Using Natural Language Processing to Improve Biomedical Concept Normalization and Relation Mining

Ning Kang

The studies presented in this thesis were financially supported by:
Erasmus MC Rotterdam
European Commission FP7 Program (FP7/2007-2013) under grant no. 231727 (the CALBC Project)

The printing of this thesis was financially supported by:
Erasmus University Rotterdam
Department of Medical Informatics, Erasmus MC
J.E. Jurriaanse Stichting

ISBN: 978-90-6464-691-1
Layout: Ning Kang
Printed by: GVO drukkers & vormgevers B.V

USING NATURAL LANGUAGE PROCESSING TO IMPROVE BIOMEDICAL CONCEPT NORMALIZATION AND RELATION MINING

Gebruik van natuurlijke taalverwerking om biomedische conceptherkenning en relatie-extractie te verbeteren

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Erasmus Universiteit Rotterdam
op gezag van de
rector magnificus

Prof.dr. H.G. Schmidt

en volgens besluit van het College voor Promoties

De openbare verdediging zal plaatsvinden op
woensdag 18 september 2013 om 15.30 uur

door

Ning Kang

geboren te Yinchuan, China

PROMOTIECOMMISSIE

Promotor:
Prof.dr. J. van der Lei

Overige leden:
Prof.dr. B. Mons
Prof.dr.ir. G.W. Jenster
Prof.dr. U. Hahn

Copromotoren:
Dr.ir. J.A. Kors
Dr. E.M. van Mulligen

Dedicated to my parents, my wife, and my daughter

谨以此博士论文献给我的父母，妻子，和我亲爱的女儿们

TABLE OF CONTENTS

CHAPTERS IN THIS THESIS ARE BASED ON THE FOLLOWING PUBLICATIONS:

Chapter 2

Ning Kang, Erik M. van Mulligen, and Jan A. Kors. Comparing and combining chunkers of biomedical text. Journal of Biomedical Informatics, 2011;44:354–60

Chapter 3

Ning Kang, Erik M. van Mulligen, and Jan A. Kors. Training text chunkers on a silver standard corpus: can silver replace gold?. BMC Bioinformatics, 2012; 13:17

Chapter 4

Ning Kang, Zubair Afzal, Bharat Singh, Erik M. van Mulligen, and Jan A. Kors. Using an ensemble system to improve concept extraction from clinical records. Journal of Biomedical Informatics, 2012:45(3):423-8

Chapter 5

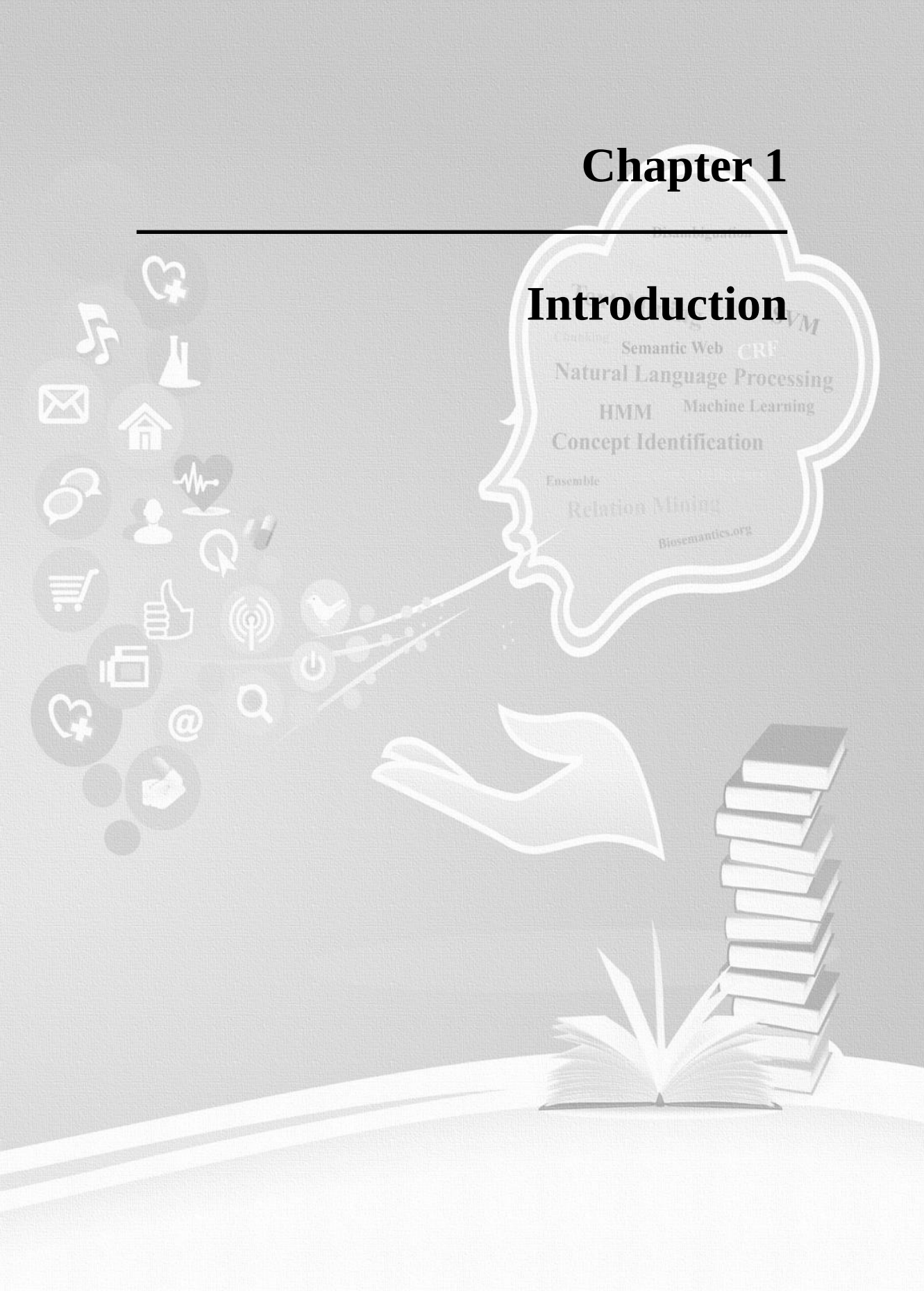
Ning Kang, Bharat Singh, Zubair Afzal, Erik M. van Mulligen, and Jan A. Kors. Using rule-based natural language processing to improve disease normalization in biomedical text. Journal of the American Medical Informatics Association, 2012, doi:10.1136/amiajnl-2012-001173

