NTNU – Trondheim
Norwegian University of
Science and Technology

Automated Annotation of Events Related to Central Venous Catheterization in Norwegian Clinical Notes

Ingrid Andås Berg

Healthcare Informatics
Submission date: March 2014
Supervisor: Øystein Nytrø, IDI

Norwegian University of Science and Technology
Department of Computer and Information Science

Norwegian University of Science and Technology

Master Thesis

Automated annotation of events related to central venous catheterisation in Norwegian clinical notes

Author:
Ingrid Andås Berg

Supervisor:
Assoc. Prof. Øystein Nytrø

A thesis submitted in partial fulfillment of the requirements for the degree of MSc in Medical Technology, specialisation Healthcare informatics

at the

Department of Computer and Information Science
Information Systems Group

March 2014

NTNU – Trondheim
Norwegian University of
Science and Technology

NORWEGIAN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Abstract

The Faculty of Natural Sciences and Technology
Department of Computer and Information Science

MSc in Medical Technology, specialisation Healthcare informatics

Automated annotation of events related to central venous catheterisation in Norwegian clinical notes

by Ingrid Andås Berg

Health personnel are required to use Electronic Health records for documentation and communication. Clinical notes from such records contain valuable information, but unfortunately this is often in narrative form, making it difficult to retrieve and extract information from them. One such problem is to get an overview of the number of patient days for patients with central venous catheter (CVC). The risk of infections increase with an increasing number of patient days. The present study examines the utility of applying NER to extract CVC related events from clinical notes. No studies have previously examined this application for Norwegian Clinical notes. Conditional random fields are used to make models based on different feature sets. The feature sets are combinations of word window, stem, synonymous and International classification for Nursing Practice (ICNP) axis. A corpus manually annotated with CVC event types was used for training and testing different models using three-fold cross-validation. Sixteen different combinations of features were tested. A factorial analysis using the three cross-fold runs as blocks was conducted to determine which features had the greatest effect on performance. Word window, ICNP axis and an interaction effect between these were found to affect performance significantly. Stem had an effect on recall, whereas no such effect was found for precision. An interaction effect between synonymous and ICNP-axis was found to effect precision. Accumulative scores of the different label types gave a precision of 56.29 %, a recall of 39.4 % and a f-measure of 46.33 for the best feature combination. Overlapping labels, errors in corpus and manual annotation are sources of error in the study. Thus, further research is necessary to draw certain conclusions about the present findings.

Preface

This research represents my Master thesis written in 2013 at the Department of Computer and Information Science (IDI) at the Norwegian University of Science and Technology (NTNU). The project is part of the Evicare project [1], Evidence-based care processes: Integrating knowledge in clinical information systems. The National Knowledge Center for Healthcare is responsible for the project, and the project is led by the director of the center, Geir Bukholm. The project is also supported by The Research Council of Norway, DIPS ASA, Datakvalitet AS, Innlandet Hospital Trust, Akershus University Hospital, Oslo University Hospital, The National Health Library and NTNU. At NTNU, the project is led by Associate Professor Øystein Nytrø from the Department of Computer and Information Sciences.

As part of integrating clinical records and clinical guidelines, this thesis will focus on Named Entity Recognition of clinical records, with a particular focus on catheter related events.

Acknowledgements

I would like to thank my supervisor Øystein Nytrø for providing the idea for the thesis, for supervision and for being kind, helpful and patient throughout the project. I would also like to thank PHD students Hans Moen, Thomas Brox Røst and Professor Rune Sætre for being helpful, answering all sorts of questions and for co-supervision. Thanks to Sætre for help with python related questions and to Moen for help with synonym handling and ICNP extraction. Thanks to Professor Laura Slaughter and PHD student Christine Tvedt for discussions and for answering ontology and annotation related questions. Thanks to Tvedt for performing manual annotation. Thanks to Haldor Husby from Akershus University Hospital for useful discussions regarding the corpus and to Dips ASA for providing data. Thanks to Associate Professor, Mette Langaas for answerig questions regarding statistics. Thanks to Maria Skeppstedt and Sumithra Velupillai for answering questions regarding CRF templates. Thanks to master students Jaspreet Singh and Roar Bjurstrøm for useful discussions.

Last but not least thanks for support from all of my family; Especially my grandfather, Einar Berg for being supportive and helpful regarding language-related questions and last but not least thanks to my fiancé, Ove-Joakim Istad for being supportive and to my four months old son for being a smiling, happy and motivating child!

. . .

Contents

List of Figures

List of Tables

Abbreviations

ACE	**A**utomatic **C**ontext **E**xtraction
AdvP	**A**dverb **P**hrase
AE	**A**dverse **E**vent
AP	**A**djective **P**hrase
CRBSI	**C**atheter- **R**elated **B**loodstream **I**nfection
CG	**C**onstraint **G**rammar
CoNLL	**C**onference on **N**atural **L**anguage **L**earning
CRF	**C**onditional- **R**andom **F**ields
DRG	**D**iagnosis- **R**elated **G**roup
EBM	**E**vidence **B**ased **M**edicine
EHR	**E**lectronic **H**ealth **R**ecord
EPR	**E**lectronic **P**atient **R**ecord
EVT	**E**vent
FS	**F**inite **S**tate grammar
GP	**G**eneral **P**ractitioner
HMM	**H**idden **M**arkov **M**odel
HPN	**H**ome **P**arenteral **N**utrition
IAA	**I**nter **A**nnotator **A**greement
ICD	**I**nternational **C**lassification of **D**iseases
ICNP	**I**nternational **C**lassification for **N**ursing **P**ractice
IOB	**I**nside **O**utside **B**egin
IR	**I**nformation **R**etrieval
LOC	**L**ocation
MC	**M**arkov **C**hain
MEMM	**M**aximum **E**ntropy **M**arkov **M**odel
MeSH	**M**edical **S**ubject **H**eadings
MRF	**M**arkov **R**andom **F**ield
NCSPNOMESCO	**C**lassification of **S**urgical **P**rocedures
NERNamed	**E**ntity **R**ecognition

MC	**N**ew **Y**ork **P**atient **O**ccurence **R**eporting and **T**racking **S**ystem
NLPNatural	**L**anguage **P**rocessing
NPNoun	**P**hrase
ORG	**O**rganization
OTH	**O**ther
PN	**P**arenteral **N**utrition
PP	**P**repositional **P**hrase
POS	**P**art **O**f **S**peech
PRS	**P**erson
REK	**R**egional **E**tisk **K**omité
UMLS	**U**nified **M**edical **L**anguage **S**ystem
VP	**V**erb **P**hrase
WRK	**W**ork of art
WSM	**W**ord **S**pace **M**odel

Chapter 1

Introduction

1.1 The current study

The main purpose of this thesis was to investigate the potential of an automatic classifier on classification of words from clinical notes that deal with Central venous catheter (CVC) events. Recognizing events related to CVC can contribute to providing clinical decision support.

1.2 Problem formulation

This thesis is part of the Evicare project. The main objective of the Evicare project is "to develop methods and technology for providing "Evidence-Based Medicine" (EBM) at the point of care, integrated with an electronic health record (EHR) or other health information systems directly involved in the clinical process, resulting in higher quality of care and a more detailed, transparent documentation of care processes" [1].

Clinical records are the foundation of healthcare documentation and communication. According to the Act of 2 July 1999 relating to health personnel [2], all people with patient responsibility are required to document their work for the purposes of patient care, reporting and registration. The majority of information is in narrative form, roughly organized according to care provider (physician, nurse, physiotherapist etc.), phase of treatment and formal role of the note (discharge, order, prescription, referral etc.). The unstructured nature makes it challenging to retrieve and extract relevant information about a patient.

The purpose of this thesis is to:

1. Evaluate the utility of Named-entity recognition for automatic annotation of Norwegian clinical notes mentioning events related to central venous catheterization (CVC).
2. Find factors that may contribute to classify CVC events
3. Provide an overview of similar, previous studies.

1.3 Motivation

Central venous catheter patients are dependent on a CVC for different medical purposes [3, 4]. According to our collaborators at Ahus there is a lack of knowledge about the prevalence and duration of CVC for Norwegian patients. Identifying the number of patient days with CVC of patients that probably or certainly have had CVC inserted is important information to the hospital. Sub-goals in achieving this is to identify starting and end dates, CVC days and days where it seems that CVCs are not present.

A possible first step to achieve information about the number of CVC days is to automatically annotate phrases of clinical notes that indicate certain events. The events may include CVC insertion and CVC removal. Identifying this information can contribute to give a faster and more accessible overview of the occurrence and duration of CVC usage. This can help monitoring the number of CVC related bloodstream infections. Also annotating placement of CVC, care of CVC and equipment that have been used can contribute to performing risk evaluations.

Increased amounts of electronically stored data and the need to efficiently extract information from these data have made researchers look in the direction of natural language processing (NLP) to solve challenges within the healthcare sector [5]. Using NER for automatic annotation of Norwegian clinical notes related to CVC has not previously been undertaken. Automatic annotation of CVC events can be utilized to provide clinical decision support by using it as an input for classifying clinical notes as CVC positive or negative, as keywords for searching clinical guidelines and literature, to index records and for tagging of guidelines. In addition, it can be used to reveal overlaps and inconsistencies in guidelines, and thus improve decisions. This project will be part of a larger pipeline providing decision support for CVC. CVC events are annotated because they may help identifying critical patients. CVC patients have an increased risk of blood infection (sepsis) that may be life-threatening without treatment. Previous studies have investigated semi-automated methods for detecting CVC related events in records and these concluded a need for further work on the theme [6].

Chapter 2

Medical Background

Electronic records are in the process of replacing paper records. Scandinavian countries have been leaders in the implementation of EHRs [7]. In Norway almost all the hospitals and general practitioners (GPs) use an EHR system on a daily basis [8]. The usage is presently estimated to be more than 90% [9, 10] among hospitals and GPs. This advance has occurred rapidly. In a study from 2001, none of the largest hospitals in Norway had completed implementing EHR systems [11]. This chapter contains medical aspects regarding CVC and associated challenges, as well as information about the structure of medical records.

2.1 Central venous catheter

CVC consists a tube inserted into one of the central veins of a patient in order to provide nutrition, medication, other fluids or for measuring central venous pressure [3, 4].

How long a patient is in need of a CVC varies from a couple of days to permanent use. Catheter-related bloodstream infection (CRBSI) is a blood infection caused by bacteria from a catheter. It is one of the most severe complications of CVC usage and also the most common reason for hospital acquired sepsis. The risk is low the first days of CVC usage, but increases with increasing number of CVC usage days. The surface of the CVC becomes covered by plasma proteins and bacteria after insertion. It is important to keep the amount of bacteria below a certain threshold to prevent CRBSI. CRBSI may cause different symptoms and complications such as fever, septic shock, organ failure and in worst case death [12].

CRBSI is often diagnosed by the presence of symptoms, growth of bacteria from the catheter and positive blood cultures which discloses the presence of bacteria in the blood above a certain threshold. CRBSI or suspected CRBSI most often causes removal of the CVC. In some cases, if an insertion cite for a new CVC is unavailable, medical treatment might be induced. Removal of the CVC may be costly, painful and cause complications for the patient. Thus, efforts are made to prevent removal of the CVC [12].

In order to prevent CRBSI strict guidelines are followed. These guidelines encompass selection of catheter type, insertion site, hygiene and aseptic technique, care and replacement of catheter, but also a number of other matters. Other efforts to prevent CRBSI is to reduce the access to the catheter to a minimum and to limit the amount of persons nursing the patient [3, 12].

2.1.1 Patients dependent of CVC for parenteral nutrition

The results of multiple studies have indicated that CVC related bloodstream infections represents one of the most common and severe complications in patients receiving long-term parenteral nutrition (PN). Parenteral nutrition (PN) is often necessary in cases of intestinal failure. For some patients parenteral nutrition is necessary for a short period of time, while other patients need parenteral nutrition permanently and thus receive home parenteral nutrition (HPN) [3].

In a retrospective study by Hojsak et al. [3], the occurrence of sepsis was recorded over a period of 21 years among children (n=62) getting long-term parenteral, hospital or home parenteral nutrition. Catheter related sepsis were discovered through positive blood culture samples of patients with suspected sepsis. CVC was removed when the parenteral nutrition was terminated and replaced in cases of sepsis, occlusion, accidental removal, local infection and death. The results showed that there were 1.7 septic episodes / 1000 days of parenteral nutrition and 0,93 deaths / 10 000 days. 12.8% of the CVCs were removed because of septic episodes. For hospital parenteral nutrition the average number of PN days for each patient was 149,7 whereas this number was 1415.3 days for HPN.

Of the CVC related sepsis episodes, 33.3% occurred in home parenteral nutrition. The occurrence of sepsis was significantly lower at home (0.94 / 1000 days of HPN) than in the hospital (2.75 / 1000 days of PN).

The rate of sepsis and deaths in the study was considered to be very low compared to other studies. Some of the explanation for their low rate of sepsis was that guidelines were strictly followed and documentation was implemented for all long-term CVCs. Also, the hospital under study had the largest pediatric PN program in Croatia [3].

2.2 Clinical text

In an EHR all texts are clinical. Clinical texts may be challenging because they often contain short phrases that are not necessarily grammatically correct [5]. Accordingly, clinical texts may differ from those used in other domains. They are often written directly as free text in narrative form [13] or transcribed or dictated [5].

Medical records consist both of structured and unstructured fields. Structured fields are easier to understand for a computer since these are well defined and often consist of a predefined set of possible choices. However, valuable information is often documented in unstructured fields and in fact approximately 40% of a medical record consist of unstructured information [14]. Clinical notes contain many domain specific terms, they are heterogeneous and often do not conform to regular grammar and abbreviations. Spelling mistakes are also common [15]. Several synonyms for diagnoses and drugs exist, as well as Latin expressions. Different pre-processing steps can be taken to increase a program's ability to understand the text. Dictionaries and standardized classification systems are helpful in this regard. Extracting information from clinical texts often includes naming entities and mapping these to vocabularies [15] - named entity recognition.

Chapter 3

Technical Background

This chapter covers natural language processing - including pre-processing of text, annotation of text (Part-of-speech tagging and named entity recognition), dictionaries and lexical resources used for matching medical terms, medical annotated corpora used for training and for making gold standards, machine learning methods and evaluation measures used to evaluate NLP systems.

The next sections will introduce topics relevant to NER systems.

3.1 Natural language processing

Natural language processing (NLP) is the process of making computers able to understand and manipulate natural languages [5]. This can be useful in the clinical domain where both documentation, guidelines and clinical descriptions are represented by a free text format, with various degree of grammatical formality and content [16]. The use of EHRs has increased, which contributes to make it easier to use natural language processing for information extraction [5].

Several tools for doing NLP exist, and often combinations of tools are applied. Common NLP steps for clinical notes are described by Jensen [15] as:

- Boundary detection - splitting text into sentences
- Tokenization - splitting sentences into meaningful units (often words)
- Normalization - converting words to base forms
- Part-of-speech tagging - tagging tokens with a grammatical information such as whether a word is a verb or a noun
- Shallow parsing - identifying syntactical units such as noun phrases
- Entity recognition - recognizing entities, mapping these to vocabularies and identify whether the entity is negated or not

3.1.1 Information Retrieval

Natural language processing can be used for information retrieval. Information retrieval may refer to retrieval of different types of information. Normally, it refers to the process of retrieving text documents from a database that are relevant to some search queries [17]. An example is a Google search hit. Different algorithms are applied in information retrieval. One of the most successfully applied is the vector space model (VSM) described below.

3.1.1.1 The Vector Space Model

The VSM was proposed by Salton, Wong and Yang [18]. It represents each document, D_i, as a vector in a document space. The document vectors consists of term weights, d_i which are in most cases weights of the words in a document. A document vector representation is shown in 3.1. The equations regarding VSM are taken from [18].

$$D_i = (d_{i1}, d_{i2}, d_{it}....) \tag{3.1}$$

Each vector consists of a set of index terms that have different weights according to their importance. The number of dimensions of the vectors depend on the number of terms in the documents. The index terms may be boolean or the terms may be weighted. Documents that are high in similarity are close to each other in the document space. The similarity between documents can be found by the inverse function of the angle between two document vectors. If the angle is zero, the documents are equal. Thus, documents represented by vectors far from each other in the vector space have a low similarity. The distribution of the vectors in the document space depends on which terms the document contains and how these are weighted. The vectors lengths may be normalized to one so that only the positioning of the vectors is kept and each of the vectors may be represented as a dot. Thus, dots clustered together represent documents similar to each other. The similarity between documents can be calculated by equation 3.2.

$$F = \sum_{i=1}^{n} \sum_{j=1}^{n} s(D_i, D_j), i \neq j \tag{3.2}$$

The purpose of this sort of indexing is to be able to return a relevant set of documents when a user query is posed. However, without knowledge about which search queries that may be posed, it is difficult to index the documents so that only relevant documents to a search query is returned. Another option is to minimize the similarity function between the documents so that the distance between document vectors is as large as possible. This increases precision because it decreases the risk of retrieving non-relevant documents to a search query. If more than one document should be retrieved, this can be obtained by discarding non-relevant documents. This increases recall. Minimizing the similarity function is however not considered a good solution because it requires n^2 vector comparisons. Salton, Wong and Yang, [18], found that the best solution is a clustered vector space where each cluster has a centroid. A centroid represents all of the documents in one cluster, c_j. It is calculated as seen in 3.3.

$$c_j = (1/m) \sum_{i=1}^{m} d_i j \tag{3.3}$$

Similarly, a centroid for the complete vector space may be calculated from the cluster centroids. Comparing document vectors to the main centroid can then reduce the complexity of the computation of density in the document space to n, see equation 3.4.

$$F = \sum_{i=1}^{n} s(C^*, D_i), \tag{3.4}$$

C^* represents the main centroid. The different clusters ideally contain similar documents. An example is that it is likely that documents containing information about CVC are clustered together, as are documents containing information about breast cancer. Documents containing information about CVC are automatically clustered together because they contain many of the same index terms / words, making their document vectors similar. As described above, similarity between documents can be maximized, increasing recall or minimized, increasing precision. In a clustered vector space with centroids, these properties can be combined. Minimizing similarity between centroids make the different clusters distant from each other, increasing the risk of returning documents only from the relevant category. At the same time, recall can be maintained by minimizing the similarity between documents inside a specific cluster. Space density is affected by weighting scheme [18].

The study by Salton et al., [18], tried to find which weighting scheme that returned the optimal retrieval. Term frequency weighting is based on the frequency of a term in a document. Terms frequently occurring in a document are weighted more than terms seldom occurring. However, words occurring often in a document may be common words that occurs often in almost all of the documents. To improve performance, Salton et al. found that a promising weighting scheme was to multiply the term frequency with the inverse document frequency. The document frequency is a measure of how many documents that contains a term, t. The inverse document frequency can be found by the equation in 3.5. This is called tf-idf weighting.

$$(IDF)_k = (log_2 n) - (log_2 d_k) + 1 \tag{3.5}$$

The IDF is relevant because it gives information on whether a term is rare or common across all documents. A word that occurs in many documents may be irrelevant to a search query. An example is the word "‘it"’ or "‘was"’. The term frequency ensures that terms occurring frequently in individual documents are weighted more. The tf-idf weighting scheme was tested on two different cluster organizations varying in overlap. Comparing tf-weighting to tf-idf weighting, it was found that including IDF decreased density in the document space and improved retrieval performance. The average improvement was about 14 %. They also tested whether increased density would result in decreased performance by applying the df instead of the idf and therefore weighting more the terms that occurred in many documents. This hypothesis could be confirmed as the retrieval performance decreased by about 10 %. The researchers also examined the effect of alternating the space density so that documents inside clusters became tighter together and clusters more separated from each other. This alteration produced the expected effect so that performance was further enhanced. These results supported the "'term discrimination"' model.

