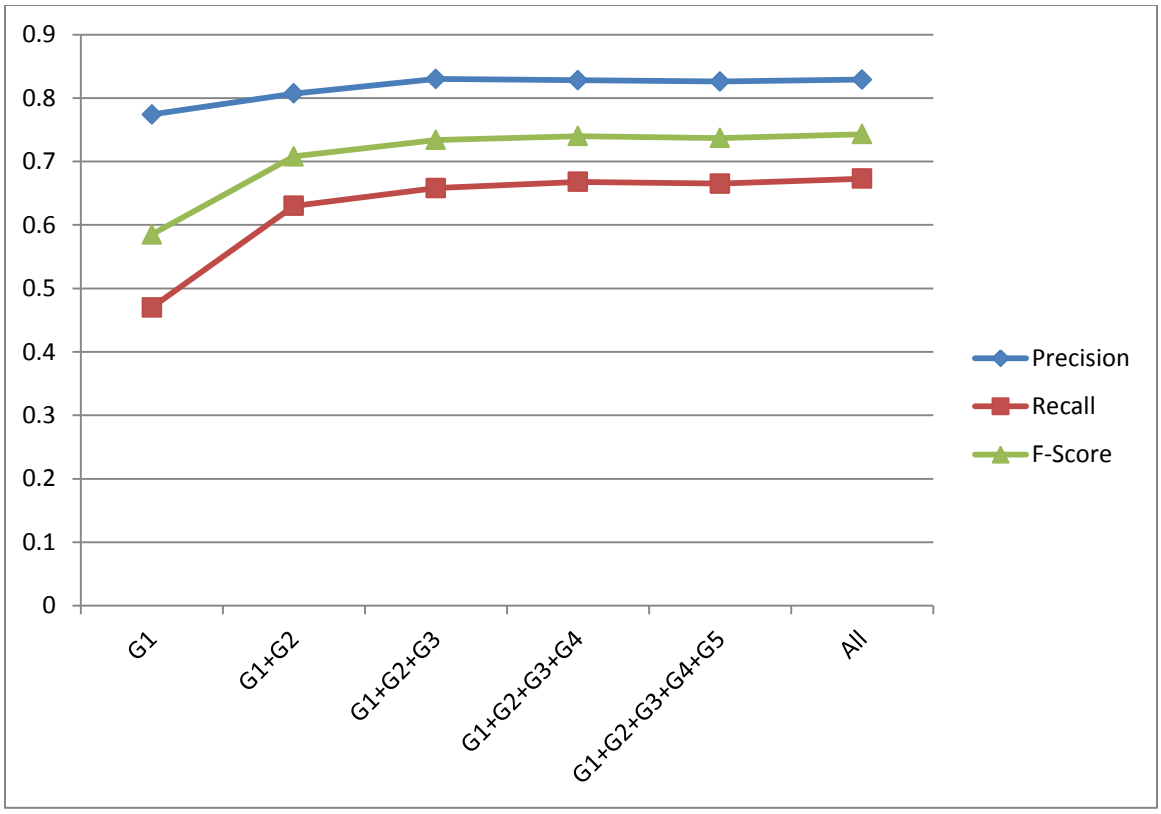

Figure 3 Diagram of performance of different sets of features

In Figure 3, the diagram of performance of system with different features in strict evaluation case is illustrated. An interesting thing here is feature "parent of the word." When this feature added to the feature set G1+G2+G3+G4, performance of the system is decreased. Not only precision has been affected, but also recall and F-score are influenced by it. This means that feature G5 of "parent of the word" might not be useful to the system. Another impressive thing in G5 feature is that, when we remove it from the set of all features, F-score does not change, but recall and precision change a little. You can see the results after excluding feature G5 from the set of all features in Table 7.

Table 7 Result including all features except "parent of the word"

S/R	Precision	Recall	F-score

Strict	0.827	0.675	0.743
Relaxed	0.958	0.799	0.871

For this reason we did ablation analysis and by excluding each feature group we achieved results shown in Table 8. Results in this table are sorted based on the importance of each group of features. Thus the most important features are in group 1 (morphological and lexical) and the least important features are abbreviations and parent of the word.