Chapter 6

Ning Kang, Bharat Singh, Chinh Bui, Zubair Afzal, Erik M. van Mulligen, and Jan A. Kors. Knowledge-based extraction of adverse drug events from biomedical text. (Submitted)

Chapter 1

Introduction



This thesis concerns the use of natural language processing for improving biomedical concept normalization and relation mining. We begin this chapter by introducing the background of biomedical text mining, and subsequently we will continue by describing a typical text mining pipeline, some key issues and problems in mining biomedical texts, and the possibility of using natural language processing to solve these problems. Finally, we end with an outline of the work done in this thesis.

BACKGROUND

What is text mining and why do we need it?

Information overload is one of the most widely felt problems in our modern society. Especially in the biomedical and clinical domain, most knowledge is only available in unstructured textual form, such as scientific literature and clinical notes [1]. Due to the fact that the amount of data in these resources is huge and expanding quickly, there is a pressing need for a more efficient approach to accessing and extracting information in a format that can be easily assimilated by humans or further processed by other automated tools. One approach is the use of computer systems to automatically process and extract useful information from these resources [2]. This approach is called text mining, sometimes alternately referred to as text data mining. It is a relatively new field that attempts to retrieve meaningful information from natural language texts. It may be loosely characterized as the process of analyzing texts to extract information that is useful for particular purposes [3]. In this way, the knowledge expressed in texts could be identified, extracted, managed, integrated, and exploited. Furthermore, new or tacit knowledge may also be discovered by using these methods [4].

The pioneering work that used text mining in biology was done by Swanson [5], who showed that text mining could help with the construction of hypotheses from associations found from vast amounts of research abstracts. Some of these hypotheses were later experimentally validated by experts. In recent years, text mining has been applied in numerous areas, such as finding interesting concepts from text, establishing functional annotations and relations among genes, discovering protein-protein interactions, interpreting microarray experiments, associating geno- and phenotypes, fact extraction, etc [6]. These areas use not only the traditional linguistic approaches, but also semantic approaches.

Why is there so much interest in the field of text mining?

As shown in Figure 1, the goal of biomedical text mining is to allow researchers to

identify information more efficiently, to find relationships between biomedical concepts, or to obtain a summary of literature. By applying algorithmic, statistical, and data management methods to the vast amount of biomedical literature as well as the free text fields of biomedical databases, the problem of information overload can be shifted from the researcher to the computer, and can be addressed more efficiently [7].

In this way, end users do not face having to read several tens of thousands of retrieved documents, but rely on text mining systems to accurately extract relevant information from these documents, and thus end users are enabled to discover interesting associations and potentially new knowledge [6]. This type of text mining has been used to automatically extract relationships between biomedical concepts (e.g. protein-protein interactions, genes and diseases, drugs and side effects, drugs and diseases) from biomedical texts.

