Lisa Deckert, BSc

Medical Entity Recognition and Semantic Type Classification from Clinical Discharge Letters

MASTER'S THESIS

to achieve the university degree of

Master of Science

Individual Master's degree programme:
Bioinformatics and Medical Informatics

submitted to

Graz University of Technology

Supervisor

Dip.-Ing. Dr.techn. Univ.-Doz. Christian Gütl

Institute of Information Systems and Computer Media

Graz, August 2016

In cooperation with

THE UNIVERSITY OF WESTERN AUSTRALIA

The University of Western Australia

35 Stirling Highway

Crawley, WA 6009

Australia

Co-Supervisor

Dr. Wei Liu

School of Computer Science and Software Engineering

Lisa Deckert, BSc

Medizinische Begriffserkennung und Klassifizierung semantischer Typen anhand von klinischen Arztbriefen

MASTERARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieurin

Individuelles Masterstudium Biomedical Engineering:
Bioinformatik und Medizinische Informatik

eingereicht an der

Technischen Universität Graz

Betreuer

Dip.-Ing. Dr.techn. Univ.-Doz. Christian Gütl

Institut für Informationssysteme und Computer Medien

Graz, August 2016

In Kooperation mit

THE UNIVERSITY OF WESTERN AUSTRALIA

The University of Western Australia

35 Stirling Highway

Crawley, WA 6009

Australia

Co-Betreuerin

Dr. Wei Liu

School of Computer Science and Software Engineering

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis dissertation.

Graz, _____ _____

Date Signature

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdocument ist mit der vorliegenden Masterarbeit/Diplomarbeit identisch.

Graz, am _____ _____

Datum Unterschrift

Abstract

The automatic extraction of valuable information from clinical documents is a main goal of medical Natural Language Processing (NLP). Identifying and classifying medical terms in unstructured text plays a fundamental role in this information extraction process.

This thesis presents a medical entity recognition and semantic type classification system. Based on the insights gained form a background study and a literature research, supervised learning and active learning approaches using Conditional Random Fields (CRF) and Inside Outside Beginning (IOB) format labels are introduced. The supervised classifier is implemented in three different ways: Classifier Version 1 (CV1) uses a single-step approach, Classifier Version 2 (CV2) adds an extra candidate phrase extraction step, and Classifier Version 3 (CV3) performs the identification of the medical entity and the semantic type classification in two separate stages. The active learning approach builds on CV1 and uses uncertainty-based sampling for query selection.

The most successful supervised approach was CV2 with a F1-score of 0.98. CV1 reached very similar performance although being more time consuming than the other approaches. Overall, the simpler approaches (CV1 and CV2) outperformed the stepwise model (CV3). The active learning approach using uncertainty-based sampling considerably outperformed a random baseline and was able to achieve a F1-score of 0.95 within 200 iteration. The active learning system was able to reach a desirable performance, while requiring significantly less training data than the supervised approaches. Future work could include extending the used feature set with semantic features or word representations, or employing alternative query sampling methods for active learning, such as diversity-based methods.

Kurzfassung

Das automatische Extrahieren von wertvollen Informationen aus klinischen Dokumenten ist ein Hauptziel des medizinischen Natural Language Processing (NLP). Die Identifizierung und Klassifizierung von medizinischen Begriffen in unstrukturiertem Text spielt eine fundamentale Rolle in diesem Prozess der Informationsextraktion.

Diese Arbeit präsentiert ein System zur medizinischen Begriffserkennung und Klassifizierung semantischer Typen. Auf Grundlage der gewonnenen Erkenntnisse der Hintergrundstudie und der Literaturrecherche wurde ein überwachtes und ein aktives Lernmodell zusammen mit Conditional Random Fields (CRF) und Inside Outside Beginning (IOB) Format vorgestellt. Überwachte Klassifizierung wird in drei Varianten umgesetzt: Classifier Version 1 (CV1) führt medizinische Begriffserkennung mittels eines einzelnen Schrittes durch, Classifier Version 2 (CV2) durch Hinzufügen eines zusätzlichen Extraktionsschritt von Kandidatenphrasen, und Classifier Version 3 (CV3) durch Ausführen der Identifikation der medizinischen Begriffe und der Klassifizierung der semantischen Typen in zwei getrennten Stufen. Der aktive Lernansatz baut auf CV1 auf und wählt die Trainingssätze basierend auf Unsicherheit aus.

Der erfolgreichste überwachte Ansatz war CV2 mit einer F1-Score von 0.98. CV1 erreichte eine sehr ähnliche Leistung, hatte jedoch einen höheren Zeitaufwand als die anderen Ansätze. Insgesamt übertrafen die einfacheren Ansätze (CV1 und CV2) das Mehrstufenmodell (CV3). Der aktive Lernansatz mit Auswahl der Trainingsdaten basierend auf Unsicherheit schnitt deutlich besser ab als eine zufällige Auswahl und konnte eine F1-Score von 0.95 innerhalb von 200 Iteration erzielen. Das aktive Lernsystem konnte eine gewünschte Leistung erreichen, während deutlich weniger Trainingsdaten erforderlich waren als für die überwachten Ansätze. Zukünftige Forschung könnte unter anderem die Erweiterung des Feature Sets durch semantische Informationen oder Wortdarstellungen, oder auch die Verwendung andere Methoden der Trainingsdatenauswahl beim aktiven Lernansatz beinhalten.

Table of Contents

List of Figures

List of Tables

Abbreviations

BILOU	Beginning Inside Last Outside Unit.
CLEF	Conference and Labs of the Evaluation Forum.
CoNLL-2003	Conference on Natural Language Learning 2003.
CRF	Conditional Random Fields.
cTakes	Clincal Text Analysis and Knowledge Extraction System.
CV1	Classifier Version 1.
CV2	Classifier Version 2.
CV3	Classifier Version 3.
FN	False Negative.
FP	False Positive.
GATE	General Architecture for Text Engineering.
HMM	Hidden Markov Models.
i2b2	Informatics for Integrating Biology and the Bedside.
IOB	Inside Outside Beginning.
L-BFGS	Limited-Memory Broyden–Fletcher–Goldfarb–Shanno.
LSP-MLP	Linguistic String Project - Medical Language Processor.
ME	Maximum Entropy.
MedLEE	Medical Language Extraction and Encoding System.
MUC-6	Message Understanding Conference-6.
NED	Named Entity Disambiguation.
NER	Named Entity Recognition.

NLM	National Library of Medicine.
NLP	Natural Language Processing.
NLTK	Natural Language Toolkit.
NP	Noun Phrase.
NP-chunking	Noun Phrase Chunking.
POS	Part-of-Speech.
SemEval	Semantic Evaluation.
SPRUS	Special Purpose Radiology Understanding System.
SVM	Support Vector Machines.
tf-idf	Term Frequency - Inverse Document Frequency.
TN	True Negative.
TP	True Positive.
UIMA	Unstructured Information Management Architecture.
UMLS	Unified Medical Language System.

1 Introduction

> *Hiding within those mounds of data is knowledge that could change the life of a patient, or change the world.*
>
> -
>
> Atul Butte, Stanford (Goldman, 2012)

This quote appeared in an online article of the Stanford Medicine Magazine from 2012 about big data analysis in medicine[1] (Goldman, 2012). The major hypothesis is that various important medical questions can be solved by analyzing the already existing medical data. According to Atul Butte in the article by Goldman (2012), many insightful answers may be concealed in the vast amount of unstructured data. The great question is: How can we access this hidden knowledge?

1.1 Motivation and Background

The amount of available medical data is growing constantly. In 2014 there were around 2.800.000 inpatient hospital stays in Austria[2], which is a third of the total population. This number has been growing constantly over the past 25 years, as shown in Figure 1.1. On top of patients who have been treated in hospitals there have been even more people searching the services of general practitioners, dentists, or some type of specialists. In Austria and in other European countries as well, the number of practising physician is steadily

[1] King of the Mountain, Stanford Medicine Magazine: http://sm.stanford.edu/archive/stanmed/2012summer/article3.html

[2] Statistik Austria: http://www.statistik.at/

growing according to Eurostat[3], which can be seen in Figure 1.2. Each of these patient-doctor-interactions produces large quantities of data which is usually stored in unstructured text. Most medical data is highly informative and could be useful for many applications, however the underlying information needs to be made accessible by computers. Humans are able to process plain text easily, but are not suited to handle the vast amount of textual data created in the medical domain every day. This challenge is the motivational foundation of the computer science field of Natural Language Processing (NLP).

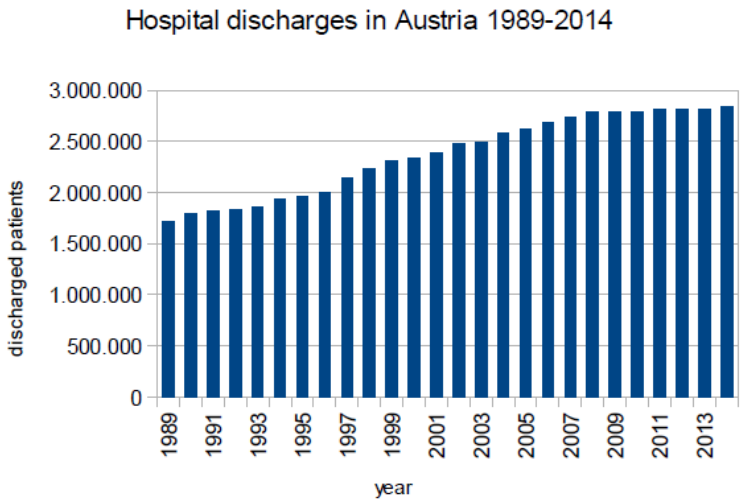


Figure 1.1: Number of hospital discharges in Austria from 1989 to 2014 according to Statistik Austria

The main ambition of NLP is to create systems for automatically handling and understanding human language. A key step in this process is the identification of certain concepts in text (Liu, Chung, Wang, Ng, & Morlet, 2015). In the medical domain, concepts such as diseases or symptoms are highly interesting to identify. This identification and classification task is called medical entity recognition and is also referred to as medical named entity recognition or even as named entity recognition in the medical domain (Abacha & Zweigenbaum, 2011). Medical entity recognition still faces many challenges due to the peculiarities

[3]Eurostat: http://ec.europa.eu/eurostat

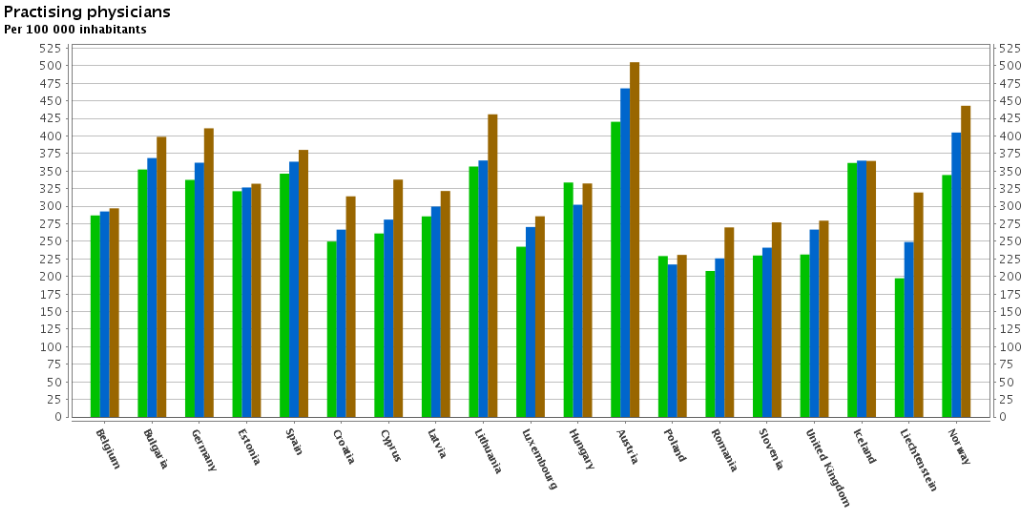


Figure 1.2: Practising physicians in European countries per 100 000 inhabitants in the years 2004, 2009, and 2014 according to Eurostat

of medical language and the poor availability of medical document collections for research (Y. Xu, Hong, Tsujii, & Chang, 2012; Cohen, 2005; Friedman, Rindflesch, & Corn, 2013).

To get a better picture of the advantages and possible applications of a medical entity recognition system, consider the following questions about patient history, decision support, and potential concept dependencies:

- Which medication did the patient receive the last time these symptoms were presented?
- How was the course of action for other patients of a similar age exhibiting similar symptoms?
- Can a certain disease combined with a specific symptom or treatment lead to another disease?

NLP can help find answers to these questions and yet more. With the data in a structured format, practical information can be gained and used for many possible applications ranging from search queries to data analysis or recommendation systems. Without doubt, there is a high demand for reliable medical entity recognition systems.

1.2 Problem Definition

According to Tao, Song, Sharma, and Chute (2013) more than 80% of the existing biomedical data is in plain text format. The valuable information encoded in these natural language texts has to be extracted using advanced tools and algorithms. The goal of this thesis is to create a system for automatically extracting medical terms and classifying the identified terms as one of a predefined semantic type. The system aims to effectively deal with the peculiarities of medical natural language in a specific domain, while still being adaptable to other fields. Basic and more elaborate machine learning approaches for this problem will be explored. After an extensive literature study, the best suited methods will be implemented, including supervised methods and less popular alternatives. This work will then conclude by presenting the findings and compare the various implemented approaches.

1.3 Outline of this Thesis

At first, Chapter 2 gives insights into theoretical concepts which form the basis of this thesis. After an introduction to NLP in general, medical entity recognition methods are described. Elementary NLP techniques are presented as well as the different machine learning approaches.

Then, the current state-of-the-art in medical entity recognition is reviewed in Chapter 3. Relevant publications and similar approaches are discussed. The insights into related work provide guidance for selecting the best suited methods for the given task.

Chapter 4 describes the methodology of this thesis which builds on the findings of the previous chapter. After explaining the fundamental design decisions, the conceptual architecture is depicted to give an overview of the idea. Then, the used tools and libraries are described.

The experimental setup is presented in more detail in Chapter 5. A data set of the medical domain is used to evaluate the implemented system. Common performance measures are calculated and the results are discussed.

This work then concludes by summarizing the most important findings in Chapter 6. At last, the difficulties and encountered problems are pointed out and further research ideas are given as an outlook.

2 Background

The goal of this thesis is creating a system for medical entity recognition and semantic type classification. In order to tackle this challenge, knowledge and expertise from various research fields have to be combined. The developed system should be based on state-of-the-art machine learning methods and well established NLP techniques. In this section the theoretical foundations of this research area are explained. The basic concepts as well as more specific approaches are described at this point. The theories and concepts presented in this chapter are later put into practice when the medical entity recognition system is designed.

2.1 Natural Language Processing

The field of NLP aims to automatically understand and manipulate human language by using intelligent computer systems. Advances in this field go back to the 1960's when the first simple NLP systems were published, e.g. ELIZA (Weizenbaum, 1966). The foundations of this research area are computer science, artificial intelligence and linguistics. Possible applications for NLP are machine translations, human-computer interaction, and information retrieval (Chowdhury, 2005).

Even though NLP may seem simple at a first glance, it is in fact a rather difficult task. Humans usually do not realize how much knowledge is needed for understanding natural language, but even babies first have to learn it after they are born. Challenges in understanding natural language are given by its great variety, expressiveness, ambiguity and vagueness (Friedman & Hripcsak, 1999).

There are several components of natural language. The syntactic information consists of the structure of the sentence, including the Part-of-Speech (POS) of the words. The semantic component contains information about the meaning of words and sentences and the order in which words are combined to form certain phrases. Another important component is the domain knowledge, which is basically information about the subject itself, e.g. specific terms of a certain medical domain (Friedman & Hripcsak, 1999).

The main aspect of this work is information extraction through NLP. Automatically extracting and annotating useful information from a collection of documents or other information sources is the major goal. An overview of the general NLP process is shown in Figure 2.1, according to Friedman et al. (2013). The right side shows the operational parts of NLP. These may involve methods and tools for classification, feature selection, or labelling formats as well as a complete system to combine various parts. On the left side, the figure shows the resources needed for a NLP system. A sample data set is commonly used for training the model. Feature engineering as well as the design of the model usually build on domain knowledge and linguistic knowledge. A successful NLP system produces a structured output which can then be used for further applications (Friedman et al., 2013).

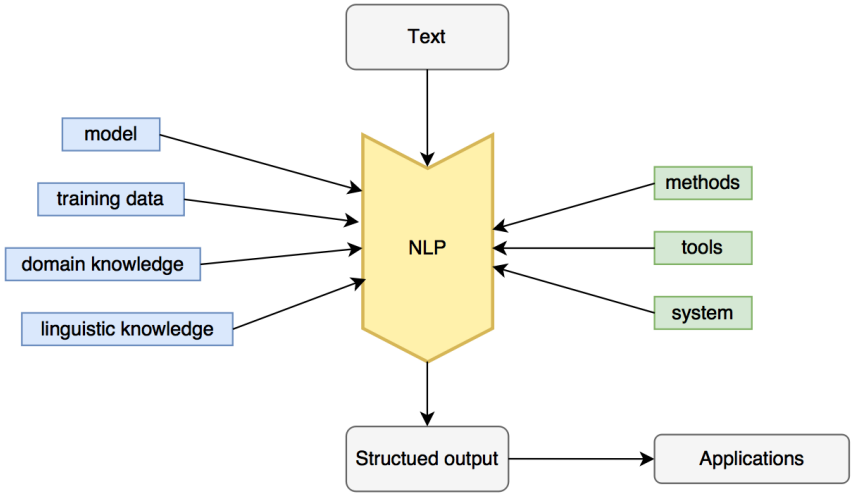


Figure 2.1: Overview of the general NLP process. Image adapted from Friedman, Rindflesch, and Corn (2013).

2.2 Medical Natural Language Processing

Twenty years ago medical NLP was considered as one of the most challenging tasks in medical information retrieval. Most of the medical information today is in the form of free text, e.g. doctors letters, discharge letters. Even though the advantages of storing information in a structured form such as databases are obvious, the larger amount of medical documentation remains in plain natural language text. A straightforward explanation would be that natural language is simply the easiest way to communicate complex information (Spyns, 1996). However, it increases the difficulty of automatically accessing the important underlying information. NLP tries to tackle exactly this problem and a crucial aspect is the entity extraction and recognition step.

2.3 Named Entity Recognition and Disambiguation

The task of Named Entity Recognition (NER), which is an important subtask of NLP (Doan, Collier, Xu, Pham, & Tu, 2012), was first introduced in 1996 at the Message Understanding Conference-6 (MUC-6) (Grishman & Sundheim, 1996). Named entities are terms of a specific type, i.e. names of things. Most studied types of named entities are *persons*, *locations*, and *organizations* (Nadeau & Sekine, 2007). The basic goal of NER is to identify all occurrences of named entities within a collection of documents (Cohen, 2005).

According to Ratinov and Roth (2009) the fundamental parts of designing a NER system are (1) finding a representation of text chunks, (2) choosing an inference algorithm, (3) modelling non-local dependencies, and (4) using external knowledge sources.

The general approaches of NER are the following (Abacha & Zweigenbaum, 2011; Cohen, 2005):

- knowledge-based
  - based on lexicons or dictionaries
  - based on expert rules

- linguistic-based
  - based on syntactic or lexical rules
- statistical
  - methods such as graph-based models, deep learning etc.
- hybrid
  - any combination of the above

Knowledge-based and linguistic-based can both be considered rule-based methods. The main advantages of these methods are that there is no learning or preprocessing step needed and the results are easily reproducible. However they both depend on prior knowledge, since they either rely on a construction of a knowledge base or a prior construction of carefully hand-crafted rules, which is both time consuming. Statistical machine learning methods on the other hand need a large amount of training data, but do not demand a knowledge base or any other prior knowledge sources (Abacha & Zweigenbaum, 2011). Hybrid approaches are relatively new and try to combine the advantages of machine learning and rule-based methods. Many modern systems belong to this category.

The goal of a NER system is to predict the type of a word or phrase, thus the usual output is a set of tags, which assign the types of named entities to the terms (Cohen, 2005). A simple example for NER is shown in Figure 2.2. This image was created with the Stanford NER online tool[1], which uses linear chain Conditional Random Fields (CRF) sequence models (Finkel, Grenager, & Manning, 2005).