Table 8 Results of ablation

	Precision	**Recall**	**F-Score**
All - G1 (Morphological and Lexical)	0.779	0.569	0.658
All - G3 (MetaMap)	0.81	0.648	0.720
All - G5 (Lemmatization)	0.825	0.666	0.737
All - G2 (Semantic)	0.824	0.669	0.738
All - G6 (Abbreviations)	0.828	0.668	0.740
All - G4 (Parent)	0.827	0.675	0.743
All	**0.829**	**0.673**	**0.743**

Another change in our system was that we added semantic group of all other word except diseases or disorders. The semantic group that we used in our system was limited to diseases. To see the results when other semantic groups are involved, we added them and results in Table 9 were obtained.

Table 9 Results including all semantic groups

S/R	Precision	Recall	F-score
Strict	0.823	0.665	0.735
Relaxed	0.957	0.789	0.864

As it is obvious in Table 9, adding other semantic groups not only did not help but also it decreases accuracy of the system.

Also in this project we ran two baselines, UMLS and MetaMap. UMLS baseline is obtained from UMLS by mapping all words directly into UMLS Metathesaurus. This baseline has very low performance against MetaMap and our proposed methods. The reason that can be said about that is mapping all words into UMLS with no limitation and restriction may lead to improper results. For example in disease/disorder named entity recognition, token "wasting" is not a disease or disorder, but it has CUI, C0235394, and a semantic group T047 that falls into disorder semantic group in UMLS. Another baseline is MetaMap that has been implemented by a Java API provided by national institute of health (NIH). It has performance much better than UMLS, but it is still not good at extracting for clinical concepts. Some of the reasons are failing to identify split noun phrases as a concept, failing to rank identified phrases high enough, and changing the identified concept to its original one [44]. The comparison between these two baselines and our proposed method is shown Table 10, and Figure 4 illustrates precision, recall, and F-score of these three approaches in strict evaluation case.

Table 10 Comparison of three methods, UMLS, MetaMap, and our proposed method

Baseline	S/R	Precision	Recall	F-score
UMLS	**Strict**	0.384	0.332	0.356
MetaMap		0.474	0.628	0.540
Proposed Method		0.827	0.675	0.743
UMLS	**Relaxed**	0.565	0.743	0.642
MetaMap		0.470	0.405	0.435
Proposed Method		0.958	0.799	0.871