Typically, text mining aims to build a system to correctly extract the intention (meaning) from a text in a machine processable form. The assumption here is that human language follows strict "rules" and that these rules have to be implemented with grammar and syntax to capture the intention of a text. However, human language is very flexible, and is difficult to grasp in a set of rules, which very much complicates the text mining task. In some controlled domains, after much training these systems can achieve acceptable performance rates. However, in the biomedical domain, with different text genres (e.g., electronic patient records, scientific texts), the performance of a text mining system drops if it has been trained on one genre, and then is applied on another genre.

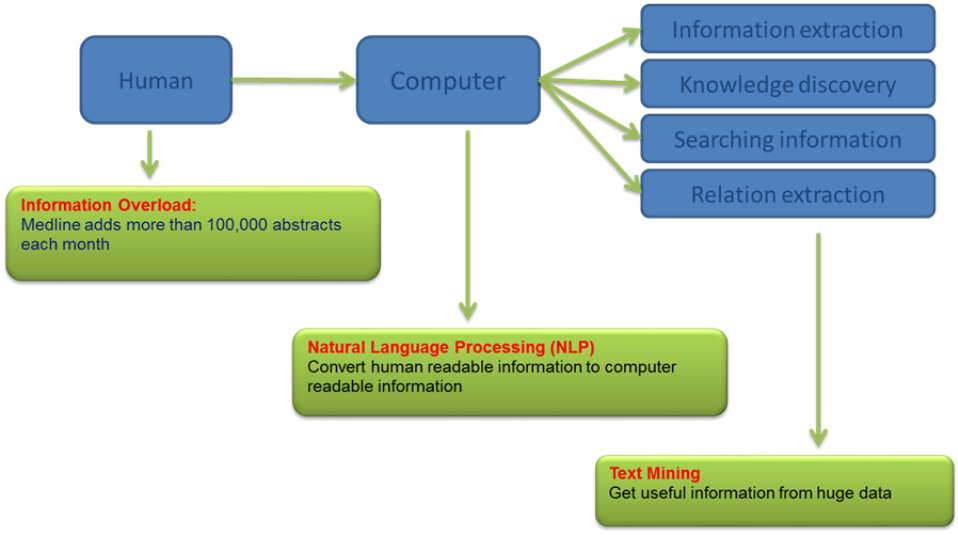


Figure 1. Using text mining to solve the information overload problem.

The usage and benefits of text mining in biology and biomedicine are being discussed in various major conferences, workshops, special tracks, and tutorials. For instance, the American Medical Informatics Association (AMIA), the Association for Computational Linguistics (ACL), the BioCreative and BioNLP initiatives, the Intelligent Systems for Molecular Biology (ISMB), the Text REtrieval Conference (TREC), and the Pacific Symposium on Biocomputing (PSB) have been addressing this topic for many years now.

TEXT MINING PIPELINE

An example of a typical text-mining pipeline

Figure 2 shows an example of a typical text-mining pipeline. Normally, the input of a text mining system is the plain text from scientific papers or clinical records. The first step of a text-mining pipeline is to make a syntactic analysis, including sentence splitting, tokenization, part-of-speech (POS) tagging, and chunking. This step is more or less domain independent, only syntax information is used. After the first step, named entity recognition is used to annotate named entities of interest with semantic groups, such as drugs or diseases, and subsequently to assign concept unique identifiers to the recognized named entities. Finally, relations between concepts are extracted. These steps are domain dependent, and semantic information is also used for analysis. Since each step uses the results from the previous steps, the performance of each step directly impacts the performance of the following steps.