The term discrimination model weights terms according to their discriminative value. That is, to what extent an index term makes a collection of documents more equal or more different from each other. To figure whether a term is a good discriminator equation 3.4 can be calculated with and without the term included in all documents. If the document space density is decreased when the term is included, it is a good discriminative term. The researchers investigated discriminative scores of terms in a collection of 450 medical documents. The documents contained 4726 terms, and these were ranked from 1 to 4726 according to their discriminative value, 1 being the most discriminative and 4726 being the least discriminative term.

The worst discriminators had a score of more than 4000 and were found for terms with a document frequency higher than 25. Terms with a very low document frequency (1-3) also had a poor discriminative score, 3000. The best discriminative scores were found for terms with a document frequency between n/100 and n/10 for n documents. If these results can be generalized, it has been concluded that a good indexing strategy would be that terms with moderate document frequency should be used for content identification directly, terms with a high document frequency could be converted to terms of lower document frequency by using them as components of indexing phrases, terms with a low document frequency could be converted to terms of higher document frequency by finding a more general term for several low frequency terms with a similar meaning. These transformations were tested on three different textual collections, one of which was a medical collection. The overall improvement in precision and recall was found to be from 18 to 50 percent, where 50 percent was for the medical collection.

3.1.1.2 Word space models

A review by Sahlgren, [19], describes word space models (WSM) as follows; The word space model is an implementation of the vector space model that also takes into account semantic similarity between words. Words that often occur close to the same words or in the same context are assumed to be similar to each other. Thus, words are represented as context vectors and words that often occur in the same context produce similar context vectors. It is most often implemented as a co-occurrence matrix where the rows corresponds to words and the columns corresponds to context, which is most often either documents or word segments.

Latent semantic indexing (LSA) is such an implementation, where the context corresponds to documents. A cell in such a matrix represents the number of times the word in row_i co-occur in the document or word segment in col_i. Thus, the columns correspond to context and the number of columns correspond to the number of dimensions of the word space. The frequencies of co-occurrences are often weighted to avoid common words to receive a too high weight and to account for different document sizes. Word space models have the advantages that they make semantic similarity measurable and make it possible to look at semantic similarity in mathematical terms. However, word space models pose some problems; the number of dimensions of the context vectors increases with increased size of a textual collection since the vector dimensions correspond to the number of documents or word segments in the collection. Also, since many words only co-occur in a small set of documents, all the documents where the word does not co-occur will have blank cells. Having a large matrix with a lot of blank cells is inefficient. Techniques for dimension reduction exist, but disadvantages of these are that the complete co-occurrence matrix has to be made before the reduction can take place, and every time new data have to be added, the complete co-occurrence matrix has to be reconstructed, which is costly both in terms of execution time and memory usage.

3.1.1.3 Random Indexing

Random Indexing (RI) is a model developed by Kanerva, Kristoferson & Holst, [20], which aims at solving the problem of large co-occurrence matrices in WSM. In RI each document is assigned a unique, high-dimensional random index vector consisting of a few -1 and 1 and the rest 0. Each time a word is found in one of the documents in the collection, the index vector corresponding to that document is added to the word's context vector in the co-occurrence matrix. Thus, the value of the elements in the context vector is edited. The dimension of the context vectors is set, so the co-occurrence matrix does not increase as the number of documents in the collection increases.

The co-occurrence matrix made by random indexing is an approximation of the original co-occurrence matrix so that the differences between context vectors will be the same as for the original co-occurence matrix. Other advantages of Random indexing is that the co-occurrence matrix does not have to be remade when new data are added, and that it can be applied to any context; also to other materials than words and documents. Random indexing is faster than doing LSA followed by dimension reduction. RI has been proved effective in several empirical studies [19, 20].

3.1.1.4 Combining Random Indexing and Random Permutation

A study by Sahlgren, Holst & Kanerva [21] showed that it is possible to capture word order information in word spaces by combining random Index vectors and permutations of vector coordinates. A recent study by Henriksson, Moen, Skeppstedt, Eklund, Daudaravicius & Hassel [22] shows how the combination of RI models and random permutation (RP) models enhances synonym extraction in clinical notes. They measured the ability to find synonym pairs, the ability to find abbreviations from the extended form and the ability to find the extended form from an abbreviation. The best results were obtained when RI and RP were combined.

3.1.2 Information Extraction

Information extraction (IE) is a way of extracting and structuring relevant information from a text. The information that is considered relevant depends on the purpose and domain [23, 24]. Information extraction aims to extract predefined specific types of information. It differs from information retrieval which retrieves documents and text mining, the purpose of which is generating new knowledge. Automatic information extraction requires either a rule-based system or the use of machine learning approaches [5].

Identifying features such as negations, temporality and events are important when working with information extraction. Pattern matching, rule based methods, statistics and machine learning are common techniques used to extract relevant features from text. After identifying such features, terminologies can be applied to help classifying tokens into different categories [5].

A review by Meystre et al. [5] has summarized results from studies on IE from EHRs from 1995-2008. They describe the relationships between IR, IE, and text mining as follows; IR involves retrieving documents, whereas information extraction is about extracting information. Text mining is about generating knowledge from text and often requires several steps, including IR and IE. After IR and IE, data mining can be applied. This entails finding relationships among pieces of information [5].

Increased amounts of data, a stronger focus on quality of care, reduction of errors and use of electronic records instead of paper records motivates to using information extraction in EHRs [5].

3.2 Medical language processing

3.3 Pre-processing

Pre-processing in the form of tokenization, spell checking, document structure analysis, sentence splitting, word disambiguation, part-of-speech (POS) tagging and parsing are common steps when working with NLP and IE.

3.3.1 Tokenization

This step consists of partitioning sequences of input, such as text, into meaningful subunits, such as words [25].

3.3.2 Lemmatization

To normalize tokens, lemmatization and stemming are common techniques. Lemmatization and stemming both aim to find a base form for different inflections of words. Having a base form for inflected words that have the same underlying meaning facilitates differentiation of words. Lemmatization often use vocabularies to find the citation form of words [17, 26, 27].

3.3.3 Stemming

The difference between stemming and lemmatization is that when stemming is applied, a simple heuristic is used to make the base form of words. Ends of words are chopped off to make a common base form rather than applying vocabularies to find the base form [17]

3.4 Part-of-speech tagging

Part-of-speech (POS) tagging is an important sub task in NLP systems where tokens are assigned tags from a predefined tag-set. The tags are often word classes such as verbs, nouns and adjectives. The tags give linguistic information about a token or word and its function in a sentence. Once a text has been tagged with word classes, these can be used for further NLP, such as phrase chunking, named entity recognition (NER) and parsing [24, 28].

3.4.1 Norwegian POS-taggers - The Oslo-Bergen tagger

The Oslo-Bergen tagger is a morphological and syntactical tagger developed for bokmål and nynorsk (the two Norwegian languages). It consists of a pre-processing module with a multi tagger and a compound analyser, a grammar module for syntactical and morphological disambiguation and a statistical module that removes remaining morphological ambiguities. The pre-processing module contains functionality for finding sentence limits and it supports several types of grammatical tags. The Oslo-Bergen tagger uses norsk ordbank for multitagging. Norsk ordbank consists of words both in conjugated and dictionary form of words from large dictionaries in nynorsk and bokmål. [29]

3.5 Named Entity Recognition

Named entity recognition (NER) is to some extent similar to part-of-speech tagging. It is the process of recognizing and categorizing expressions into some predefined classes. A named entity is a token or expression categorized into a set of defined categories in a manner that benefits a particular purpose. What categories that are relevant depends on the purpose of the information extraction. One example of NER is to define categories such as drug, diagnose and symptom and categorize tokens and phrases accordingly. An example of a named entity category applied in the current study is CareCVC, denoting all phrases referring to care of CVC:

”‘ <CareCVC >CVC care <CareCVC >was performed at three.”‘

NER was first coined by the message understanding conference (MUC) number 6. MUC started as a conference to foster research on automated analysis of military texts. One of the main goals of MUC-6 was to find information extraction functions that were of practical use, domain independent and automated. Named entity recognition was therefore defined as a task involving “identifying the names of all the people, organizations and geographic locations in a text” [30].

An example category used at the MUC was the tag ENAMEX for entity name expression. An example tag used for NER is < ENAMEX TYPE=”PERSON” > Jim < /ENAMEX >. For numerical expressions, NUMEX was applied as tag. [30].

The NER applied in the MUC-6 was on test sets from the Wall Street Journal, and the results were very good, over 90 percent recall and precision.

Word-by-word sequence labelling is a common approach of how to do NER. Words that fulfil the requirements of entity tags are labelled sequentially. IOB encoding is used to indicate the boundaries of an entity chunk. B is used for the beginning token, while I is used for tokens inside a chunk. O is used for tokens outside the entity.

3.5.1 Features

Features are applied for recognition of entities. Different features can be applied to find and classify Named Entities (NE). This commonly includes word-level features. Common features are lexical items, stemmed lexical items, shape, character affixes, parts of speech, syntactic chunk labels, list look-up features (gazetteers), predictive tokens and bags of words/bag of n-grams.

Surrounding or predictive words and n-grams can also be applied as features since context is useful for recognising entities. Shape features are the orthographic pattern of a word such as whether the word is capitalized, lower cased, mixed case or written in caps. When applying list look-up features, stemming and lemmatization are often used in addition in order to include inflected words as matches to the dictionary. Edit-distance/fuzzy-matching can also be used to decide if two similar words should be counted as matches. Document and corpus features such as multiple occurrences of words and meta-information in a document can also be used. When a set of features has been selected, it is common to use these for annotating a corpus that can be used as a training set for a classifier [24, 31].

3.5.2 Relationships between entities

Another part of the NLP process can be to assign, or find, relations between entities. An example of such a relation could be ”‘part of”’ to indicate that a blood test is part of an examination. Supervised learning can be applied to detect such relations. Annotations on training sets are used to detect new relations. Different algorithms to detect new relations exist. The simplest one consists of two steps; Step one is to figure out if a relation exists between an entity pair. Step two is to label the relation. Techniques for detecting relations are Decision trees, Naive Bayes and Maximum entropy. Disease-treatment relations have been specifically studied, and hidden Markov models and discriminative neural network models have been applied successfully [24]

When NER has been performed, two types of ambiguities arise. Two equal words labelled with the same entity label may refer to two different entities of the same type. An example is two persons with the same name. The other ambiguity that may arise is when a word may refer to two different entity types. An example is that a word may be the same for a location and an organization [24].

Sekine [31] surveyed research on NER from 1991 to 2006. The survey indicated that the first NER systems used handcrafted rule-based algorithms to recognize entities, while for newer systems, supervised learning is most commonly used. One drawback of supervised learning is that it requires an annotated corpus for training.

Unsupervised learning applies unannotated data and performs NER based on clustering, lexical resources, lexical patterns and statistics. [31].

A combination of supervised and unsupervised learning can also be applied. This is called semi-supervised learning because it is partly supervised using bootstrapping, meaning that the application of some seeds start the learning process. The seeds can be lexical features, patterns, seed entity examples, syntactic relationships and existing NER systems. According to Sekine Semi-supervised learning seems to be equally effective as supervised learning in some cases [31].

The survey by Sekine [31] indicated that a great amount of research on NER exists in different languages and with different entity types, but the research on different genres and domains is limited. NER has been successful in the domain of molecular biology and specifically for detecting genes.

Making NER general has been an unsuccessful approach since NER made for one domain produces poor results in another domain. Thus, making a domain specific NER for clinical texts, taking into consideration medical knowledge is an important task in the domain of healthcare informatics.

3.6 Dictionaries and lexical resources

Dictionaries and lexical resources are relevant as these can be applied as features that contribute to classification of phrases.

3.6.1 Wordnet

Wordnet is a dictionary in English. Words are grouped on meaning and synonyms are clustered. Also relationships between concepts are expressed at wordnet. In addition words in Wordnet are grouped according to their part-of-speech so that adjectives are grouped together and so on.

3.6.2 ICD-10

ICD-10 is an international coding system for classifying diseases. The classification system can be used for statistical purposes. It was originally used as a registration system used to register causes of death and was expanded later to include diseases and other health problems [32]. In ICD-10, diseases are named and encoded. Each disease contains a name, an ICD-10 code and relationships to other diseases. Diseases are classified into seventeen different classes of diseases [33]. ICD-10 codes contain a letter that describes the class of diseases and a number that describes its sub-category more specifically, and then a second letter that indicates the specific disease [33]. The classification system is managed by the WHO [34]. Contrary to many other lexical resources, ICD-10 has been translated into Norwegian and is applied daily in Norwegian hospitals. It is used by the Norwegian Central Bureau of Statistics. The Norwegian version is available online [35].

3.6.3 SNOMED CT

SNOMED CT is a standardized terminology containing medical concepts organized as a hierarchical way. Each concept is assigned a semantic class and a name. SNOMED CT has been translated to Swedish, but the Swedish version does not contain synonyms for concepts.

3.6.4 MeSH

Medical Subject Headings (MeSH) is a vocabulary of medical concepts developed by the National Library of Medicine. MeSH started at the National Library of Medicine in the USA and has now been translated into most European languages. One of its main purposes is to be used to index medical literature [36]. Synonymous concepts are given a common id number and one common concept term. The vocabulary is often used for indexing of literature databases such as PubMed and SveMed+ since the MeSH terms ensures searches for synonyms, thus provides better retrieval. The MeSH also contains hierarchies and relationships between concepts, and can therefore also be considered an ontology [37]

MeSH facilitates communication between medical and non-medical personnel. MeSH has also contributed both to better indexing and retrieval of literature as educational institutions apply MeSH terms when registering books and literature in their databases and also students and health personnel are trained in the use of MeSH terms for searches. Norwegian educational institutions have benefited from the Swedish MeSH as the Swedish terms are often more similar to Norwegian than the English ones. However, the use of the Swedish MeSH for Norwegian educational purposes suggests a need for a Norwegian translation of the MeSH. Thus, in 2010 the Norwegian Health library, Helsebiblioteket, started a Norwegian translation of the MeSH. Its first release is planned to be completed in 2013. The Norwegian MeSH has already been integrated into the European Health terminology/Ontology Portal (EHTOP) and in the Swedish literature database SveMed+ [37]. The Norwegian version of MeSH will be available through SveMed+ advanced search from January 2013. Svemed+ is a Swedish database, but since it is also extensively used by Norwegians, 17 000 Norwegian MeSH terms have been included in the advanced search version of the database.

Another advantage of the MeSH is the possibility of translating terms into different languages. MeSH also has the possibility of being linked with other encoding systems such as the ICD-10 [37].

3.7 Machine learning techniques and classifiers

Machine learning techniques are useful to detect statistical patterns in data and to predict new data based on previous ones. Machine learning techniques are either supervised or unsupervised. Supervised learning are based on labelled data whereas unsupervised learning are based on unlabelled data. That means that unsupervised learning tries to detect patterns in data without any predefined properties to look for [15].

3.7.1 Supervised learning techniques:

Supervised machine learning techniques are based on a data set for training the algorithm. The data set is labelled with a set of features and the algorithm is trained to label entities with the given feature set. Feature vectors are often applied so that each token can be assigned a set of features that contribute to the NER [15].

Some common supervised learning algorithms are Naive Bayes; Artificial Neural Networks; Support Vector Machines and random forests [15].

Advantages of supervised learning is its robustness to random errors in features or labels when applied on large data sets and that a model can be reused for new data sets. However, it is poor in recognizing systematic biases in the data and it is prone to over-fitting.

3.7.2 Statistical vs. deterministic models

Deterministic models make models of observations by using known properties of observations. Once the known properties are found the model can be defined. Given an input, the deterministic model will always produce the same output. Statistical models on the other hand are constructed from statistical variables [38].

3.7.3 Generative models

Generative models typically maximize the joint probability of observation and label sequences when training models for predictions. They are also typically poor at accounting for multiple interacting features and dependencies of observations. Observations are assumed to be independent of each other [39].

3.7.3.1 Markov Models

MM are used to predict states given observable states. Given observable events, a matrix for state transitions can be made. An example is whether the weather is sunny, cloudy or rainy. Based on observations of these states, we can make a model to predict states in the future. In a Markov model for this example, the prediction of a next weather state would be based on the previous state and transition probabilities. Given a sunny day, the transitions to the next possible states, sunny, cloudy and rainy have known probabilities. So, to find a next state one can look solely on what the current state is and the probability of transitions [40].

3.7.3.2 Markov chains

A Markov chain (MC) is a chain consisting of a fixed number of states of a process. The process starts in one state and proceeds from state to state. Each state, s_i has a set of possible next states. The transition from a state, s_i to each possible next state has a transition probability. The transition probability, from state s_i to state s_{i+1} is independent of the previous states s_{i-1}, s_{i-2}, ..., s_{i-n}. The next state only depends on the current state. A probability distribution for start states is also found to define a start state for the process [41].

Given a MC with transition probabilities, a transition matrix can be made for the chain. Each element of the transition matrix represents the probability of going from one state to another state. The possible states of a process is given as rows, and for each row there are columns of possible next states. Thus, a future state depends on the current state, but none of the other states. To find the probability of a state that is several steps into the future, one can summarize the conditional probabilities. So, state s_{i+1} depends on state s_i and state s_{i+2} depends on state s_{i+1}. This gives two conditional probabilities; $P(s_{i+1} \mid s_i)$ and $P(s_{i+2} \mid s_{i+1})$. By summarizing these two conditional probabilities it is possible to find the probability of state s_{i+2} after two steps [41].

The dependencies in a Markov chain form a linear structure [42].

3.7.3.3 Hidden Markov Models

HMM are stochastic models of observations. When HMM is used to find correct labels of words, the joint probability of paired observation and label sequences is maximized. Thus, the basis for HMM is the equation for joint probability, (3.6). In the equation, s refers to states or labels and x refers to observations or words. In a first order HMM two simple assumptions are made;

1. a state s_i depends only on the previous state s_{i-1}
2. an observation, x_i depends solely on the current state, s_i ...

The first assumption is the Markov assumption. The first order Markov assumption is that a state of a process is conditioned only on the previous state, s_{i-1}. For the second order Markov assumption, the two previous states are taken into account, s_{i-1} and s_{i-2}. This way, the Markov assumption can be extended to higher-order Markov assumptions. [42, 43].