2.3.1 Named Entity Disambiguation

The goal of Named Entity Disambiguation (NED) is to assign the identified entities in the text to the concepts in a knowledge base or ontology (Y. Li et al., 2013). The ambiguity of terms presents the major challenge for this task. A proper name may refer to more than one named entity, e.g. different people may have the same name (Bunescu & Pasca, 2006). To tackle this task

[1]Stanford Named Entity Tagger: http://nlp.stanford.edu:8080/ner/

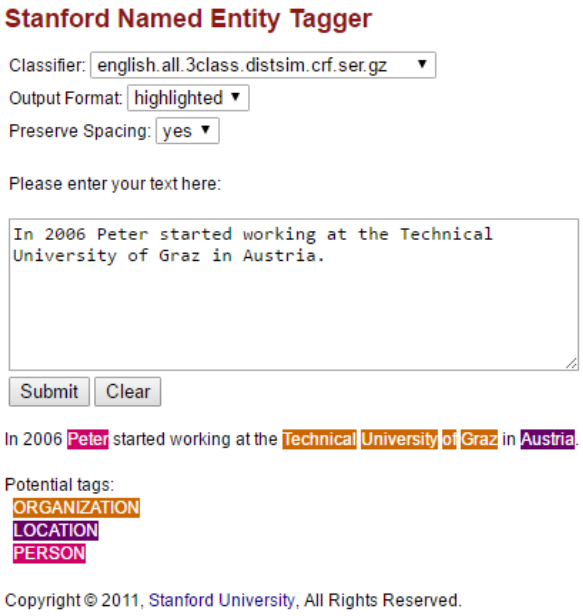


Figure 2.2: Basic example of NER using the Stanford NER tool

contextual clues and prior background knowledge can be very useful. Take the sentence *"Anna went to Columbia."* for example. The mention of "Columbia" could have multiple meanings. It could be the capital city of South Carolina, Columbia University, or even the commonly misspelled country Colombia. On the other hand, consider the sentence *"Anna is studying at Columbia."*. In this case it is clear through the context and the knowledge that one studies at a university that the mentioned entity is Columbia University.

NED is an interesting and challenging field. For this thesis however, it does not have much importance because this work deals only with data from a very specific medical domain.

2.3.2 Multilingual Named Entity Recognition

While English is the most common language, NLP is also interesting for other languages. An important factor for multilingual NER is to include language-specific knowledge in the system. However, a general problem for this task is

the low amount of labeled data. According to Faruqui and Padó (2010) the only available annotated data set in German is the data from the Conference on Natural Language Learning 2003 (CoNLL-2003) shared task. Nevertheless, Wikipedia is a well-known resource for multilingual approaches, because it contains editions in about 200 languages, is fast growing and dynamic (Bunescu & Pasca, 2006). Another popular data set for multilingual NLP are the proceedings of the European Parliament. The Europarl corpus is freely available and contains parallel texts in 11 different languages including German (Faruqui & Padó, 2010; Koehn, 2005).

2.4 Medical Entity Recognition and Semantic Type Classification

In the medical field many automatic applications could be developed using clinical information. However, most of the available data is in the form of unstructured text. To be able to use the textual data and access the information in a reliable way, it is essential to convert it to structured information. Medical entity recognition plays an important role here. The goal is to identify and classify medical entities in text. In the literature it is sometimes also mentioned as medical concept identification or concept mapping (S. Zhang & Elhadad, 2013). The main tasks of medical entity recognition and semantic type classification are (1) the identification of the entity and its boundaries within the sentence, and (2) the type classification of the located entity usually using a set of pre-defined categories (Abacha & Zweigenbaum, 2011; S. Zhang & Elhadad, 2013). Examples are given in Table 2.1 to show how these two steps are applied to plain sentences. Identification and classification of medical terms, can either be performed consecutively or simultaneously.

Previously, the dominating problem with medical information used to be the collection of sufficient data. Today however, the amount of medical data, such as clinical letters or patient records is immense, thus leading to the challenge of making use of this mostly unstructured data. As a consequence, the demand for systems which can process this data into useful information is high (Liu et al., 2015). Due to the sensitivity of medical data, accessibility of data is another

Example 1	
Sentence	Today her visual acuity was improved up to 6/6 in each eye and I ordered her some new glasses at her request.
Term identification	Today her [visual acuity] was improved up to 6/6 in each eye and I ordered her some new [glasses] at her request.
Term classification	Today her [visual acuity]$_{measurement}$ was improved up to 6/6 in each eye and I ordered her some new [glasses]$_{treatment}$ at her request.
Example 2	
Sentence	His eye movements were much more free, and he suffers less diplopia now.
Term identification	His [eye movements] were much more free, and he suffers less [diplopia] now.
Term classification	His [eye movements]$_{sign}$ were much more free, and he suffers less [diplopia]$_{symptom}$ now.

Table 2.1: Examples of medical entity recognition. Sentences are taken from the ophthalmology data set, which was provided for this thesis.

obstacle. For future research in this field an important aim is to provide greater accessability to full text collections Cohen (2005).

While NER may be considered as solved in many domains, medical entity recognition is still a challenging task as a result of the irregularities and ambiguities of medical terms (Gong, Yang, Feng, & Yang, 2015). The problem of boundary detection for medical concepts also increases the complexity (Uzuner, South, Shen, & DuVall, 2011). Further difficulties for medical entity recognition and type classification include different terminological variations, abbreviations, and multi-word names (Abacha & Zweigenbaum, 2011).

2.4.1 Nested biomedical entities

A common problem with NER and especially medical entity recognition are multi-word terms and nested entity terms, e.g. “inferior retinal vein occlusion”. These terms might include one or more other medical entities. A general starting point for entity recognition is to only process the term with the most words without addressing the problem of nested terms (Gong et al., 2015). However,

this might result in overlooking some interesting patterns or relationships between entities. In a study on the Informatics for Integrating Biology and the Bedside (i2b2)/VA 2010 Pittsburgh corpus which contains clinical notes and entities of the categories Problems, Test, and Treatments the authors found that 31% of the medical terms are nested terms (S. Zhang & Elhadad, 2013).

For instance, consider the medical term "inferior retinal vein occlusion". The full term, all its components and their semantic types can be seen in Table 2.2. This example is taken from a corpus of clinical letters which is explained in more detail in Section 5.2. An interesting observation which can be drawn from this example is that the whole term is of type diagnosis, but the compounds are of type anatomy followed by type diagnosis. This could possibly be a re-occurring pattern which would not have been detected if nested entities were ignored. Extra care has to be taken with this approach as to not count all of these entities as separate occurrence. This would lead to incorrect statistical measures, since the term still occurred only once in the document.

Medical Term	Category
vein	Anatomy
inferior retinal	Anatomy
retinal vein	Anatomy
vein occlusion	Diagnosis
retinal vein occlusion	Diagnosis
inferior retinal vein occlusion	Diagnosis

Table 2.2: Entities extracted from the term *inferior retinal vein occlusion*

2.4.2 Tagging format

The Inside Outside Beginning (IOB) format, sometimes also named BIO format, was first introduced by Ramshaw and Marcus (1995) for the application of Noun Phrase Chunking (NP-chunking). A Noun Phrase (NP) is a group of words with a noun as a head word and NP-chunking refers to identifying NPs in a sentence. By adding IOB labels to the NPs the authors transformed the chunking problem into a classification problem. With this format words are labelled B if at the beginning of an entity, I if they are inside, and O if they

are outside some entity. These IOB labels are added to the real label of the word.

Different variations or extensions of this format have been proposed. One of the most popular ones being the Beginning Inside Last Outside Unit (BILOU) format, which is applied for labelling in a similar fashion as the IOB format (Ratinov & Roth, 2009). The BILOU format is occasionally referred to as BIESO, which stands for beginning, inside, end, single, and outside. For explanatory purpose, consider at the sentence *"Tom has a black cat."*. The following example shows and how the IOB and the BILOU format can be used for NP-chunking:

chunked:	[(NP) Tom] has [(NP) a black cat] .
IOB tagged:	Tom/B-NP has/O a/B-NP black/I-NP cat/I-NP .
BILOU tagged:	Tom/U-NP has/O a/B-NP black/I-NP cat/L-NP .

2.5 Feature Space

In NLP the textual data is usually represented by features. A feature, in the context of entity recognition, is an attribute which describes a certain property of a word or phrase. Features can be numeric, boolean, or categorical. During feature engineering each word in the text is represented by a number of meaningful features, called a feature vector. The extracted features of the word are then used as input for the entity recognition algorithm. Different types of features are significant for NLP. According to Nadeau and Sekine (2007) and Suakkaphong, Zhang, and Chen (2011) features can be generally divided into lexical, syntactic, semantic, and document- and corpus-based features. Some examples for these feature categories are given in Table 2.3.

Lexical features both relate to word-level information and represent the characteristics and appearance of terms. Syntactic features are concerned with the structural properties of the words. They contain more information and are more difficult to model. Semantic features represent the meaning of the word or the affiliation to a certain pre-defined category. While these features are highly informative, the dictionaries, gazetteers, lexicons or lookup lists are very difficult to create and maintain. The lists are usually not complete and some categories might overlap. Further, semantic features are domain specific and

Types of Features	Examples
lexical features	word itself
	lowercase
	uppercase
	punctuation
	hyphens
	digits
	lexical patterns
	suffixes
	prefixes
syntactic features	POS-tags
	noun phrases
	bigrams
	n-grams
semantic features	semantic category
	list lookup
	dictionary matching
document and corpus features	term frequency
	tf-idf
	meta information
	position in document

Table 2.3: Examples of general feature categories

not easily transferable to other domains. Document and corpus based features go beyond the word level, as they are defined over the whole document or collection of documents, i.e. corpus. Depending on the classification task at hand, these features can contain valuable information (Nadeau & Sekine, 2007; Suakkaphong et al., 2011).

The features, or data representation, have a key influence on the performance and quality of the entity recognition model (Abacha & Zweigenbaum, 2011). A common risk is to select more features than necessary or features that are too complicated, which is referred to as overfitting. It can lead to increasing complexity without any benefits or even to worse performance by adding irrelevant features. Another downside of overfitting to a given data set is low portability to other domains (Hawkins, 2004). Thus, the overall goal of feature design is to chose highly informative features without overfitting.

2.6 Performance Measures

In order to conduct an organized experiment meaningful evaluation is a key element. The most common measures for evaluation are precision and recall, and the combination of the two, the F1-score (Cohen, 2005). The precision measure is the percentage of selected items that are correct, while the recall is the percentage of correct items that are selected. To calculate these measures the True Positive (TP), False Positive (FP), False Negative (FN) and True Negative (TN) values are needed. These can be seen in Table 2.4.

	correct	incorrect
selected	TP	FP
not selected	FN	TN

Table 2.4: Contingency table

The precision and the recall are computed with Equation 2.1 and 2.2 respectively. Another common measure is the accuracy which can be seen in Equation 2.3.

$$Precision = \frac{TP}{TP + FP} \tag{2.1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2.2}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.3}$$

The F1-score is calculated using the values for precision P and recall R which is shown in Equation 2.4 for the general case with a constant β, which is chosen in a way to give more weight to either precision or recall. Equation 2.5 shows the balanced F1-measure with $\beta = 1$.

$$F = \frac{(\beta^2 + 1)PR}{\beta^2 P + R} \tag{2.4}$$

$$F = \frac{2PR}{P + R} \tag{2.5}$$

2.7 Rule-based Methods for Entity Recognition

Rule-based systems require expert knowledge in order to develop classification rules. These rules are usually difficult to create and often not very robust (Brill, 1992). The first systems used carefully hand crafted rules, but most modern systems use machine learning to automatically create the rules. The great advantage of rule-based systems is that they have no need for a labelled training data set and thus are the best option when no annotated data is available (Nadeau & Sekine, 2007). Further, the common if-then rules are usually simple for humans to comprehend. Some example methods for automatically creating rules are extraction through the sequential covering algorithm or inducing rules from decision trees (Han, Pei, & Kamber, 2011).

2.7.1 Rule Extraction Using a Sequential Covering Algorithm

Sequential covering algorithms extract rules directly from the training data set. Rules are created sequentially and ideally address as many samples of the desired class as possible while not covering any samples from other classes. The rules should be high in accuracy and usually follow the basic if-then pattern. The rules are sequentially created until the desired quality is reached or until there is no more training data available (Han et al., 2011).

2.7.2 Rule Extraction from a Decision Tree

Another way of creating a rule-based system for NLP is inducing if-then rules from a decision tree. This is done in a simple manner by retracing all paths from the root node to each leaf and creating simple if-then rules connected through a logical *and* for every path. The extracted rules are then joined using the logical *or*. By the nature of a decision tree the created rules are mutually exclusive and unordered. Inducing rules from a decision tree is very straight forward, but can lead to an exhaustive set of rules. Further processing might be needed in order to decrease the amount of rules and to result in a clearly structured system (Han et al., 2011).

2.8 Machine Learning Methods for Entity Recognition

The first attempts in NER were rule-based systems. Lately however, machine learning techniques have become more popular, as long as there is a sufficient amount of data available. The different statistical approaches may be supervised, unsupervised or semi-supervised (Nadeau & Sekine, 2007).

2.8.1 Supervised Learning

Supervised learning approaches are the most prevalent techniques at the moment (Nadeau & Sekine, 2007). The foundation of all supervised learning methods is a large labelled training data set from which the model infers knowledge, i.e. the model is trained. The key elements for a high performance are the availability of a large labelled training data set and a carefully selected feature set (Abacha & Zweigenbaum, 2011). In NLP lexical and syntactic characteristics are exploited for extracting meaningful features from the text, such as POS tags. Some examples of supervised learning approaches are naive Bayes (Rish, 2001), Decision Trees (Quinlan, 1986), Hidden Markov Models (HMM) (Rabiner, 1989), Support Vector Machines (SVM) (Cortes, 1995), and CRF (Lafferty, McCallum, & Pereira, 2001). These methods all have the need for a large annotated document collection in common. Based on the annotations, the system is then able to create lists of entities and disambiguation rules (Nadeau & Sekine, 2007). The main challenge with supervised learning is the need for a large amount of annotated training data, which is often difficult, expensive, or time consuming to obtain and requires manual effort of human experts (Carlson, Betteridge, Wang, Hruschka, & Mitchell, 2010).

Generally, supervised learning models can be either generative or discriminative. Generative approaches model the joint distribution and make predictions by using the Bayes rule for calculating the conditional probability. Discriminative models, on the other hand, directly model the conditional probability distribution (Jordan & Ng, 2002; Sutton & McCallum, 2011). For most classification tasks, discriminative models are the preferred approach (Jordan & Ng, 2002) as they are better for handling rich and overlapping features (Sutton & McCallum, 2011). The following are brief descriptions of selected supervised learning approaches.

Naive Bayes

The naive Bayes classifier is a very basic generative approach. The Bayes' theorem serves as theoretical foundation and is applied to the classification problem, as shown in Equation 2.6.

$$P(C_k|X) = \frac{P(X|C_k)P(C_k)}{P(X)} \tag{2.6}$$

The class is denoted by C_k, given k different classes, and X represents the feature vector $X = (x_1, x_2, ... x_n)$. Given the feature vector X, the naive Bayesian classifier predicts the class with the highest posterior probability $P(C_k|X)$ using the Bayes' theorem in Equation 2.6. Since $P(X)$ is constant, only $P(X|C_k)P(C_k)$ needs to be maximized. To reduce the complexity of this computation, the classification algorithm builds on the simple assumption that the features are conditionally independent of each other given the class label, i.e. they exhibit class-conditional independence. This assumption facilitates the calculation of $P(X|C_k)$ as can be seen in Equation 2.7 (Han et al., 2011).

$$P(X|C_k) = \prod_{i=1}^{n} P(x_i|C_k) = P(x_1|C_k) \cdot P(x_2|C_k) \cdot ... \cdot P(x_n|C_k) \tag{2.7}$$

The naive assumption made by Bayesian classifiers is mostly unrealistic in practice, but greatly simplifies the computation. Naive Bayes classifiers are usually high in speed and are often able to compete with more sophisticated methods (Rish, 2001; Han et al., 2011).

Decision Tree Induction

The application of decision trees for classification tasks goes back to the 1980's. The characteristic of these methods is that the knowledge is represented in form of a decision tree. Typically starting at a root node, the internal nodes represent a test on an attribute, the branches denote the outcome of the test and the leaves symbolize the class labels. After the construction, classification rules can be easily induced from the decision tree. In general, these classifiers are intuitive, generative and computationally fast (Han et al., 2011; Quinlan, 1986).

HMM

Markov chains are the foundation of HMM and the original theoretic concepts were published by Baum and Petrie (1966). A HMM is a finite model based on generating the joint probability distribution of possible sequences (Eddy, 1996). In this generative approach, the probability of the current state is only dependent on the probability of the previous state, which makes it particularly useful for sequential data, such as DNA or sentences. The goal is to design a HMM capable of explaining an observed sequence without knowing the hidden stochastic process (Han et al., 2011). For a more detailed explanation of the underlying theory see Rabiner (1989).

SVM

SVM are a type of discriminative model originally intended for two class classification problems. The idea is to classify data points by constructing an optimal separation hyperplane. Apart from classification, regression tasks are also a common application for SVM (Cortes, 1995).

CRF

The main idea behind CRF is to build a discriminative probabilistic model for labelling sequence data. The advantage is that the conditional probability can depend on non-independent or arbitrary features of the observed sequence. Features may include different granularity levels of the same observation, such as characters, n-grams or words. Further, the conditional probability may also depend on the past and future observation which is taken into account as well. CRF use an exponential model for the probabilities of the entire label sequence given the observation sequence. In general, CRF perform very well in cases where the data distribution has dependencies of an higher order than the model, which is the case in natural language processing and in many other real life problem settings (Lafferty et al., 2001; Sha & Pereira, 2003; Sutton & McCallum, 2011).

2.8.2 Unsupervised Learning

On the contrary to supervised learning, unsupervised learning techniques are applied to unlabelled data and build on lexical structures and statistics. Unlabelled data is usually easy to collect, but not as useful and can often lead to poor performance (Carlson et al., 2010).

Clustering

Clustering is the most common unsupervised method for NER (Nadeau & Sekine, 2007). Basically clustering means to group unlabelled data points in a suitable way (Jain, Murty, & Flynn, 1999). One of the most prevalent algorithms for clustering is the k-means algorithm (MacQueen, 1967).

2.8.3 Semi-Supervised Learning

Semi-supervised learning tries to overcome the problem of collecting a large amount of labelled data by using only a small number of annotated data combined with a larger amount of unlabelled data. This approach is promising and can be very useful if not enough labelled data is available. However, most systems have not achieved an accuracy similar to the supervised approaches (Carlson et al., 2010). Some examples of wrapper-based semi-supervised machine learning methods are bootstrapping, co-training, and disagreement-based approaches.

Bootstrapping

Bootstrapping, or self-training, is an iterative wrapper method. The model uses the annotated data for training and classifies unlabelled data. The confident labels are added to the training data set and so forth. With this method the supervised model needs no modification. The main advantage of bootstrapping is its simplicity, however incorrect labels are carried on and have increasing impact with every iteration (Suakkaphong et al., 2011; Zhu & Goldberg, 2009).

Co-Training

Co-training is also an iterative wrapper technique for semi-supervised machine learning, which was first introduced by Blum and Mitchell (1998). Two or more classifiers are trained on different views of the data, which can be achieved by training different types of classifiers or by training the same classifier on different subsets of the data set. Then, the classifiers predict unlabelled data and the most confident labels are added to the training data set of the other classifier or classifiers. This process continues until all the unlabelled data has been labelled (Suakkaphong et al., 2011; Zhu & Goldberg, 2009).

Disagreement-based Learning

A further approach to semi-supervised learning is based on disagreement. Multiple learning algorithms are trained and collaborating with each other. The idea behind disagreement-based semi-supervised learning is to look at the instances where the learners disagree and let the learner with the highest confidence teach the other learner (Zhou & Li, 2010).

2.8.4 Ensemble Methods

Ensemble methods try to combine the results of two or more different classifiers. Ideally an ensemble of classifiers should perform better than the individual classifiers (Saha & Ekbal, 2013). In order to achieve a good performance the individual classifiers should differ from each other in where they are prone to make errors (Opitz & Maclin, 1999). Some examples for ensemble methods are voting, bagging, boosting, and stacking. By combining classifier results, more entities will be found, however also more false positive and false negative cases will also arise (Keretna, Lim, Creighton, & Shaban, 2014).

Voting

Different classifiers are known to perform better for a certain class than others. The ides of voting is that each classifier is allowed to vote for the cases in

which it performs well. An extension for this approach would be to add weights according to the confidence level of the classifier (Saha & Ekbal, 2013).

Bagging

In bagging (Breiman, 1996) the classifiers are trained on different subsets of the training data set. These subsets are usually randomly chosen from the data set. (Opitz & Maclin, 1999).

Boosting

Boosting (Freund, 1995; Schapire, 1990) is similar to bagging, with the exception that the training data subsets are chosen depending on the performance of an earlier classifier. Boosting creates a series of classifiers and presents each one with previously incorrectly predicted examples to increase the performance. On a downside boosting methods may overfit noisy data sets (Opitz & Maclin, 1999).