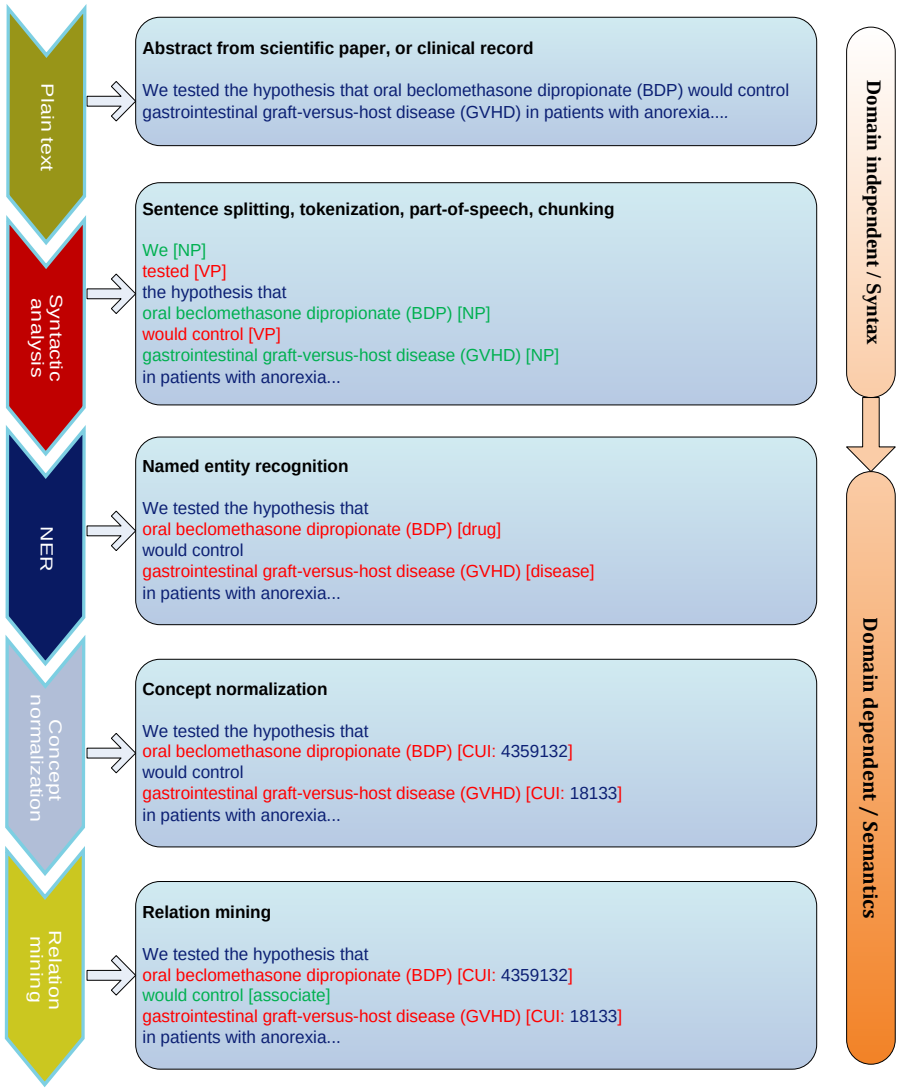


Figure 2. An example of the text-mining pipeline.

Detailed steps in a typical text mining pipeline

In general, document annotations are comments, notes, explanations, or other types of external remarks of the document. In the biomedical text mining domain, document annotations follow the principles set in natural language processing (NLP) by adding annotations at multiple levels of syntactic analysis, such as grammatical and semantic annotations [8]. There are mainly three approaches for the annotation of biomedical text:

a complete manual annotation that is based on annotators' knowledge; or pre-annotation by an annotation system, and then manual correction by domain experts; or fully automatic annotation by annotation systems. Each of these approaches has its strengths and weaknesses. For instance, manual annotations are usually more accurate, but they could be very expensive because of the time required from domain experts. Automatic annotations are usually much quicker, the size of the annotated corpora could be larger, but the quality is in general lower. Automatic annotation systems are dictionary-based, rule-based, machine learning-based, or hybrid. Normally, machine learning-based systems always need an annotated corpus for training, therefore in the traditional way, manual annotation is always needed to create such a training corpus. In the biomedical domain, corpora mainly consist of annotated journal and conference abstracts that are provided by MEDLINE [9], which is the first, and one of the most important databases in the biomedical domain. The 2011 MEDLINE contains over 18 million references from over 5,500 journals worldwide.

Depending on its purpose, different systems have different components. The performance of a text mining application depends on the performance of each component in the pipeline. Although each of these components has its own features, they achieve their goals by employing similar methods. Below are the steps of a typical text mining pipeline described in more detail.

Sentence splitting

Sentence splitting is the first step in a typical text mining pipeline. It is the process of splitting abstracts or paragraphs into sentences. For instance, in the example below, the sentence splitter splits the text into three sentences.

We tested the hypothesis.[Sentence]
The result showed that oral beclomethasone dipropionate (BDP) would control gastrointestinal graft-versus-host disease (GVHD).[Sentence]
GVHD exists in patients with anorexi.[Sentence]

Recognizing the end of a sentence may not be a trivial task for a computer. In English, punctuation marks that usually appear at the end of a line may not indicate the end of a sentence, but could be part of an abbreviation or acronym, a decimal number, or part of a bracket of periods surrounding a Roman numeral. For some unstructured texts such as clinical records, sentence splitting may even be much more complex [10].

Tokenization

Tokenization is the step after the sentence splitting. It is the process of splitting a sentence into words, phrases, symbols, or other meaningful elements called tokens. For instance, in this example, the tokenizer splits the sentence into word tokens.