Given these assumptions, the equation (3.6) can be rewritten to (3.7). Second order, third order and higher order HMM are also possible, meaning that the number of previous states that a state s_i is dependent on is increased with the order number. See equation (3.8) for an example of the formula for a second order HMM [43]. The equations below are taken from Ponomareva et al. [43].

$$P(s,x) = P(x|s)P(s) \tag{3.6}$$

$$P(s,x) = \prod_{i=1}^{n} P(x_i|s_i)P(s_i|s_i - 1) \tag{3.7}$$

$$P(s,x) = \prod_{i=1}^{n} P(x_i|s_i)P(s_i|s_i - 1, s_i - 2) \tag{3.8}$$

HMM assumes that observations are independent of context, an assumption that does not hold true for sequences of words [43].

State machines where each state is hidden, but the transitions between states are visible can be applied to describe HMM. An example is given in figure 3.1 For HMM the output of each state is visible, but states are hidden [40].

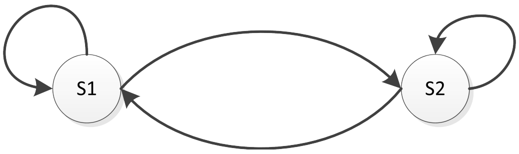

Figure 3.1: A HMM consists of a number of states, S1...SN and transitions between states. The process of the states is hidden, whereas the output of each state is known.

A HMM consists of

1. A number of states, N 2. The number of distinct outputs at each state, M 3. The probability of a transition from one state to another. 4. The probability distribution for the distinct outputs at each state. 5. The probability of starting at a state.

[40]

To make a HMM the elements above have to be specified. Then, the HMM can be applied to generate a sequence of observations and also to explain how a sequence of observations was generated [40].

A HMM has to take into account the following problems:

1. Finding a way to estimate how well a model explains a sequence of observations. 2. Finding the state sequence that best explains the observations. This means trying to find the sequence of the hidden states. Some optimality criterion often based on the intended use of the model is applied to find this. 3. Finding a way to adjust the elements of the model to maximize the probability of an observation sequence given the model. Elements of the model are adjusted based on training sequences to the model. Thus a model can be adjusted for observations from a specific domain. [40]

3.7.4 Conditional models

Conditional models predict sequences of labels given sequences of observations. An advantage of conditional models is that they do not assume independence of observations, they can handle correlated features and allow features of observations at different levels of granularity. Past and future observations can also be accounted for [39].

3.7.4.1 Maximum Entropy Markov Models

Maximum Entropy Markov Models (MEMM) are conditional. For MEMM an exponential model is applied to find sequences of labels given sequences of observations. A disadvantage of MEMM is the Label bias problem. [39].

3.7.4.2 Conditional Random Fields

Conditional random fields predict sequences of states/labels given sequences of observations, $P(y|x)$ [39, 43]. CRFs share several properties of HMM. The main difference between CRF and HMM is that CRF maximizes a conditional probability whereas HMM maximizes a joint probability [43]. Other differences are that CRF is discriminative, undirected and allows non-probabilistic sub models [42]. CRFs are based on Markov random fields (MRFs). MRFs are similar to MC, but in contrast to MC, the dependencies of MRF do not form a linear structure. The dependencies of MRF can have any structure and be represented by an undirected graph. Majoros [42] provides a list of the components of MRF:

1. an alphabet of possible labels
2. a set of variables that can be assigned values from the alphabet
3. a probability distribution for assignment of labels to variables, P_{M}
4. an undirected graph representing dependencies between variables ...

For the assignment of labels to variables, P_{M}, the dependency graph given in point four must always hold true. Each variable depends on its direct neighbours, except itself [42]. This is similar to MC where predictions of next variables depend only on the current variable.

The Hammersley-Clifford theorem is useful to MRF. It states that the likelihood of an assignment, x, of a label to a variable X (under model M) is given by

$$P_M(x) = \frac{1}{Z} e^Q(x) \tag{3.9}$$

The equation above, 3.9, as well as the equations 3.10 and 3.11 are taken from Majoros, [42].

The formula presumes that no assignment has a probability less than zero. In the formula, Z is a normalization factor that can be written as:

$$\sum_{x'} e^{Q}(x') \tag{3.10}$$

‘Q(x) can be expanded to:

$$\begin{aligned}Q(x_0, x_1 ... x_{n-1}) = \sum_{0<=i<n} x_i \Phi_i(x_i) + \sum_{0<=i<j<n} x_i x_j \Phi_{i,j}(x_i, x_j) + ... \\ +x_0 x_1 ... x_{n-1} \Phi_{0,1...,n-1}(x_0, x_1, ..., x_{n-1})\end{aligned} \tag{3.11}$$

The Φ functions are called potential functions and represent cliques of the undirected dependency graph of MRF. A clique is either a singleton (one vertex) or a sub graph where an edge exists between all vertices in the graph. Φ functions that do not represent a clique are set to zero. Since a clique can be any sub graph, overlapping cliques may occur. The Hammersley-Clifford theorem makes it possible to calculate probabilities based on Φ functions. This means that training a MRF can be subdivided into training on potential functions [42].

CRF extends MRF. CRF contains both observable and unobservable variables and its formula is based on MRF and the Hammersley-Clifford theorem. Like MRF, CRF consists of an alphabet of labels, a set of variables, potential functions and an undirected graph describing dependencies between variables. One of the main differences is that CRF contains both observable and unobservable variables;

”‘ A CRF may be described as a MRF plus a set of “external” (observable) variables X, which are not considered variables of the MRF but are globally visible (as fixed constants) to the MRF’s potential functions, Φ_c [42]. ”‘

Since the observable variables of the CRF are external to the MRF, only the cliques of the unobservable variables (u-cliques) are used in the Hammersley-Clifford theorem. It is assumed that observable variables are given.

CRF is conditional because only the cliques of the unobservable variables are applied in the Hammersley-Clifford potential functions, when computing $P(y|x)$, whereas the observable variables are assumed given.

”‘Since the observables X are fixed, the conditional probability $P(y|x)$ of the unobservable given the observables is:

$$P_M(y|x) = \frac{1}{Z(x)} e^{Q(y,x)} = \frac{1}{\sum_{y'} e^{Q(y',x)}} e^{Q(y,x)} \tag{3.12}$$

where Q(y,x) is evaluated via the potential functions—one per u-clique in the dependency graph:

$$Q(y, x) = \sum_{c \in C} \Phi_c(y_c, x) \tag{3.13}$$

"‘ In the formula above, the $\Phi_c(y_c, x)$ is a function of a u-clique. X is included because the X's can be considered constants and the $y_c's$ are vertices included in the clique. Since the X's are included in the formula, the CRF model is not available until after the X's are inputted. This is in contrast to generative models where the model is available prior to input."’ [42]

Simplifications can be made to the CRF formula.

"‘In practice the potential functions Φ_c are very often decomposed into a weighted sum of “feature sensors” f_k , producing:

$$P(y|x) = \frac{1}{Z} e^{\sum_{c \in C} \sum_{i \in F} \lambda_i f_i(c,x)} \tag{3.14}$$

where F is a family of feature sensors, which are specific to individual cliques [42]."‘

3.7.5 CRF vs. SVM

A study by Li et al. [44] compared CRF and SVM for NER in clinical notes. They used a standardized set of named entities so that the two machine learning methods could be compared. They found that CRF gave the best results with an average f-measure=0,86 compared to SVM which gave an average f-measure of 0,64.

3.7.6 Brat rapid annotation tool

Different tools assisting the process of making annotated corpora exist. One of them is the Brat rapid annotation tool [45] that was applied in the current study. The Brat rapid annotation tool [45] provides a graphical interface that facilitates the annotation process. Phrases or entities from data files can easily be marked, and annotations attached to them. Once the annotation process is ended, Brat makes an annotation file that contains all annotations made for a specific file. Thus, each data file has a corresponding annotation file. Brat annotation files are based on the Standoff format, which is a way of specifying annotations. An annotation in the Standoff format consists of an annotation identifier such as "‘T1"’ for tag number one, an Offset start number - defining at which character number the annotation starts, an offset end number - defining where the annotation stops and the phrase that has been annotated. An example annotation is given below:

"‘T1 Carecvc 0 44 Care of central venous catheter was provided"’

3.8 Evaluation methods and measures

NER systems are often compared to gold standards. A gold standard corpus is a collection of texts that have been annotated by humans. The gold standard represents the desired output of the automatic classification system. Thus, it can be used to evaluate the performance of a classification system. For gold standard corpora it is common to calculate annotator agreement to ensure reliability of the corpus. Annotator agreement means that two annotators annotate the corpus equally according to decided classification classes. If the differences in annotations are large, the reliability is low.

It is also common to make a baseline class, a random assignment, that can be used to compare with the results of other classification classes. [46].

When evaluating NER systems, precision/positive predictive value, recall/sensitivity and F-score are commonly reported. Relevant to these measures are

- true positives (TP)
- false positives (FP)
- true negatives (TN)
- false negatives (FN)

Entities labelled with a given class/label by the classification system are called positives. These can be either true or false. True positives are entities that were correctly assigned the given class, meaning that the classifier imparted the same class as the gold standard. False positives are entities that were assigned to the given class even though this assignment was not in accordance with the gold standard. Negatives are the entities not assigned to a given class. These can also be true or false. True negatives are entities that are predicted not to belong to a class both by the classification system and by the gold standard. False negatives on the other hand are entities that were predicted to not belong to a class by the classification system, but which according to the gold standard, should have been assigned the given class [47] .

Recall/Sensitivity (R) is the proportion of correctly classified instances of all positive instances, R=TP/(TP+FN) or R=true positives (TP) / real positives (RP). Precision/Confidence is the proportion of predicted positives (PP) that are true positives (TP), P=TP/(TP+FP) or P= TP / predicted positives (PP). Increasing recall will result in a decrease of precision. Therefore, the harmonic mean, F, which accounts for both by adding a weight for precision and recall, is used to measure overall performance. [31, 46–48].

Sekine [31] summarizes evaluation techniques applied for the largest NER conferences; MUC, IREX, CONLL and ACE. A common evaluation technique is to compare the performance of the NER system output to the result of a human linguist. Sekine et al. describes three different sort of evaluation techniques, MUC evaluation, Exact-match evaluations (IREX and CONLL) and ACE evaluations. MUC evaluation credits partially correct matches found by the classification algorithm. This is performed by evaluating on two axes, one for the correct type, such as "'PERSON"' and one for selecting the correct text and its text boundaries. Both precision and recall are accounted for when calculating the final MUC score, the harmonic mean of precision and recall. The advantage of this evaluation is that partial correct recognitions are credited and that all types of errors are included.

In contrast, IREX and CONLL require exact matches. These evaluation methods are too strict for some systems. ACE evaluation includes a weight for each entity type that contributes to a total maximum value. Costs for errors and missed entities are included. For the total evaluation, the contributions of the different entities are calculated and the costs of errors are subtracted. ACE is the most powerful evaluation method, but the drawback is that it is also the most complex one, which makes error analysis more difficult. Sekine [31] indicated that evaluation results are highly dependent on the choice of evaluation method.

3.8.1 Cross-validation

Cross-validation is a statistical method used to train and estimate the performance of a model. It is highly accepted and used for learning algorithms. Refaeilzadeh, Tang & Liu, [49], have made a literature survey of Cross-validation and its usages. In cross-validation the data are divided into training data that are used to train a model, and testing data that are used to test the model. It is called cross-validation because the training and test data will somehow cross over, being substituted by each other, in different test rounds. The simplest form of cross-validation is resubstitute validation where the same data are used both to train the model and to test the model. Quite obviously, this method easily leads to over-fitting. Over-fitting is comparable to giving a medical student all the exam questions in advance when the goal is to test his/her performance in medicine. The student is trained on the exam questions and thus performs well when tested on the exam questions. This may lead to the false assumption that the student's performance in medicine is very good in general. However, when he/she is given unseen questions, he/she performs poorly. When testing a trained algorithm the goal is to test how well the model would perform on unseen data. Thus, over-fitting should be avoided. [49]

Another simple form of cross-validation is hold-out cross-validation which is better than resubstitution cross-validation. The data are divided into two sets, one used for training and one for testing. The model is trained on the former and subsequently tested on the latter, unseen data set. Thus, over-fitting is prevented. However, a drawback with this method is that less data are available for training, since one set has to be reserved for testing. Another drawback is that the results are highly dependent on an even distribution of data within training and test sets. If the distribution of data is skewed, the results would be poor [49].

K-fold cross-validation solves the problems with hold-out cross-validation and is the most commonly used and accepted type of cross-validation to evaluate and compare learning algorithms. In K-fold cross-validation, the data are partitioned into k folds, most commonly k = 10 is used. One of the folds is held out for testing, while the remaining k-1 folds are used for training. This procedure is repeated k times. Each time, the test fold is changed to a new fold so that after k rounds each fold has been the test fold once. Each round gives a result, and after k rounds, the k results are averaged. K-fold cross-validation ensures that in each round, the training is performed only on the k-1 folds and the test fold remains unseen to the model. At the same time, all of the available data are used once for testing and k-1 times for training. Thus, as opposed to hold-out cross-validation, no data are "‘wasted"’ for training and since all data are included in training in one of the rounds, the way the partitioning is made matters less than in hold-out cross-validation. A drawback of a high k is that the overlap in the different training sets increases. As an example, the overlap in the training folds is 8/9 with 10-fold cross-validation, whereas it is only 1/2 with 3-fold cross-validation. Also, a higher k gives a lower amount of data for testing in each round, 1/10 with 10-fold cross-validation and 1/3 for testing with 3-fold cross-validation. Ten-fold cross-validation has been found to be a good compromise in the text mining domain. Stratification to ensure equal amount of different data types in each folder is often used in combination with k-fold cross-validation. The drawback of k-fold cross-validation is that it may underestimate the variance in performance in the different rounds because the training sets have overlap. This may lead to an increased risk of type-1 error. However, k-fold cross-validation is still considered the preferred method because the test sets are kept independent and the amount of training data is kept as large as possible. Several values of k may be chosen, and the authors also discuss which number for k is the optimal. The benefit of a large k is that the amount of training data is kept high. As an example, with 10-fold cross validation, only 1/10 of the data are held out in each round, whereas with 3-fold cross-validation, 1/3 of the data are held out for training in each round. Another benefit of a large k is that the number of estimates for the model's performance is higher than for a lower k. Ten-fold cross-validation produces 10 estimates of the performance of the model whereas 3-fold cross-validation produces only 3 estimates. [49]

A special version of K-fold cross-validation is leave-one-out cross-validation. This is k-fold cross-validation where k is equal to the number of data instances, k=n. In each round, the test set contains only one instance. As can be expected, this is a very time-consuming method. An example related to the current study would be that for 4500 clinical notes, 4500 folds would be used and the experiment would have to be repeated 4500 times. It can be useful if n is very small. [49]

Different applications of cross-validation are also discussed by the authors [49]. In addition to being used for performance estimation, cross-validation can also be used to compare algorithms and to tune parameters of a model. When comparing algorithms, pair-wise comparisons have been found to give good results. Also, it is better to perform pair-wise t-tests applying k as samples of each algorithm than to compare the average scores between the models directly.

Chapter 4

Previous studies

4.1 Medical corpora

As described in chapter 3.7.1, an annotated corpus is necessary for supervised learning. Automatic annotation by a system can be compared to human annotations.

Some medical corpora, such as the i2b2 corpus, are available for English, but Scandinavian corpora are almost non-existent [48, 50]. Such corpora are important for progress in research - to evaluate and/or train new NLP systems in the medical domain. This thesis will focus mostly on corpora developed for Scandinavian since the goal of this research is to do NER for Norwegian medical records.

4.1.1 i2b2

i2b2, "'Informatics for integrating Biology & the bedside"', is a National center for biomedical computing [51]. Their goal is to improve healthcare systems. They therefore propose i2b2 challenges and make available clinical data (the i2b2 corpus) for research purposes. Many researchers apply these data to test their systems. An example is the master thesis by Bruce [50]. Unfortunately, these resources are available in English only and the domains in which they are relevant may be limited.

4.1.2 Swedish medical corpora

Kokkinakis [48] describes how a large medical corpus, MEDLEX, of Swedish texts were prepared for bio-medical text mining. Annotations were performed by applying a generic entity recognizer in combination with a terminology recognizer. The generic NER categorized the named entities in eight categories and sixty subtype categories. The NER system had previously been thoroughly tested and evaluated for performance for each single entity.

An annotator based on MeSH was applied to recognize medical terms. Words from the original MeSH were converted and normalized to fulfill the purpose of the annotator. The annotator was improved by adding symptoms, names of pharmaceutical products, drugs, Greek and Latin terms. These features were added by adding sources for pharmaceutical products and Greek and Latin terms and by finding a self-made way of making entities for symptoms.

To add grammatical annotations a parsing module based on Cascaded analysis of syntactic structure (Cass) was applied. Cass annotate the data with grammatical labels such as multiword-expressions and conjoined compounds.

The system gave good results on recall and precision when tested on articles from the weekly edition of “The Swedish Medical Association’s magazine”. Annotations made on these articles were manually checked for correctness against the online MeSH. They also found that unambiguous terms could contribute to finding the meaning of ambiguous neighboring terms.

The authors pointed out the need for more research on the following areas; considering the need of a human evaluator in the process loop, the need for a larger number of texts for evaluation and the need to investigate the impact of processing step order, such as trying doing parsing before annotation. Using unambiguous terms to help disambiguating neighbouring terms should also be further investigated.

4.1.3 Norwegian medical corpora

Available resources for testing Norwegian NLP systems for medicine are very limited. Huseth & Røst [52] have been working on making such resources available. They made a semi-automated tool for annotation. The collection of data was based on patient histories from general practice. Initially, these were annotated with base forms, POS tags and phrasal tags. Five phrasal tags were used, NP (Noun phrase), VP (Verb phrase), PP (Prepositional phrase), AP and AdvP (Adverb phrase). The IOB format was used to annotate if a word was in the beginning, inside or outside a phrase. An example of a phrase tag was NP-B. Words could also be annotated as sensitive for later de-identification or as unsure if the annotator was unsure.

A previously developed POS-tagger was integrated with the annotation system so that tags were suggested for each word and the human annotator could check suggested tags for errors. The POS tagger was based on trigram Hidden Markov Models which estimates the most probable tag sequence (T) given a word sequence (W), P(T | W). According to Bayes theorem this can be rewritten as P(T)P(W | T). P(T) was estimated using trigrams, meaning that the probability of a tag was estimated based on the three previous tags. Then linear interpolation was used for smoothing. Huseth & Røst handled compounding by using the last word for tagging because making a dictionary of compounded words in Norwegian is difficult due to the many occurrences of such words in Norwegian. Human and machine annotation benefited from each other by incrementally training the POS tagger. Human annotation speed increased because the suggestions by the automated annotations improved.

Similarly, the POS tagger benefited from training on human annotated data. The text was also automatically split into sentences, tokens, tags for base form and phrase tags were automatically selected. NorKompleks computational lexicon was used to find base forms of words. If the word was not in the lexicon, the word itself was used as base form. Phrase tags were based on POS tags and assigned based on static rules. Tokenization was based on white spaces. Some medical constructs could not be handled by the standardized tokenizer. Exceptions were made to handle these. Abbreviations were annotated by the human annotator. The annotation tool was made by using the Python language and the Django web framework. The annotation tool had three major components, a module for sentence and tokenization review (to add missing punctuations and other necessary modifications), an interface with the text and base form words listed with drop-down selections for POS tags, phrase tags, sensitivity and a module for cases where the annotator was unsure. The results were promising as a tagger trained on the medical corpus became better for medical documents than a tagger trained on a more general corpus [52].