Stacking

The principles of stacking (Wolpert, 1992) are based on letting a higher level classifier correct the errors of multiple lower level classifiers. The final stage classifier makes a prediction combining the results of the previous classifiers (Hendrickx & Bosch, 2003).

2.8.5 Active Learning

Active learning, sometimes also known as human-in-the-loop learning, is based on the idea of using a human expert during the training process. By incorporating human knowledge in this way, the necessary amount of annotated training data can be significantly reduced. Basically the goal is to use less training samples while achieving high a accuracy through choosing the training data intelligently (Settles, 2010).

Active learning is an iterative approach and can be described as a series of steps. According to K. Zhang et al. (2014) the basic steps of an active learning system are as following:

1. train model with a small amount of annotated training data
2. use the model on unlabelled data
3. present some the data to an expert for labelling
4. add the new labelled data to the training data set
5. iterate steps 1-4 until convergence

A graphical representation of these steps as an active learning cycle (Settles, 2010) is shown in Figure 2.3. The key step in active learning is to chose which data is presented to the expert (step 3 in this case). A baseline approach is to choose randomly. However the goal is to select data that will contribute the most useful information to the training process. Approaches could be based on selecting the data with the highest uncertainty or with the most diversity (Chen, Lasko, Mei, Denny, & Xu, 2015). The large impact on time needed for training makes this step an essential task in active learning.

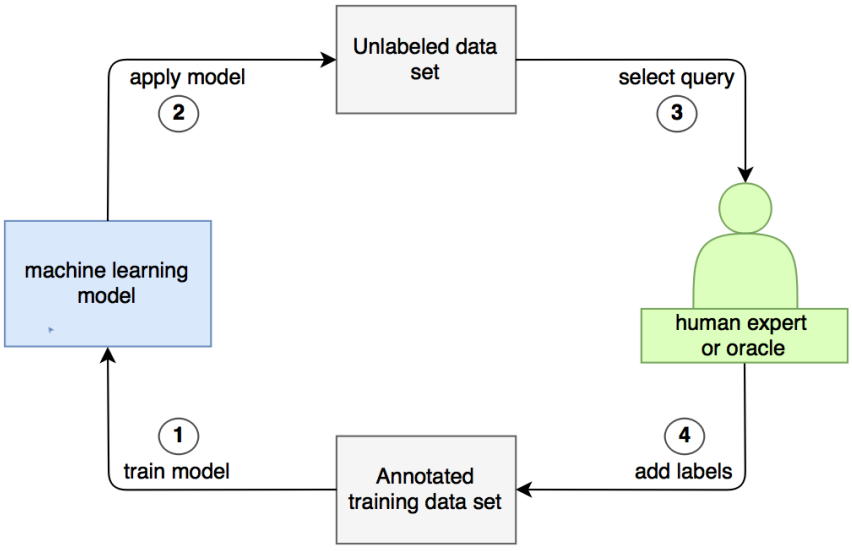


Figure 2.3: Representation of the active learning cycle. Image adapted from Settles (2010) combined with the basic steps according to K. Zhang et al. (2014).

Uncertainty-Based Sampling

Uncertainty sampling is based on the assumption that the most uncertain examples in the data set are also the most informative. To present the most uncertain data point to the expert would give the classifier the most useful information. There are different methods in how to determine the uncertainty. E.g. based on the posterior probability output from the CRF classifier, based on the entropy of the probability distribution, or based on the entropy of the individual words (Chen et al., 2015).

Diversity-Based Sampling.

As opposed to uncertainty sampling, diversity sampling is not dependent on the model and its results. It is based on the assumption that adding similar examples to the training data set does not bring much improvement in performance and thus the best strategy is to find query examples that differ to the already labelled data. Example approaches include word similarity, syntax similarity, and semantic similarity (Chen et al., 2015).

2.9 Summary

Generally, NLP is concerned with automatically understanding natural language. Many different research fields play a role in solving this problem, e.g. machine learning and linguistics. Information extraction from plain text is a major goal of NLP. An important subtask here is NER, which aims to automatically extract certain terms, such as *persons*, *locations*, and *organizations* from plain text.

Medical entity recognition, on the other hand, is concerned with extracting medical terms from text. The two main tasks in this field are (1) identification of the medical term and its boundary in the sentence and (2) classification of the term as one of a pre-defined semantic categories, such as *anatomy*, *disease* or *symptom* (Abacha & Zweigenbaum, 2011; S. Zhang & Elhadad, 2013).

Different approaches for medical entity recognition may be rule-based or based on machine learning. While rule-based methods have a need for expert knowledge (Brill, 1992), machine learning approached require a large annotated data set (Nadeau & Sekine, 2007). Different machine learning approaches are supervised learning, unsupervised learning, semi-supervised learning, ensemble methods, and active learning. The basic concepts have been described above and serve as a theoretical foundation of the task at hand.

3 Related Work

Essentially, the aim of any information extraction system is to turn data into information, information into knowledge, and finally knowledge into wisdom (Rowley, 2007). Medical entity recognition aims to extract information by identifying and classifying medical terms in text. This chapter on related work highlights published literature that is relevant for this thesis. The selected papers give an outline of previous work from the general and the medical domain. Finally, state-of-the-art NLP systems for the medical domain are described shortly to complete the overview of related work. The new insights gained from researching related work will help making mindful system design decisions.

3.1 Related Work in Named Entity Recognition for the General Domain

A lot of relevant research has been conducted in the general domain. In 2007, Nadeau and Sekine (2007) published a survey of named entity recognition and classification, which serves as a good starting point for of this section. In general, adaptability is a desirable feature for NER systems and thus much of the discoveries made in other domains are still useful for the medical problem setting.

3.1.1 Tagging Format

Ratinov and Roth (2009) did a comparison of several methods for the separate steps of NER in their work. Considering representation of terms, the authors

found the BILOU format was performing superior to the IOB format. The same results could be obtained in another study comparing the two different tagging formats by Tang, Cao, Wu, Jiang, and Xu (2012). However, the authors also stated that the more complicated BILOU format requires more training time than the simpler IOB format.

3.1.2 Semi-Supervised

Carlson et al. (2010) showed that by constraining the learning task by coupling the training of many extractors of categories, the performance of a semi-supervised system can improve greatly. The input of their system is an ontology describing the target categories and relations, some seed examples for each and a lot of unlabelled data. The iterative method starts by training multiple classifiers with the seed examples and then uses these classifiers for the unlabelled data. Confident new examples are then added to the seed examples for and the classification step is performed again.

3.1.3 Active Learning

Recently, K. Zhang et al. (2014) combined the approach of active learning with CRF for sentiment identification from online reviews. The authors used syntactic and semantic features for the CRF model and presented two different active learning strategies for choosing which data is to be presented to the human oracle. Both strategies performed better than the baseline approaches with an accuracy greater than 65%. Further adding and tuning of features and also incorporating the topic of the data is proposed for future research.

3.1.4 Multilingual Entity Recognition

The CoNLL-2003 shared task focused on the challenge of language-independent NER (Tjong Kim Sang & De Meulder, 2003). The best performing system in this task for both English and German used a combination of Maximum Entropy (ME) models, transformation-based learning, HMM, and robust risk minimization. Interestingly, this system achieved a recall of 89% for the English

data, but only 64% using the German data (Florian, Ittycheriah, Jing, & Zhang, 2003). A possible reasons for this rather big difference might be the higher morphological complexity of German which makes lemmatization more difficult. Another example for why NER might be harder for German data, is that capitalization is a good predictor for named entities in English, but does not have much use in German because all nouns are capitalized (Faruqui & Padó, 2010).

Apart from the CoNLL-2003 shared task, advances were made using Wikipedia as a knowledge source. Yosef, Spaniol, and Weikum (2014) developed a system for the Arabic language called AIDArabic. In another study, Nothman, Ringland, Radford, Murphy, and Curran (2013) classified each Wikipedia article into named entity types for 9 languages, including English, German, Spanish, and Russian. Hereby the authors produced a multilingual silver-standard corpus of training annotations for further research.

3.2 Related Work in Medical Entity Recognition

In medical entity recognition, there are some added challenges compared to the general domain, such as abbreviations, Latin words, and multi-word terms. The approaches and concepts described in the following papers have been designed for medical entity recognition specifically.

3.2.1 Previous Work on the Ophthalmology Data Set

Recently, Liu et al. (2015) presented a system for unsupervised medical term extraction from clinical letters. The experiment was performed on the same data set as used in the research of this thesis. The system consists of three complementary unsupervised approaches, which extract medical terms from documents and a genetic algorithm for integrating the extraction results to learn the best parameters. The used approaches were namely PrefixSpan (Pei et al., 2004), C-Value (Frantzi, Ananiadou, & Mima, 2000), and TextRank (Mihalcea & Tarau, 2004). The PrefixSpan algorithm uses n-gram frequencies for term extraction based on sequential pattern mining. The C-Value approach is based on statistical and linguistic information, and ranks the terms considering their

length, frequency and the frequency of their sub-phrases. The term extraction in the TextRank algorithm is performed by using a co-occurrence graph for candidate ranking. The three different approaches were combined using a genetic algorithm ensemble method. The proposed system achieved a f-score of 72.47% with minimal annotation, thus performed considerably better than the extraction approaches by themselves.

3.2.2 Conditional Random Fields and Feature Selection

In an article presented by Abacha and Zweigenbaum (2011) different methods for medical entity recognition were compared: (1) a semantic and rule-based method which builds on MetaMap, (2) a statistical method with a SVM classifier, (3) a statistical method using the IOB format and a CRF classifier, and (4) a hybrid method using the features from method 1 and combining it with the statistical methods from 3. The best performing method was the hybrid approach. The system uses the output of the domain knowledge based approach, transforms it into the IOB format and trains the CRF with these tags as features.

Wang (2009) presented a CRF model which uses IOB formatting. The features used in the study include lexical, orthographic, semantic features, but no syntactic features as they were dealing with a specific type of medical notes without much grammatical structure. The system was able to outperform the baseline method. For future work the author mentioned making changes to the semantic classes and dividing some of the classes into further categories.

For the 2013 Conference and Labs of the Evaluation Forum (CLEF) eHEALTH challenge, Bodnari, Deléger, Lavergne, Névéol, and Zweigenbaum (2013) presented a supervised CRF model for the identification of disorder entities from electronic health records. They chose to use lexical and morphological features containing information about the lemma of the terms in form of unigram and bigram features, syntactic features, such as POS tags. Another feature was the type of the document in general and the authors further included Unified Medical Language System (UMLS) features and also features generated with Wikipedia. Additionally, the authors also proposed to use more textual data and to include the Brown word clustering information as features for future work.

Keretna, Lim, and Creighton (2015) proposed a feature extraction technique based on graphs. The idea is to create a graph representing the given unstructured medical text and extract helpful features from this graph. In addition to the novel features the authors also used POS tags, suffixes, Term Frequency - Inverse Document Frequencys (tf-idfs), orthographic features, preceding and following words as contextual features. The approach could increase performance in 5 out of 6 cases with the only exception being the CRF classifier.

3.2.3 Ensemble methods

An approach for dealing with the peculiarities of medical language was presented by Y. Xu et al. (2012). The authors created two CRF models for entity recognition, one for standard natural language and one specifically designed for telegraphic sentences which are typical in many medical letters, patient records, or discharge summaries. The proposed system dynamically switches between the two models and performed better in combination than either of the two by itself. For the standard language model the authors chose lexical, syntactic, ontological and word features. For the model for telegraphic sentences the features consisted of lexical, syntactic, ontological and sentence information.

Keretna et al. (2014) proposed an ensemble approach, which combines a CRF with a ME classifier. The presented system performed considerably better than the two classifiers applied separately. Different features are used for the different classifies in order to achieve a diversity which leads to better ensemble results. For the ME classifier the used features were of linguistic nature and consisted of n-grams, the word itself, and the frequency of the word in the data set, while the features for the CRF classifier were contextual and contained the shape of the word, prefixes, suffixes, and the previous and following word. For further work the authors propose to use a higher number of classifiers for each feature set and to increase the input features as well. Additionally, ensemble methods other than voting could be applied and studied.

3.2.4 Semi-Supervised

A disease entity recognition system was developed by Suakkaphong et al. (2011). Bootstrapping, a semi-supervised technique, is combined with CRF and implemented in a sequential fashion. Lexical features and syntactic features are used for classification. The authors claim that this combination outperforms supervised CRF for disease name recognition.

3.2.5 Active Learning

In a study on active learning, Chen et al. (2015) compared different sample selection strategies for named entity recognition in clinical text. They used a CRF classification systems which has already been presented in a previous study (Jiang et al., 2011). The results showed that uncertainty-based approaches outperformed diversity-based methods as well as the baseline approaches.

3.2.6 Nested Biomedical Entities

A biomedical entity recognition system dealing with nested terms was proposed by Gong et al. (2015). The system is presented with previously extracted noun phrases. A window size is set to one word and entity recognition is performed for all single-word terms in the noun phrase. The window size is then increased by one and the process is repeated until window size is equal to the number of words of the noun phrase.

3.3 Relevant Competitions of the Natural Language Processing Community in the Medical Domain

Objectively comparing results from different research publications with each other is hardly possible as the experimental conditions vary greatly. Different data sets, cross validation techniques, evaluation measures and other factors

make a correct comparison of NLP systems difficult. In order to promote research and enable objective comparison of NLP systems, various competitions have been held in the past. The Semantic Evaluation (SemEval) series and the CoNLL-2003 challenge (Tjong Kim Sang & De Meulder, 2003) incorporated general and medical tasks, while the i2b2 tasks and the CLEF eHEALTH challenges strictly focused on the biomedical domain.

3.3.1 i2b2 challenge 2010

In 2010 the i2b2 challenge presented three tasks: (1) medical concept extraction from patient reports, (2) assertion classification of theses concepts, and (3) relation extraction. The best performing systems of the first task used CRF or a hybrid approach training CRF with the output of a rule-based NER system. The best performing system for task 1 achieved an overall F1-score of 0.852. In the other tasks the most effective systems were based on SVM where a few also combined this with the output of a rule-based system (Uzuner et al., 2011).

One of the promising hybrid system combined CRF with heuristic rule-based models for entity recognition and a SVM for assertion classification (Jiang et al., 2011). Another participating team, Minard et al. (2011), tried different approaches and also concluded in their work that the approach with the highest performance was of a hybrid kind.

3.3.2 CLEF eHealth challenges 2013 and 2015

The goal of the 2013 CLEF eHealth challenge was disease entity recognition from electronic medical records. Bodnari et al. (2013) developed a supervised system based on a CRF model. Additionally the system used Wikipedia combined with biomedical terminologies as a knowledge source. The system achieved a F1-score of 0.711.

The 2015 CLEF eHealth evaluation lab task 1b participants tried to extract 10 types of entities, including anatomy and disorder, from French biomedical text. The best performing system yielded a F1-measure of 0.756. A major difficulty for this task lies in developing a system for a language other than English

for which there are not many resources available for the biomedical domain (Grouin et al., 2015).

3.4 Overview of Relevant Natural Language Processing Tools for the Medical Domain

In order to give an overview, this section provides a short description of selected tools for medical NLP. Aside from the tools below which have been designed for the medical field specifically, there are some comprehensive NLP libraries which contain components useful for both the general and the medical domain, such as OpenNLP[1], the Stanford NLP Toolkit[2], Natural Language Toolkit (NLTK)[3] (Loper & Bird, 2002), General Architecture for Text Engineering (GATE)[4] (Cunningham, Maynard, Bontcheva, & Tablan, 2002) and Unstructured Information Management Architecture (UIMA)[5] (Ferrucci & Lally, 2004).

3.4.1 LSP-MLP

According to Meystre, Savova, Kipper-Schuler, and Hurdle (2008) the first tool for medical NLP was the Linguistic String Project - Medical Language Processor (LSP-MLP)[6] (Sager, Friedman, & Lyman, 1987), which was developed at New York University. It focused on extracting symptoms, drugs, and medication side effects from clinical documents.

[1] Apache OpenNLP: https://opennlp.apache.org/
[2] The Stanford NLP Group: http://nlp.stanford.edu/
[3] NLTK: http://www.nltk.org/
[4] GATE: https://gate.ac.uk/
[5] UIMA: https://uima.apache.org/
[6] LSP-MLP: http://www.cs.nyu.edu/cs/projects/lsp/

3.4.2 UMLS

The UMLS[7] (Lindberg, 1990) is a repository of biomedical vocabularies and also integrates integrates terminology, classification and coding standards. The UMLS is a large-scale project of the National Library of Medicine (NLM), e.g. including the SPECIALIST system (McGray, Sponsler, Brylawski, & Browne, 1987). Many NLP applications are based on the UMLS.

3.4.3 SPRUS

The Special Purpose Radiology Understanding System (SPRUS) (Haug, Ranum, & Frederick, 1990), later called SymText (Haug et al., 1995) and then MPLUS (Christensen, Haug, & Fiszman, 2002), was developed at the University of Utah. It first focused only on semantics and then added syntactic and probabilistic analysis to the system.

3.4.4 MedLEE

The Medical Language Extraction and Encoding System (MedLEE) was also one of the first medical NLP systems developed (Friedman, Johnson, Forman, & Starren, 1995). It is based on semantics and is used for NLP from clinical reports to be applied for decision support. MedLEE has been commercialized in 2012 (BusinessWire, 2012)by health fidelity[8].

3.4.5 MetaMap

MetaMap[9] was developed by the NLM. The system uses a knowledge-based approach for mapping terms to the concepts of the UMLS Metathesaurus (Aronson, 2001).

[7]UMLS: https://www.nlm.nih.gov/research/umls/
[8]Health Fidelity: http://healthfidelity.com/
[9]MetaMap: https://metamap.nlm.nih.gov/

3.4.6 cTakes

The Clincal Text Analysis and Knowledge Extraction System (cTakes)[10] (Savova, Kipper-Schuler, Buntrock, & Chute, 2008) is based on the UIMA framework. It consists of a full NLP pipeline, including a NER model, for the clinical domain. The approach combines machine learning methods with rule-based methods.

3.4.7 MedEx

MedEx[11], a medication extraction system, was developed in 2010 with a major focus on identifying medication entities, including temporal entities, from clinical notes and also discharge summaries. The system is based on UIMA and was created at the Vanderbilt University Center (H. Xu et al., 2010).

3.5 Summary

The amount of computer-processable medical data in the form of plain text, such as electronic health records or clinical discharge letters, is steadily increasing, and might render a valuable source for analysis. However, the unstructured nature of plain text makes it difficult to extract useful information. The challenge here lies in understanding natural language and converting the underlying information into a structured format. Classical NER tackles the problem of identifying names of persons, organizations, or locations in plain text. While this has been studied extensively, there are still many challenges to overcome in the medical area.

The statistical machine learning approach has been used in many of the published papers on entity recognition in general. One of the most popular classifiers in the medical domain is the CRF classifier (Abacha & Zweigenbaum, 2011; Bodnari et al., 2013; Keretna et al., 2015; Wang, 2009). An essential part of the learning task is to select the best features for the given data set. These

[10] cTakes: http://ctakes.apache.org/
[11] MedEx: https://sbmi.uth.edu/ccb/resources/medex.htm

features may include lexical, syntactic, semantic, or corpus-based features. The representation of term labels, i.e. the tagging format, also plays an important role. Research in the general domain shows that while BILOU labels generally outperform the more basic IOB labels, they also lead to an increase in training time (Tang et al., 2012). A relatively new machine learning approach for entity recognition is active learning. This method has shown favourable results in the general domain (K. Zhang et al., 2014) and the medical domain (Chen et al., 2015). The idea is to incorporate a human expert, in this case a doctor or medical specialist, into the learning process. This iterative approach is promising, but more research is needed in this area (Chen et al., 2015).

To conclude, the idea of using a machine learning classifier for medical entity recognition is not new, however combining this classifier with carefully chosen preprocessing steps, an informative feature set, and a suitable tagging format for entity boundary detection could lead to a better performing system. Further, techniques other than supervised learning could be interesting to implement and subsequently compare the results. Generally, medical entity recognition is an interesting challenge which builds on different research fields and it could have a large impact on the development of new medical applications.

4 Methods and Development

After extensively studying the literature on medical entity recognition, some conclusions can be drawn. Most importantly, there is definitely a high demand for efficient medical entity recognition systems, seeing that a large portion of the existing medical data is in plain text format. Therefore, automatically extracting information from unstructured data is the main goal in this field of research (Liu et al., 2015).

This work is focused on the two main tasks of medical entity recognition: (1) trying to identify medical terms in natural language text documents, and (2) classifying the extracted terms as certain semantic types (Abacha & Zweigenbaum, 2011). There are many approaches to this challenge as has been shown in Chapter 2 and Chapter 3. Supervised approaches have been widely used and have continuously shown good results as long as there is a sufficient amount of labelled training data available (Friedman et al., 2013). In this thesis a promising supervised learning model will be implemented and combined with other techniques for achieving high performance. A common challenge in the medical domain is the poor availability of annotated training data (Cohen, 2005), thus an alternative approach to the supervised method will be explored.