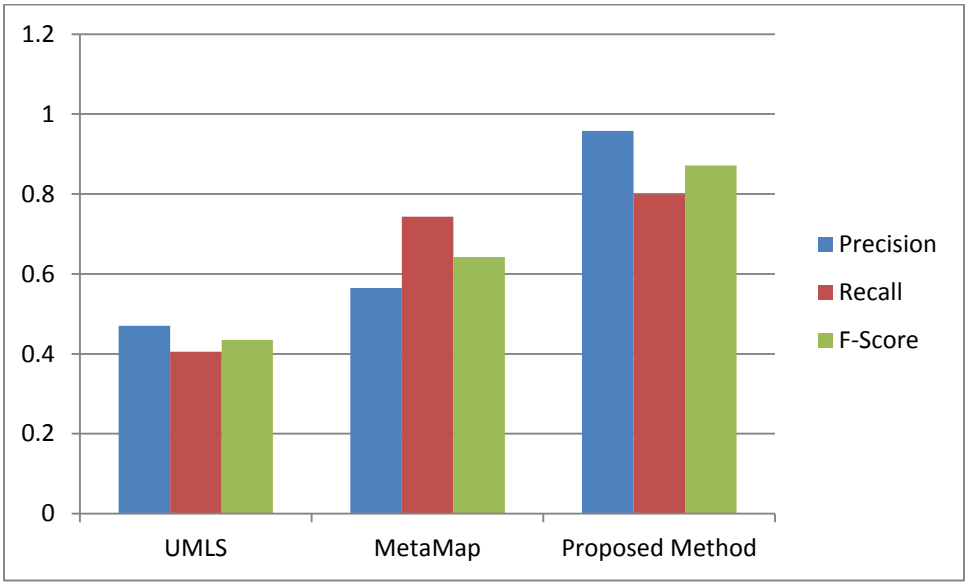

Figure 4 Diagram of different measures in UMLS, MetaMap, and our proposed method

4.3 Results on Test Data Set in SemEval 2014

On April 11th, 2014 results of SemEval 2014 were released by the organizers. The system was run on test data set containing 133 reports. Interesting thing about the test data set is that it only contains discharge summaries, and there are no other types of reports such as echocardiogram and radiology reports. Moreover we have used not only training data set

but also development data set to train the CRF. Results achieved by our team in SemEval 2014 are shown in Table 11.

Table 11 Results of system run on test data set in SemEval 2014

S/R	Precision	Recall	F-score
Strict	0.787	0.726	0.755
Relaxed	0.911	0.856	0.883

Because gold standard data are not provided by the committee of SemEval 2014, we could not run our system with different groups of features.

Also after seeing results we found that our team ranked 3rd among 19 teams around the world.

4.4 Error Analysis

In error analysis there were some issues that our system failed to recognize. The first example is: *t1 & t2 signal*. In this example our system identified it as a disease by labeling *t1/B t2/I signal/I*. But in gold standard, it says *t1/B signal/I* is a disease, and *t2/B signal/I* is another one although there is only one signal in the sentence. This issue was not common, and we only saw once.

Another issue was with body parts. For example in the sentence "*left atrium is moderately dilated*", left atrium is a part of a disease labeled as *left/B atrium/I dilated/I*. But in many cases, there were parts of body that were not part of disease but our system detected them as a disease, like *left/B atrium/I*.

Adverbs before disease names are also another issue that we had. For example sever pain should be labeled as *server/O pain/B*, but our system labeled it as *server/B pain/I*. the reason for that is because sometimes adjectives are parts of body, “chest pain” labeled as *chest/B pain/I*. Thus our system in many cases failed to recognize it.

5.Conclusion and Future Work

As described in chapter 2, there have been several approaches for named entity recognition in biomedical and clinical domain. The most accurate approaches for NER system use machine learning methods. Because of existence of disjoint spans of clinical entities in clinical reports, machine learning methods that are used for sequence labeling are widely used to detect clinical concepts. Among machine learning methods conditional random fields or CRFs are used as the basis of many NER systems for sequence labeling. In our study that is based on CRF as well, we applied some novel features to feed CRF. These new features such as MetaMap matches, abbreviations, and semantics improved performance of the system that shows an NER system can be enhanced with features that are highly related to semantics. As shown in Table 7, performance of the system is highly increased when semantic features from UMLS were added. After adding these features, by embedding MetaMap features it was improved more as well. These show that semantic features related to tokens can be highly effective to enhance performance of the system.

Studying on dependency trees and how they relate to NER systems is the topic for our future work. In fact this research area is the one that is not considered in many NER systems. By using the dependencies between tokens and finding to what other tokens they are related, a significant improvement might be achieved. We are currently working on this improve our system.

In Figure 5, an example of a dependency tree is illustrated. In this example diseases or disorders (headache and dizziness) are included in the last right sub tree. An idea here is to remove other sub trees that do not have any significant information. In this example, we can only keep the last right sub tree.