4.2 Previous studies

Several previous studies on NLP in the medical domain exist. However, the medical domains investigated in each study vary and are not always comparable. A recurrent problem in the medical domain is the lack of available data sources [15]. EHRs are generally strictly treated and special permissions are required to access them. Huseth & Røst [52] have made one annotated medical corpus available for Norwegian, but otherwise, the availability of such resources seems to be non-existent for the Norwegian language. The lack of available resources has caused some researchers to try to find alternative resources such as web pages like PatientsLikeMe where patients share detailed information about themselves.

4.3 Medical entity recognition

A study by Abacha & Zweigenbaum, [53], compared three different approaches to Medical entity recognition (MER). Three categories were used in the study, "‘Problem"’, "‘Treatment"’ and "‘Test"’. The different approaches used for MER were the following:

1. MetaMap - a mapping tool used to map terms to UMLS concepts
2. MetaMap with rules used for categorization
3. TreeTagger for noun phrase chunking and SVM for categorization
4. CRF with BIO tags to annotate beginning (B), inside (I) and outside (O) of a phrase
5. CRF with BIO tags and MetaMap to use UMLS concepts as features for terms

The different approaches were tested on data from the i2b2 2010 corpus which consisted in discharge summaries and progress notes. Precision, recall and f-measure were measured. The results showed that approach number five gave the best results (f-measure=77,55), followed by approach number four (76,17), number two (52,28), number three (45,33) and number one (15,8).

This order was the same when looking at precision and recall individually as well. For each of the three categories, approach number five gave the highest f-scores for all of them. The five different approaches were also compared when applied to another corpus, consisting of scientific abstracts rather then clinical notes. The order of which approach was best remained the same, but the differences were lower. Secondly, approach number five was tested with a combination of the two corpora as training data. This gave better results then using only one of the corpora for training. They discussed the pros and cons of using rule-based methods compared to statistical methods. Advantages of rule based method were that these do not require a learning step and that the mapping to UMLS concepts is more available. However, rule-based methods were dependent on a good chunker to find correct boundaries of phrases. The CRF algorithm were tested with different features. It was found that also for the statistical approaches, the entity boundaries were important. Adding UMLS concepts as features to the algorithm without using the BIO tags decreased performance, while using CRF with BIO-tags and semantic features (UMLS concepts) gave the best performance.

4.3.1 Automatic detection of adverse events in clinical texts

NLP has been applied with promising results for detection of adverse events (AE) in clinical notes [54–56]. An adverse event is an unwanted effect of a medical intervention [57, 58]. As mentioned in the medical background chapter, 2.1, adverse events such as CRBSI is related to the usage of CVC. A study by Penz et al. [6] compared natural language processing and phrase matching for semi-automated detection of adverse events related to CVC. A combination of the two methods was also evaluated.

In the latter study, CVC was chosen for the study because the number of adverse events related to CVC was limited. Thus, the number of ways that these events could be described in clinical notes was limited. Regular expressions (a way to specify a search pattern) was used for phrase matching. A local lexicon used by nurses and doctors at the hospital, UMLS synonym phrases as well as phrases reported as common for CVC AEs by three surgeons were used to make regular expressions for the phrase matching algorithm. Simulated CVC related AEs and a scoring system were used to test and improve the phrase matching algorithm. The regular expressions were improved step by step each time errors were made on simulated CVC related AEs. The word distance between a CVC expression and an AE was used as a measure of how probable it was that the AE was related to the CVC expression.

A natural language processing program, MedLEE, was compared to the phrase matching. Some pre-processing of the notes was necessary because MedLEE required the input to be in a specific format. The pre-processing module was iteratively improved in a similar way as the phrase matching algorithm. Records containing different types of clinical notes (physician progress notes, consultation notes, nursing notes, procedure notes, operative records and discharge summaries) were used in the study. A selection criterion for records was that they contained at least one CVC placement. CPT and ICD-9 procedure codes indicated whether a note contained CVC, and these were used for the selection. Records were selected from a five years period, which resulted in 365 CVC records. Each record was converted to a large text file where the temporal order of notes were kept to be able to investigate causality of AEs and CVCs.

Fourty of the records were manually reviewed by two surgically trained physicians to form a reference standard. The physicians manually read the records in order to decide if the records contained CVC related AEs. The collection contained 30 records that had a high probability of containing AEs (distance between AE and CVC was 6-13) based on the scoring system and 10 records that were very unlikely to contain AEs (score 0). This ensured that both records with a high probability of including CVC related AEs and records with a low probability of having CVC related AEs were included in the standard. The manual review showed that certain information such as blood culture results was seldom present in the clinical notes. Thus, it could be difficult to decide with certainty if a record contained CVC related AEs. The physicians rated the probability that the records contained CVC related AEs using a scale with the values ”‘possible”’, ”‘highly likely”’, ”‘likely”’ and ”‘certain”’. A board certified surgeon did an evaluation for records where the two physicians disagreed.

The two semi-automated methods were compared to the manually made reference standard. To decide if CVC related AEs identified by one of the semi-automated methods were true positives, the findings were correlated to the manual reference standard and considered a true positive if the manual rating was at least ”‘possible”’. MedLEE was found to be better at specificity then sensitivity, while the opposite result was found for the phrase matching algorithm. Thus, a third experiment combining records with a high score at the phrase matching with the results of MedLEEs predictions of AEs was performed.

It was found that one of the commonest reasons for failure of a semi-automated method to detect a CVC related AE or falsely reporting a CVC related AE was spelling errors and abbreviations even though usual abbreviations were included as part of the pre-processing of both methods. Suggested methods to handle spelling errors were to use dictation, professional transcription, automated spelling check including suggestions for abbreviations and an additional pre-processing module able to handle abbreviations and spelling errors that their methods did not handle. Spelling errors were a greater problem in this study than in previous studies, and this was explained by the type of clinical notes included in the study. The authors argued that physician entered notes probably have a much higher rate of spelling errors than discharge summaries.

Phrases in which physicians had documented that information about risks was given to the patient was often falsely detected as positive findings of CVC related AEs by the phrase matching algorithm. Phrases often related to CVC AE that occurred closely to CVC synonyms lead to similar problems for MedLEE.

The number of CVC records seemed low and a crosscheck was performed to check the reliability of the selection. Cardiac and aortic procedures (always involving CVC) documentation could be checked against the administrative selection method, and it was expected that the patients that went through these procedures would be present both by the administrative selection method and by the surgical documentation. However, of 1423 surgical procedures, only 163 were captured by the administrative selection method.

The study indicated that detection of CVC related AEs by the semi-automated methods was lower, but comparable to detection of such events by humans. The results also revealed that in some cases the physicians making the reference standard overlooked CVC-related events if these were mentioned very briefly. In these cases, the semiautomatic methods performed better. The combination of the phrase-matching algorithm and MedLEE gave the best results with a sensitivity of 72 %, a specificity of 80.1 % and a predictive value of 64.3 %. The estimated rate of CVC AEs for a life of a catheter in the complete set of records was calculated using the results for the 316 records and their PPV. The estimate for the phrase matching was 6.4 %, for MedLEE, 6.2 % and for the combined method, 10.4 %. NLP was better at specificity, but not on sensitivity, whereas phrase matching was better at sensitivity, but worse at specificity. The best results were obtained when the two methods were combined. Combining the two methods gave a sensitivity score of 72% and a specificity score of 80,1 %. The positive predictive value of AEs found in the 316 records by each method was 41 % for the phrase matching method, 70.5 % for the MedLee and 64.3 % for the combined method. The specificity, sensitivity and PPV were considered high enough to be useful for surveillance purposes since it would highly reduce the number of negative charts to process.

Another conclusion of the study was that text-based methods to find CVC patients may be more valid than using administrative methods and that this should be studied further.

MedLEE was also applied in a study by Melton & Hripcsak,[13] for detection of 45 types of AEs in discharge summaries. The AEs were predefined in New York Patient Occurrence Reporting and Tracking System (NYPORTS), an adverse event reporting system used in New York. The authors pointed out that an advantage of using MedLEE rather than specific trigger words to detect events was that MedLEE converts the text into a coded format that yields information about negation, uncertainty, timing, synonyms and abbreviations. In the experiment, MedLEE was used to code discharge summaries and in addition queries were made to extract events as a list from the coded discharge summaries. These queries were improved iteratively. 100 charts were initially read by two reviewers, a physician and a informatician. The inter-rater reliability was found to be very high, a chance-corrected agreement of 0.94. Thus, it was decided that the physician alone was sufficient to do the manual review of the remaining charts. Also, reliability between different data sources was evaluated; 1000 discharge summaries and 1000 electronic charts were manually reviewed and compared to figure out whether discharge summaries was a valid source to find adverse events. The agreement was found to be 0.96 and it was concluded that discharge summaries could be used to detect adverse events [13].

The system was initially tested on 1000 discharge summaries and the results were compared to a human review. Then the system was tested for all discharge summaries of a period of four years, 57452 cases. The results were reviewed by the physician. Detection of both cases and event types was reported. The manual review by the physician was used to decide true positives for events and cases. For both cases and events, sensitivity was only fair (event sensitivity=0.25, case sensitivity=0.28), while specificity was very high for both (event specificity=0.9996 case specificity=0.982). However, the sensitivity was much higher than the traditional detection system which had a sensitivity of 0.086. The automated system missed some of the events reported by the traditional system (110/322), but detected 594 new events that the traditional system had missed. Thus, in total, the number of events detected was increased. The system performed better than several previous studies that had applied simple search strings rather than NLP. The system's ability to detect both events and event types was also considered new and unique. Other findings in the study was that it was hard to make queries with respect to time when these were more complicated than figuring that one event happened before another. Also, looking at the different event types, the results for each event type was highly variable. This was explained by the number of each event type being variable and by some queries being harder to implement than others. Events that were not explicitly described were also hard to detect. Possible biases in the study was that only electronic records were included and that patients staying shorter than 48 hours were excluded from the study. Including events of these patients could potentially lead to another result. The authors points out several potential areas of use for their system such as national screenings, adverse event prevention, automated diagnosis coding, real-time clinical guidance, computer-assisted documentation and feedback to clinicians [13].

Gurulingappa et al. [59] developed a system for detection of adverse events related sentences in medical case reports. The system combined a maximum entropy based classifier and dictionary-based NER. The classifier applied morphological and syntactic features to classify sentences as positive or negative. Sentences classified as positive were those that contained drug-related adverse effects, whereas negatives did not. The dictionary-based NER was used on the positive sentences to normalize words such as names of drugs. A corpus, ADE, consisting of 2972 Pubmed case reports was partitioned into a training (80 %)and a test set (20%). Annotations were made for drugs, adverse effects, doses and relationships between them. Annotator agreement had been calculated between three different annotators. Different sets of features and different supervised learning algorithms were tested. The combinations of features consisted in words, lemmatized tokens, lexicon-token-matches; Drugbank and MedDRA single word lexicon matches, lemmatized-token-bigrams, lemmatized-token trigrams, noun-character-affixes, preceding and succeeding lemmatized verbs of drug- or condition matches, lemmatized-tokens-in-window; window of +-5 lemmatized tokens of drug- and condition matches, Standford-token-dependencies; a parser finding dependencies among words.

The researchers started with only the words as features and then sub sequentially extended the feature set by adding features one by one. For each additional feature that was added, the performance was tested by cross-validation on the training set for all of the algorithms used in the experiments. F-scores were used as metric. If the addition of a feature lead to worsened results, the feature was removed before adding additional features. All features except lemmatized-token-trigrams lead to improved F-scores. The maximum entropy classifier gave the highest F-score during the comparison of the algorithms. It was therefore decided to use this algorithm in further tests. The maximum entropy algorithm was tested with the ADE test set. The results of this test were compared to a baseline consisting of only words as features. Scores for precision, recall, f-score and macro-averaged f-scores of both positively and negatively classified sentences of the maximum entropy algorithm were compared to base line scores.

To reveal possible sources of errors in the classification system, a manual analysis of false positives and false negatives was performed. Three sources of false positives were mentioned;

1. Adverse events referring to drug type and not to the corresponding specific drugs.
2. Adverse effects of a drug mentioned in general terms, without specifying what the adverse event consisted in
3. Adverse effects associated with various forms of medical treatment such as adverse effects associated with the removal of the thyroid gland (thymus)

Similarly, erroneously classified negative sentences were manually reviewed. Three reasons are also mentioned for false negatives;

1. The adverse effects were not described in the dictionaries applied
2. Long sentences where adverse effects were hidden in the text
3. The relationship between drug and adverse effect was incompletely described

The NER system was also analysed. A test was performed to check if co-occurrence of drug and adverse effect could be used as a basis to decide if a sentence was positive. The co-occurrence of drug and adverse effect was not enough to classify the sentence. Words missing in the dictionaries applied by the NER system was the most common reason for false negative entities. Words that were missing were often abbreviations.

To evaluate the trained maximum entropy classifier on unseen data, an exam corpus was applied. The classifier was found to be helpful in identifying relationships between drugs and adverse effects. It was concluded that the system was promising in terms of detecting new cases of adverse effects and knowledge extraction from medical texts

4.4 Named entity recognition system for Scandinavian languages

In the large Nomen Nescio NER project several different strategies to make NER systems for Norwegian, Swedish and Danish were investigated. The data sets were not medical, but their results are interesting because they use NER with a main focus on how NER can perform efficiently for Scandinavian languages. These are considered as dialects of the same language and are therefore comparable since similar corpora were used for the systems. Six different strategies were tested and compared. Three of the systems were made for Norwegian, two for Danish and one for Swedish. Some of the systems were based on statistical methods and some of them on rule-based methods [60].

Six different named entities were applied in the project; person (PRS), location (LOC), organization (ORG), event (EVT), work of art (WRK) and other (OTH). These categories were selected because they were considered as being of possible use for information retrieval (IR) at the World wide web. It was found that selecting correct semantic category of a word was more difficult than expected. For instance, country names might represent political organizations, sports teams or locations. To deal with the categorization problems, two annotation strategies called "'Function over form"' and "'Form over function"' were applied and compared [60].

The "'Form over function"' strategy entailed that words of a specific name form were always treated the same way, independent of setting. This strategy was dependent on word lists/entity dictionaries (gazetteers), which causes inaccuracies when words are not found in the gazetteers. In cases where the word is not found in the gazetteers, the context is used to categorize the word, which might lead to inconsistencies in the labelling. An example of the "'Form over function"' strategy is that Israel is tagged as a location even though in a specific sentence it may refer to the Israelian army, which is an ORG and not a LOC. This strategy was applied for the Danish and Swedish systems [60].

The "'Function over form"' strategy prioritized the function of the word for categorization. This means that the same word could be given the LOC tag in a sentence where the function of the word was a location, and the ORG tag in another sentence where it had the function of a political organization. The "'Function over form"' strategy was applied for the Norwegian systems. For this strategy errors in labelling also occurred when the context was insufficient to correctly decide the function of the word [60].

The Danish and one of the Norwegian systems in the project were rule based and used constraint grammar (CG) and CG tags. The Swedish system applied another rule based method, shallow parsing with context sensitive finite state grammars (FS). For two of the Norwegian systems, statistical methods were applied; one of them applied maximum entropy and the other used memory-based learning. Gazetteers of some form was applied for all the systems [60].

A small scale manual method and a larger scale automatic method were used for evaluation of the systems. The small scale method aimed at adjusting evaluation criteria for each system to make the systems comparable despite that they were made for different languages and with different strategies. An example is that tags were made differently for "'Function over form"' systems than for "'Form over function systems"'. Thus, the number of correctly identified tags had to be counted taking the strategy into consideration. This evaluation showed largest recall (91 %) and precision (93%) for the Swedish FS system [60].

For the larger scale evaluation, the systems were tested separately on larger corpora. For this evaluation, the Danish CG system performed best (95% recall and 95% precision) [60].

The results indicated that the number of items in the gazetteers and the way the gazetteers were used affected the success of the systems. It was hypothesized that the systems would perform worse when removing gazetteers and that the reduction in performance would be largest for the "'Form over function"' systems since these applied larger gazetteers than the "'Function over form"' systems. The Swedish FS system was compared to the Norwegian CG system and their hypothesis could be confirmed. The Swedish system dropped from a recall of 91% to a recall of 53% while the Norwegian system increased recall from 72% to 83% [60].

The results also showed that whether the system was rule-based or statistical did not affect performance systematically [60].

4.5 Annotation and classification of Swedish Medical Records

Annotation and classification of medical records in Scandinavian languages have also been performed at Stockholm University. These researchers de-identified medical records and collected them as a corpus so that it could also be applied by other researchers. They studied how affirmed, negated and speculated information is expressed in Swedish medical records and looked at annotation and classification of these data.

Three different corpora were made, one consisting of records annotated with identifiable information, one annotated for sentence level uncertainty expressions and one annotated for diagnostic statement level uncertainty [46].

The Stockholm Electronic Patient Record (EPR) corpus was used to develop the different gold standards. The Stockholm EPR consists of EHRs from the Stockholm county council from 2006-2008 [46].

For the sentence level certainty annotation the researchers chose assessment (bedømning) fields from some of the records in the Stockholm ERP corpus. Assessment fields were chosen as these fields contain the largest amount of reasoning and it is probable that fields containing reasoning might contain expressions of doubt and uncertainty. The goal was to find out how uncertainties were expressed in these fields. Three persons manually annotated the data with the following annotation classes; certain expression, uncertain expression, negation, speculative words, undefined expression, undefined speculative words. Inter-annotator agreement (IAA) between annotators were calculated. IAA was found to be high for certain expressions and negations and considerably lower for uncertain expressions and speculative words [61].

Skeppstedt et al. [36], used named entity recognition for assessment data from a Swedish emergency unit from the Stockholm EPR. Annotated data were used to test a rule- and terminology-based entity recognition system. Body structure, disorder, and finding, all corresponding to semantic classes from the SNOMED CT were used as entities. An experienced physician performed the annotations.

The authors tested the possibility that SNOMED CT could be used as a resource for automatic retrieval of medical entities and to what extent the SNOMED CT covers clinical expressions from clinical records. Rule-based lexical look up in one to five different terminologies were applied to recognize the three entities in the clinical records. The preannotated corpus was used as gold standard. Different kinds of preprocessing were applied to investigate how that influenced the results. Different experimental conditions were tested. Some included lemmatization, removal of stop words, including words with a levenshtein distance of 1, including the terminology ICD-10, the MeSH, Wikipedia: Project Medicin, Medical abbreviations and acronyms.

Words from the clinical notes could be looked up in the SNOMED CT (and for some of the experiments also including other terminologies) and then the SNOMED CT semantic class (body part, finding, disorder) for the looked up word could be found and used to annotate the data. When ICD-10 was included as terminology, chapter 1-17 and 19, except T357-T629 were used for disorder, chapter 18 was used to match findings. When MeSH was included, category F03 and

C were used for disorder, A01-A10 were used for body structure. Also, a list of diseases from the Wikipedia Projekt Medicin was included as terminology used to match disorder Medical abbreviations and acronyms were included to also include abbreviated terms. Lists of abbreviations for disorders, body structure and findings were made. For evaluation of the results, a script from the CoNLL shared tasks was applied. This calculated precision, recall and F-score for exact matches.