This chapter describes the elementary methodology of this thesis. Considering the insights that have been gained from studying the related work, basic system design decisions are made. Then, the conceptual architecture is described and an overview is shown. Later, the chosen methods are explained in more detail. After a brief description of the tools, libraries, and resources needed for the implementation, the chapter ends with a short summary of the presented information.

4.1 Requirements and Goals

A data set of clinical doctors letters together with a list of labelled medical terms has been provided. The main goal is to create a simple medical entity recognition system, which is capable of identifying and classifying medical terms in these letters. Supervised learning techniques have demonstrated promising results as long as enough annotated data is accessible, thus a supervised method will be selected as a first approach for this task (Friedman et al., 2013).

While the classifier itself is a crucial part of the medical entity recognition system, there are other techniques which can also have a large impact on the performance. NP-chunking and special tagging formats are examples of such methods (Ramshaw & Marcus, 1995; Ratinov & Roth, 2009). As a result, another suggestion is to add further steps before the actual classification in order to enhance the performance. For research purposes this could be implemented as an additional version of the system to obtain an interesting comparison.

Feature engineering plays an important part in any classification task. The aim is to find the best representation of the given data. Features for the medical term identification and semantic type classification problem could include lexical, syntactic, semantic, document based, or corpus based features. The challenge presents itself in selecting informative features without suffering from overfitting (Garla & Brandt, 2012; Y. Xu et al., 2012).

Friedman et al. (2013) stated that interesting future research directions in medical entity recognition involve finding alternatives to supervised learning. The time and labour excessive annotation of a training data set as well as the poor adaptability to other domains are remaining challenges for supervised learning approaches. Hence, there is a demand of finding alternative methods which require less supervision (Friedman et al., 2013).

Visualization is often neglected in other experiments. A simple and understandable visualization tool for marking medical terms in text would encourage analysis before and after the actual experiment. During an initial analysis it could show some patterns and give interesting insights into where the terms are located in the text. On the other hand, medical entity recognition often involves medical experts who could certainly profit from seeing the results in an understandable way. Consequently, the creation of a simple web interface is

also a desirable objective. The idea is to plainly highlight the identified and classified medical terms in text. A straightforward web interface could be useful for analysis and visualization.

Taking everything into account, the goals of this thesis can be summarized in the following points:

- creating a medical entity recognition system capable of handling clinical doctors letters
- selecting features that are highly informative, without overfitting the data
- implementing different versions of a supervised classifier
- implementing an alternative classifier with less supervision
- evaluating and comparing the different classifiers
- creating a final list of medical entities extracted from the given data set
- creating a simple web interface for highlighting identified medical entities in text

4.2 Conceptual Architecture

In order to develop a successful system, the individual parts, such as preprocessing and classification, have to work together well. Based on the identified requirements (see Section 4.1), a medical entity recognition system is proposed. A general overview of the conceptional architecture of this system is depicted in Figure 4.1.

The input of the system is a collection of raw text documents. These files are subject to a series of preprocessing steps, more precisely, cleaning, word and sentence tokenization, and POS tagging. The next step is the classification of the medical entities. This part is the most complex part and is also implemented in different ways for an interesting comparison. The different approaches all use training data, a list of known medical entities, the IOB tagging format, and feature engineering. The system produces output in a structured format, which can then be used for evaluation and analysis. Moreover, the output list can serve as input for the web interface and possibly be used for future applications.

Going into more detail on the preprocessing steps, basic data cleaning is necessary as the input is in raw text format. Further, it is important to keep

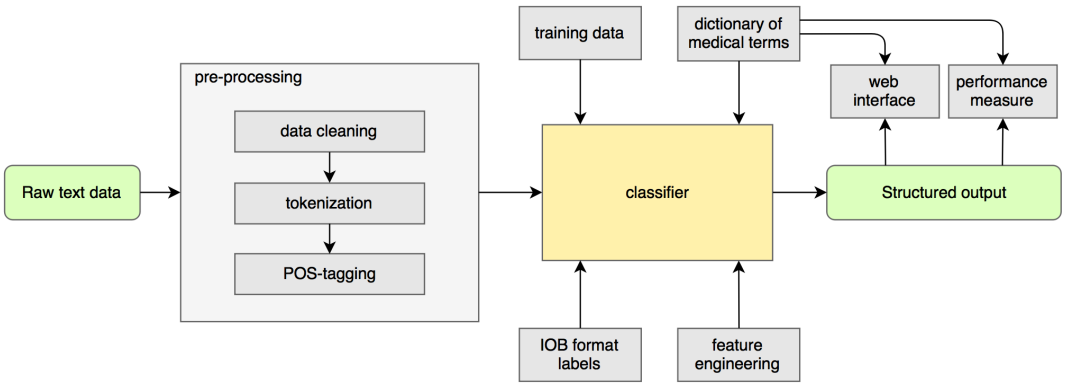


Figure 4.1: Conceptional architecture of the medical entity recognition system

in mind that word level information is most interesting for this task. For performing sentence and word level NLP, techniques such as sentence and word tokenization should be carried out first. Taking some of the previous work into account, POS tags have proven to be informative features for medical entity recognition (Doan et al., 2012; Y. Xu et al., 2012). Therefore, the preprocessing steps will consist of data cleaning, sentence and word tokenization, and POS tagging.

A classifier in a medical entity recognition system requires labelled training data. The format of these labels might also have an impact on performance. The established IOB format and the more complex BILOU format would both be good options for labelling. However, a comparison showed that the more basic IOB format requires less training time than the more advanced format (Tang et al., 2012). For time efficiency reasons, the IOB format is chosen for implementation. Further, a dictionary of known medical terms is used for various reasons. The dictionary enables annotating the training data, serves as input for a initial analysis through the web interface and can be used for evaluation purposes.

Considering the classification, an informative feature set can greatly increase performance (Abacha & Zweigenbaum, 2011) and particular attention should be paid to feature engineering with focus on linguistic features, such as lexical and syntactic features (recall Section 2.5). Since textual data can be seen as a sequence of words, the best supervised approach for this task would be some kind of sequential labelling machine learning model. CRF is well

suited for this task and even considered as the best performing supervised learning model for medical entity recognition according to Wang (2009) (see Section 3.2.2). Consequently, CRF is selected for classification. Supervised classification can be performed at once or in separate stages and further step can be added to the approach to enhance performance. Since various methods seem promising, different variations of the supervised approach will be implemented for comparison. The supervised classification approaches are discussed in more detail in the following Subsection 4.2.1.

As an alternative to the supervised learning approach, the active learning classification, as explained in Section 2.8.5, is promising. The hypothesis is that by adding an expert into the learning process of the classifier, the system will achieve high accuracy while being faster and requiring no or little training data (Chen et al., 2015). Hence, an active learning approach will also be implemented in this work. A more detailed discussion of concepts of the active learning approach is given in the Subsection 4.2.2.

A web interface will be created for visualization purposes. The web application will enable the analysis of the initial observations as well as the final findings of the study. The popular Stanford NER online tool[1] will serve as a role model for the design of the web page. The main idea of the web interface is to highlight the identifies medical terms in text using a colors corresponding to the semantic types.

Finally, performance measure will show which variation of the medical entity recognition system performs best. According to Cohen (2005), the most common evaluation measures for entity recognition systems are precision and recall, and F1-score. For this reason and to enable objective comparison, precision, recall and F1-score will be used for evaluation.

4.2.1 Supervised Classification Approaches

Medical entity recognition consists of two main steps, namely (1) identifying the medical term and its boundary in text, and (2) classifying this term as a certain semantic category (Abacha & Zweigenbaum, 2011; S. Zhang & Elhadad, 2013). Considering that CRF is a sequential labelling model, these two steps

[1]Stanford Named Entity Tagger: http://nlp.stanford.edu:8080/ner/

could be performed in a single process (S. Zhang & Elhadad, 2013). In order to handle the two tasks at once, a suitable label format, such as IOB, is essential. As an example, a CRF classifier combined with the IOB format labels has been presented by Abacha and Zweigenbaum (2011). With this in mind, the supervised Classifier Version 1 (CV1) is designed, which performs medical entity recognition in a single step using CRF.

In a study by Y. Xu et al. (2012) NP-chunking was incorporated as a preprocessing step and the identified NPs were then used as features for the classifier. After a brief analysis of the data set however, it became clear that some of the medical terms are not NPs, but rather verb phrases, adjective phrases or foreign words. The idea was formed to denote all phrases that could be possible medical terms as candidate phrases. Thus, identifying these candidate phrases using a simple rule-based chunker and then feeding this additional information to the classifier could improve the overall performance. Thereof, for Classifier Version 2 (CV2), the additional step of candidate phrase extraction is added before the actual classification step.

Previous research has shown that boundary detection of multi-word medical terms is one of the more difficult tasks of medical entity recognition (Uzuner et al., 2011). Even though the CRF classifier is capable of handling entity boundary detection and entity classification in a single step (S. Zhang & Elhadad, 2013), it would nevertheless be interesting to see how performance and execution time changes if the two steps were executed separately. Thus, for Classifier Version 3 (CV3) of the supervised system, the classification is performed in two stages, resulting in a stepwise classification. After candidate phrase chunking, the medical entity is only recognized at first and then classified in the next step.

Three different supervised learning systems are implemented in this work. An overview of the three approaches is shown in Figure 4.2. To summarize, the popular classifier CRF is implemented in three different ways: (1) using a single-step approach, (2) adding an extra candidate phrase extraction step before classification, and (3) performing the identification of the medical entity and the type classification in two separate steps. This work will find out if the more elaborate approaches actually exhibit an increase in performance as would be assumed.

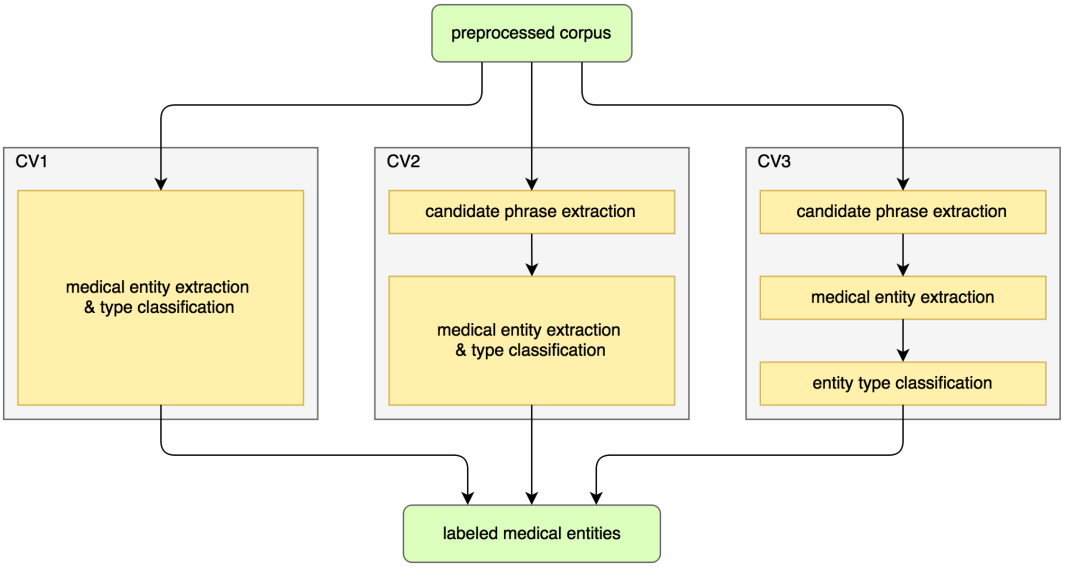


Figure 4.2: Overview of three variations of a supervised CRF classifier for medical entity recognition

4.2.2 Active Learning Approach

In addition to the supervised approach, an active learning system is implemented. The intention of active learning is to overcome the problem of poor availability of annotated data sets in the medical domain (Cohen, 2005). Directly integrating a medical specialist into the training process is the main advantage of active learning systems. The basic idea here is that high accuracy can be reached by carefully selecting the training samples to present to the expert while the necessary amount of training data is reduced (Settles, 2010). The basic steps of an active learning system presented by K. Zhang et al. (2014) will serve as a role model for implementing this approach.

The resulting active learning model can be seen in Figure 4.3. The single-step CRF approach (CV1), as explained in the section above, is used for classification. The model is trained with a small initial data set at first. Next, a subset of data samples is selected and presented to the expert for labelling. Uncertainty-based approaches are promising for selecting the best query samples, since they outperform other methods according to Chen et al. (2015) (see Section 3.2.5). Consequently, the active learning approach will perform query sampling based

on uncertainty. Because CRF is the model of choice, the uncertainty can simply be estimated through the prior probability output of the classifier after prediction. The chosen query sampling method is strongly related to the uncertainty-based approach explained by Chen et al. (2015). The selected and labelled samples are then added to the training data set and the iterative cycle continues until convergence or the desired performance is reached. Finally, the active learning approach will be compared to the more established supervised methods. The findings will be analysed and discussed.

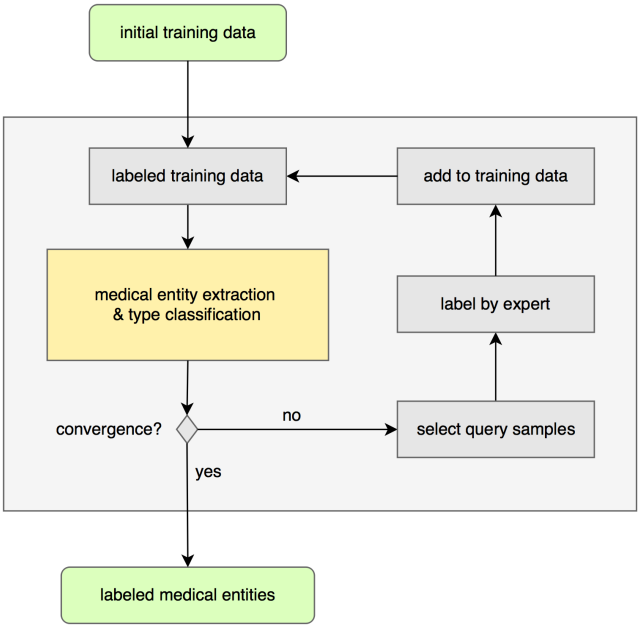


Figure 4.3: Outline of the active learning classification steps

4.3 Conditional Random Fields

CRF will be used for classification in this work and therefore the theoretical basis is briefly described. The idea of CRF was first published in 2001 (Lafferty et al., 2001). For classification tasks, the aim is to create a model for maximizing the conditional probability to predict the most likely class. While generative methods, such as HMM, model the joint probably, CRF directly models the

conditional probability (Sutton & McCallum, 2011). In the following definition of a general CRF by Lafferty et al. (2001), **X** represents a random variable over the data sequence, i.e. the feature vector, and **Y** is the corresponding random variable over the label sequence.

Definition. *Let $G = (V, E)$ be a graph such that $Y = (Y_\upsilon)_{\upsilon \in V}$, so that Y is indexed by the vertices of G. Then (X, Y) is a conditional random field in case, when conditioned on X, the random variables Y_υ obey the Markov property with respect to the graph: $p(Y_\upsilon | X, Y_\omega, \omega \neq \upsilon) = p(Y_\upsilon | X, Y_\omega, \omega \sim \upsilon)$, where $\omega \sim \upsilon$ means that ω and υ are neighbours in G.*(Lafferty et al., 2001)

According to the original definition by Lafferty et al. (2001) and to Sutton and McCallum (2011), who have written a very useful tutorial on CRF, Equation 4.1 depicts the conditional distribution of a CRF.

$$P(y|x) = \frac{1}{Z(x)} \prod_{\Psi_A \in G} exp \left\{ \sum_{k=1}^{K(A)} \theta_{ak} f_{ak}(y_a, x_a) \right\} \tag{4.1}$$

The normalization function (or partition function) **Z** is defined according to Equation 4.2.

$$Z(x) = \sum_{y} \prod_{\Psi_A \in G} exp \left\{ \sum_{k=1}^{K(A)} \theta_{ak} f_{ak}(y_a, x_a) \right\} \tag{4.2}$$

4.4 Used Tools and Libraries

Several existing tools and libraries are employed for the implementation of the presented approaches. The tools are selected based on the requirements and the conceptual architecture described in Section 4.1 and Section 4.2 respectively. This section provides a brief overview and description of all tools and libraries used for this thesis.

4.4.1 Bootstrap

Bootstrap[2] is a popular framework for developing HTML, CSS, and JS applications. It also provides free design templates and thus was the framework of choice for giving the web application a simple design.

4.4.2 CRFsuite and python-crfsuite

The CRFsuite[3] software is a fast implementation of CRF. The tool provides fast training and tagging, different training methods, linear-chain CRF. The python-crfsuite[4] library is a very simple Python binding to CRFsuite. It uses Cython and the CRFsuite C++ API. The combination of these tools is used for training and testing the classifier model.

4.4.3 Django

Django[5] is a Python framework for building Web applications. The open source library is fast, secure, scalable and takes care of Web development basics. Django was used for creating the web application.

4.4.4 FuzzyWuzzy and python-Levenshtein

FuzzyWuzzy[6] is a small python package for fuzzy string matching. The similarity of two sequences is calculated using the Levenshtein distance and the corresponding package python-Levenshtein[7].

[2]Bootstrap: http://getbootstrap.com/
[3]CRFsuite: http://www.chokkan.org/software/crfsuite/
[4]python-crfsuite: https://pypi.python.org/pypi/python-crfsuite
[5]Django: https://www.djangoproject.com/
[6]FuzzyWuzzy: https://pypi.python.org/pypi/fuzzywuzzy
[7]python-Levenshtein: https://github.com/ztane/python-Levenshtein/

4.4.5 NLTK

NLTK[8] is an extensive Python package. It includes over 50 lexical resources and provides many libraries and tools for working with natural language. The tool kit includes libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning.

4.4.6 Pandas

Pandas[9] is another useful Python package equipped with data structures and data analysis tools. The pandas library makes data manipulation simple and effective.

4.4.7 Scikit-learn

The Python package scikit-learn[10] contains tools for data mining and analysis. Applications include classification, regression, clustering, model selection, preprocessing, and dimensionality reduction.

4.4.8 Simplejson

Simplejson[11] is a fast and simple Python library for decoding and encoding JSON files. This package is used for transforming a csv file into a JSON file to make it compatible with the Django framework.

[8]NLTK: http://www.nltk.org/
[9]Pandas: http://pandas.pydata.org/
[10]Scikit-learn: http://scikit-learn.org/stable/
[11]Simplejson: https://pypi.python.org/pypi/simplejson/

4.4.9 Stanford POS Tagger

The Stanford POS tagger[12] is implemented in Java. After tokenization, it assigns the according POS tag to each word in a plain text sentence. This state-of-the-art tool is a log-linear POS tagger as described by Toutanova, Klein, and Manning (2003) and Toutanova and Manning (2000). The software also contains two trained models for the English language. The POS tags used are defined by the University of Pennsylvania (Penn) Treebank tag set[13] (Santorini, 1990).

4.5 Summary

The goals of this work have been further refined and an appropriate medical entity recognition system has been proposed based on the insights that have been gained from studying the literature. The system is able to handle medical text data through combining proven methods and techniques, such as text preprocessing, POS tagging, and IOB format labelling. For classification, the popular CRF model will be applied since it has shown to work well with sequential data (Wang, 2009). Moreover, feature engineering has significant impact on the performance of the classifier. The features will be designed to best represent the data while taking care not to cause overfitting.

The two central challenges of medical entity recognition are the identification of medical terms in text and the classification of the semantic type of these terms (Abacha & Zweigenbaum, 2011). To tackle these tasks, three supervised approaches and an active learning approach have been presented.

Section 4.2.1 described three different versions of a supervised classifier. The first version follows a basic single-step approach of performing the two tasks at once. For a second version of the classifier an additional step of extracting possible candidate phrases is integrated and this additional information is added to the feature vector for classification. In the third version of the classifier, the identification and the classification are performed in two separate steps.

[12] Stanford POS Tagger: http://nlp.stanford.edu/software/tagger.shtml
[13] Penn Treebank tag set: http://www.comp.leeds.ac.uk/amalgam/tagsets/upenn.html

Considering that annotation of medical text documents is costly and involves expert knowledge (Friedman et al., 2013), an alternative approach to the supervised method has also been presented in Section 4.2.2. The active learning model is an iterative approach which directly incorporates a domain specialist into the learning process. An uncertainty-based approach will be used for selecting the query samples. By iteratively presenting the chosen samples to the expert, this approach should be able to reduce the necessary training data.