The[T] result[T] showed[T] that[T] oral[T] beclomethasone[T] dipropionate[T] ([T]BDP[T]) [T] would[T] control[T] gastrointestinal[T] graft-versus-host[T] disease[T] ([T]GVHD[T]) [T].

The tokenizer uses white space such as blanks and tabs as the primary clue for splitting the text into tokens. Punctuation marks are split from the initial tokens. This is not as easy as it sounds. For example, when should a token containing a hyphen be split into two or more tokens? When does a period indicate the end of an abbreviation or a number? Some domain-dependent tokens, such as chemicals, are even more difficult to annotate because they can be very complex, with many special characters.

Part-of-speech tagging

Part-of-speech (POS) tagging is the process of marking up a word in a text as corresponding to a particular part of speech, based on both its definition, as well as its context, i.e., its relationship with adjacent and related words in a phrase, sentence, or paragraph. In the example below, all tokens are annotated with different POS tags from a POS annotation scheme. For instance, [DT] means definite article, [NN] means noun, and [VB] means verb. More information about these POS tags can be found elsewhere [11].

The[DT] result[NN] showed[VBD] that[IN] oral[JJ] beclomethasone[NN] dipropionate[NN] ([-LRB-] BDP[NNP])[-RRB-] would[MD] control[VB] gastrointestinal[JJ] graft-versus-host[JJ] disease[NN] ([-LRB-] GVHD[NNP])[-RRB-] .[.]

POS tagging is based on both the meaning of the word and its positional relationship with adjacent words. A simple list of the POS includes adjectives, adverbs, conjunctions, nouns, prepositions, pronouns, and verbs. Over the years, many POS annotation schemes have been developed and applied to different systems and corpora. Compared with more advanced text mining tasks, the tasks of sentence splitting, tokenization, and POS tagging are much easier. Most systems can achieve F-scores of more than 95% [12].

Chunking

Chunking (also called shallow parsing or chunk parsing) is a technique that attempts to provide some machine understanding of the structure of a sentence [13]. It splits text into groups of words that constitute a grammatical unit, such as noun phrase (NP), verb phrase (VP), or prepositional phrase (PP). In the following example, all tokens are grouped with different chunking annotations.

We [NP]
tested [VP]
the hypothesis [NP] that
oral beclomethasone dipropionate (BDP) [NP]
would control [VP]
gastrointestinal graft-versus-host disease (GVHD) [NP]
in patients with anorexia...

Chunking annotations are based on the annotation information of sentence, token, and part-of-speech (POS) [14]. There are several well-known chunking systems available, such as the GATE chunker [15], the Genia Tagger [16], Lingpipe [17], MetaMap [18], OpenNLP [12], and Yamcha [19]. Only few chunkers are rule-based, and all the others are machine learning-based. These chunkers also use different annotation schemes, similar to the different schemes used for POS. Most chunking systems can achieve F-scores of between 80% and 90%. There are a few chunking corpora available, such as the GENIA Treebank corpus [20], and the PennBioIE corpus [21]. These corpora also include the annotation information of sentence splitting, tokenization, and POS. Previous comparisons of chunking systems include CoNLL-2000 [15, 23].

The use of chunkers for the biomedical domain is mainly on the annotation of noun phrases and verb phrases. Noun phrases are important for the recognition and identification of biomedical entities, such as diseases and genes [18, 23]. Patterns of noun phrases and verb phrases can be used for mining relations between biomedical entities [24]. Although chunking is an essential pre-processing step in information extraction systems, no comparative studies of chunking systems are available for the biomedical domain.

Named entity recognition

Named entity recognition (NER) (also known as entity identification or entity extraction) is a subtask of information extraction that seeks to locate and classify atomic elements

in a text into predefined categories such as the names of persons, organizations, locations, etc. In the biomedical domain, NER has been used to annotate concepts such as genes, proteins, drugs, and diseases. Recognizing a concept does not require the annotation system to name it with a specific concept id, but just to determine if it belongs to a specific concept type [25]. For instance, in this example, drug and disease have been annotated as named entities.

We tested the hypothesis that
oral beclomethasone dipropionate (BDP) [drug]
would control
gastrointestinal graft-versus-host disease (GVHD) [disease]
in patients with anorexia...

For NER, semantic information may also need to be involved to categorize the entities. There are several biomedical knowledge resources available for getting semantic information, such as the Unified Medical Language System (UMLS) [9], and the Linking Open Drug Data (LODD) [26]. Among these resources, the UMLS is the most important. It is maintained by the NLM, and provides more than 100 dictionaries, terminologies, and ontologies in its Metathesaurus [9]. It also contains a semantic network that contains relations between semantic types. The UMLS Metathesaurus has been widely used by different biomedical text mining systems, especially by dictionary-based systems.