The results showed that the total F-score was best for body structures (0.77), which improved recall when stop word filtering was applied to SNOMED CT terms for body structures, as body structures can also be included in descriptive expressions. For disorder, it was 0.63 and for finding, it was 0.41. Also, body structure was most influenced by pre-processing.

Disorder was most influenced by adding terminologies. Compared to other studies, the researchers found the results to be low, but they point out that a possible explanation could be that the texts they applied may be less formal than for example discharge summaries as applied in some other studies. The researchers suggest that the results indicate a limited coverage by the SNOMED CT for clinical terms in the records applied in the study. Long expressions were not always discovered by the system as entities and many of the false negatives were abbreviations.

They also found that body structures were most often annotated as one-token expressions. Disorders were often annotated as two-token expressions. Findings were often annotated as two-token or three-token expressions. The system did not correctly recognize expressions longer than two tokens. Almost none of the correctly recognized terms were abbreviations. Which of the classes finding and disorder that a word was assigned was found to be context dependent.

For future work, the researchers recommend measurement of IAA, testing of the system on other clinical texts, usage of machine learning methods to recognize entities in the clinical texts, and usage of the output from the rule-based systems as a feature of a machine learning based system.

They also conclude that a rule-based system that applies existing terminologies is insufficient to do NER for clinical texts with good results in terms of precision and recall [36].

Velupillai [62] studied automatic classification of factuality levels for Swedish diagnostic statements. They applied machine learning techniques with an automatic classifier using conditional random fields (CRF) which had been trained on a corpus of assessment fields from the Stockholm EPR corpus. They used local context features, word, lemma and part of speech tags for classification. A general POS-tagger for Swedish was applied. Positive and negative factuality levels were applied for classes, and these were graded as certain, probable and possible. F-score, recall and precision were calculated using the CoNLL shared task script. The word itself was used to calculate baseline levels. Certainly positive received the highest results (F-score 0,742). The best results were obtained when words, lemma and POS-information were used with a context window of +-4, meaning that the four preceding and posterior words were used as features. They found that preceding context gives valuable information. POS information was also found to be most useful as a feature when used in combination with words and lemmas.

A recent study by Skeppstedt, [63], used CRF for NER of the entities disorder, finding, body structure and pharmaceutial drug in clinical notes. This study was different from others by using Swedish texts and also by separating disorder and finding instead of combining them in the entity "'Problem"' which had been done in several other studies. They applied CRF++ with IOB-tags. The clinical notes were from the Stockholm EPR Corpus. Only parts of the notes were applied in the study, namely "'Assessment"' texts as these contained information about disorders and findings.

Manual annotation guidelines were developed by a physician with annotation experience and a computer linguist. As an example phrases to be annotated were made as short as possible, excluding severity. Compound words were not splitted. A second phycisian was used to validate the annotation guidelines. The features used to classify terms were terminologies; MeSH, ICD-10, SNOMED-CT and FASS. Compound words were splitted in two parts if one gave a match in any of the terminologies. Lemmas, POS-tags and ortographic features were also applied in the study. The features were added one by one, and only features that improved performance were kept in each iteration.

They used 30-fold crossvalidation and argued that the advantage of this compared to 10-fold crossvalidation was that more data can be used for training. The optimal feature set included, lemma with a window of -1, POS tag including two previous and one posterior POS tag, terminology match including the previous terminology match, compound splitting features and orthographic features without window. This feature set gave a recall of 0,759, precision of 0,832 and an f-score of 0,794. A separate evaluation set was used for a final evaluation of the optimal feature set. To figure out to which extent the different features improved the results, features were removed one at a time to see how much the performance decreased. The current lemma followed by terminology were the features that had the greatest effect. The results for the evaluation set were similar to those obtained during development.

Common errors were ambiguity in categories, spelling errors, jargon, phrases where too much or too little had been included, abbreviations, compound words and errors in the manual annotation. The authors concluded that NER approaches applied for English are transferable to Swedish. However, compound words are common in Swedish and poses an extra challenge compared to English. Contrary to other studies it was found that small word windows yielded the best results. The authors explained that this may be because the set of features were too large relative to the size of the corpus. Another source of error was that the lemmatiser was not properly adapted for the medical domain. Inter-annotator agreement results were lower for finding and disorder then for the other categories, indicating that these were harder to separate from each other and that the question of whether these should be merged or not depended on the use of the NER system.

4.6 Synonym handling in Norwegian Clinical notes

A study by Henriksson et al. [22] used word space models based on random indexing and random permutations to find synonym pairs in clinical text. Random indexing and random permutations were applied to reveal semantic relationships between words. Abbreviations could be mapped to the full form of the word in the same way.

4.7 NER in the clinical domain

Bruce [50] experimented on ontology-based information extraction in the clinical domain. The ontologies SNOMED CT and RxNorms were used to find entities in clinical texts. The results were compared to the results of the i2b2 shared tasks.

The cTakes framework was applied, and two home-made modules were developed. The research indicated that ontologies may be used to identify named entities in clinical texts.

With a medical corpus available, supervised learning could be applied.

4.8 Concept normalization

Bashyam & Taira (2009) [64] studied how to compare lengthy medical concepts that are orthographically different, but which have the same meaning (e.g. heart attack and cardiac attack). Phrases were transformed from free text to a normalized dependency vector space representation. Phrases were firstly tokenized. After tokenization, syntactic parsing was performed. Syntactic parsing involved making a syntactic dependency tree which contained syntactic relations between words. In a relation, one word is head and the other is modifier. A word can be the modifier only once, but it can be the head several times in a phrase. One relation is represented as a link between the words involved.

After syntactic parsing, link reduction was performed. The links between words could be bi-lexical or tri-lexical, where a bilexical link was a strong dependency link between two words, whereas a tri-lexical link was a link between three words where a mediator word was often in between the two related words (e.g. nucleus-of-thalamus, where of was the mediator word). Link reduction involved removing mediator words when possible to reduce a tri-lexical link to a bi-lexical link (nucleus-thalamus). When a tri-lexical link was converted to a bi-lexical link, the new link was tagged with the mediator word that was removed.

The tokens in the reduced dependency tree was then normalized to their base form such that for example thalmic was normalized to thalamus. ”‘The normalized dependency parse tree is represented as in a vector space as a bag-of-links.”’. When these steps were performed both for a phrase from a clinical text and for a phrase from a taxonomy, it was possible to correctly match phrases from clinical notes to phrases from taxonomies even though the phrases were differently before doing the normalization.

Chapter 5

Methods

5.1 Data

Twenty-eight EHRs, consisting of 4533 clinical notes, were selected from the Dips EHR database of the Akershus University hospital (Ahus). Since a patient could have stayed at the hospital for different periods of time, each record contained multiple periods of care. Each period of care contained several clinical notes, including nursing notes, surgical notes, medical notes written by physicians and laboratory examinations. The relationships between patient records, periods of care and clinical notes is illustrated by figure 5.1.

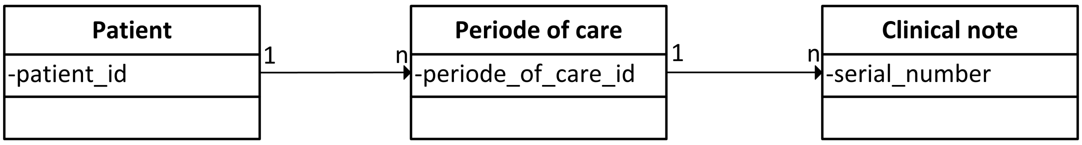

FIGURE 5.1: Each patient has a patient Id. One patient can have multiple periods of care, each of which also has an Id. Each period of care contains several clinical notes. Each clinical note has a unique serial number.

5.1.1 Permissions and storage of data

The study was approved by the Norwegian Regional Committees of Medical Research Ethics (REK). Unique personal acquaintance characters were removed from the data, but an exception from confidentiality was approved because the researchers could potentially recognize persons known to them privately. Use of data was only permitted to members of the Evicare project. A non-disclosure agreement was signed by all members of the project.

The data were kept and stored confidentially in a secure zone. Three machines disconnected from the Internet could be used to remotely access a server with an external hard drive containing data.

5.1.2 Selection criteria

The study population was selected from a prevalence survey from September 2011, in which some medical attributes of the hospital's patients were examined and reported on a specific date. The selection was based on this survey and not on the EHR system. On this specific day the survey identified 28 patients with CVC, which constitute the present study population.

Patients where the CVC had been removed before the prevalence survey or had CVC inserted after the prevalence survey, were not included in the study population. The 28 records included all the clinical notes on a patient during his/her stay at the hospital. They therefore contained much information other than that pertaining to CVC. In fact, the 28 records contained 4533 clinical notes. This number was considered adequate for research purposes, and was also possible to explore within the time limitations of the project.

The large amount of clinical notes for each patient ensured that notes containing as well as not containing CVC-related events were included in the study. Figure 5.2 illustrates the interrelationship between CVC during a period of care and a prevalence survey. Figure 5.3 gives an example of a patient who has clinical notes containing CVC information, but who was not registered with CVC in a prevalence survey.

A patient could also have CVC inserted after the prevalence survey. In that case, the patient's clinical notes from after the survey would probably contain information about CVC, but the patient would have a negative result for CVC at the prevalence survey. Notes that could possibly contain CVC information were not read if the patient did not have CVC at the prevalence survey (see 5.3). A requirement for selection of records was a care period of more than three days for each patient.

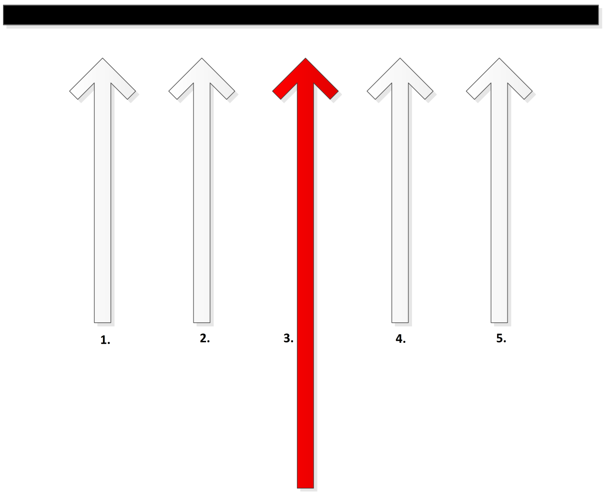

FIGURE 5.2: The line on top of the figure illustrates a time line. The first arrow marks the start date of a care period (1). The second arrow (2) illustrates CVC insertion. The third arrow (3) illustrates a prevalence survey. The fourth arrow (4) illustrates removal of CVC and the fifth arrow illustrates discharge of the patient. Figure taken and modified from Christine Tvedt's presentation.

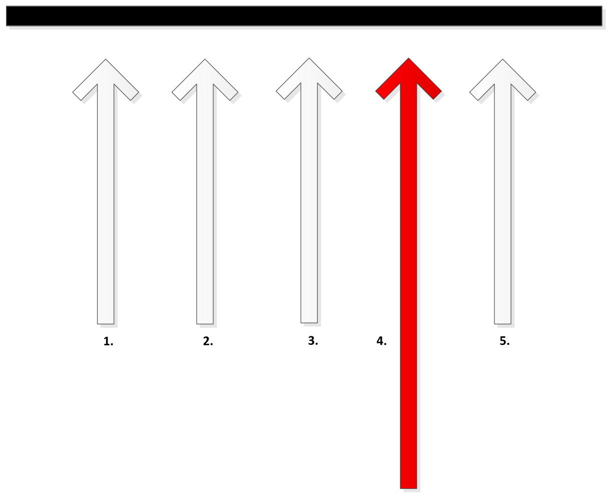

FIGURE 5.3: The line on top of the figure illustrates a time line. The first arrow marks the start date of a care period (1). The second arrow (2) illustrates CVC insertion. The third arrow (3) illustrates removal of CVC. The fourth arrow (4) illustrates the day of a prevalence survey. The fifth arrow (5) illustrates discharge of the patient. Figure taken and modified from Christine Tvedt's presentation.

5.1.3 Manual annotation

Human-performed reading and annotation of clinical notes was performed in order to make a corpus of data annotated with CVC events available for the study. This was performed by PHD student Christine Tvedt, employee at Kunnskapssenteret [65]. She has a background as a nurse with a special competence in infection control and thus has domain knowledge necessary to perform annotation of CVC related events. Brat rapid annotation tool [45] was applied for the human-performed annotation. Table 5.1 shows and explains annotations applied in the annotation process. Tvedt, in cooperation with Laura Slaughter, a domain expert in natural language processing, defined the set of annotations that were used to annotate CVC-related events. By applying Brat annotation tool, one annotation file was produced for each clinical note. The annotation file was named by the same file name as its corresponding clinical note and contained all annotations belonging to that clinical note. The format of the annotation file is called ann-format or stand off format.

5.2 Training a classifier to automatic annotate journals

5.2.1 Preparing clinical notes for learning

The data were sent to NTNU in text format, converted from rich text format by Haldor Husby in cooperation with DIPS [66]. Clinical notes were identified by a serial number, a patient number, a care period number, document type and the date when the document was made. Serial number could be used as a unique identifier for a clinical note. Annotation files were provided separately to the clinical notes.

Table 5.1: Annotations

Annotation	Description
Carecvc	Care, observation or assessment of CVC
PlanCarecvc	Care of CVC has not been performed, but has been booked or planned
PlanInscvc	Admission of CVC not performed, but planned, desired or ordered for the future
Inscvc	CVC has been inserted
Remcvc	CVC has been removed
PlanRemvcvc	Removal of CVC has been planned
Symptom	statements indicating that there may be a blood system infection (BSI) contains the words "blood culture", "infection" coupled with "CVC" and the like
Sepsis	Sentence containing the word sepsis
Device: CVC, Hickmann, VAP, other	Type of CVC
Site: Jugular, subclavian, femoralis	Site of the vein for CVC insertion
Possiblecvc	Sentences in which CVC is discussed without mentioning the word "CVC". Information possibly relevant to CVC

A C++ implementation of CRF, CRF++, [67] was used as training algorithm in the current study. CRF++ required training and test files to be represented in CONLL format, see [68] for an example. Each token had to be on a single line with its features and label on the same line separated by tab. The tokens represented words to be labelled, the features belonged to the tokens and helped finding the correct label of a token. Words from the clinical notes were used as tokens. For training files, the correct label of a token was known and assigned to the token prior to running the algorithm.

For testing files, an attempt was made to predict labels. Sentences were separated by vertical space/empty lines. A phrase given a specific label caused all the words of that phrase to be assigned with that label. Word level tagging was applied and IOB/BIO tags were used to define boundaries of start and end of a phrase. This was made by assigning a letter, I, O or B to each label that were attached to a token. This indicated whether a token of a phrase was in the beginning (B) or inside (I) the phrase. Words that were not part of any phrase were labelled O for outside. The CONLL format and the BIO tags are exemplified in table 5.2.

A template file had to be specified before training. The template file denoted features and context to be applied in training and testing. Each element in the template file consisted of a row and column number that informed the algorithm that the feature in that position of the input files should be included as a feature during training. An example template file for the Conll file in 5.2 is shown in 5.3. Each line have a unique identifier, "'UXX"', a row and a column. [0,1] means that when training on the files given in conll format, row 0, column 1 should be included as one of the features.

In this case this is the stem of a token. [-1,1] still means column number one. The -1 means that we are using the row that is prior to the current token. Thus, [-1,1] means the stem of the token

Table 5.2: Conll format

Token	Stem	Synonymous	ICNP axis	Label
Stell	stell	0	J	B-Carecvc
av	av	inaktiv	DS	I-Carecvc
CVK	cvk	CVC	J	I-Carecvc
ble	ble	fremkomme	J	I-Carecvc
utført	utfør	avholde	J	I-Carecvc
Pasienten	pasient	behandlingstrengende	J	O

Table 5.3: Template file example

```
U01:%x[0,1]
U02:%x[-1,1]
U03:%x[0,2]
U04:%x[0,3]
B
```

before the current token. The B in the end of the template file tells the CRF algorithm that the template is a bigram template, meaning that output of the current token and the previous token is applied when predicting the label of the current token.

A tool for NLP-assisted text Annotation, called AnnToConll, [69], was applied for converting clinical notes and annotation files into CONLL format. In the CONLL format, words are printed as a vertical column and sentences are split by adding a space between the words. Columns are also generated for offset numbers, and labels from the ann-files are added in the file as a separate column. The tool had to be slightly modified in order to be applicable for Norwegian language and in the format required by CRF++. The tool with modifications is included on the DVD attached to this thesis. The modified anntoconll tool converted tokens and labels to CONLL format. A separate module was made to select files for conversion and to run the script for all files. A module was also made to add stemming, synonyms and ICNP codes as features for training. The stemming was based on a standard snowball stemmer for Norwegian. The synonym matching was performed by Hans Moen using the code applied for synonym extraction in the work by Henriksson et al. [22]. Here three semantic vector space models were trained.

As mentioned previously, Velupillai [62] used different terminologies as features for tokens. In that study sentences were used to do look-ups in the terminologies and then assigned the best matching terminology term to each of the tokens in the sentence. The same strategy was applied in the current experiment. Sentences were stemmed and matched against terms from ICNP by applying a sentence similarity method. For this we used the code by Hans Moen that previously had been used to generate the Random Indexing based features used in Marsi et.al [70] and Moen et.al [71]. Sentence vectors were created by summing normalized term context vectors for the constituent terms. In addition to TF-IDF weights, double weight was given to context vectors of terms matching a dictionary of medical terms. This dictionary contained a collection of terms derived from ICD-9, ICD-10, NCSP, NIS treatment codes and DRG codes. For sentences that matched ICNP terms, the ICNP axis corresponding to the ICNP term was returned. ICNP axis was later applied as features of tokens to the CRF algorithm.

The data files were separated into training files and test files. A small program was made to make a randomized selection of files from the total amount of clinical notes available (see attached DVD). The partitioning of files into training and test files was done similar to other studies applying CRF++ [62, 72], dividing the files into 20 % for testing and 80 % for training.

Training required use of the training files and a CRF template. A template containing all features was tested once with Unigram template and once with Bigram template to decide which of these templates should be used for further experiments. Bigram template gave clearly the best results and was therefore used for further experiments. The template defined features to be included in an experiment, as well as whether bigrams and context windows of words should be included. Including a context window of for example +-2 words implied that two tokens prior and two tokens posterior to a token were included as context features of a token.

The CONLL format, including token, features and labels, was also applied for test data, but for test data the label input was used merely as a comparator to the label predicted by the algorithm. After a test run, CRF++ provided a list of results including precision and recall for the sum of all labels, as well as precision and recall for each label separately.

5.2.2 Design and statistical method

A full factorial design, a 2^4 experiment, was applied to investigate the effect of features possibly relevant to annotation of CVC related events. This type of design was appropriate since it requires few experimental runs to indicate the effect of various factors on the response variable. A full factorial design includes all combinations of features on/off and allows the possibility to investigate both the effect of each feature alone, as well as interactions between features [73].