In conclusion, three versions of a supervised medical entity recognition system and an active learning system will be implemented. Lastly, the performance of these approaches will be measured and compared, which gives rise to the following two research questions:

1. Will the more complex supervised approaches perform better than the basic single-step approach?
2. Will the active learning system be able to significantly reduce the amount of data needed to reach the same or higher performance scores than the supervised approaches?

5 Experiment Design and Results

The aim of this work is to design the experiments, implement and compare the approaches of medical entity recognition which have been proposed in Chapter 4. Essentially, medical entity recognition enables converting plain text into structured information (S. Zhang & Elhadad, 2013). The presented systems aims to identify and classify medical terms in text. The structured output then consists of a list of medical terms with their corresponding semantic type.

Three supervised classifiers using CRF and an active learning classifier as an alternative are implemented. The different supervised approaches include a simple single-step approach (CV1), an approach with additional candidate phrase extraction (CV2), and a stepwise approach performing candidate phrase extraction, medical term identification, and type classification sequentially (CV3). The active learning method utilizes the classification model from the supervised CV1 and uncertainty-based sampling.

The detailed information of the setup of the study is described in this chapter. The precise descriptions of the used data set and of how the selected methods are implemented will also be provided. After the detailed design of this work has been discussed, the obtained results will be presented and the most interesting findings will be reviewed.

5.1 Development environment

The presented medical entity recognition and semantic type classification is implemented and evaluated on a Nectar server[1]. The server provides scalable

[1]Nectar: https://nectar.org.au/research-cloud/

computing power and allows running programs rapidly, which is a major advantage as the training of the machine learning model is usually time consuming. Access to a Nectar instance has been provided by co-supervisor of this theses Wei Liu. An Ubuntu 14.04.1 LTS machine runs on the Nectar server. The X2Go Client[2] is used to access the server remotely using a Windows 10 operating system. The medical entity recognition system is implemented in Python[3] (Version 3.4.3).

The web interface is implemented and run directly on the Windows 10 machine using Anaconda[4] (Version 2.5.0). Anaconda is a data analytics platform with Python and major Python packages pre-installed. The used version of Anaconda includes Python (Version 3.5.1) as well as the Spyder IDE[5] for implementing Python programs.

5.2 Data Set

The data set used in this experiment consists of 29175 clinical discharge letters between ophthalmologists and general practitioners. The data set has been provided by co-supervisor Wei Liu, who has used the same data in a previous study on an unsupervised medical entity recognition system (Liu et al., 2015), which was presented in Section 3.2.1. The medical terms extracted in the unsupervised experiment were presented to three domain experts for revision and the results were used for annotating the data set.

The medical field of ophthalmology is concerned with the anatomy and physiology of the eye, eye diseases and vision disorders. The letters have been collected during a period of 10 years and were written by five different specialists. The names and addresses of patients have been removed by anonymisation algorithms for privacy reasons and thus no conclusions can be drawn on the actual patients. Further, the letters are in unstructured natural text in the English language.

[2]X2Go: http://wiki.x2go.org/
[3]Python: https://www.python.org/
[4]Anaconda: https://www.continuum.io/why-anaconda
[5]Spyder: https://github.com/spyder-ide/spyder

An initial analysis of the data set shows that it is comprised of exactly 29175 non-empty letters. Each letter consists of approximately 7.61 sentences and each sentence in turn is made up of about 17.01 words, which leads to an average number of 129.44 words per letter.

5.3 Semantic Types

The identified medical terms have to be classified as one of a pre-defined semantic category. Common semantic categories for medical entity recognition are the SOAP types, which stands for *subjective (S)*, *objective (O)*, *assessment (A)* and *plan (P)* (Cameron & Turtle-song, 2002). Other semantic classes have been used in the research also. As future research options, Wang (2009) mentioned that dividing some categories further could lead to less ambiguity and more coverage of terms.

During a study on the same data set, (Liu et al., 2015) consulted with domain experts and agreed on seven semantic types, which are also used for the task at hand. These seven categories can be seen in Table 5.1 and are used for classifying medical terms in this work.

Number	Category
0	Symptom
1	Anatomy
2	Sign
3	Test
4	Measurement
5	Diagnosis
6	Treatment

Table 5.1: Semantic types for classification of medical terms

5.4 Dictionary of Known Medical Terms

The ophthalmology discharge letters data set was used in a previous experiment by Liu et al. (2015) who extracted medical terms using an unsupervised approach which was explained in more detail in Section 3.2.1. The obtained medical terms were then presented to three domain experts for evaluation. In the first step, the correctness of the extracted entity was marked, meaning that the experts stated if the extracted term was an actual medical entity.

In the second evaluation step, the domain experts classified the correctly extracted medical entities as one of the seven semantic types which have been described in Section 5.3, and more specifically in Table 5.1. The medical terms were only considered as correct, if all three domain experts agreed on what type of semantic category the term belonged to. The Python library pandas enabled fast list manipulations through the use of data frames. The resulting list of correctly extracted and classified medical terms will be used for labelling the training data and evaluating this work. The list of known medical terms consists of 2179 medical terms of a length between one and six words per term and serves as a lookup dictionary for evaluating the classifier. The first 10 lines of the dictionary of known medical terms are shown in Table 5.2.

phrase	category
a-/b-scans	4
a-scan	4
abducens nerve palsy	6
ablation	7
abnormal blood test	4
abnormal head posture	3
abnormal intraocular pressure	3
abnormal macular reflex	3
abnormal vascular pattern	3
abscess	6

Table 5.2: Extract of the first 10 lines of the dictionary of known medical terms. Terms are in alphabetical order.

5.5 General System Architecture

Based on the conceptual architecture presented in Section 4.2, the actual architecture of the system incorporating existing tools and frameworks is now presented. The components of the medical entity recognition system are displayed in Figure 5.1. In order to keep the overview simple, the web interface and the tools used exclusively for the web interface are described in more detail in the following Section 5.6.

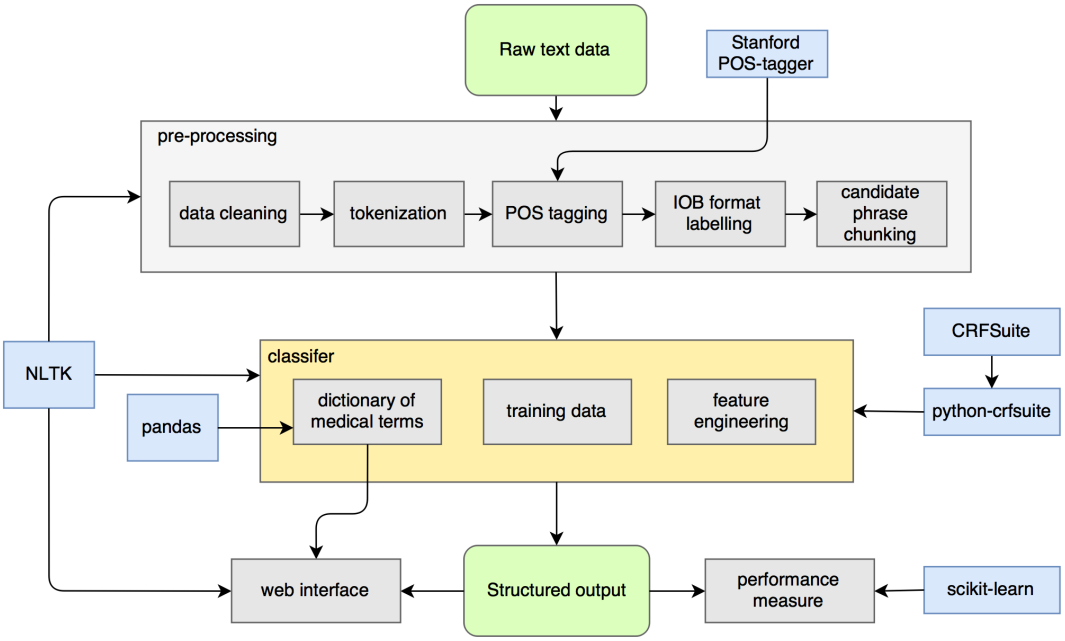


Figure 5.1: System architecture of the medical entity recognition system

The Python library NLTK (see Section 4.4.5) is used for handling the corpus of clinical discharge letters, tokenization, POS tagging, candidate phrase chunking, during the classification and also for the web interface. The popular Stanford POS tagger, as mentioned in Section 4.4.9, is applied for POS tagging. The Python library pandas (see Section 4.4.6) is used for the creation of the dictionary of known medical terms, which has been described in Section 5.4. The classification is performed using CRFSuite and the Python binding python-crfsuite, as explained in Section 4.4.2. The scikit-learn library (recall Section 4.4.7) is used for cross-validation and for calculation of precision,

recall and F1-score. The components of the system are described in more detail in the following sections.

5.6 Web Interface

Another objective of this thesis was to create a simple web interface for visualisation of recognized medical terms in text. The design of the web application is shown in Figure 5.2. The application takes plain text as input and identifies medical entities. In the output text, the terms are highlighted in different colors according to the seven semantic types *symptom*, *anatomy*, *sign*, *test*, *measurement*, *diagnosis*, and *treatment*. The Python web framework Django (see Section 4.4.3) is used for the implementation of the web page. In order to match the terms with medical entities, a medical term lookup list is used. This list can either be the dictionary of known medical terms or the output of one of the supervised or active learning approaches. For making the list in CSV format compatible with the Django framework, the Python package simplejson (recall Section 4.4.8) is used. Components of the application are word and sentence tokenization, stemming, and fuzzy matching. The used tokenizers and stemmers are part of the NLTK library, which has been described in Section 4.4.5. The Python packages FuzzyWuzzy in combination with python-Levenshtein were used for fuzzy string matching, as mentioned in Section 4.4.4. A basic bootstrap template (see Section 4.4.1) was used for a clear and simple design of the application.

Figure 5.3 shows the initial page of the created web interface. Text input can take place through simple copying and pasting or by uploading up to three files in 'txt' or 'zip' format. The web interface allows users to chose which categories of medical entities should be highlighted. By default, all seven categories are highlighted in the output text in different colors. In this process the longest matching term is preferred over shorter terms. Further if terms of the same length would overlap, the later term is chosen to be highlighted. Overlapping highlighting has not been implemented, but could easily be added if wished for. Below the input text field, the user has the possibility to choose between exact matching and various algorithms of fuzzy matching, with exact matching being the default setting. Additionally the user can choose which type of stemming

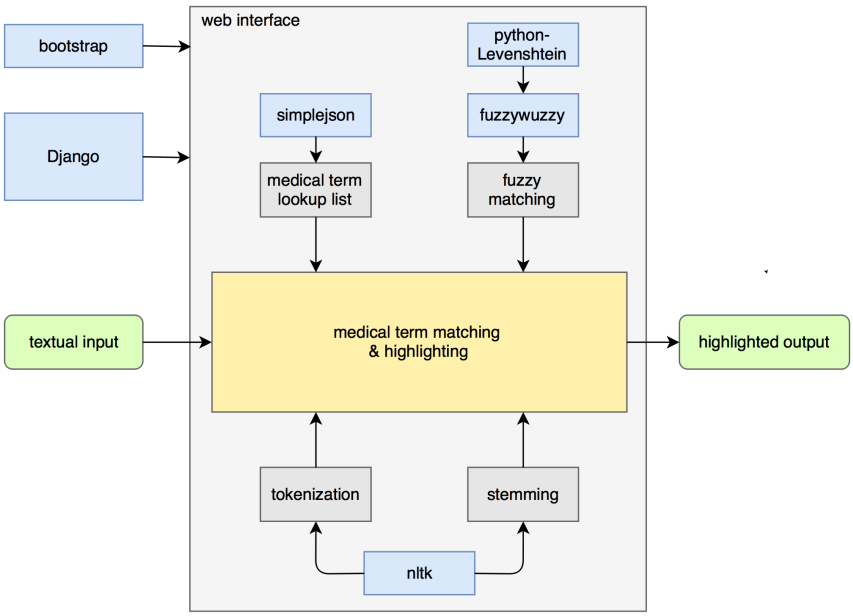


Figure 5.2: Architecture of the medical entity recognition web application including used tools

should be performed before matching the terms, while no stemming is selected by default.

Fuzzy matching is performed using the FuzzyWuzzy library. It uses the Levenshtein distance to calculate differences between string sequences. If fuzzy matching is selected, the user is able to choose the matching threshold as well. In more detail, the web interface allows the user to make the following fuzzy matching algorithm choices:

- **exact:** No fuzzy matching is applied.
- **ratio:** Simple comparison of two strings with a measurement of edit distance.
- **partial ratio:** This method also considers partial matches.
- **token sort ratio:** Considers out of order strings by sorting the tokens alphabetically, and then joining them back together.
- **token set ratio:** Matches strings which include each other by splitting them into an intersection and a remainder string.

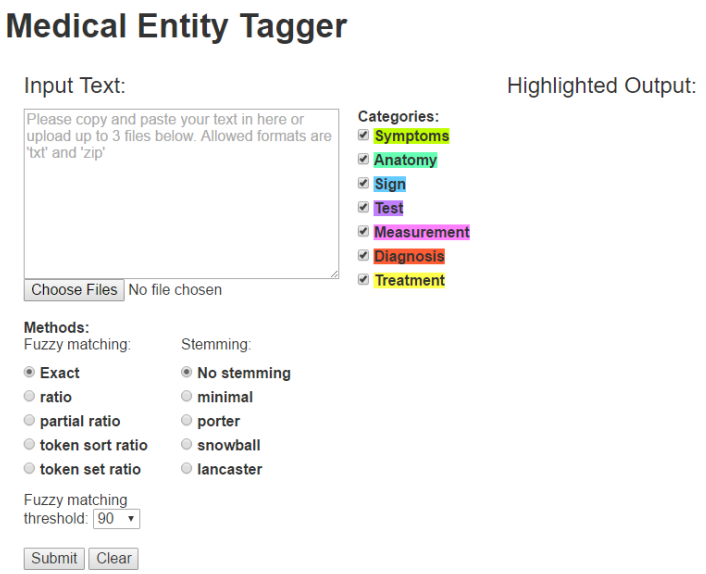


Figure 5.3: Initial page of the medical entity recognition web application

Stemming is performed using the stem package of the NLTK library. The web page gives the user a choice of the following stemming algorithms:

- **no stemming:** No stemming is performed.
- **minimal:** A stemmer that uses regular expressions to identify morphological affixes. This minimal version only removes the substring 's'.
- **porter:** A word stemmer based on the original Porter stemming algorithm presented in Porter (1980).
- **snowball:** The English Snowball stemmers developed by Martin Porter[6].
- **lancaster:** A word stemmer based on the Lancaster stemming algorithm presented in Chris (1990).

Figure 5.4 shows the output of the web application using an example letter from the ophthalmology data set. None of the extra options are selected and the default settings of exact matching and no stemming stay in place. The seven categories are identified and highlighted. As can be seen in the figure, three medical terms are recognized, namely the diagnosis *macular degeneration*, the anatomy term *intraocular* and the treatment term *catarct surgery*.

[6]http://snowball.tartarus.org/algorithms/english/stemmer.html

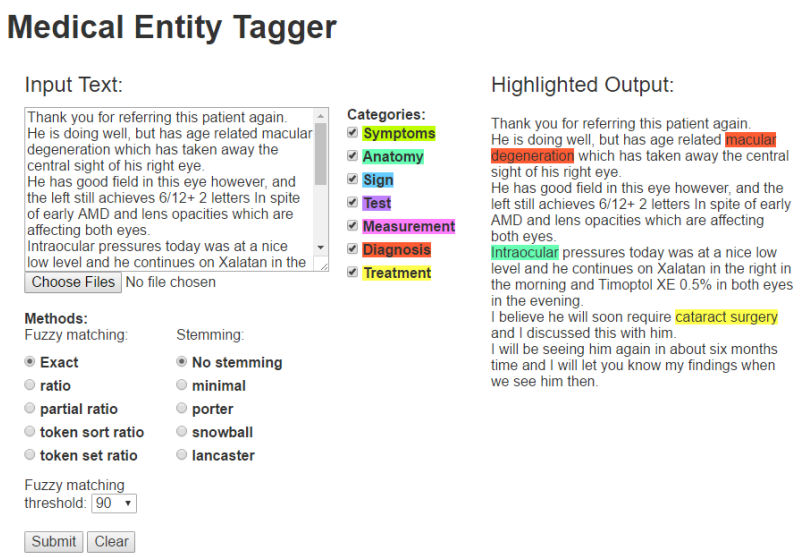


Figure 5.4: Medical entity recognition web application using an example clinical letter with the default options

After choosing the more advanced option of fuzzy matching using the ratio algorithm with a threshold of 95, the results slightly change, as can be seen in Figure 5.5. The application now also recognizes the diagnosis term *age related macular degeneration* and the measurement term *intraocular pressures* in addition to the treatment term *cataract surgery*. The reason for these newly found terms is that in the lookup list, these terms are denoted as *age-related macular degeneration* with a hyphen and *intraocular pressure* without the plural 's'. Since fuzzy matching is selected and these terms only differ by one character, they are matched.

The option of stemming is selected in the example given in Figure 5.6. Using porter stemming before matching the terms, the application is able to find the diagnosis term *macular degeneration*, the sign term *lens opacities*, the measurement term *intraocular pressures*, and the treatment term *cataract surgery*. The new term of *lens opacities* is recognized using this method since the lookup dictionary contains the term *lens opacity*. The fuzzy matching option did not find it, as there is more than one character difference of the two terms. However the word stem is exactly the same and thus it is recognized using the porter stemmer.

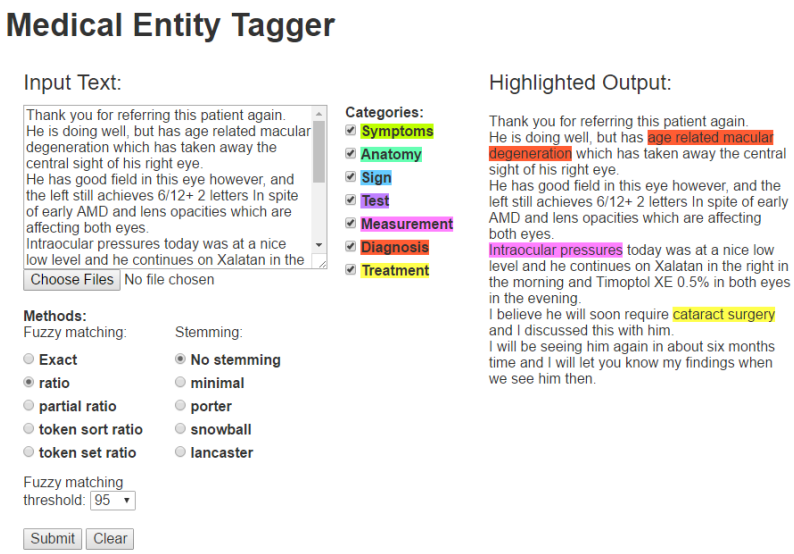


Figure 5.5: Medical entity recognition web application using an example clinical letter with fuzzy string matching selected using the ratio algorithm

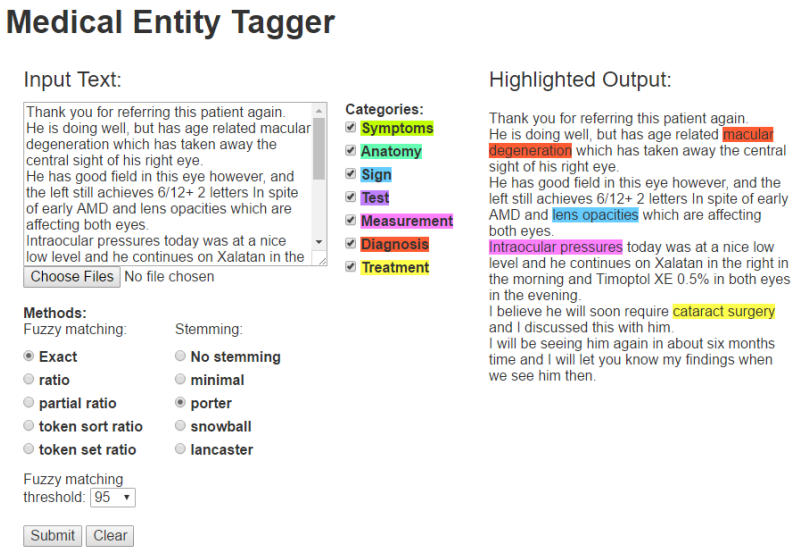


Figure 5.6: Medical entity recognition web application using an example clinical letter with porter stemming selected

5.7 Preprocessing

The first step of the medical entity recognition pipeline is preprocessing of the raw data. The doctors letters are in plain text, have been anonymised and contain some noise which makes preprocessing necessary. The following sections briefly explain the individual steps.