There are many named-entity recognition challenges and shared tasks in the biomedical domain, such as BioCreAtivE [27–29], BioNLP [30, 31], i2b2 [32–34], JNLPBA [35], TREC [36, 37], and CALBC [38]. These challenges and shared tasks developed many corpora for machine learning-based named entity recognition systems, such as the Collaborative Annotation of a Large Biomedical Corpus (CALBC) [38], the i2b2/VA corpus [34], the BioCreative corpora [39–41], the Colorado Richly Annotated Full-Text (CRAFT) corpus [42], and the BioNLP corpus [30, 31].

Although there are many NER systems and corpora available, they achieve similar performances: around an F-score of 80% for exact matching, and around 90% for loose matching. An exact match means that the gold standard and chunker annotations are identical, i.e., both annotations have the same start and end location in the corpus. A loose match means that at least the start position or the end position from the chunker annotations has to be the same as the gold standard position.

Further improvement of a single system or algorithm may be difficult. This is the reason for the introduction of ensemble-based systems. Systems that combine different classifiers are called ensemble-based systems, also known by various other names, such

as multiple classifier systems, a committee of classifiers, or a mixture of experts [43]. The general idea is that the combined methodology of multiple classifiers reduces the risk of errors, and performs better than the best individual classifier for a broad range of applications and under a variety of scenarios. There are many popular ensemble based algorithms, such as bagging [44], boosting [45], AdaBoost [46], stacked generalization [47], and a hierarchical mixture of experts [48], as well as commonly used combination rules, including algebraic combination of outputs or voting based techniques.

Multiple classifier systems have been applied to many domains, including biomedical text mining and information extraction domains. For instance, Smith et al. [40] combined the results of 19 systems for gene mention recognition and found that the combined system outperformed the best individual system by 3.5 percentage points in terms of its F-score. Kim et al. [30] combined eight systems for event extraction and showed that the performance of the combined system increased by 4 percentage points as compared to the best individual system.

Although ensemble systems appear to work well in various biomedical domains, it has not yet been investigated whether the approach is also effective for concept recognition in clinical records, which is regarded as more difficult than concept recognition in scientific literature [49]. The outcome is uncertain because it is still unclear which characteristics of individual systems contribute best to an ensemble system.

Apart from using ensemble systems to improve performance, it is also an innovative way to use ensemble systems for automatically generating a so-called silver standard corpus (SSC), which is closer to a gold standard corpus (GSC) for annotation quality. The creation of a GSC is tedious and expensive: annotation guidelines have to be established, domain experts must be trained, the annotation process is time-consuming, and annotation disagreements have to be resolved [38]. An SSC, on the other hand, is much easier to generate, and the size could be much larger. For machine learning-based systems, a training corpus is essential, thus the quality and size of a training corpus could have a direct impact on the performance of a machine learning-based system. Unfortunately, there are very few annotated corpora available for each sub-domain of the biomedical text mining field [7], so an SSC is a possible approach to tackle the minimal availability of relevant GSCs.

The notion that a combination of systems can be used to create an SSC has been explored in the CALBC (Collaborative Annotation of a Large Biomedical Corpus) project [38], in which the NLP community has been invited to annotate a large biomedical corpus with a variety of named-entity recognition systems. However, it is still not clear if a machine learning-based system which is trained on an SSC could get the same performance as

when it is trained on a GSC, and if an SSC could be a viable alternative, or supplement, to a GSC when training text mining systems in a biomedical domain.

Concept normalization

Although NER can extract entities from text and classify them into semantic categories, without precise identification of textual representation of particular entities, it is difficult for computers to understand the meaning and delivered message from articles and reports [50]. To solve this problem, some information retrieval (IR) and information extraction (IE) systems use concept identification tools to recognize useful terms from texts, such as diseases, genes, proteins, drugs, and chemical compounds, etc [51]. The concept identification task labels the terms with concept identifiers from a resource that contains further information about the terms. It is considered a more difficult task than the named entity recognition task [2]. Below is an example of the concept identification of a drug and a disease with UMLS concept identifiers.

We tested the hypothesis that
oral beclomethasone dipropionate (BDP) [UMLS id: 4359132]
would control
gastrointestinal graft-versus-host disease (GVHD) [UMLS id: 18133]
in patients with anorexia...

Much research has been done in named entity recognition, but fewer studies have addressed the more difficult task of concept normalization. Concept normalization systems are often dictionary-based, i.e., they try to find concept occurrences in a text by matching text strings with concept names and their corresponding identifiers in a dictionary. The dictionary is composed of entries from one or more knowledge sources, such as Gene Ontology [52], Entrez Gene [53], or UMLS [54]. Typically, dictionary-based systems use little or no linguistic information to find concepts.