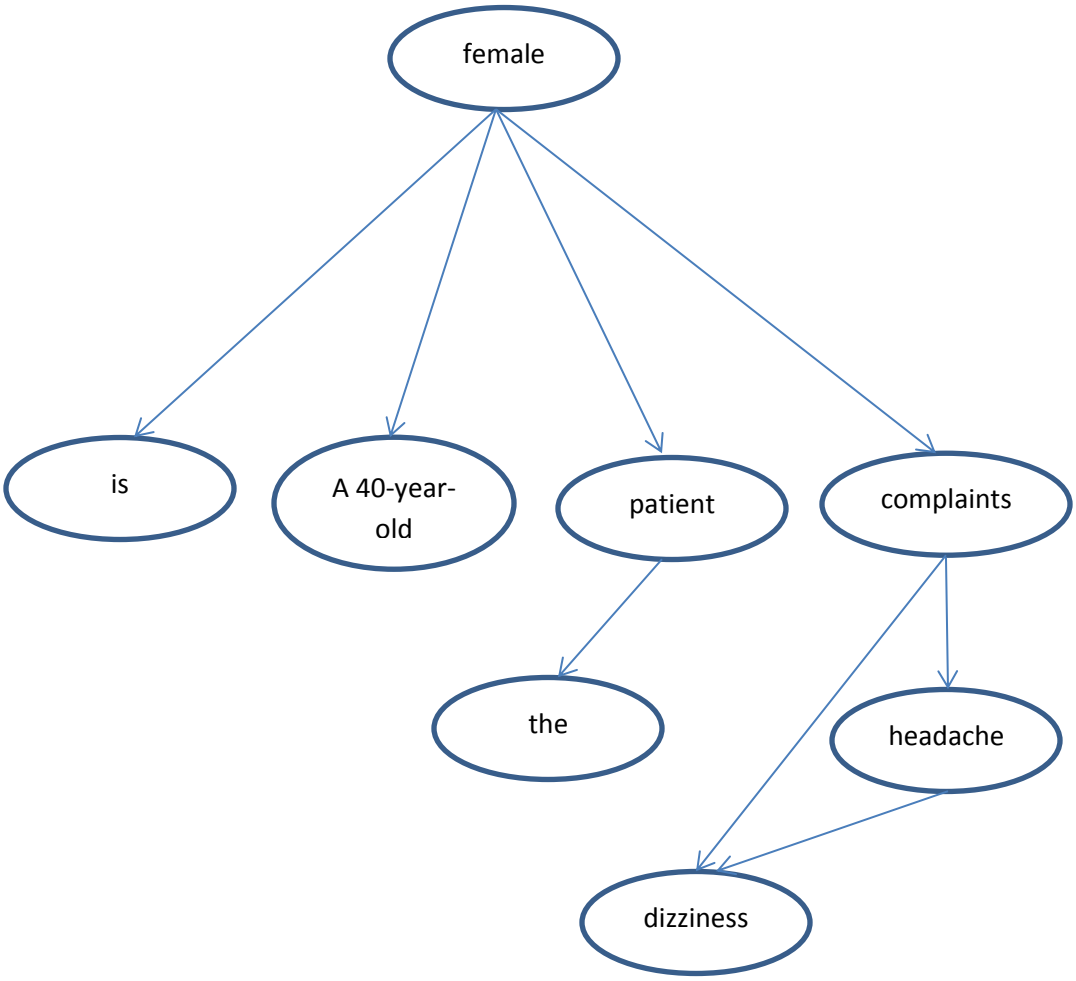

Figure 5 Collapsed dependency tree of example: The patient is a 40-year-old female with complaints of headache and dizziness

Moreover we are going to add an acronym system to improve our results. This system will disambiguate acronyms and find proper extensions of them. A kind of this system can be found in [47].

6.References

[1] Ulf Leser, Jörg Hakenberg, What makes a gene name?Named entity recognition in The biomedical literature, BRIEFINGS IN BIOINFORMATICS. VOL 6. NO 4. 357–369. 2005.

[2] Sanaa Chtioui, Evaluation of gene/protein name recognition Programs, Masters in Proteomics and Bioinformatics, University of Geneva, Geneva 2008.

[3] Jari Bjorne, Suwisa Kaewphan and Tapio Salakoski, UTurku: Drug Named Entity Recognition and Drug-Drug Interaction Extraction Using SVM Classification and Domain Knowledge, Proceeding of SemEval-2013, York, 2013.

[4] Carreras, Xavier; Màrquez, Lluís; Padró, Lluís (2003). "A simple named entity extractor using AdaBoost". CoNLL.