5.7.1 Data Cleaning

Data cleaning is performed with basic string replacement using regular expressions. Additionally, empty files are discarded. The following cases are regarded in this order during preprocessing:

1. all none printable strings were replaced with a single space
2. unwanted white space characters were replaced with a single space
3. patterns such as "## # ####", which were due to anonymization, were replaced with a single "#"
4. multiple "." were replaced with a single "."
5. for better sentence tokenization, a space was added after the ".", if the "." was followed by a capital letter

5.7.2 Tokenization and POS-Tagging

After the data cleaning step some basic NLP is performed using the NLTK library for Python. The documents are split into sentences, which are then split into words using NLTK tokenizers. The tokenized sentences are further processed by the Stanford POS-tagger, as mentioned in Section 4.4.9. After the POS-tagging step each word and the assigned POS-tag are saved to files in the format *token*/*POS-tag*.

5.7.3 IOB Format Labelling

In this next step, the IOB labels are added to the terms. The list of known medical entities is assumed to be ground truth and used as a lookup table.

All terms are compared with the dictionary and in case of a match, the term is labelled according to the IOB format. The sentences are scanned through systematically and preference is given to the longest (term with most words) and also to the later term in case of multiple matches. As there are seven different semantic categories, there are 15 different label in IOB format (O; B-1,B-2, ... B-7; I-1,I-2, ... I-7).

5.7.4 Candidate Phrase Chunking

Many NER and medical entity recognition systems incorporate a NP-chunker, which identifies groups of words with a head noun. In the given ophthalmology data set, some medical terms are not NPs, but rather verb phrases, adjectives or foreign words. Classic NP-chunking would rule these medical terms out, which is not desirable. Thus, a candidate phrase chunking step is introduced in the pipeline. Candidate phrases can be medical terms or not, but words that are not classified as part of a candidate phrase are not further processed.

This step is implemented using a simple regular expression parser from the NLTK library. The candidate terms are extracted based on the following four grammar rules:

1. <FW>?<RB>?<DT>?<(VBG|VBN|VBZ)>?<JJ>$*$<(NN|NNS|NNP)>+<VBZ>?
2. <(VB|VBG|VBN|VBP|VBZ)>
3. <JJ>
4. <FW>

The results of the preprocessing steps are saved to text files. Table 5.3 shows an example of a preprocessed sentence. Each line consists of (1) the token, (2) the POS tag with the possible extension -CP to mark it as a candidate phrase, and (3) the label in IOB format. Sentences are separated by blank lines.

5.7.5 Preprocessing Findings

During a first analysis of the data, some obvious patterns became clear. In general, it is very common for multi-word terms to include another medical entity. A multi-word diagnosis or measurement frequently often includes an anatomy term at the beginning. And a multi-word treatment often contains a

I	PRP	O
will	MD	O
pick	VB-CP	O
this	DT	O
up	RP	O
when	WRB	O
I	PRP	O
see	VBP-CP	O
her	PRP$	O
after	IN	O
the	DT	O
weekend	NN-CP	O
and	CC	O
commence	VB-CP	O
her	PRP$	O
on	IN	O
antiviral	NNS-CP	B-7
agents	NNS-CP	I-7
.	.	O

Table 5.3: Example sentence at the end of all the preprocessing steps. Each line contains the token, the POS tag (and possible extension -CP), and the IOB label.

diagnosis term at the beginning. To summarize, some examples are given in Table 5.4.

For confirmation and visualization purpose, the web application described in Section 5.6 was used for the same clinical letter example as before. Interesting observations could be made by using different settings of the web interface. Some screen shots can be seen in Figure 5.7. The ratio algorithm and a threshold of 95 are selected for fuzzy matching and stemming is performed using the porter stemmer. The output of the web interface considering all seven semantic type categories is shown in Subfigure 5.7a. As would be expected, by combining fuzzy matching and stemming, the application recognizes the best results of the two individual selections (as compared to only fuzzy matching in Figure 5.5 and only stemming in Figure 5.6). The identified medical entities are *age related macular degeneration* of type diagnosis, *lens opacities* of type sign, *intraocular pressures* of type measurement, and *cataract surgery* of type treatment.

When less categories are chosen to highlight, interesting observations can be made. Looking at Subfigure 5.7b, all categories except measurement are selected.

Text fragment	Annotation
has had pterygium surgery on the right	pterygium, Diagnosis pterygium surgery, Treatment
considers cataract surgery on that eye	cataract, Diagnosis cataract surgery, Treatment
intraocular pressure and fundi were normal	intraocular, Anatomy intraocular pressure, Measurement
to have age related macular degeneration	macular, Anatomy macular degeneration, Diagnosis

Table 5.4: Examples for multi-word medical terms including another medical term

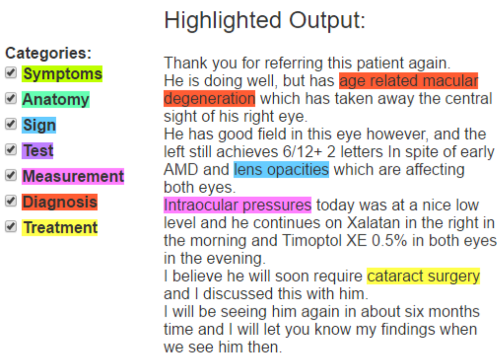
The measurement term *intraocular pressures* is not recognized any more, but instead the anatomy term *intraocular* is highlighted. The system behaves correspondingly for the other categories. Subfigure 5.7c shows what happens when all types are selected except diagnosis. Instead of the diagnosis term *age related macular degeneration*, the anatomy term *macular* is recognized. On the last Subfigure 5.7d in the bottom right corner, all classes except treatment are chosen. Alternatively to the treatment term *cataract surgery*, the diagnosis is highlighted. These observations agree with the finding from Table 5.4.

5.8 Feature Extraction

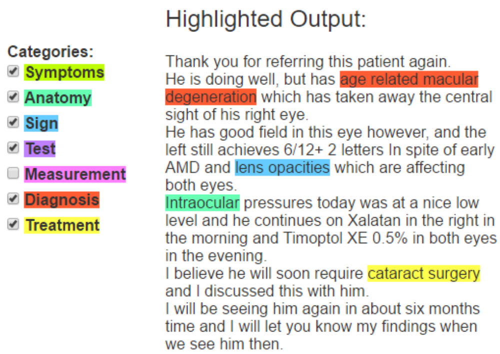
A set of 13 different features is used for this experiment. The features are of lexical and syntactic nature and there is one sentence based feature, which can be seen in Table 5.5. These features are used for the word itself as well as for the previous and the following word. If there is no previous or following word, the feature *BOS* (beginning of sentence) or *EOS* (end of sentence) is appended respectively. Hence, up to 39 features are used per term all together.

The POS tags are created using the Stanford POS-tagger, see Section 4.4.9 and Section 5.7.2. The University of Pennsylvania (Penn) Treebank tag set[7] is used for labelling the POS tags. The POS tags usually have two or three letters. The first two letters denote the general word class, such as 'NN' for common noun, and the third letter describes the more specific type, e.g. 'NNS'

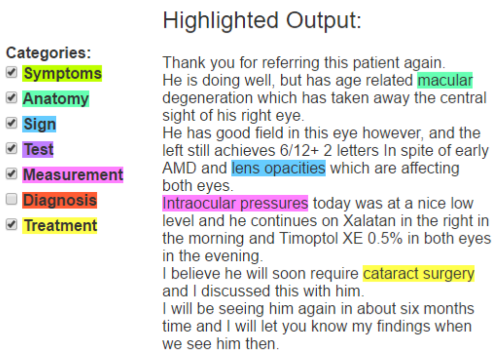
[7]Penn Treebank tag set: http://www.comp.leeds.ac.uk/amalgam/tagsets/upenn.html



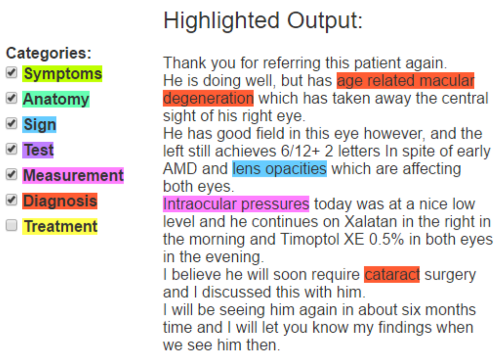
(a) all semantic types are selected



(b) measurement is not selected



(c) diagnosis is not selected



(d) treatment is not selected

Figure 5.7: Examples of nested medical terms visualized by using the web application to highlight different categories. Used settings were the ratio algorithm for fuzzy matching with a threshold of 95 and the porter stemmer.

for plural common noun. Thus, the full POS tag and the first two letters alone are used as features.

lexical features:	**syntactic features:**
word in lowercase	POS tag (e.g. NNS)
word length	first 2 characters of POS tag (e.g. NN)
word stem (porter stemmer)	candidate phrase (boolean)
prefix (2, 3, 4 letters)	
suffix (2, 3, 4 letters)	**sentence based features:**
uppercase (boolean)	position of the word in the sentence
title case (boolean)	
alpha characters only (boolean)	
digits only (boolean)	

Table 5.5: Features used for the experiment

5.9 Classification

This work implements three supervised classifier version and an active learning version. All variations of the medical entity recognition system use data cleaning, POS tagging, and IOB format labelling for preprocessing, which have been described in Section 5.7.

The classification step of the medical entity recognition and semantic type classification system is performed using CRF (see Section 4.3). For training the model and predicting the class labels, the CRFsuite[8] was used in combination with the python binding python-crfsuite[9], as described in Section 4.4.2. The selected methods from the CRFsuite are the linear-chain CRF for training and testing, the Limited-Memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) method (Nocedal, 1980) for optimization, and the forward-backward algorithm using the scaling method (Rabiner, 1989) for calculating the posterior marginal distribution.

[8]CRFsuite: http://www.chokkan.org/software/crfsuite/
[9]python-crfsuite: https://pypi.python.org/pypi/python-crfsuite

5.9.1 Supervised Classification

Three different versions of a supervised classifier are implemented based on the concepts described in Section 4.2.1. The exact implementations of the different classifier versions using selected tools and libraries are described in this section.

Classifier Version 1

CV1 performs medical entity identification and semantic type classification in a single step. The training data is represented by the features which are extracted from the preprocessed documents. Each word in the data set is labelled with IOB format labels. The use of this labelling format enables classification in a single step. In CV1 every word of the document is used for classification. This results in a very imbalanced data set, as most of the words are no medical terms and thus labelled *O*.

Classifier Version 2

Additionally to the methodology of CV1, the second version of the classifier (CV2) applies the step of candidate phrase chunking before classification. The terms which are identified as candidate phrases are marked with *-CP* as extension to the POS tag. Further, the boolean feature candidate phrase is added to the feature set. For training and testing, only the candidate phrase terms are used. Thus, the amount of training data is reduced. The remaining approach is equivalent to CV1.

Classifier Version 3

Besides the techniques of CV2, the classification in CV3 is divided into two separate steps. The classifier is basically used twice. The first model is trained using the candidate phrases (according to CV2), but only using the labels *I*, *O* and *B* without the semantic categories. As a next step, the extracted multi-word terms are joined through inserting '_' instead of a space between

terms. As a result, the data set now only contains terms labelled as O or a medical entity marked as B. In another classification execution, the identified medical terms are now classified as a certain semantic category. This time, the second model is trained using only the medical terms. The prediction of the test data also occurs in two steps accordingly. For evaluation, the multi-word terms connected by '_' are separated again to make the performance measures comparable to the other approaches.

5.9.2 Active Learning Classification

The active learning classifier, which has been introduced in Section 4.2.2, is implemented in a similar manner as CV1. The only difference is the selection of the training data. As there is no human expert present for the experiment, the dictionary of known medical terms from the ophthalmology domain is used as an oracle. Initially, five letters are randomly selected for training the model. This small number is chosen, since the active learning system should function as an alternative for situation with very little or no training data. In every iteration, five sentences are selected to be presented to the expert. In regard to the theoretical human expert, five sentences have to be enough, since annotation is costly and time consuming.

After the model is trained, the remaining documents of the data set are used for evaluation. As long as the stopping criteria is not reached, the iterative cycle continues. Reasons for ending the classification are reaching a desirable F1-score or reaching the maximum amount of 200 iterations. The F1-score of the best performing supervised system is chosen as stopping criteria. The selection of query samples to be presented to the oracle is based on uncertainty. The CRF algorithm provides the posterior probability of each sequence, i.e. sentence, of the test data set. The five sentences with the lowest posterior probability, and thus the highest uncertainty, are selected for expert labelling.

5.10 Performance Evaluation

10-fold cross-validation is used for statistical analysis of the three supervised classifiers. The full data set is randomly divided into 10 folds. 9 parts are used

as training data set and the testing data set is comprised of the remaining part. The 10 folds are iterated through in a way that the testing part is always a different one. The averages over all 10 iterations are calculated for the final measures of precision, recall and F1-score. This technique allows objective evaluation of the system. The Python package scikit-learn (see Section 4.4.7) is used for implementing cross validation.

5.11 Results and Discussion

A series of experiments have been performed to evaluate the three supervised approaches and the one active learning approach presented. The systems are evaluated using a data set of 29175 clinical discharge letters. Performance measures used are precision, recall and F1-score. The following sections show the outcome of the study and discuss interesting findings.

5.11.1 Results of the Supervised Classification Approaches

The performances of the three variations of the supervised classifier are shown in Table 5.6 for data set sizes ranging from 25 documents to the full data set of 29175 documents. For each data set size and for each classifier version, the precision, recall and F1-scores are presented. The highest measures for every data set size are marked in bold. In addition to the results in tabular form, the F1-scores of the different classifiers are shown in a graph over the data set size in Figure 5.8. Overall, a maximum F1-score of 0.98 is achieved using CV2 for data set sizes 10000 and 25000.

CV1, CV2 and CV3 show very similar performance, with CV2 slightly leading for larger data sets. For very small data sets, CV3 performs best. With increasing data set size however, the performance of CV3 decreases compared to the other two. It is somewhat surprising that the most advanced classifier performs the worst. Possible explanations could be that some medical entities are not identified in the first step and don't even reach the second step. Thus, early mistakes progress. Another possible issue might be the combination of the multi-word terms to a single word. Through combining terms, the features

selected for the previous and following words change which could have an impact on performance.

	CV1			CV2			CV3		
files	P	R	F1	P	R	F1	P	R	F1
25	0.782	0.617	0.665	0.780	0.605	0.652	**0.827**	**0.633**	**0.685**
50	**0.873**	0.683	0.753	0.869	0.678	0.747	0.842	**0.713**	**0.759**
100	**0.882**	**0.736**	**0.793**	0.881	0.722	0.783	0.823	0.718	0.757
250	**0.936**	0.844	**0.882**	0.931	0.841	0.876	0.892	**0.850**	0.865
500	0.956	**0.897**	0.920	**0.957**	0.894	**0.924**	0.918	0.884	0.895
1000	0.962	**0.930**	**0.946**	**0.970**	0.929	**0.946**	0.939	0.916	0.925
2500	0.978	**0.954**	0.964	**0.979**	0.952	**0.967**	0.955	0.947	0.949
5000	0.981	**0.967**	**0.975**	**0.982**	0.966	0.972	0.961	0.951	0.955
10000	0.982	0.970	0.979	**0.986**	**0.970**	**0.980**	0.967	0.960	0.962
25000	0.981	0.971	0.976	**0.987**	**0.978**	**0.980**	0.963	0.951	0.956
29175	0.982	0.971	0.976	**0.984**	**0.973**	**0.979**	0.968	0.960	0.964

Table 5.6: Performance of the three variations of the supervised classifier for different data set sizes

Another observation can be made by comparing execution times, as is shown in Table 5.7 for a data set size of 2500 documents. These times can not be viewed as absolute values, since many other things would have to be considered, such as the development environment (see Section 5.1), or other applications running simultaneously. However, the ratios of the presented times are suitable to compare the different supervised classifier variations. CV2 and 3 are fastest, because they perform prior candidate phrase selection which leads to using less words for training. CV1 needs almost double the time as the other versions. This shows that the simplest classifier is the most time consuming, since it is trained on all words and not only on the candidate phrases.

	CV1	CV2	CV3
time in s	1341	758	619

Table 5.7: Execution time of the three variations of the supervised classifier for a data set of 2500 documents

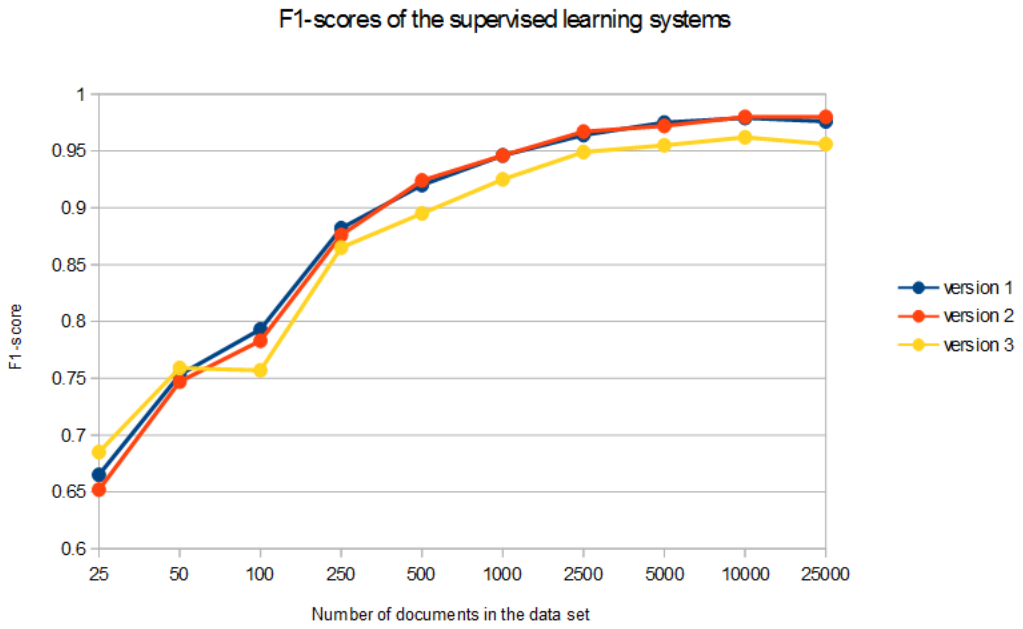


Figure 5.8: F1-scores of the three different supervised learning approaches over various data set size

5.11.2 Results of the Active Learning Approach

The active learning classifier is evaluated using data set sizes of 100, 1000, 10000, and 29175 clinical letters. For evaluation, precision, recall and F1-scores are measured for every iteration. The overall performance results are presented in Table 5.8 for every fifth iteration until a maximum of 200 iterations.

As expected, the performance steadily increases with every iteration. Through a suitable selection of query samples for the expert to label, the overall amount of training data needed is significantly reduced. For the full data set, the system was able to reach a F1-score of 0.92 after 100 iterations and a F1-score of 0.95 after 200 iterations. However, such high numbers of iterations might not be realistic considering that a human expert should normally be used for the labelling task.

Another observation is that the performance is smaller for larger data sets. This is probably due to the fact that the training data to testing data ratio decreases for larger data sets. The training data consists of the initial five documents and the additional five sentences per iteration, which is both independent of the overall data set size. The remaining sentences in the data set, which are not used for training, are then used for evaluation. Thus the number of sentences in the testing data set increases with the size of the overall data set.

For visualization purpose, the performance measures of the first 100 iterations using the full data set of 29175 clinical letters are shown in Figure 5.9. First of all, the graph shows how the F1-score is a weighted average of precision and recall, since the F1 curve is situated between the precision and recall curves. The graph also clearly illustrates that the increase in performance is of logarithmic shape rather than linear, meaning that the performance increases rather fast in the beginning and then slowly converges towards the maximum value.