The text files were randomly divided into three equal sized partitions so that three-fold cross validation could be applied (see code attached on DVD). These three partitions were used for all the different templates/feature combinations so that each partition was used as test set once and training set twice. Minitab 16.0 [73] was used for statistical analysis. Three blocks were used in the factorial analysis, one for each of the three cross validation data sets so that each block consisted of the sixteen feature combinations.

The CRF algorithm was trained and tested for each of the sixteen feature combinations. One feature combination run resulted in recall-, precision and F-scores for each of the named entity categories. Also, cumulative recall-, precision- and f-measure score were returned for each feature combination run. The cumulative f-measures were used as response variables in the factorial design so that each of the sixteen feature combinations were assigned one f-measure, resulting in forty-eight responses, sixteen for each cross validation data set.

CRF returned f-measure, recall and precision for each model trained using different feature sets. Result scores were obtained for each annotation label as well as an overall result score for all

Table 5.4: Features included in the experiment

	+	−
Word window	+/− Four words	+/− No words
Stem	Included	Not included
ICNP axis	Included	Not included
Synonym	Included	Not included

annotation labels. The accumulative scores for the sixteen feature combinations were analyzed using a factorial analysis in Minitab 16. Three factorial analysis were performed; for recall, precision and f-measure. Three blocks were applied, one for each cross-validation run. Terms in the model up to second order was included. Third and fourth order interactions are often results of random noise and were therefore excluded. An F-test (denoted by F, but different from f-measure) compares differences in variance and can reveal whether some of the factors have a significant effect on the overall result scores. The α-level for significance was set to 0.05 in all tests. In cases of large residuals, an Anderson Darling test was performed to check if residuals were normally distributed.

5.2.3 Features

Features considered relevant to annotation of CVC related events were the word itself, word window of +/− four words, the stem, possible ICNP matches and synonymous of tokens, see 5.4 for an overview of feature combinations included. The word itself always has to be part of the NER process and is therefore not analysed as a separate feature.

A word window of +-4 words were applied. The stemmer was based on a Snowball stemmer.

For synonyms, all words were converted to lowercase. The top ranked word from the combined models was used as synonym for query words. When combining the word similarity scores from each model, these were averaged and normalized. Minimum term frequency of synonym candidates was set to 50. Words with a synonym value larger than 2 were not included and the token was given the synonym "‘-"’.

Search for ICNP matches was performed sentence by sentence so that a possible ICNP match was returned for a whole sentence. All words of a sentence were then given the same ICNP axis in the CRF template file. The match value for the ICNP axis had to be >0.35. Axes with a lower value were not included, and these tokens were given the axis "‘-"’.

Experiments were performed with bigram features (taking the previous output into account).

Chapter 6

Results

A search in the annotation files yielded information about the match frequency of each annotation label in the 4533 clinical notes, shown in table 6.1.

The factorial analysis of f-measures indicated significant effects of "'word window"' (F=963.38, p<0.001), "'ICNP axis"' (F=199.30, p<0.001) and an interaction effect between these (F= 176.61, p<0.001), see pareto chart in figure 6.1. Also, the effect of blocks was significant (F=31.57, p<0.001), indicating that there were significant differences between the three cross validation runs. Main effects plots, see figure 6.2 illustrate the differences in f-measures between presence and absence of the different features. Interaction plots illustrate interactions between factors, see 6.3. Parallel lines illustrate no interaction. A Difference between lines in an interaction plot illustrates interaction. Thus, an interaction between word window and ICNP axis is present.

Large residuals were found for two of the observations, indicating that these were lower than expected by the regression model. These two were observation number 2 (f-measure=18.43, residual=-4.42) and 14 (f-measure=17.21, residual=-4.61). An Anderson-Darling test on stored

Table 6.1: Frequency of each label in the clinical notes

Label	Frequency
Carecvc	392 matches in 341 files
PlanCarecvc	58 matches in 52 files
PlanInscvc	96 matches in 77 files
Inscvc	86 matches in 73 files
Remcvc	65 matches in 54 files
PlanRemvcvc	28 matches in 21 files
Symptom	143 matches in 105 files
Sepsis	78 matches in 38 files
Device: CVC, Hickmann, VAP, other	0 matches in 0 files
Site: Jugular, subclavian, femoralis	0 matches in 0 files
Possiblecvc	55 matches in 36 files.
CVC	44 matches in 40 files.

residuals indicated that the residuals were not significantly different from a normal distribution (AD = 0.458, p = 0.253).

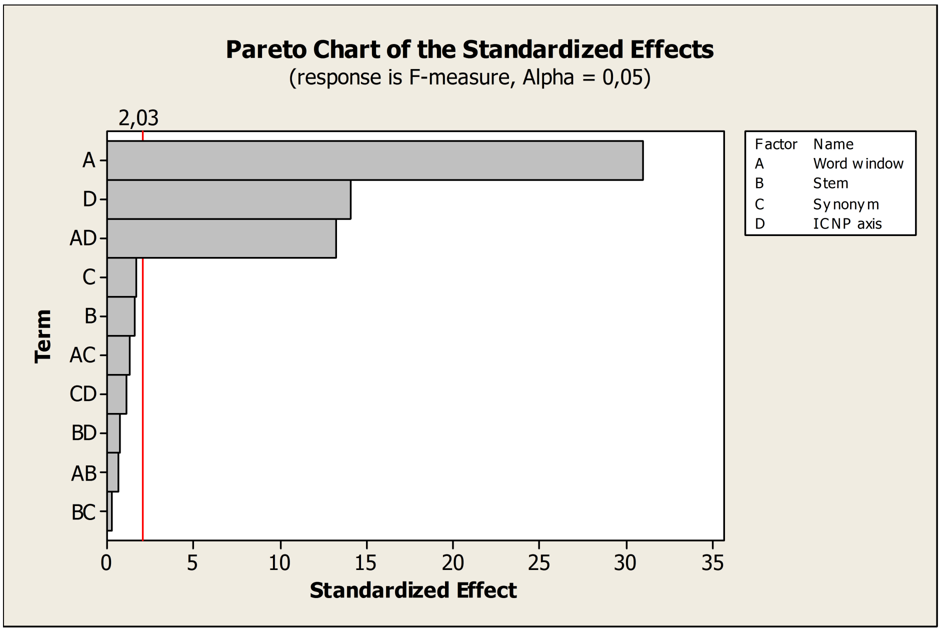

FIGURE 6.1: The chart displays the absolute values of the effects. Effects passing the reference line are significant.

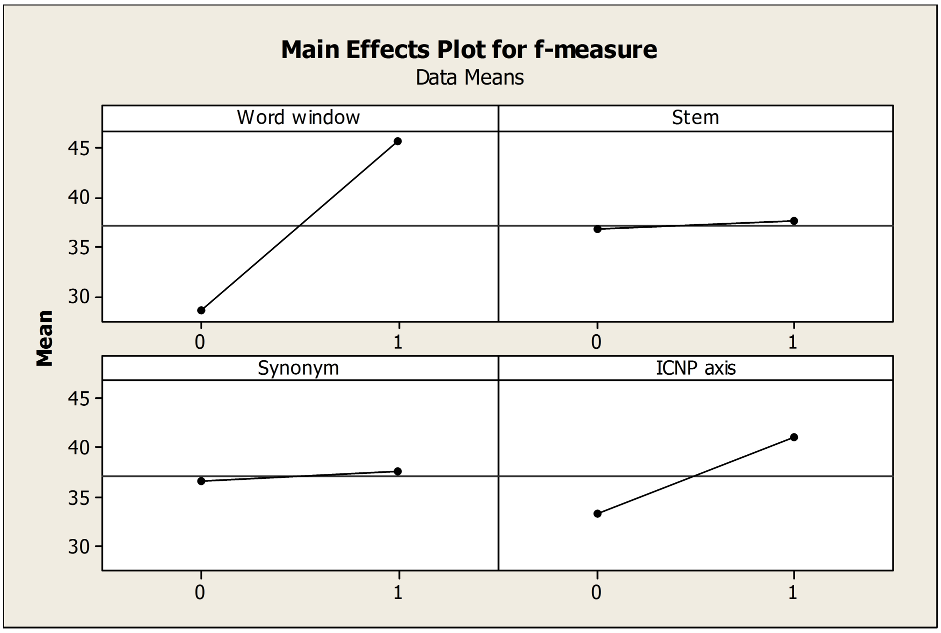

FIGURE 6.2: The figure displays how the presence of the four main effects, word window, stem, synonym and ICNP axis effect the f-measures when they are either on (1) or off (-1). Word window has the steepest slope between being present and not to being present and thus constitutes the greatest effect on the f-measure. Also ICNP axis has a significant effect.

A corresponding analysis regarding recall indicated significant effects of word window (F=761,85, p=0,000), ICNP axis (F=85,21, p=0,000), stem (F=4,79, p=0,035) and the interaction between word window and ICNP axis (F= 76,02, p=0,000), see pareto plot in figure 6.4. Blocks were significantly different from each other (F=40,65, p=0,000). For main effects plot and interaction

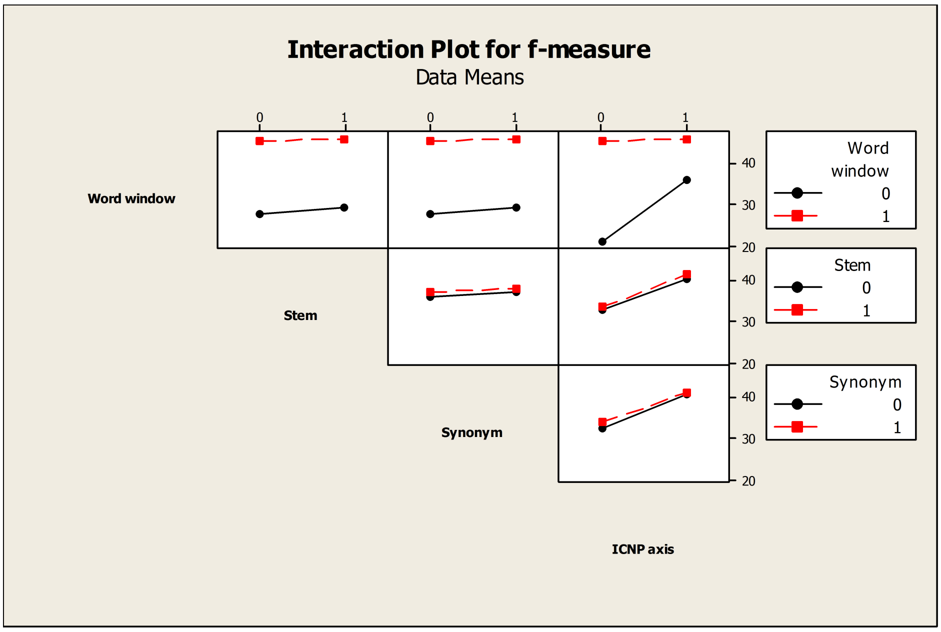

FIGURE 6.3: The figure displays interactions between features. An interaction effect exists between word window and ICNP axis.

plots, see figure 6.5 and figure 6.6. Three observations were a bit lower than expected by the regression model, but an Anderson darling test indicated that the data were not significantly different from a normal distribution (AD=0,634, p=0,093).

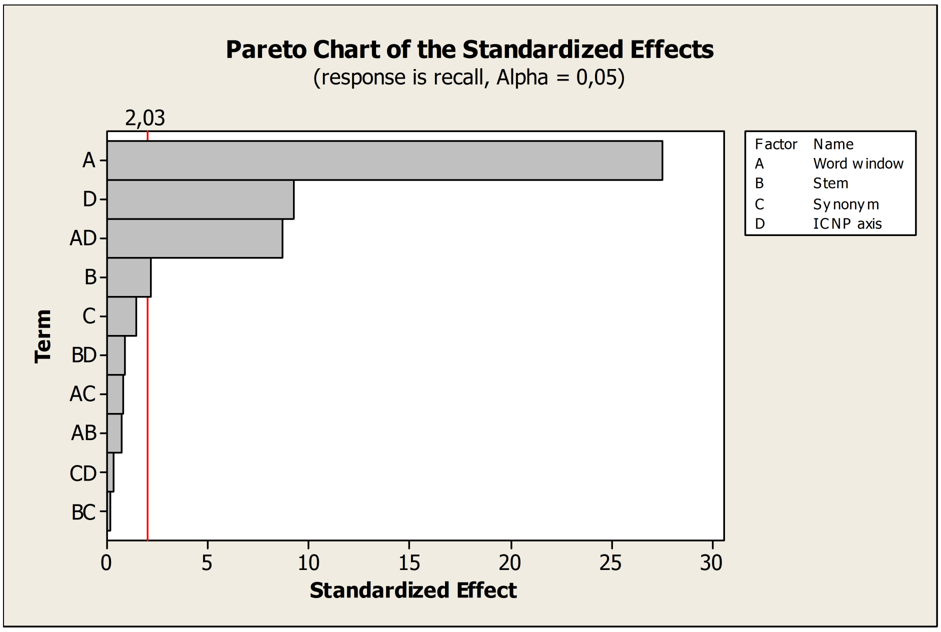

FIGURE 6.4: The chart displays the absolute values of the effects.

The analysis of precision indicated significant effects of word window (F=1093.73, p<0.001) and ICNP axis (F=459.29, p<0.001). Two interaction effects were also found, between word window and ICNP axis (F=405.88, p<0.001) and between synonymous and ICNP axis (F=5.43, p=0.026). Also for the precision analysis the effect of blocks was significant (F=19.59, p<0.001). Pareto chart for precision is shown in figure 6.7, main effects plot in figure 6.8 and interaction

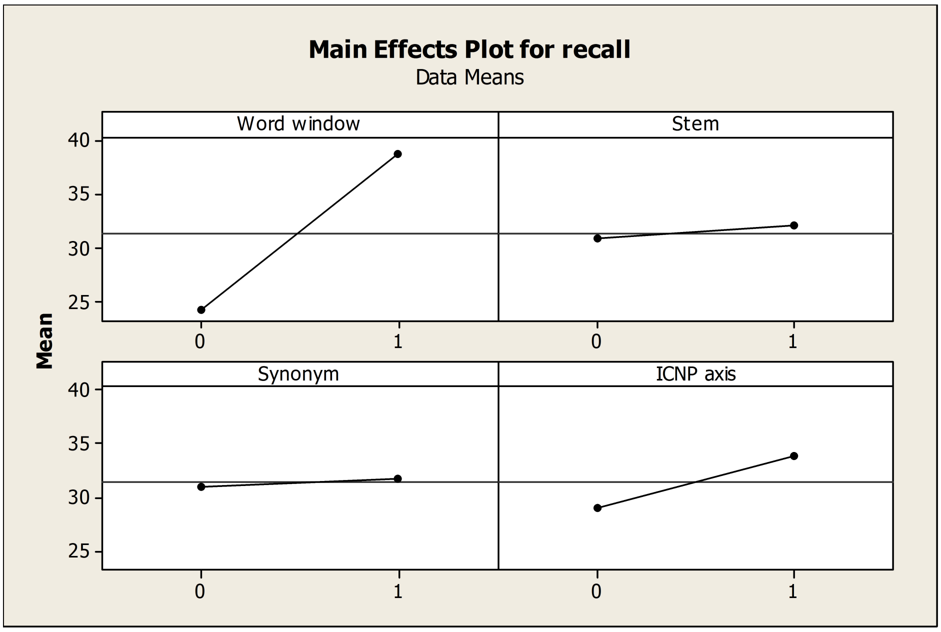

Figure 6.5: The figure displays how the presence of the four main effects, word window, stem, synonym and ICNP axis effect recall when they are either on (1) or off (-1). As for the f-measure analysis, word window and ICNP axis had a significant effect. Stem also has a significant effect in this analysis and the slope for stem is therefore steeper in this analysis then in the f-measure analysis.

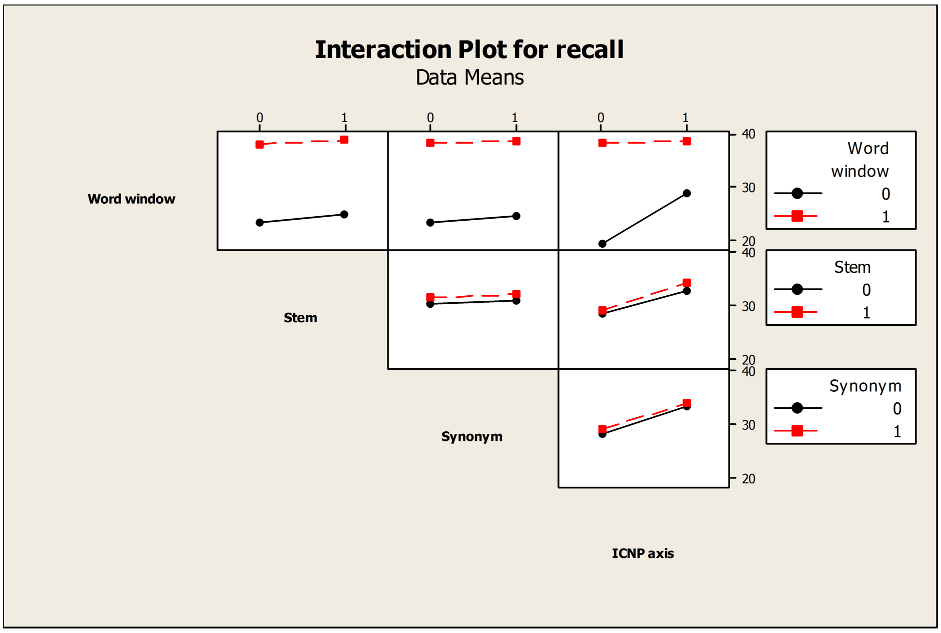

Figure 6.6: The figure displays interactions between features. An interaction effect exists between word window and ICNP axis.

plots in figure 6.9. Two observations were lower than predicted by the regression model, but an Anderson Darling test indicated that also for this analysis, the residuals were not significantly different from a normal distribution (AD=0.652, p=0.139).