There are only a few corpora in the biomedical domain that incorporate concept annotations, notably the Arizona Disease Corpus (AZDC) [55], the BioCreative gene normalization corpora [39–41], the Colorado Richly Annotated Full-Text (CRAFT) corpus [42], and the Gene Regulation Event Corpus (GREC) [56]. These corpora are widely used to train and test concept normalization systems.

Most well-known concept normalization systems are dictionary based. These systems include MetaMap [18], Mgrep [57], Negfinder [58], Peregrine [59], and Whatizit [60]. Although several systems, such as MetaMap, perform some lexical analysis in the

normalization process, more advanced NLP techniques such as chunking are mostly not considered. Even though dictionaries contain many concepts and terms, it is nearly impossible to cover all term variations, or to keep these resources complete as science progresses. Furthermore, most concept normalization systems have difficulties in dealing with linguistic constructs such as coordination, with abbreviations, disambiguation, and finding the precise term boundaries.

Compared with the variety of named-entity recognition challenges and shared tasks in the biomedical domain, there are only a few challenges and shared tasks for concept normalization. Substantial work on gene normalization has been done in a series of gene normalization tasks that were part of the BioCreative competitions [39–41]. In these challenges, the best systems achieved F-scores of about 70%.

Relation extraction

Chunking, NER and concept identification of well-defined terms, such as genes or proteins, have achieved a good level of maturity such that it can form the basis for the next step: the extraction of relations that exist between the recognized terms, entities and concepts [61]. The goal of the relation extraction task is to identify occurrences of particular types of relationships between pairs of given entities.

Although common concept classes (e.g., genes or drugs) are normally very specific, the types of relationships between concepts may be broad, including any type of biomedical association. In the biological domain, many studies have been done on the extraction of relations between genes and proteins, or protein-protein interaction [62–65]. Other associations of interest include interactions between proteins and single nucleotide point mutations [66], proteins and their binding sites [67], and genes and diseases [68]. In the clinical domain, relationships between drugs and diseases [69] and drugs and adverse effects [70] are becoming increasingly important, now that more and more of these data are stored in electronic health record systems. Below is an example of a relation between a drug and a disease.

Oral beclomethasone dipropionate (BDP) [UMLS id: 4359132]
would control [associate]
gastrointestinal graft-versus-host disease (GVHD) [UMLS id: 18133]
in patients with anorexia...

There are many issues that still need to be solved for extracting relations from biomedical texts. One of the biggest problems is the performance of the current approaches.

Compared with the performance of other biomedical text mining tasks, the performance of relation extraction is still quite low [71, 72]. In order to satisfy the demands of specific tasks, such as automatically building high-quality biomedical relation databases, the performance of current approaches needs to be improved.

In recent years, many relation extraction systems have been developed, such as JRex [73], Semantic Knowledge Representation (SKR) [74], java Simple Relation Extraction (jSRE) [75], etc. These systems use approaches such as simple co-occurrence approaches, rule-based approaches, machine learning-based approaches, and NLP. There have been several relation extraction challenges in the biomedical domain, e.g., BioCreative [29, 76, 77] and BioNLP [30, 31]. Performances in these challenges are quite different depending on the complexity of the given tasks. For instance, in the Biocreative II protein-protein interaction task, the highest F-score of 78% was obtained for the interaction detection subtask [76]. In BioCreative III, a maximum F-score of 55% was achieved [77] for detecting protein-protein interactions in full-text articles. There are several publicly available corpora in the biomedical domain that incorporate relation annotations, notably the corpora generated for the Biocreative [29, 76, 77] and BioNLP [30, 31] challenges, the EDGAR corpus[78], the GENIA event corpus[20], the CLEF corpus[79], the PharmGKB knowledge base[80], the EU-ADR corpus[81], and the ADE corpus[70].

In the domain of relation mining, both entities such as proteins, genes, diseases, drugs and their effects have to be correctly identified to find a relationship between them [82]. One possibility of improving performance of relation mining is to use existing knowledge. This is, however, a new research area that has not yet been addressed in much detail.

RESEARCH QUESTIONS

Although text mining has been widely used in the biomedical domain and yields promising results, there are still limitations, issues and problems that need to be addressed. The performance of text mining systems is probably the biggest issue. Most systems, especially machine learning-based systems, are tuned for a particular domain but performance degrades when applied to another domain. Another problem is how to analyze unstructured texts such as clinical records that often have frequent spelling errors and do not follow grammatical rules. These texts are more difficult to analyze than well-written texts such as research papers. The third issue is how to cope with the lack of annotated corpora for training machine-learning algorithms. Although many systems use syntactic information to improve their performance, there are limitations because of

grammatical errors, spelling mistakes, abbreviations, etc.