To evaluate the effect of the uncertainty-based query selection approach, random sampling has also been implemented as a baseline. Figure 5.10 shows the F1-scores evaluated on the full data set of 29175 clinical letters. The F1-scores are presented as a function of the number of iterations. After 100 iterations, the uncertainty-based approach achieves a F1-score of 0.92, while the random sampling baseline only reached a F1-score of 0.71 within the same number of

	100 files			1000 files			10000 files			29175 files		
I	P	R	F1	P	R	F1	P	R	F1	P	R	F1
0	0.44	0.12	0.18	0.52	0.15	0.21	0.58	0.22	0.31	0.40	0.09	0.14
5	0.81	0.39	0.52	0.88	0.35	0.48	0.88	0.46	0.59	0.79	0.32	0.42
10	0.93	0.56	0.68	0.89	0.49	0.60	0.90	0.58	0.70	0.83	0.43	0.54
15	0.96	0.67	0.78	0.91	0.59	0.70	0.93	0.64	0.75	0.87	0.61	0.70
20	0.93	0.70	0.79	0.91	0.71	0.79	0.95	0.71	0.80	0.92	0.69	0.77
25	0.96	0.72	0.81	0.93	0.75	0.83	0.93	0.79	0.84	0.93	0.74	0.81
30	0.96	0.74	0.83	0.95	0.77	0.85	0.94	0.82	0.87	0.94	0.74	0.82
35	0.99	0.77	0.86	0.95	0.81	0.87	0.96	0.82	0.88	0.93	0.77	0.84
40	0.98	0.82	0.88	0.96	0.82	0.88	0.96	0.81	0.87	0.93	0.80	0.85
45	0.97	0.85	0.91	0.97	0.83	0.89	0.95	0.86	0.90	0.95	0.81	0.87
50	0.98	0.89	0.93	0.96	0.84	0.90	0.97	0.83	0.89	0.95	0.83	0.88
55	0.97	0.90	0.93	0.97	0.87	0.91	0.97	0.86	0.90	0.95	0.84	0.88
60	0.98	0.91	0.94	0.98	0.87	0.92	0.96	0.87	0.91	0.95	0.84	0.89
65	0.98	0.92	0.95	0.98	0.88	0.92	0.97	0.87	0.92	0.95	0.85	0.89
70	1	0.94	0.97	0.98	0.88	0.93	0.97	0.88	0.92	0.96	0.85	0.90
75	1	0.93	0.96	0.98	0.90	0.93	0.97	0.89	0.93	0.96	0.86	0.91
80	1	0.96	0.98	0.98	0.91	0.95	0.97	0.90	0.93	0.96	0.87	0.91
85				0.99	0.91	0.95	0.97	0.91	0.94	0.97	0.87	0.91
90				0.99	0.92	0.95	0.98	0.92	0.95	0.97	0.88	0.92
95				0.99	0.93	0.96	0.98	0.92	0.95	0.97	0.88	0.92
100				0.99	0.93	0.96	0.98	0.92	0.95	0.96	0.89	0.92
105				0.99	0.93	0.96	0.98	0.92	0.95	0.97	0.89	0.93
110				0.99	0.94	0.96	0.98	0.92	0.95	0.97	0.90	0.93
115				0.99	0.94	0.96	0.98	0.92	0.95	0.97	0.90	0.93
120				0.99	0.94	0.96	0.98	0.93	0.95	0.97	0.90	0.93
125				0.99	0.95	0.97	0.98	0.93	0.95	0.98	0.90	0.93
130				0.99	0.95	0.97	0.98	0.94	0.96	0.98	0.90	0.93
135				0.99	0.95	0.97	0.98	0.94	0.96	0.98	0.90	0.94
140				0.99	0.95	0.97	0.98	0.94	0.96	0.97	0.91	0.94
145				0.99	0.95	0.97	0.99	0.94	0.96	0.97	0.91	0.94
150				0.99	0.95	0.97	0.99	0.94	0.96	0.98	0.91	0.94
155				0.99	0.96	0.97	0.99	0.94	0.96	0.98	0.91	0.94
160				0.99	0.96	0.98	0.99	0.94	0.96	0.97	0.92	0.94
170							0.99	0.95	0.96	0.98	0.93	0.95
180							0.99	0.95	0.97	0.98	0.92	0.95
190							0.99	0.95	0.97	0.98	0.92	0.95
200							0.99	0.95	0.97	0.98	0.93	0.95

Table 5.8: Overall performance measures of the active learning approach for different data set sizes

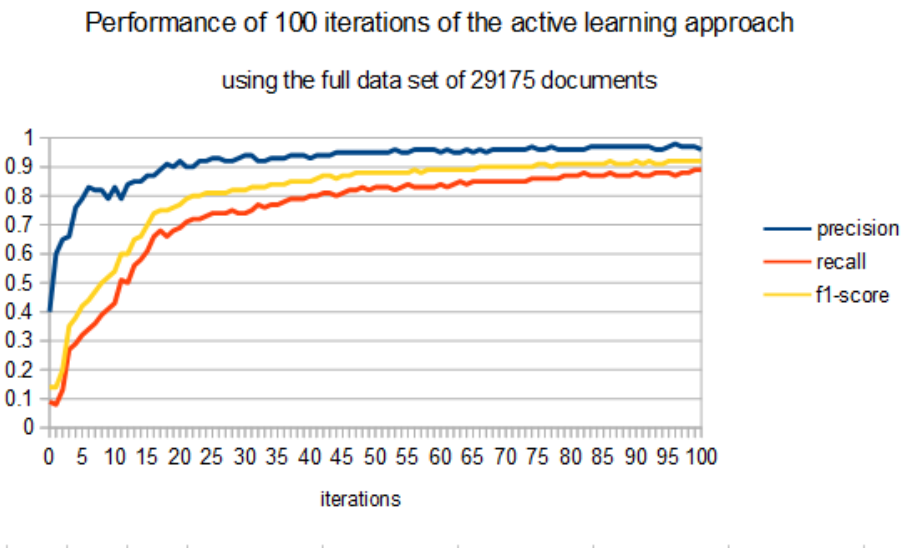


Figure 5.9: F1-score, precision and recall of 100 iterations of the active learning system using the full data set of 29175 documents

iterations. The uncertainty-based approach clearly outperforms the random selection baseline.

5.11.3 Comparison of the Different Approaches

In order to objectively compare the two approaches, it is important to remember that each clinical letter consists of approximately 7.61 sentences. The active learning approach randomly chooses five documents as an initial training data set and five sentences are added to the training data per iteration. The supervised learning systems use 10-fold cross-validation and hence the training data set is always nine-tenths of the full data set. Taking this information into consideration, it is easily possible to convert training data set sizes given as number of documents or iterations into number of sentences for objective comparison.

Table 5.9 shows the direct comparison of CV1 of the supervised approach and the active learning approach. CV1 has been chosen for comparison, because the active learning approach trains the model in the same manner as CV1. For

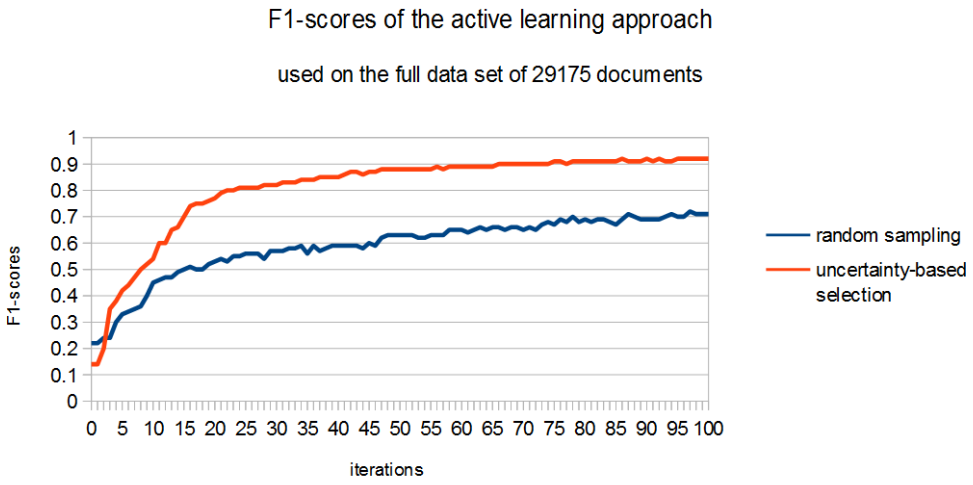


Figure 5.10: F1-score of 100 iterations of the active learning system using the full data set of 29175 documents. The implemented uncertainty-based approach for sampling is compared to random selection.

the case of using 100 files the supervised CV1 achieves a F1-score of 0.79 using a training set of 90 documents, or about 685 sentences. The active learning approach evaluated on 100 files is able to reach the same F1-score after only 20 iterations, which is equal to 138 sentences. As a consequence, the active learning approach requires less training data while achieving the same performance as the supervised approach.

training sentences	supervised learning			active learning 100 letters			active learning 29175 letters		
	P	R	F1	P	R	F1	P	R	F1
138	0.67	0.60	0.61	0.93	0.70	0.79	0.92	0.69	0.77
168	0.78	0.62	0.67	0.96	0.72	0.82	0.93	0.74	0.81
343	0.87	0.68	0.75	0.98	0.92	0.95	0.96	0.83	0.89
685	0.88	0.74	0.79	1	$\geq$0.96	$\geq$0.98	0.97	0.91	0.94

Table 5.9: Results of the supervised approach CV1 for data set sizes 20, 25, 50, and 100. Directly compared with the active learning approach used on a data set of 100 clinical letters and on the full data set.

Fur further analysis and visualization, two comparisons of the supervised CV1

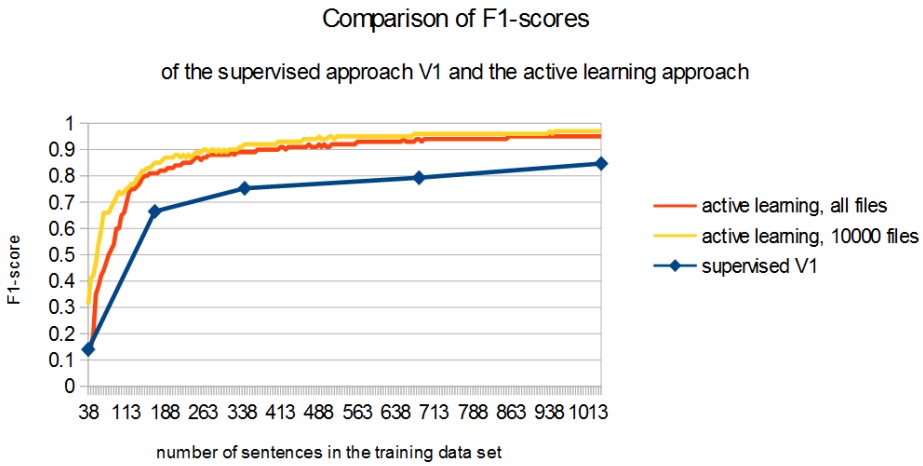
and the active learning classifier are presented in Figure 5.11. Subfigure 5.11a shows the F1-score as a function of the number of sentences in the training data set and Subfigure 5.11b presents the number of sentences needed to reach a certain F1-score. Again, it becomes clear that the active learning approach is able to achieve higher performance while requiring significantly less labelled data.

5.12 Summary

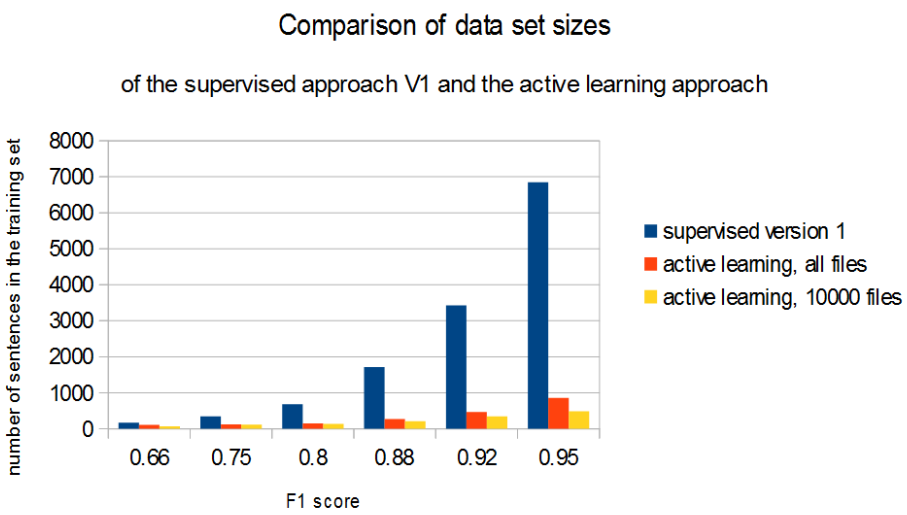
This work deals with medical entity recognition and semantic type classification. Three variations of a supervised classifier and one active learning classifier have been presented and evaluated. A data set of 29175 clinical discharge letters was used for evaluation. Performance measures consisted of precision, recall, and F1-score.

Of the three supervised approaches, CV1 and CV2 outperformed CV3. The best results were achieved by CV2 with an F1-score of 0.98. The identification of the medical terms and the classification of the semantic types are executed in separate steps in CV3. Due to the separation of the two tasks, mistakes made in the first step are transferred to the second step, which is a possible reason for the inferior performance of CV3. Directly comparing the other two approaches, CV1 performed slightly better for smaller data set and CV2 achieved slightly better results for larger data sets. However, CV2 is more time efficient than CV1. CV2 performs candidate phrase chunking before the actual classification, which leads to less training data samples and makes this approach less time consuming.

The active learning system was able to reach a competitive performance with less labelled training data. Using the full data set, the active learning system achieves a F1-score of 0.92 after 100 iterations. The active learning approach uses uncertainty-based sampling for selecting the data samples to present to the expert for labelling. As a baseline, random sampling was also implemented. The uncertainty-based active learning approach was able to significantly outperform the random sampling baseline.



(a) F1-score as function of training data set sizes



(b) Number of sentences in training data set to reach F1-scores

Figure 5.11: Comparison of the supervised classifier CV1 and the active learning system. The active learning approach is evaluated on the full data set and on a smaller data set of 10000 clinical letter.

For training the model of the active learning system, the supervised CV1 was used, which allows for interesting comparison. Using a training data set of about 685 sentences, the supervised CV1 achieved a F1-score of 0.79. The active learning approach reached the same F1-score after only 20 iterations using a total of 138 training sentences. Consequently, the active learning system was able to reach the same performance while reducing the training data by 80%.

Finally, the two research questions which have been asked in Section 4.5 can now be addressed. The insights gained from this study lead to the following answers:

1. Will the more complex supervised approaches perform better than the basic single-step approach?

   A: Based on the setup and experimentation, results indicate that CV1 and CV2 outperformed the more complex CV3. Overall, CV2 demonstrated the best performance. The medical entity recognition approach CV2 combined prior candidate phrase chunking with single-step identification and classification.

2. Will the active learning system be able to significantly reduce the amount of data needed to reach the same or higher performance scores than the supervised approaches?

   A: In consideration of the setup used in this experiment, the active learning approach reached the same F1-score as the supervised approach CV1 while using decidedly less training data. For the example of a F1-score of 0.79, the active learning approach was able to reduce the necessary training data by 80%.

6 Conclusion

The majority of medical data is only available in unstructured natural language text format (Liu et al., 2015). Clinical letters contain a great amount of useful information, which is hidden in unstructured text. Consequently, there is a high demand for systems with the ability to extract this valuable information. Medical entity recognition aims to identify and classify informative medical terms in text documents, thus making it an essential part of medical NLP and text analysis(Abacha & Zweigenbaum, 2011).

In this thesis, different supervised and active learning medical entity recognition and type classification systems have been presented. The approaches were evaluated using a collection of clinical discharge letters and demonstrated successful performance. The following section gives an overview of the most interesting findings from literature survey, development and experimentation of this work. Finally, a summary about possible future research directions concludes this thesis.

6.1 Summary of this Thesis

The main objective of this thesis is the creation of a system for automatic medical term extraction and classification of the corresponding semantic type. During the course of this work, many interesting observations have been made and the most relevant findings and results are now discussed.

The foundations of medical entity recognition are computer science, artificial intelligence and linguistics. Compared to the closely related field of NER, medical entity recognition has to deal with additional challenges, such as multi-word terms, abbreviations and other peculiarities of medical language. In general medical entity recognition consists of two tasks: (1) identifying the

medical entity and its boundaries within the sentence, and (2) classifying the semantic type of the extracted entity based on a set of pre-defined categories (Abacha & Zweigenbaum, 2011; S. Zhang & Elhadad, 2013). The insights gained from the literature survey of the general area can easily be adapted to the medical field, e.g. IOB format labelling. Supervised machine learning and the CRF classifier are popular for both general and medical applications. Research also shows that active learning is a promising alternative to the more established supervised approach. Moreover, the literature review demonstrates that there are still many challenges to overcome in the field of medical entity recognition.

Building on the observations gained through the background research and the literature survey, three different supervised learning variations and one active learning approach are designed. All approaches employ the CRF model for classification combined with IOB format labels and common preprocessing steps (e.g. POS tagging). The three versions of the supervised system are composed of a single-step approach (CV1), an additional step of candidate phrase extraction step before classification (CV2), and a stepwise approach which performs the identification and the type classification of the medical entity separately (CV3). The active learning model builds on CV1 and selects the training sample to present to the expert based on uncertainty.

The experiments are carried out using a data set of 29175 clinical discharge letters. Precision, recall and F1-score are employed for evaluation. Overall CV2 achieves the best results with an F1-score of 0.98. The more simple CV1 shows a similar performance, but is more time consuming. CV3 is outperformed by the other approaches, because the mistakes made in the entity identification step are carried on to the semantic type classification step. The performance of the uncertainty-based sampling approach for active learning clearly exceeds the baseline of random sampling. The comparison of the supervised approaches with the active learning approach confirms that through selecting training samples iteratively and based on uncertainty, the amount of labelled data can be dramatically reduced.

In summary, the best performing supervised system (CV2) uses candidate phrase extraction and IOB labels combined with CRF. The active learning approach uses uncertainty-based sampling together with IOB format labels and

CRF. The insights gained in this study give rise to interesting ideas for further research, which will be briefly discussed in the following section.

6.2 Future Outlook

Based on the presented medical entity recognition approaches, further research could include (1) exploring other alternatives to supervised learning, such as semi-supervised learning, (2) adding further steps to the preprocessing pipeline, e.g. spell checking, (3) using the more advanced BILOU format instead of the IOB labels , (4) adapting the presented systems to a multilingual medical entity recognition problem, or (5) using a different machine learning model for classification, such as SVM. These points just give an idea of further research possibilities. The encountered limitations of the presented medical entity recognition and semantic type classification approaches are now discussed and specific alternatives for future work are described.

The ophthalmology data set used in this work was originally unannotated and labels were added using the medical terms that have been extracted during an unsupervised study on the data set by Liu et al. (2015). As a result, the annotations are not exactly ground truth, but still suitable for testing. For future work, it would be interesting to evaluate the presented approaches on a different data set. However, it is difficult to access annotated data sets in the medical domain for privacy reasons.

The findings of a first analysis of the data after preprocessing (see Section 5.7.5) demonstrated that many of the multi-word medical terms contain one or more other medical terms which might be of a different type. A study presented by S. Zhang and Elhadad (2013) showed that nested medical terms occur frequently in clinical notes. Gong et al. (2015) proposed a biomedical entity recognition system for handling these nested terms. The multi-word terms are iterated through using of a window with increasing size to find all medical entities contained in the multi-word terms. This approach is intriguing and would also be suitable for the ophthalmology data set. Thus, including the nested entity approach into the presented systems is another future research possibility.

The used features in this study consist of lexical features, syntactic features, and sentence based features. Semantic features contain semantic type information from dictionaries or ontologies, such as the UMLS. Adding the semantic categories provided by dictionary lookup to the feature set used for this work could lead to an increase in performance. Nevertheless, incorporating domain specific information will lead to less adaptability to other areas. Other features that have shown to increase performance are word clusters (Y. Xu et al., 2012) or distributional based word representations (Tang, Cao, Wu, Jiang, & Xu, 2013). Hence, a further option for future research is adding extra features for classification.

The initial documents selected for training of the active learning classifier have an impact on overall performance. The implemented classifier chooses these documents randomly. Other initial selection methods could be explored, e.g. training the classifier with one example for each semantic category. Considering the query selection of the active learning classifier, a downside of uncertainty-based sampling by using the posterior probabilities of the CRF is that the model needs to train and predict labels for every iteration which is time consuming. Hence, alternative methods for selecting the training samples could be investigated. Even though the diversity based approach has been outperformed by the uncertainty based approach in a study by Chen et al. (2015), it would anyhow be interesting to implement it as an alternative. A more advanced sampling method was proposed by Huang, Jin, and Zhou (2014), which selects informative and representative sample, thus combining uncertainty and diversity based approaches. Adapting this approach for the presented active learning system in this thesis is another option for further research.

Bibliography

Abacha, A. B. & Zweigenbaum, P. (2011). Medical entity recognition: a comparison of semantic and statistical methods. In *2011 workshop on biomedical natural language processing* (2, pp. 56–64).

Aronson, A. R. (2001, January). Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. *Proceedings of the AMIA Symposium*, 17–21.