[5] http://www.medlingmap.org/node/2

[6] http://metamap.nlm.nih.gov/

[7] https://ctakes.apache.org/

[8] Andreea Bodnari, Louise Deleger, Thomas Lavergne, Aurelie Neveol and Pierre Zweigenbaum, A Supervised Named-Entity Extraction System for Medical Text**,** Online Working Notes of the CLEF 2013 Evaluation Labs and Workshop, 23 - 26 September, Valencia – Spain.

[9] James Cogley, Nicola Stokes and Joe Carthy, Medical Disorder Recognition with Structural Support Vector Machines, Online Working Notes of the CLEF 2013 Evaluation Labs and Workshop, 23 - 26 September, Valencia – Spain.

[10] Jung-wei Fan, Nav deep Sood and Yang Huang, Disorder Concept Identification from Clinical Notes: an Experience with the ShARe/CLEF 2013 Challenge,

Online Working Notes of the CLEF 2013 Evaluation Labs and Workshop, 23 - 26 September, Valencia – Spain.

[11] James Gung, Using Relations for Identification and Normalization of Disorders: Team CLEAR in the ShARe/CLEF 2013 eHealth Evaluation Lab, Online Working Notes of the CLEF 2013 Evaluation Labs and Workshop, 23 - 26 September, Valencia – Spain.

[12] Lucia Hervas, Victor Martinez, Irene Sanchez and Alberto Diaz , UCM at CLEF eHealth 2013 Shared Task1, Online Working Notes of the CLEF 2013 Evaluation Labs and Workshop, 23 - 26 September, Valencia – Spain.

[13] Robert Leaman, Ritu Khare and Zhiyong Lu, NCBI at 2013 ShARe/CLEF eHealth Shared Task: Disorder Normalization in Clinical Notes with DNorm, Online Working Notes of the CLEF 2013 Evaluation Labs and Workshop, 23 - 26 September, Valencia – Spain.

[14] Hongfang Liu, Kavishwar Wagholikar, Siddhartha Jonnalagadda and Sunghwan Sohn, Integrated cTAKES for Concept Mention Detection and Normalization, Online Working Notes of the CLEF 2013 Evaluation Labs and Workshop, 23 - 26 September, Valencia – Spain.

[15] John David Osborne, Binod Gyawali and Thamar Solorio, Evaluation of YTEX and MetaMap for Clinical Concept Recognition, Online Working Notes of the CLEF 2013 Evaluation Labs and Workshop, 23 - 26 September, Valencia – Spain.

[16] Jon D. Patrick, Leila Safari and Ying Ou, ShARe/CLEF eHealth 2013 Named Entity Recognition and Normalization of Disorders Challenge, Online Working

Notes of the CLEF 2013 Evaluation Labs and Workshop, 23 - 26 September, Valencia – Spain.

[17] S. V. Ramanan, Shereen Broido and P. Senthil Nathan, Performance of a Multi-class Biomedical Tagger on Clinical Records, Online Working Notes of the CLEF 2013 Evaluation Labs and Workshop, 23 - 26 September, Valencia – Spain.

[18] Mitchell, T. (1997). *Machine Learning*, McGraw Hill. ISBN 0-07-042807-7

[19] Tim Finin, Will Murnane, Anand Karandikar, Nicholas Keller and Justin Martineau, Annotating named entities in Twitter data with crowdsourcing, Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk**,** Pages 80-88, Stroudsburg, PA, USA, 2010.

[20] Buzhou Tang, Yonghui Wu, Min Jiang, Joshua C. Denny and Hua Xu**,** Recognizing and Encoding Discorder Concepts in Clinical Text using Machine Learning and Vector Space Model, Online Working Notes of the CLEF 2013 Evaluation Labs and Workshop, 23 - 26 September, Valencia – Spain.

[21] http://en.wikipedia.org/wiki/SemEval

[22] http://alt.qcri.org/semeval2014/

[23] http://alt.qcri.org/semeval2014/task7/

[24] Thomas Lavergne, Olivier Capp, and Fran cois Yvon. Practical very large scale CRFs. In ACL Proc, pages 504-513, 2010.