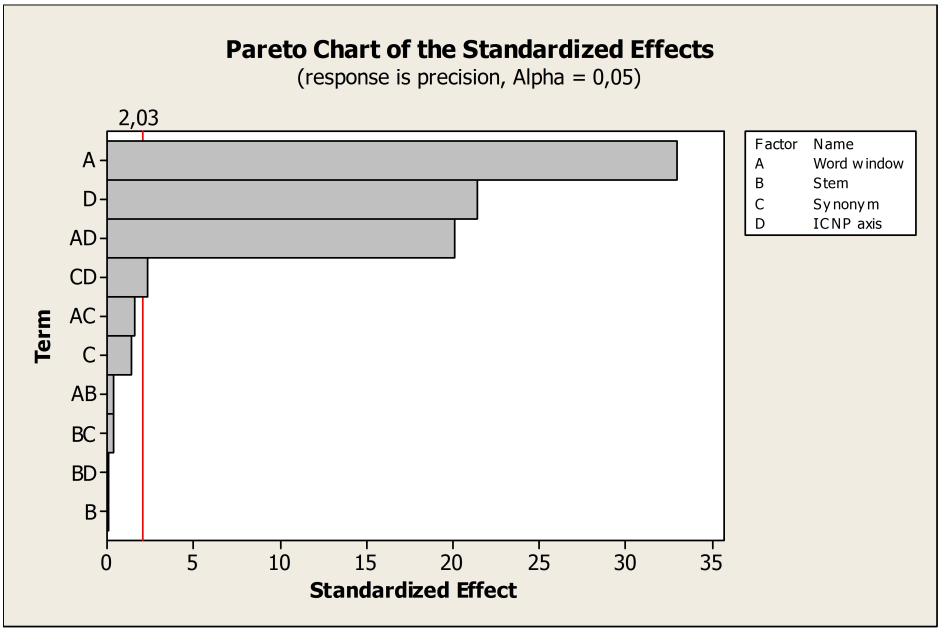

FIGURE 6.7: The chart displays the absolute values of the effects. Effects passing the reference line are significant. Differently from the analysis of recall is that the interaction between synonymous and ICNP axis is significant, and that the other significant effects apparently have a greater effect on precision then on recall.

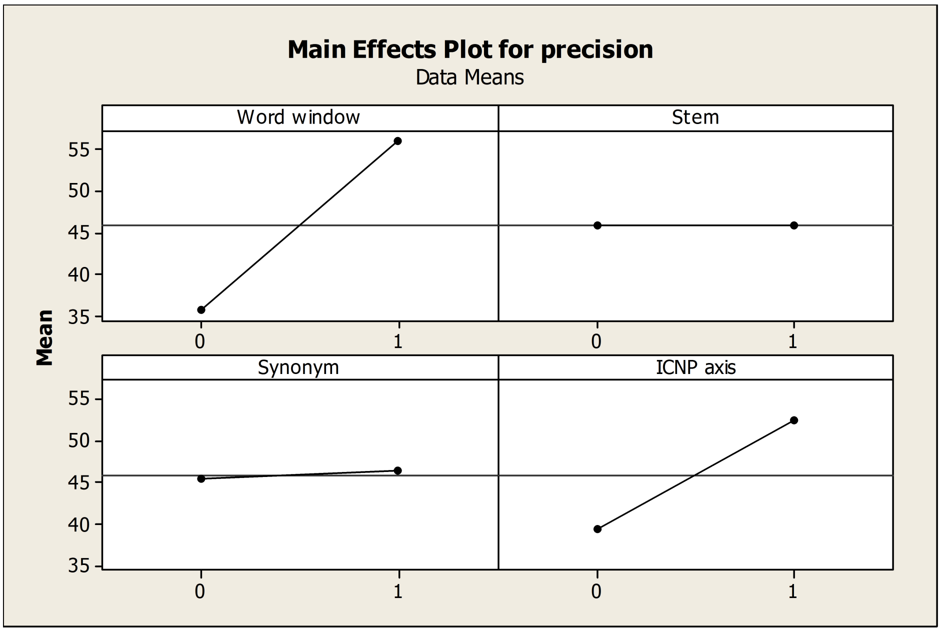

FIGURE 6.8: The figure displays how the presence of the four main effects, word window, stem, synonym and ICNP axis effect precision when they are either on (1) or off (-1). Stem and synonymous seem to have no main effect on precision.

The lowest result scores were the scores of the baseline feature set where none of the four features were on. Only the token itself was used as input for the algorithm in these experiment runs.

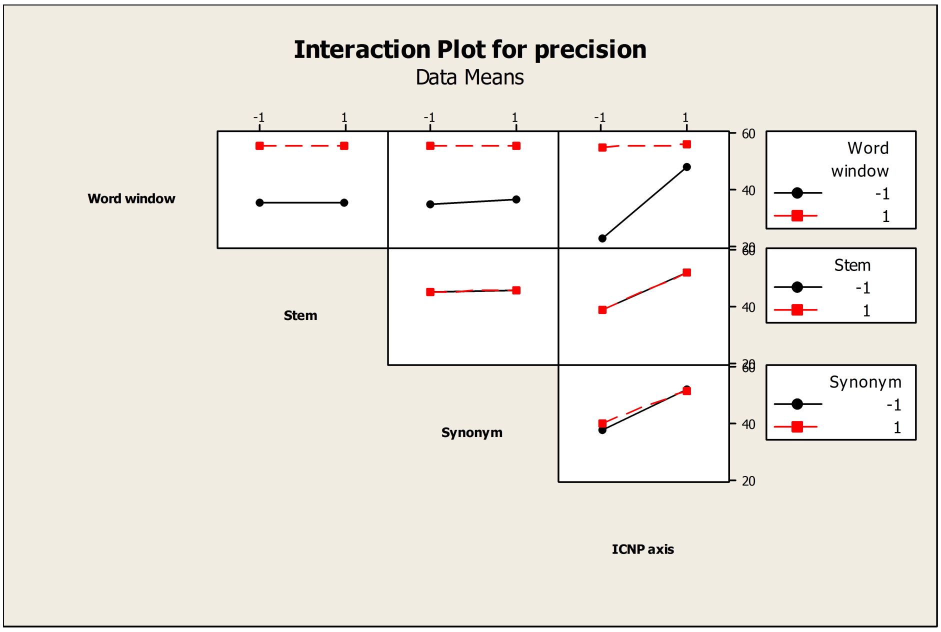

Figure 6.9: An interaction effect exists between word window and ICNP axis. Also, an interaction effect exist between synonymous and ICNP axis.

Table 6.2: Results for each category - baseline

Category	Precision	Recall	$F_{\beta=1}$
CVC	0.00%	0.00%	0.00
CareCVC	25.10%	27.96%	26.38
Hickman	0.00%	0.00%	0.00
Inscvc	27.63%	23.99%	24.83
PlanCarecvc	6.94%	3.30%	4.37
PlanInscvc	29.43%	21.71%	24.44
PlanRemcvc	0.00%	0.00%	0.00
PossibleCVC	14.14%	3.97%	6.00
Remcvc	13.47%	6.11%	8.34
Sepsis	28.93%	10.36%	14.54
Symptom	19.51%	10.74%	13.21
Overall	20.21%	18.47 %	19.27

The baseline result set is shown in table 6.2. The best F-score was obtained when all features were included. Table 6.3 shows the results for each annotation category for the template where all features were included. Table 6.2 and table 6.3 show the average values of the three cross-validation runs. After finding the best feature set some experimentation was performed by editing the word window. It was found that decreasing the word window to +-2 had no effect on the results; precision 56.51, recall 39.44 and F-score=46.44.

Table 6.3: Results for each category when all of the four features were included

Category	Precision	Recall	$F_{\beta=1}$
CVC	0.00%	0.00%	0.00
CareCVC	57.26%	61.54%	59.29
Hickman	100.00%	100.00%	100.00
Inscvc	47.16%	29.39%	35.85
PlanCarecvc	48.33%	12.87%	20.17
PlanInscvc	62.03%	48.06%	54.14
PlanRemcvc	38.89%	13.89%	20.45
PossibleCVC	63.33%	18.76%	28.78
Remcvc	25.72%	8.40%	12.62
Sepsis	35.56%	8.30%	13.46
Symptom	64.68%	24.53%	35.55
Overall	56.29%	39.4%	46.33

Chapter 7

Discussion

7.1 Clinical relevance of the study

The current study is relevant because no previous studies have investigated the possibility of using NER to detect CVC events on Norwegian clinical notes. Successfully detecting such events can contribute to making decision support based on already existing data. Detecting adverse events related to CVC may be both time-saving and economic. Also, real-time detection of potential risks may prevent AEs [6]. A corpus with annotations of CVC related events was made by Tvedt, since no such corpus existed previously for the Norwegian language.

Having a Norwegian corpus was important because Norwegian differs in many respects from English. The study can be compared to similar studies for Swedish since the Scandinavian languages may be considered dialects of the same language [60]. As in Swedish, compound words are very common in Norwegian. Other specialties of Norwegian clinical notes are that there are two written languages that are only slightly different from each other; they often contain self-made abbreviations and incomplete sentences.

7.2 Methods

7.2.1 Data selection

Identification of the number of patient days with CVC was important to Ahus. To identify this, all types of clinical notes were considered important for the study because all of these could contain information about CVC and CRBSI. As an example, care during CVC is probably noted in the nurse's notes, while infections and blood culture test results may be noted in physician's notes or laboratory notes.

An advantage of applying all types of clinical notes was that the amount of training and test data was increased. This may be positive because it increases the amount of training data. A possible drawback is that the variation in the data may also increase. Thus, the ambiguity in the data may increase and the effect may be that even more data would be required in order to compensate for the variation.

The effect of including or excluding some types of clinical notes in the study could not be predicted in advance. Another possibility could have been to include only nurse's notes since nurses often do both the insertion, care and the removal of CVC. However, information about CRBSI and blood cultures could also be of importance if for instance removal of CVC was not properly documented. Once CRBSI is suspected or blood cultures are tested, laboratory and physician's notes could also be of relevance. It was therefore decided to include all types of notes so that no information possibly relevant to the study was missed.

As described in the method section 5.1, records from patients not registered with CVC were excluded from the study. In future studies it might be interesting to include additional records from such patients. The records from patients that did not have CVC had not been manually read at the time when the present study had started.

Since the present study is focused on understanding sentences and words, thus not focusing on separating clinical notes from each other, it is unlikely that excluding non-CVC patients would affect the study in any way. If the amount of clinical notes had been greater, the percentage of CVC related words would be lower and therefore less relevant to the CRF algorithm. Applying notes from CVC patients only is therefore reasonable in the present study to ensure a large enough amount of CVC related words to discover how different features effect the named entity recognition.

Ideally, the sample should have been larger, but because the clinical notes had to be manually read and annotated, the sample size was limited by the time interval of the project. The selection of data was based on a prevalence survey rather than on a search in records. Advantages of selecting the data based on the prevalence survey was that with the clinical notes, it is more difficult to ascertain which patients that had CVC if the documentation was unstructured. There is a risk that even though a patient is registered with CVC in the prevalence survey, he/she may have no trace of CVC related events in the clinical notes. However, the risk that none of his/her clinical notes contains CVC related events seems unlikely. Thus, selection of data based on prevalence surveys seems like a good approach when the goal is to have some control of whether the notes are from CVC patients.

Another possible weakness of the selection method was that manually annotated records were selected based on a prevalence survey that is independent of the EHR system. That implies that even if a patient was registered as a CVC patient in the prevalence survey, this does not guarantee that CVC events are documented in the patient's record.

7.2.2 Evaluation of Method

At the start of the project an annotated corpus was not available, and therefore both unsupervised, semi- and supervised learning algorithms were considered. However, during the project

4533 clinical notes were annotated (by Tvedt), and it was therefore decided to utilize this annotated corpus. With an annotated corpus available, a supervised learning algorithm was a natural choice since these types of algorithms utilize annotated data for training.

Different methods for sampling and evaluation were considered for the study. Hold-out validation such as selecting 20% of the data for testing and 80% of the data for training, has been used by several authors in the literature and seemed like a time efficient and simple evaluation algorithm. With 16 different experiments time efficiency was considered important in order to make the experiment runs achievable.

Some initial experiments were performed, applying the above-mentioned evaluation method. 20 % of the data were randomly selected from the whole data set and spared for testing, while the remaining 80 % were used for training. The results of these test runs indicated that the number of different label types became very sparse in the test set.

Thus, to decrease the risk of a non-representative training and test sample, cross-validation was considered. Replications of experimental runs also contribute to increase the statistical power of the design. As mentioned in the background chapter, 10-fold cross validation is one of the standard evaluation methods used in text mining. Having 10 iterations for each experiment would increase the amount of data used for training, as well as increasing the number of estimates of the model's performance.

However, each experiment iteration took about half an hour. Consequently, running 10-fold cross-validation for one single experiment would require 5 hours in the laboratory. With 16 different experiments, this was considered too time-consuming. 10-fold cross-validation would also, as mentioned in the background chapter, lead to a lot of overlap in the training data and small test-sets. 3-fold cross-validation was considered a satisfactory compromise in order to utilize the benefits of cross-validation and at the same time make each experiment iteration achievable within a reasonable time.

Thus, a 3-fold cross-validation was chosen. The next challenge was to decide whether the same 3 folds should be used for each experiment or if the random sampling into the three different segments should be rerun for each experiment. Since an important goal of running different experiments was to decide which parameters that were best suited for event detection, it was considered important that other variables than the feature sets were kept static. Thus, the same three folds were used for each experiment. Another risk was that the results of the different experiments could be dependent on how the three fold partitioning was performed. However, this risk was minimized by applying a randomized selection method.

7.3 Choice of annotation labels

As mentioned in the section about motivation for the project 1.3, one of the goals of Ahus was to obtain a better overview of the number of patient days with CVC. It was therefore important to annotate insertion, removal and care of CVC. Symptoms and specific CVC types were also of importance because insertion and removal were sometimes not registered in the journals.

However, the results of the manual annotation indicated that some of the annotation labels, such as insertion site, were too specific as these were not applied to any of the phrases in the medical collection. Also, the "'CVC"' annotation label had very few matches and partly overlapped with some of the other annotation labels. Planning a removal of CVC had only 28 matches. In cases of overlapping labels, the anntoconll tool chose the annotation label that contained the largest proportion of characters and discarded the other.

As expected, CareCVC had a higher number of matches than the other annotation labels since care is performed and documented every day, while for instance planning a removal of CVC occurs less frequently. Some of the annotation labels, such as planning a removal of CVC had few matches. This may have influenced training and testing if the number of an annotation label was partitioned in such a way that the number of training or testing entities was too low, or if the partitioning of each annotation label to the three cross folds was skewed. The distribution of files to the three folders used in the three fold cross-validation was performed randomly, because this decreases the risk of a skewed partitioning.

Ambiguous annotation labels may have influenced the results negatively. The number of annotation labels was 11, which is much higher than in similar studies. For instance, the studies by Skeppstedt [36, 63] contained only three and four categories respectively, and the study by Penz, [6], contained four categories. Also, the studies by Skeppstedt defined categories corresponding to the SNOMED CT; disorder, finding, body structure and pharmaceutical drug.

Even though they found some ambiguity between disorder and finding, these categories seemed less overlapping than the categories presently applied. As an example it seems reasonable to believe that some ambiguity may exist between the categories PossibleCVC and Symptom. Tests for overlap have not been developed, but some overlapping annotations (<10) were discovered when the anntoconll script was run. This script chose the annotation that covered the largest amount of characters in the clinical notes. Skeppstedt [63] developed clear guidelines for annotations for each category. The guidelines were controlled by two physicians. Also, two annotators were applied and inter-annotator agreement was calculated. Such controls decrease the risk of ambiguous and overlapping categories, as well as decreasing the risk of erroneous annotations.

Since several annotators were not available for the present study, such controls could not be made. However, a strength in the selection of categories is that they cover a large spectre of CVC events and that they have been made in cooperation with a domain expert in NLP, with a wide experience in making categorization and ontologies. A possibility for future studies is to use the CVC event labels as features rather than as labels to a classifier. That is, making broader and more general classification categories, such as "'CVC"' and "'not CVC"', and applying the present annotation labels as features of tokens to be classified into the broader categories.

Discussions with Haldor Husby from Ahus 3 weeks ago revealed that unfortunately errors may exist in the manual annotation. In the corpus generated at Akershus University Hospital, which has provided the data for the present study, the predefined definitions for the annotation labels had not been followed. CVC and Hickmann were both sub categories of Device according to the definitions of the annotation labels, but both Device, Hickmann and CVC had been used as labels in the manual annotation. As an example, a phrase containing "'Hickmann"' could be annotated as "'Hickmann"' in some cases and as "'Device"' in other cases. It was also discovered a note containing a phrase that should have been tagged with "'Hickmann"' but erroneously had been tagged with "'CVC"'.

Furthermore, some of the clinical notes seemed to be summary notes, as they contained a large amount of annotations regarding both insertion, care and removal of CVC. Somatic supervisory notes (Tilsynsnotat somatikk) were examples of such a type of note. Such summary notes should perhaps be excluded in future research as they may contain repetition of other notes. Repetitive text may cause the model to be trained too much on some word sequences.

However summary notes may summarize all types of CVC events, such that each event type will be repeated an equal number of times. Therefore it might not influence the probability of the different event types for the model. Future researchers should be cautioned that label types such as PossibleCVC may not be mentioned in the summary notes. Accordingly, it seems preferable to exclude summary notes altogether.

Another important source of error regarding the data generated at Ahus was discovered at the same time. Some of the 4533 clinical notes looked like duplicates. It was discovered that every time a clinical note is reopened, a new clinical note is generated. Therefore, some of the 4533 files are different versions of the same clinical note. This may have effected training so that words or sentences that are present in the start of a clinical note is read several times by the CRF algorithm since these are present in all the different versions of a clinical note.

Because of time limitations of the present study it has not been possible to investigate how widespread this problem is. However, it seems unlikely that it was very widespread as it was not discovered until the end of the project, it was not detected even after looking through several notes during conversion of data to conll format at an earlier stage. Also, nurses rarely edit clinical notes (personal communication by nurse and Phd student Tvedt with expertise in this field).

However, 4533 clinical notes may seem a lot for 28 patients, even though each patient has several notes for each care period. Accordingly, it cannot be precluded that the large amount of notes may be caused by the fact that several versions of the same file may exist. An educated guess may indicate that 15-25 % of the clinical notes have more than one version (personal communication by Tvedt). Further discussion will be made on the assumption that the results are still valid.

7.4 Features

The features applied to classify words were word window, stem, synonymous and ICNP axis. Other features were also considered, such as POS-tags. Velupialli [46] applied POS-tags and lemmas as features. In the present study, stemming was applied instead of lemmatization because the clinical notes contained multiple spelling mistakes, unusual/self-made abbreviations and possibly some notes in other languages such as Swedish. The use of a synonym handler based on RI and RP consider words often used in similar contexts synonyms. Thus, it may be able to recognize that such words have the same semantic meaning. POS-tags were not included because it seemed unlikely that knowing the grammatical function of words would contribute much to finding the CVC event type of a word.

Word window had been successfully used in similar studies and it seemed reasonable that the context of a word would contribute in finding its label since the labels annotated phrases. Velupillai [46] performed a similar study and found that a word window of +-4 words gave the best results. Thus, a context window of +-4 was also applied in this study.

ICNP was applied to find generalized terms for medical words. Other medical collections, such as SNOMED CT, have been applied successfully in other studies[36]. The reason that ICNP was chosen in the current study, was that it consists of nursing terms, and CVC related events are typically performed by nurses.

The four chosen features were considered to be those that would have the greatest effect on detection of CVC related events.

7.5 Models

The resultant scores of the present study are lower than in other studies, such as the study by Abacha & Zweigenbaum, [53], where the f-measure was 77,55. The overall recall was 39.4% and the precision was 56.29% for the best feature set. This results in an overall f-score of 46.33. Several factors may have affected the results negatively. One reason may be that 28 records were insufficient to capture a representative sample of each CVC event type. In comparison, the previously mentioned study by Penz et al., [6], used 365 records that according to CPT and ICD-9 procedure codes contained CVC related events. Fourty-nine of these did not include any note about CVC, even though they were supposed to do so according to the CPT and ICD-9 procedure codes. They were therefore excluded.