Baum, L. E. & Petrie, T. (1966, December). Statistical Inference for Probabilistic Functions of Finite State Markov Chains. *The Annals of Mathematical Statistics*, *37*(6), 1554–1563. doi:10.1214/aoms/1177699147

Blum, A. & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on computational learning theory* (pp. 92–100). New York, New York, USA: ACM Press. doi:10.1145/279943.279962. arXiv: arXiv:1011.1669v3

Bodnari, A., Deléger, L., Lavergne, T., Névéol, A., & Zweigenbaum, P. (2013). A supervised named-entity extraction system for medical text. In *Clef workshop proceedings* (Vol. 1179, pp. 1–8).

Breiman, L. (1996). Bagging Predictors. *Machine Learning*, *24*, 123–140.

Brill, E. (1992). A simple rule-based part of speech tagger. In *Proceedings of the workshop on speech and natural language - hlt '91* (pp. 112–116). Morristown, NJ, USA: Association for Computational Linguistics. doi:10.3115/1075527.1075553. arXiv: 9406010 [cmp-lg]

Bunescu, R. & Pasca, M. (2006). Using Encyclopedic Knowledge for Named Entity Disambiguation. *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, (April), 3–7.

BusinessWire. (2012). Columbia Grants Health Fidelity Exclusive License to MedLEE NLP. Retrieved July 25, 2016, from http://www.businesswire.com / news / home / 20120111006135 / en / Columbia - Grants - Health - Fidelity-Exclusive-License-MedLEE

Cameron, S. & Turtle-song, I. (2002). Learning to write case notes using the SOAP format. *Journal of Counseling & Development*, *80*(3), 286–292. doi:10.1002/j.1556-6678.2002.tb00193.x

Carlson, A., Betteridge, J., Wang, R. C., Hruschka, E. R., & Mitchell, T. M. (2010, February). Coupled semi-supervised learning for information extraction. In *Proceedings of the third acm international conference on web search and data mining - wsdm '10* (pp. 101–110). New York, New York, USA: ACM Press. doi:10.1145/1718487.1718501

Chen, Y., Lasko, T. A., Mei, Q., Denny, J. C., & Xu, H. (2015, December). A study of active learning methods for named entity recognition in clinical text. *Journal of biomedical informatics*, *58*, 11–8. doi:10.1016/j.jbi.2015.09.010

Chowdhury, G. G. (2005). Natural language processing. *Annual Review of Information Science and Technology*, *37*, 51–89. doi:10.1002/aris.1440370103. arXiv: 0812.0143v2

Chris, D. P. (1990). Another stemmer. In *Acm sigir forum* (Vol. 24, *3*, pp. 56–61).

Christensen, L. M., Haug, P. J., & Fiszman, M. (2002). MPLUS: a probabilistic medical language understanding system. In *Proceedings of the acl-02 workshop on natural language processing in the biomedical domain* (July, pp. 29–36). doi:10.3115/1118149.1118154

Cohen, A. M. (2005, January). A survey of current work in biomedical text mining. *Briefings in Bioinformatics*, *6*(1), 57–71. doi:10.1093/bib/6.1.57

Cortes, C. (1995). Support-Vector Networks. *Machine Learning*, *20*, 273–297.

Cunningham, H., Maynard, D., Bontcheva, K., & Tablan, V. (2002). GATE, a General Architecture for Text Engineering. *Computers and the Humanities*, *36*(2), 223–254. doi:10.1023/A:1014348124664

Doan, S., Collier, N., Xu, H., Pham, H. D., & Tu, M. P. (2012, January). Recognition of medication information from discharge summaries using ensembles of classifiers. *BMC medical informatics and decision making*, *12*(1), 36. doi:10.1186/1472-6947-12-36

Eddy, S. R. (1996, June). Hidden Markov models. *Current Opinion in Structural Biology*, *6*(3), 361–365. doi:10.1016/S0959-440X(96)80056-X

Faruqui, M. & Padó, S. (2010). Training and evaluating a German named entity recognizer with semantic generalization. In *Proceedings of konvens 2010* (pp. 129–133).

Ferrucci, D. & Lally, A. (2004). UIMA: An Architectural Approach to Unstructured Information Processing in the Corporate Research Environment. *Journal of Natural Language Engineering*, *10*(3-4), 327–348.

Finkel, J., Grenager, T., & Manning, C. (2005, June). Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics - acl '05* (pp. 363–370). Morristown, NJ, USA: Association for Computational Linguistics. doi:10.3115/1219840.1219885

Florian, R., Ittycheriah, A., Jing, H., & Zhang, T. (2003, May). Named entity recognition through classifier combination. In *Proceedings of the seventh conference on natural language learning at hlt-naacl 2003* - (Vol. 4, pp. 168–171). Morristown, NJ, USA: Association for Computational Linguistics. doi:10.3115/1119176.1119201

Frantzi, K., Ananiadou, S., & Mima, H. (2000, August). Automatic recognition of multi-word terms:. the C-value/NC-value method. *International Journal on Digital Libraries*, *3*(2), 115–130. doi:10.1007/s007999900023

Freund, Y. (1995). Boosting a weak learning algorithm by majority. In *Information and computation*.

Friedman, C. & Hripcsak, G. (1999). Natural language processing and its future in medicine. *Acad Med*, *74*(8), 890–895.

Friedman, C., Johnson, S. B., Forman, B., & Starren, J. (1995). Architectural requirements for a multipurpose natural language processor in the clinical environment. In *Proceedings of the symposium on computer applications in medical care* (pp. 347–51). American Medical Informatics Association.

Friedman, C., Rindflesch, T. C., & Corn, M. (2013, October). Natural language processing: state of the art and prospects for significant progress, a workshop sponsored by the National Library of Medicine. *Journal of biomedical informatics*, *46*(5), 765–73. doi:10.1016/j.jbi.2013.06.004

Garla, V. N. & Brandt, C. (2012, October). Ontology-guided feature engineering for clinical text classification. *Journal of biomedical informatics*, *45*(5), 992–8. doi:10.1016/j.jbi.2012.04.010

Goldman, B. (2012). King of the Mountain. *Stanford Medicine Magazine*, (2012 Summer). Retrieved from http://sm.stanford.edu/archive/stanmed/2012summer/article3.html

Gong, L., Yang, R., Feng, J., & Yang, G. (2015, July). A combined approach for the extraction of the multi-word and nested biomedical entity. In *2015*

ieee international conference on digital signal processing (dsp) (pp. 708–711). IEEE. doi:10.1109/ICDSP.2015.7251967

Grishman, R. & Sundheim, B. (1996). Message Understanding Conference-6: A Brief History. In *Proceedings of the 16th conference on computational linguistics* - (Vol. 1, p. 466). Morristown, NJ, USA: Association for Computational Linguistics. doi:10.3115/992628.992709

Grouin, C., Tannier, X., Hamon, T., Kelly, L., Goeuriot, L., & Zweigenbaum, P. (2015). CLEF eHealth Evaluation Lab 2015 Task 1b : clinical named entity recognition. In *Clef workshop proceedings*.

Han, J., Pei, J., & Kamber, M. (2011). *Data mining: concepts and techniques*. doi:10.1002/1521-3773(20010316)40:6⟨9823::AID-ANIE9823⟩3.3.CO;2-C

Haug, P. J., Koehler, S., Lau, L. M., Wang, P., Rocha, R., & Huff, S. M. (1995). Experience with a mixed semantic/syntactic parser. *Proceedings of the Annual Symposium on Computer Application n Medical Care*. 284. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/8563286%20http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC2579100

Haug, P. J., Ranum, D. L., & Frederick, P. R. (1990, February). Computerized extraction of coded findings from free-text radiologic reports. Work in progress. *Radiology*, *174*(2), 543–548. doi:10.1148/radiology.174.2.2404321

Hawkins, D. M. (2004, January). The Problem of Overfitting. *Journal of Chemical Information and Computer Sciences*, *44*(1), 1–12. doi:10.1021/ci0342472

Hendrickx, I. & Bosch, A. V. D. (2003). Memory-based one-step named-entity recognition: Effects of seed list features, classifier stacking, and unannotated data. In *Proceedings of conll-2003* (Vol. 4, pp. 176–179). Morristown, NJ, USA: Association for Computational Linguistics. doi:10.3115/1119176.1119203

Huang, S. J., Jin, R., & Zhou, Z.-H. (2014). Active Learning by Querying Informative and Representative Examples. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*(10), 1936–1949. doi:10.1109/TPAMI.2014.2307881

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999, September). Data clustering: a review. *ACM Computing Surveys*, *31*(3), 264–323. doi:10.1145/331499.331504

Jiang, M., Chen, Y., Liu, M., Rosenbloom, S. T., Mani, S., Denny, J. C., & Xu, H. (2011, January). A study of machine-learning-based approaches to extract clinical entities and their assertions from discharge summaries.

Journal of the American Medical Informatics Association : JAMIA, *18*(5), 601–6. doi:10.1136/amiajnl-2011-000163

Jordan, M. I. & Ng, A. Y. (2002). On generative vs. discriminative classifiers: A comparison of logistic regression and naive bayes. *Advances in Neural Information Processing Systems*, *14*, 841. doi:10.1007/s11063-008-9088-7. arXiv: /dx.doi.org/10.1007/s11063-008-9088-7 [http:]

Keretna, S., Lim, C. P., & Creighton, D. (2015, October). Enhancement of Medical Named Entity Recognition Using Graph-Based Features. In *2015 ieee international conference on systems, man, and cybernetics* (pp. 1895–1900). IEEE. doi:10.1109/SMC.2015.331

Keretna, S., Lim, C. P., Creighton, D., & Shaban, K. B. (2014, October). Classification ensemble to improve medical Named Entity Recognition. In *Systems, man and cybernetics (smc), 2014 ieee international conference on* (pp. 2630–2636). IEEE. doi:10.1109/SMC.2014.6974324

Koehn, P. (2005). Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of the mt summit* (Vol. 5, pp. 79–86). doi:10.3115/1626355.1626380

Lafferty, J., McCallum, A., & Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML '01 Proceedings of the Eighteenth International Conference on Machine Learning*, *8*(June), 282–289. doi:10.1038/nprot.2006.61. arXiv: arXiv:1011.4088v1

Li, Y., Wang, C., Han, F., Han, J., Roth, D., & Yan, X. (2013). Mining evidences for named entity disambiguation. In *Proceedings of the 19th acm sigkdd international conference on knowledge discovery and data mining* (pp. 1070–1078). doi:10.1145/2487575.2487681

Lindberg, C. (1990, May). The Unified Medical Language System (UMLS) of the National Library of Medicine. *American Medical Record Association*, *61*(5), 40–2.

Liu, W., Chung, B. C., Wang, R., Ng, J., & Morlet, N. (2015). A genetic algorithm enabled ensemble for unsupervised medical term extraction from clinical letters. *Health Information Science and Systems*, *3*(1), 5. doi:10.1186/s13755-015-0013-y

Loper, E. & Bird, S. (2002). NLTK: the Natural Language Toolkit. In *Proceedings of the acl-02 workshop on effective tools and methodologies for teaching natural language processing and computational linguistics* (Vol. 1,

pp. 63–70). Morristown, NJ, USA: Association for Computational Linguistics. doi:10.3115/1118108.1118117

MacQueen, J. (1967). Some Methods for Classification and Analysis of Multivariate Observations. In *Proceedings of the fifth berkeley symposium on mathematical statistics and probability* (pp. 281–297).

McGray, A. T., Sponsler, J. L., Brylawski, B., & Browne, A. C. (1987). The Role of Lexical Knowledge in Biomedical Text Understanding. In *Proceedings of the annual symposium on computer application in medical care* (p. 103). American Medical Informatics Association.

Meystre, S. M., Savova, G. K., Kipper-Schuler, K., & Hurdle, J. F. (2008). Extracting Information from Textual Documents in the Electronic Health Record: A Review of Recent Research. *IMIA Yearbook of Medical Informatics*, *35*(1), 128–44.

Mihalcea, R. & Tarau, P. (2004, July). TextRank: Bringing Order into Texts. In *Proceedings of emnlp-04 the 2004 conference on empirical methods in natural language processing*. Association for Computational Linguistics.

Minard, A.-L., Ligozat, A.-L., Abacha, A. B., Bernhard, D., Cartoni, B., Deléger, L., ... Grouin, C. (2011, January). Hybrid methods for improving information access in clinical documents: concept, assertion, and relation identification. *Journal of the American Medical Informatics Association : JAMIA*, *18*(5), 588–93. doi:10.1136/amiajnl-2011-000154

Nadeau, D. & Sekine, S. (2007). A survey of named entity recognition and classification. *Lingvisticae Investigationes*, *30*(1), 3–26. doi:10.1075/li.30.1.03nad

Nocedal, J. (1980). Updating quasi-Newton matrices with limited storage. *Mathematics of Computation*, *35*(151), 773–782. doi:10.1090/S0025-5718-1980-0572855-7

Nothman, J., Ringland, N., Radford, W., Murphy, T., & Curran, J. R. (2013, January). Learning multilingual named entity recognition from Wikipedia. *Artificial Intelligence*, *194*, 151–175. doi:10.1016/j.artint.2012.03.006

Opitz, D. & Maclin, R. (1999). Popular Ensemble Methods: An Empirical Study. *Journal of Artificial Intelligence Research*, *11*, 169–198.

Pei, J., Han, J., Mortazavi-Asl, B., Wang, J., Pinto, H., Chen, Q., ... Hsu, M.-C. (2004). Mining Sequential Patterns by Pattern–Growth: The PrefixSpan Approach. *IEEE Transactions of Knowledge and Data Engineering*, *16*(11), 1424–1440.

Porter, M. F. (1980, April). An algorithm for suffix stripping. *Program*, *14*(3), 130–137.

Quinlan, J. R. (1986). Induction of Decision Trees. *Machine Learning*, *1*(1), 81–106. doi:10.1023/A:1022643204877

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257–286. doi:10.1109/5.18626. arXiv: arXiv:1011.1669v3

Ramshaw, L. A. & Marcus, M. P. (1995, May). Text Chunking using Transformation-Based Learning. *arXiv preprint cmp-lg/9505040*. arXiv: 9505040 [cmp-lg]

Ratinov, L. & Roth, D. (2009, June). Design challenges and misconceptions in named entity recognition. In *Conll '09 proceedings of the thirteenth conference on computational natural language learning* (pp. 147–155). Association for Computational Linguistics.

Rish, I. (2001). An empirical study of the naive Bayes classifier. In *International joint conferences on artificial intelligence 2001 workshop on empirical methods in artificial intelligence* (pp. 41–46).

Rowley, J. (2007, February). The wisdom hierarchy: representations of the DIKW hierarchy. *Journal of Information Science*, *33*(2), 163–180. doi:10.1177/0165551506070706

Sager, N., Friedman, C., & Lyman, M. (1987). *Medical Language Processing: Computer Management of Narrative Data*. Reading, Mass: Addison-Wesley.

Saha, S. & Ekbal, A. (2013). Combining multiple classifiers using vote based classifier ensemble technique for named entity recognition. *Data & Knowledge Engineering*, *85*, 15–39. doi:10.1016/j.datak.2012.06.003

Santorini, B. (1990). Part-of-Speech Tagging Guidelines for the Penn Treebank Project (3rd Revision). *Technical Reports (CIS)*, (Paper 570). doi:10.1017/CBO9781107415324.004. arXiv: arXiv:1011.1669v3

Savova, G. K., Kipper-Schuler, K., Buntrock, J., & Chute, C. G. (2008). UIMA-based clinical information extraction system. *Towards Enhanced Interoperability for Large HLT Systems: UIMA for NLP*, *39*.

Schapire, R. E. (1990, June). The strength of weak learnability. *Machine Learning*, *5*(2), 197–227. doi:10.1007/BF00116037

Settles, B. (2010). Active Learning Literature Survey. *Machine Learning*, *15*(2), 201–221. doi:10.1.1.167.4245

Sha, F. & Pereira, F. (2003). Shallow parsing with conditional random fields. In *Proceedings of the 2003 conference of the north american chapter of the*

association for computational linguistics on human language technology (pp. 134–141). Morristown, NJ, USA: Association for Computational Linguistics. doi:10.3115/1073445.1073473

Spyns, P. (1996). Natural Language Processing in Medicine: An Overview. *Methods of information in medicine*, *35*(4), 285–301.

Suakkaphong, N., Zhang, Z., & Chen, H. (2011, April). Disease Named Entity Recognition Using Semisupervised Learning and Conditional Random Fields. *Journal of the American Society for Information Science and Technology*, *62*(4), 727–737. doi:10.1002/asi.21488

Sutton, C. & McCallum, A. (2011). An Introduction to Conditional Random Fields. *Foundations and Trends in Machine Learning*, *4*(4), 267–373. doi:10.1561/2200000013. arXiv: arXiv:1011.4088v1

Tang, B., Cao, H., Wu, Y., Jiang, M., & Xu, H. (2012, October). Clinical entity recognition using structural support vector machines with rich features. In *Proceedings of the acm sixth international workshop on data and text mining in biomedical informatics - dtmbio '12* (p. 13). New York, New York, USA: ACM Press. doi:10.1145/2390068.2390073

Tang, B., Cao, H., Wu, Y., Jiang, M., & Xu, H. (2013, January). Recognizing clinical entities in hospital discharge summaries using Structural Support Vector Machines with word representation features. *BMC medical informatics and decision making*, *13 Suppl 1*(1), S1. doi:10.1186/1472-6947-13-S1-S1

Tao, C., Song, D., Sharma, D., & Chute, C. G. (2013, October). Semantator: semantic annotator for converting biomedical text to linked data. *Journal of biomedical informatics*, *46*(5), 882–93. doi:10.1016/j.jbi.2013.07.003

Tjong Kim Sang, E. F. & De Meulder, F. (2003). Introduction to the CoNLL-2003 shared task. In *Proceedings of the seventh conference on natural language learning at hlt-naacl 2003* - (Vol. 4, pp. 142–147). Morristown, NJ, USA: Association for Computational Linguistics. doi:10.3115/1119176.1119195. arXiv: 0306050 [cs]

Toutanova, K., Klein, D., & Manning, C. D. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 conference of the north american chapter of the association for computational linguistics on human language technology - volume 1 (naacl '03)* (pp. 252–259). doi:10.3115/1073445.1073478

Toutanova, K. & Manning, C. D. (2000). Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the*

2000 joint sigdat conference on empirical methods in natural language processing and very large corpora held in conjunction with the 38th annual meeting of the association for computational linguistics (Vol. 13, pp. 63–70). doi:10.3115/1117794.1117802

Uzuner, Ö., South, B. R., Shen, S., & DuVall, S. L. (2011, January). 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association : JAMIA*, *18*(5), 552–6. doi:10.1136/amiajnl-2011-000203

Wang, Y. (2009, August). Annotating and recognising named entities in clinical notes. In *Proceedings of the acl-ijcnlp 2009 student research workshop* (August, pp. 18–26). Association for Computational Linguistics. doi:10.3115/1667884.1667888

Weizenbaum, J. (1966, January). ELIZA-a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, *9*(1), 36–45. doi:10.1145/365153.365168

Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*, *5*(2), 241–259. doi:10.1016/S0893-6080(05)80023-1

Xu, H., Stenner, S. P., Doan, S., Johnson, K. B., Waitman, L. R., & Denny, J. C. (2010). MedEx: a medication information extraction system for clinical narratives. *Journal of the American Medical Informatics Association : JAMIA*, *17*(1), 19–24. doi:10.1197/jamia.M3378

Xu, Y., Hong, K., Tsujii, J., & Chang, E. I.-C. (2012, January). Feature engineering combined with machine learning and rule-based methods for structured information extraction from narrative clinical discharge summaries. *Journal of the American Medical Informatics Association : JAMIA*, *19*(5), 824–32. doi:10.1136/amiajnl-2011-000776

Yosef, M. A., Spaniol, M., & Weikum, G. (2014). AIDArabic A Named-Entity Disambiguation Framework for Arabic Text. In *The emnlp 2014 workshop on arabic natural language processing* (pp. 187–195).

Zhang, K., Xie, Y., Yang, Y., Sun, A., Liu, H., & Choudhary, A. (2014, October). Incorporating conditional random fields and active learning to improve sentiment identification. *Neural networks : the official journal of the International Neural Network Society*, *58*, 60–7. doi:10.1016/j.neunet.2014.04.005

Zhang, S. & Elhadad, N. (2013, December). Unsupervised biomedical named entity recognition: experiments with clinical and biological texts. *Journal of biomedical informatics*, *46*(6), 1088–98. doi:10.1016/j.jbi.2013.08.004

Zhou, Z.-H. & Li, M. (2010, May). Semi-supervised learning by disagreement. *Knowledge and Information Systems*, *24*(3), 415–439. doi:10.1007/s10115-009-0209-z

Zhu, X. & Goldberg, A. B. (2009, January). *Introduction to Semi-Supervised Learning*. Morgan & Claypool Publishers. doi:10.2200/S00196ED1V01Y-200906AIM006. arXiv: 1412.6596