[25] http://clear.colorado.edu/compsem/

[26] Danushka Bollegala, Naoaki Okazaki, Mitsuru Ishizuka. A Bottom-Up Approach to Sentence Ordering for Multi-Document Summarization. Thierry Poibeau and

Horacio Saggion and Jakub Piskorski and Roman Yangarber (eds.), *Multi-source, Multilingual Information Extraction and Summarization*, ISBN: 978-3-642-28568-4, chapter 12, pp. 253—276, September 2012. New York.

[27] http://www.chokkan.org/software/crfsuite/

[28] Carpenter, Bob and Breck Baldwin. 2011. Text Analysis with LingPipe 4.

[29] Leaman, R., Gonzalez, G.: BANNER: an executable survey of advances in biomedical named entity recognition. Pac. Symp. Biocomput. pp. 652-663 (2008).

[30] Leaman, R., Miller, C., Gonzalez, G.: Enabling Recognition of Diseases in Biomedical Text with Machine Learning: Corpus and Benchmark. Proceedings of the 2009 Symposium on Languages in Biology and Medicine, pp. 82-89 (2009).

[31] http://cbioc.eas.asu.edu/banner/

[32] J. Patrick, Y. Wang, & P. Budd, "An automated system for conversion of clinical notes into SNOMED clinical terminology", in Proceedings of the fifth Australasian symposium on ACSW frontiers - Volume 68, Australian Computer Society, Inc.: Ballarat, Australia. pp. 219-226, 2007.

[33] Sergei Nirenburg, Victor Raskin, The sub-world concept lexicon and the lexicon management system, Computational Linguistics, Volume 13, Numbers 3-4, July-December 1987.

[34] http://npjoint.com/annotate.php

[35] *Zoubin Ghahremani, An Introduction to Hidden Markov Models* and Bayesian Networks, International journal of pattern recognition and artificial intelligence, 15(1):9-42, 2001.

[36] Charles Sutton, Andrew McCallum, An Introduction to Conditional Random Fields, Foundations and Trends in sample, ol. 4, No. 4 pages 267–373, 2011.

[37] Rustem Takhanov, Vladimir Kolmogorov, Inference algorithms for pattern-based CRFs on sequence data, Proceedings of the 30th International Conference on Machine Learning, Atlanta, Georgia, USA, 2013.

[38] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. Proc. 18th International Conf. on Machine Learning, 2001.

[39] Unified Medical Language System (UMLS) Manual, National Library of Medicine, USA, 2009.

[40] http://www.chokkan.org/software/crfsuite/

[41] Jesse Davis, Mark Goadrich, The Relationship between Precision-Recall and ROC-Curves, Proceeding of 23rd international conference, on machine learning, Pittsburgh, PA 2006.

[42] Asif Ekbal, Sivaji Bandyopadhyay, A Conditional Random Field Approach for Named Entity Recognition in Bengali and Hindi, Linguistic Issues in Language Technology, Volume 2, Issue 1, 2009.

[43] Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple,Graham Katz, James Pustejovsky, SemEval-2007 Task 15: TempEval Temporal Relation Identification, Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007) , pages 75–80, Prague, June 2007.

[44] Wanda Pratt and Meliha Yetisgen-Yildiz, A Study of Biomedical Concept Identification: MetaMap vs. People, AMIA Annul Symposium Proceeding, pages 529-533, Washington DC, USA, 2003.

[45] Ki-Joong Lee, Young-Sook Hwang, Seonho Kim, Hae-Chang Rim, Biomedical named entity recognition using two-phase model based on SVMs, Journal of Biomedical Informatics, Volume 37, Issue 6, December 2004, Pages 436–447.

[46] http://en.wikipedia.org/wiki/List_of_medical_abbreviations

[47] Xu H, Stetson PD, Friedman, C. Combining corpus-derived sense profiles with estimated frequency information to disambiguate clinical abbreviations. AMIA Annual Symposium Proceeding, 2012. 1004-13.