Only 56 of the CVC records in their study contained procedure notes describing the placement of CVC. Since so many records lacked notes describing CVC events in that study, it is probable that this problem may also exist in the present study, since in this the selection of records also relied on information outside the EHR system, namely the prevalence survey. Throughout this study, Tvedt has been working on increasing the corpus, so it should be easy to repeat the study with a larger corpus.

The synonym handler could also be a source of error because it was not properly tested in advance. Examples of obvious erroneously synonyms detected was; cardiology for gastro and necrotic for skin. Such errors are probably caused by the use of RI and RP in the synonym handler. Words co-occurring with the same words are assumed similar by these models [20]. The reason for the chosen synonym handler was that it was easily available and that synonym handling based on RI and RP had been successfully applied in other studies; Kanerva et al., [20], used RI for solving the synonym part of the "'Test of English as a Foreign Language"' (TOEFL). The threshold for whether a word should be considered a synonym of a query had to be set by trial and error because no experiments existed for finding the optimal threshold. Finding the optimal threshold for the synonym handler could improve classification. Future studies should also consider trying other synonym handlers. However, a strength of the synonym handler is that the correct form of a word is considered a synonym of a misspelled word. As an example, ”‘sykepleier”’, meaning nurse was noted as the synonym of ”‘sykepleoer”’ which is a misspelled version of ”‘sykepleier.

Even though the performance of the best feature set was not as high as for other studies, significant differences were found between the sixteen different feature combinations. In accordance with several other studies [62, 74], context significantly improved classification performance.

ICNP axis also had a significant effect on performance. ICNP axis has not previously been tested as a feature for Norwegian clinical notes. This feature provides the algorithm with semantic and general general information such as whether a term is a judgment or an action. Providing the algorithm with a feature that specifies generally what a term refers to may help building a model that connects two more specific phrases to the same action, such as care of CVC. The results are similar to the results of Abacha & Zweigenbaum, [53], mentioned in chapter 4.3. They also found satisfactory results when CRF was combined with IOB tags and semantic information.

An interaction effect was also found between word window and ICNP axis, meaning that when these features were applied together the performance was increased more than the sum of both of them alone.

Stem had an effect on recall, but not on precision. The reason for that may be that normalizing words make them more general, making it easier to recognize that two slightly different words should be assigned the same label. However, normalizing the words make them less precise. The different effect of stem on recall and precision can be envisioned by comparing the main effects graphs for stem between recall and precision.

Synonymous apparently had a greater effect for precision then for recall as an interaction effect was found between ICNP axis and synonymous. This also seems reasonable because synonymous make information more detailed and less general.

A significant effect of blocks indicate that the three cross-fold runs were significantly different from each other. The reason for the difference is that each of the three cross-fold runs applied different folder for testing. A skewed partitioning of the different annotation labels to the three cross-folds may be another reason for this result.

There is a great variation in performance for the different annotation labels. Hickmann has an f-score of 100% while CVC has a f-score of 0%. These two extreme values seem a bit odd, perhaps because there were too few phrases annotated with Hickmann. As an example, if there is one phrase annotated with Hickmann in the test set and the model recognizes this one phrase correctly, the scores will be 100%. It seems reasonable that CareCVC obtains a higher f-score than the other categories since this is probably the most often used annotation label and therefore one of the labels that the model have the most training in recognizing.

Further research should include compound handling. As mentioned in chapter 4.1.3, Huseth & Røst, [52], applied the last word in cases of compound words.

7.5.1 Conclusion

This thesis is a study NER for automatic annotation of Norwegian clinical notes mentioning events related to CVC. CRF was applied to train models where different combinations of the features word window, stem, synonymous and ICNP axis were used as input. The results indicate that context and ICNP axis had a significant effect on classification performance. Stem had an effect on recall, but not on precision. Recall, precision and F-score of the best feature set were lower than for similar studies which may be due ambiguous annotation guidelines, errors in the corpus and overlapping annotation labels. Improvements in annotation guidelines and less overlapping annotation labels may improve performance in future studies. Future studies should study the effect of including context of features, improve compound handling and applying less overlapping categories. The categories applied in the present study may also be used as input to a classifier that use broader classification categories such as "'CVC"' and "'not CVC"'.

Bibliography

[1] Evicare. URL https://sites.google.com/site/evicare1/.

[2] Act of 2 july 1999 relating to health personnel, 2013. URL http://lovdata.no/dokument/NL/lov/1999-07-02-64/KAPITTEL_8#%C2%A739.

[3] I. Hojsak, H. Strizic, Z. Misak, I. Rimac, G. Bukovina, H. Prlic, and S. Kolacek. Central venous catheter related sepsis in children on parenteral nutrition: A 21-year single-center experience. *Clin Nutr*, 31:672–675, 2012. doi: 10.1016/j.clnu.2012.02.006.

[4] A. S. Graham, C. Ozment, K. Tegtmeyer, S. Lai, and D. A. V. Braner. Central venous catheterization. *The New England Journal of Medicine*, 356(21):21–23, 2007.

[5] S. M. Meystre, G. K. Savova, K. C. Kipper-Schuler, and J. F. Hurdle. Extracting information from textual documents in the electronic health record: a review of recent research. *Yearbook of medical informatics*, pages 128–144, 2008. ISSN 0943-4747. URL http://view.ncbi.nlm.nih.gov/pubmed/18660887.

[6] J. Penz, A. Wilcox, and J. Hurdle. Automated identification of adverse events related to central venous catheters. *Journal of Biomedical Informatics*, 40(2):174–182, 2007. URL /brokenurl#http://publication.wilsonwong.me/load.php?id=233282194.

[7] Health information technology adoption, programmes and plans: global perspectives. URL http://www.openclinical.org/hitGlobal.html.

[8] V. Heimly, A. Grimsmo, T. P. Henningsen, and A. Faxvaag. Diffusion and use of Electronic Health Record systems in Norway. *Stud Health Technol Inform.*, 160:381–385, 2010.

[9] G. Ellingsen and E. Monteiro. Big is beautiful: electronic patient records in large Norwegian hospitals 1980s-2001. pages 381–385.

[10] G. Ellingsen. *Global reach, local use. Design and use of electronic patient record systems in large hospitals.* PhD thesis, Norwegian University of Science and Technology, Department of Computer and Information Science, 2003.

[11] H. Lærum, G. Ellingsen, and A. Faxvaag. Doctors' use of electronic medical records systems in hospitals: cross sectional survey. *BMJ*, 323:1344–1348, 2001.

[12] S. Fletcher. Catheter-related bloodstream infection. *Oxford Journals*, 5:49–51, 2005. doi: 10.1093/bjaceaccp/mki011.

[13] Genevieve B. Melton and George Hripcsak. Automated Detection of Adverse Events Using Natural Language Processing of Discharge Summaries. *J Am Med Inform Assoc*, 12(4): 448–457, 2005. doi: 10.1197/jamia.M1794. URL http://www.jamia.org/cgi/content/abstract/12/4/448.

[14] Hercules Dalianis, Martin Hassel, and Sumithra Velupillai. The stockholm epr corpus - characteristics and some initial findings. to be published. In *in Proceedings of the 14th International Symposium for Health Information Management Research*, pages 14–16, 2009.

[15] Peter B. Jensen, Lars J. Jensen, and Søren Brunak. Mining electronic health records: towards better research applications and clinical care. *Nat Rev Genet*, 13:395–405, 2012.

[16] Gobinda G. Chowdhury. Natural language processing. *Annual Review of Information Science and Technology*, pages 51–89, 2003.

[17] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008. ISBN 0521865719, 9780521865715.

[18] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, November 1975. ISSN 0001-0782. doi: 10.1145/361219.361220. URL http://doi.acm.org/10.1145/361219.361220.

[19] Magnus Sahlgren. An introduction to random indexing. In *In Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering, TKE 2005*, 2005.

[20] Pentti Kanerva, Jan Kristoferson, and Anders Holst. Random indexing of text samples for latent semantic analysis. In *In Proceedings of the 22nd Annual Conference of the Cognitive Science Society*, pages 103–6. Erlbaum, 2000.

[21] Magnus Sahlgren, Anders Holst, and Pentti Kanerva. Permutations as a means to encode order in word space. In V. Sloutsky, B. Love, and K. Mcrae, editors, *Proceedings of the 30th Annual Conference of the Cognitive Science Society*, pages 1300–1305. Cognitive Science Society, Austin, TX, 2008. URL http://www.sics.se/~{}mange/papers/permutationsCogSci08.pdf.

[22] Aron Henriksson, Hans Moen, Maria Skeppstedt, Ann-Marie Eklund, Vidas Daudaravicius, and Martin Hassel. Synonym extraction of medical terms from clinical text using combinations of word space models. In *5th International Symposium on Semantic Mining in Biomedicine (SMBM), 3rd-4th September, 2012, Zurich*, volume 2012, pages 10–17, 2012. URL http://dx.doi.org/10.5167/uzh-64476.

[23] Jerry R. Hobbs. Information extraction from biomedical text. *J. Biomed Inform.*, pages 260–264, 2002.

[24] Daniel Jurafsky and James H. Martin. *Speech and language processing*. Pearson Education, Inc., Upper Saddle River, New Jersey 07458, 2nd edition, 2009. ISBN 0-13-504196-1.

[25] Nizar Habash and Owen Rambow. Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop, 2005.

[26] *Arabic Morphological Tagging, Diacritization, and Lemmatization Using Lexeme Models and Feature Ranking*, Columbus, Ohio, 06/2008 2008. Association for Computational Linguisitics. URL http://www.aclweb.org/anthology/P/P08/P08-2030.

[27] Tuomo Korenius, Jorma Laurikkala, Kalervo Järvelin, and Martti Juhola. Stemming and lemmatization in the clustering of finnish text documents. pages 625–633, 2004.

[28] Jung-wei Fan, Rashmi Prasad, Rommel M. Yabut, Richard M. Lommis, Daniel S. Ziook, John E. Mattison, and Yang Huang. Part-of-speech tagging for clinical text: wall or bridge between institutions? *AMIA Annu Symp Proc.*, page 382–391, 2011.

[29] Oslo-bergen taggeren. URL http://tekstlab.uio.no/obt-ny/index.html.

[30] Ralph Grishman and Beth Sundheim. Message understanding conference-6: a brief history. In *Proceedings of the 16th conference on Computational linguistics - Volume 1*, COLING '96, pages 466–471, Stroudsburg, PA, USA, 1996. Association for Computational Linguistics. doi: 10.3115/992628.992709. URL http://dx.doi.org/10.3115/992628.992709.

[31] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Linguisticae Investigationes*, 30(1):3–26, January 2007. URL http://www.ingentaconnect.com/content/jbp/li/2007/00000030/00000001/art00002. Publisher: John Benjamins Publishing Company.

[32] Icd-10, June 2013. URL http://www.helsedirektoratet.no/kvalitet-planlegging/helsefaglige-kodeverk/icd-10/sider/default.aspx.

[33] Store medisinske leksikon, icd-10, June 2013. URL http://sml.snl.no/ICD-10.

[34] International classification of diseases (icd), June 2013. URL http://www.who.int/classifications/icd/en/.

[35] Icd-10, June 2013. URL http://sml.snl.no/ICD-10.

[36] Maria Skeppstedt, Maria Kvist, and Hercules Dalianis. Rule-based entity recognition and coverage of snomed ct in swedish clinical text. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may 2012. European Language Resources Association (ELRA). ISBN 978-2-9517408-7-7.

[37] S. E. Aasen and M. F. Nylund. Opprydding i begrepsjungelen. *Tidsskrift for Den norske legeforeningen*, (23), December 2012. doi: 10.4045/tidsskr.12.0968.

[38] Vladimir M. Krasnopolsky and Michael S. Fox-Rabinovitz. 2006 special issue: Complex hybrid models combining deterministic and machine learning components for numerical climate modeling and weather prediction. *Neural Netw.*, 19(2):122–134, March 2006. ISSN 0893-6080. doi: 10.1016/j.neunet.2006.01.002. URL http://dx.doi.org/10.1016/j.neunet.2006.01.002.

[39] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. ISBN 1-55860-778-1. URL http://dl.acm.org/citation.cfm?id=645530.655813.

[40] Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286, 1989.

[41] Charles M. Grinstead and Laurie J. Snell. *Grinstead and Snell's Introduction to Probability*. American Mathematical Society, version dated 4 july 2006 edition, 2006. URL http://math.dartmouth.edu/~{}prob/prob/prob.pdf.

[42] W. H. Majoros. *Conditional Random Fields.*, chapter Supplement 1, pages 257–286. Cambridge University Press, 2007.

[43] Natalia Ponomareva, Paolo Rosso, Ferran Pla, and Antonio Molina. Conditional random fields vs. hidden markov models in a biomedical named entity recognition task. In *Proc. of Int. Conf. Recent Advances in Natural Language Processing, RANLP*, pages 479–483, 2007. URL http://users.dsic.upv.es/~prosso/resources/PonomarevaEtAl_RANLP07.pdf.

[44] D. Li, K. Kipper-Schuler, and G. Savova. Conditional random fields and support vector machines for disorder named entity recognition in clinical texts. In *Proceedings of the HLT Workshop on Current Trends in Biomedical Natural Language Processing*, Ohio, USA, 2008.

[45] Brat rapid annotation tool. URL http://brat.nlplab.org/.

[46] Sumithra Velupillai. *Shades of certainty - Annotation and Classification of Swedish Medical Records*. PhD thesis, Stockhold University, 2012.

[47] David M. W. Powers. Evaluation: From Precision, Recall and F-measure to ROC, Informedness, Markedness & Correlation. *Journal of Machine Learning Technologies*, 2:37–63, 2007.

[48] Dimitrios Kokkinakis. Developing resources for swedish bio-medical text mining, 2006.

[49] Payam Refaeilzadeh, Lei Tang, and Huan Liu. Cross-validation. In Ling Liu and M. Tamer Özsu, editors, *Encyclopedia of Database Systems*, pages 532–538. Springer US, 2009. ISBN 978-0-387-39940-9. URL http://dblp.uni-trier.de/db/reference/db/c.html#RefaeilzadehTL09.

[50] Lars-Erik Bruce. Ontology-driven information extraction and structuring in the clinical domain. Master's thesis, University of Oslo, 2012.

[51] i2b2, February 2014. URL https://www.i2b2.org/.

[52] O. Huseth and T. B. Røst. Developing an annotated corpus of patient histories from the primary care health record.

[53] Asma Ben Abacha and Pierre Zweigenbaum. Medical entity recognition: A comparison of semantic and statistical methods. In *Proceedings of BioNLP 2011 Workshop*, BioNLP '11, pages 56–64, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-932432-91-6. URL http://dl.acm.org/citation.cfm?id=2002902.2002911.

[54] G. B. Melton and G. Hripcsak. Automated detection of adverse events using natural language processing of discharge summaries. *J Am Med Inform Assoc.*, 12:448–457, 2005. doi: 10.1197/jamia.M1794.

[55] X. Wang, G. Hripcsak, M. Markatou, and C. Friedman. Active computerized pharmacovigilance using natural language processing, statistics, and electronic health records: a feasibility study. *J Am Med Inform Assoc.*, 16:328–337, 2009. doi: 10.1197/jamia.M3028.

[56] D. W. Bates, S. Evans, H. Murff, P. D. Stetson, L. PizziFerri, and G. Hripcsak. Detecting adverse events using information technology. *J Am Med Inform Assoc.*, 10:115–127, 2003. doi: 10.1197/jamia.M1074.

[57] What is a serious adverse event? URL http://www.fda.gov/safety/medwatch/howtoreport/ucm053087.htm.

[58] Adverse event information. URL http://indigo.gcrc.sunysb.edu/aeinfo.aspx.

[59] Harsha Gurulingappa, Juliane Fluck, Martin Hofmann-Apitius, and Luca Toldo. identification of adverse drug event assertive sentences in medical case reports. In *ECML PKDD 2011 Workshop on Knowledge Discovery in Health Care and Medicine*, pages 16–27, 2011.

[60] Janne Bondi Johannesen, Kristin Hagen, Åsne Haaland, Björk Jónsdottir, Anders Nøklestad, and Dimitris et al. Kokkinakis. Named entity recognition for the mainland scandinavian languages. *Literary and Linguistic Computing*, 20(1):91, 2005.

[61] Hercules Dalianis and Sumithra Velupillai. How certain are clinical assessments? annotating swedish clinical text for (un)certainties, speculations and negations. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may 2010. European Language Resources Association (ELRA). ISBN 2-9517408-6-7.

[62] Sumithra Velupillai. Automatic Classification of Factuality Levels – A Case Study on Swedish Diagnoses and the Impact of Local Context. In *Proc. The Fourth International Symposium on Languages in Biology and Medicine – LBM 2011*, Singapore, December 2011. URL http://people.dsv.su.se/~sumithra/publications/LBM2011/lbm_factuality_velupillai_camera_ready3.pdf.

[63] Maria Skeppstedt, Maria Kvist, Gunnar H. Nilsson, and Hercules Dalianis. Automatic recognition of disorders, findings, pharmaceuticals and body structures from clinical text: An annotation and machine learning study. *Journal of Biomedical Informatics*, (0):–, 2014. ISSN 1532-0464. doi: http://dx.doi.org/10.1016/j.jbi.2014.01.012. URL http://www.sciencedirect.com/science/article/pii/S1532046414000148.

[64] Vijayaraghavan Bashyam and Ricky K. Taira. Incorporating syntactic dependency information towards improved coding of lengthy medical concepts in clinical reports. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*, BioNLP '09, pages 125–132, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. ISBN 978-1-932432-30-5. URL http://dl.acm.org/citation.cfm?id=1572364.1572382.

[65] Kunnskapssenteret, June 2013. URL http://www.kunnskapssenteret.no/.

[66] Dips. URL http://www.dips.no/.

[67] Crf++. URL http://crfpp.googlecode.com/svn/trunk/doc/index.html?source=navbar.

[68] Conll-2012 shared task. URL http://conll.cemantix.org/2012/data.html.

[69] Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. brat: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations Session at EACL 2012*, Avignon, France, April 2012. Association for Computational Linguistics.

[70] Erwin Marsi, Hans Moen, Lars Bungum, Gleb Sizov, Björn Gambäck, and André Lynum. Ntnu-core: Combining strong features for semantic similarity.

[71] Hans Moen, Erwin Marsi, and Björn Gambäck. Towards dynamic word sense discrimination with random indexing. *ACL 2013*, page 83, 2013.

[72] Rakesh Ch. Balabantaray, Suprava Das, and Kshirabdhi T. Mishra. Case Study of Named Entity Recognition in Odia Using Crf++ Tool. *IJACSA*, 4(6):213–216, 2013.

[73] Minitab. *version 16.0*. Minitab Inc., 2010.

[74] Bryan Rink, Sanda M. Harabagiu, and Kirk Roberts. Automatic extraction of relations between medical concepts in clinical texts. *JAMIA*, 18(5):594–600, 2011. URL http://dblp.uni-trier.de/db/journals/jamia/jamia18.html#RinkHR11.