A number of approaches can be followed to mitigate this situation.

1. Assume that different systems have different strengths and weaknesses, and that performance should be improved if we build a kind of consensus between a number of systems - ensemble approaches. This approach is not new, but has never been applied in some domains, such as chunking.

2. Increase the training set of text mining systems. This approach increases the chances that we encounter many of the language constructs that we want to capture with text mining. However, it makes the task very expensive because domain experts need to be involved. If we could use an automatic system to build such a large corpus and/or extend an existing corpus with acceptable quality, then we could also automatically generate corpora for the different sublanguages and domains.

3. Exploit the use of existing knowledge resources that can help us to understand the texts, such as the implicit relations mentioned in a text, correct concept identification in case of homonyms and abbreviations, and anaphora resolution.

Improve text mining

The performance of relation mining is directly impacted by the quality of chunking and concept identification. In order to obtain high precision relation mining we need high precision chunking and concept identification. Chapter 2 contains an analysis of the performance of six chunkers trained and evaluated on the GENIA corpus. In order to improve the quality of chunking the results of the chunkers are combined using a simple voting scheme. The combination of these systems shows an improvement in performance beyond any of the individual systems.

All text-mining systems that are based on machine learning need a corpus for training. The size of most GSCs is small due to the tedious and expensive creation work. The research question is to see if we can automatically combine and harmonize with sufficient quality the results from different concept identification systems into a single SSC. Although we can automatically generate a large SSC, we still need to investigate the differences between an SSC and a GSC, and the possibilities of using this approach to supplement a GSC.

In chapter 3 we address these research questions and explore two scenarios using an SSC. In the first scenario, a chunker has to be trained for a biomedical subdomain for which a GSC is not available. Rather than creating a new GSC, we generate an SSC for this

domain and use this SSC for training the chunker. In the second scenario, a GSC from the domain of interest is available but its size is small and a chunker trained on it gives suboptimal performance. Rather than expanding the GSC, we supplement the GSC with an SSC derived for the same domain and train the chunker on the combination of GSC and SSC to improve chunker performance.

In Chapter 4, we leverage the performance of a number of systems that recognize medical concepts in clinical records by combining the output of the individual systems. Apart from performance improvement, we show that with this approach the balance between precision and recall of the ensemble system can be easily adjusted. This feature makes it ideally suited for tasks that require either a high precision or a high recall. We test our approach by participating in the concept extraction task of the 2010 i2b2/VA challenge on clinical records.

Integrate biomedical knowledge

To improve the performance of concept normalization we describe in chapter 5 the contribution of NLP techniques to biomedical concept normalization. We present a set of rules that utilize NLP information, and show that these rules substantially improve the performance of two concept normalization systems, MetaMap and Peregrine, in recognizing and normalizing diseases in biomedical texts. In chapter 6, we investigate the combination of a knowledge base and NLP techniques to improve biomedical relation mining. We present a knowledge base system that utilizes known relations between biomedical concepts and show that the system substantially improves the performance of a standard NLP and machine learning-based biomedical relation mining system for mining drugs and adverse effects in biomedical texts.

Finally, in chapter 7, we provide a general discussion of the results described in this thesis and provide suggestions for future research.

REFERENCES

1. Zweigenbaum P, Demner-Fushman D, Yu H, Cohen KB: **Frontiers of biomedical text mining: current progress**. *Brief. Bioinform* 2007, **8**:358–75.
2. Chen H: *Medical informatics: knowledge management and data mining in biomedicine*. Springer Science+ Business Media; 2005, **8**.
3. Witten IH, Bray Z, Mahoui M, Teahan B: **Text mining: A new frontier for lossless compression**. In *Data Compression Conference (DCC'99)*. Snowbird, USA: 1999:198–207.
4. Kao A, Poteet SR: *Natural language processing and text mining*. Springer; 2007.
5. Swanson DR: **Medical literature as a potential source of new knowledge.** *J Med Lib Asso* 1990, **78**:29.
6. Ananiadou S, McNaught J: *Text mining for biology and biomedicine*. Artech House London; 2006.
7. Cohen AM, Hersh WR: **A survey of current work in biomedical text mining**. *Brief. Bioinform* 2005, **6**:57–71.
8. Wynne M: *Developing linguistic corpora: a guide to good practice*. Oxbow Books; 2005.
9. Srinivasan S, Rindflesch TC, Hole WT, Aronson AR, Mork JGC-P: **Finding UMLS Metathesaurus concepts in MEDLINE.** *Proceedings of the AMIA Symposium* 2002:727–31.
10. Warrer P, Hansen EH, Juhl-Jensen L, Aagaard L: **Using text-mining techniques in electronic patient records to identify ADRs from medicine use**. *Brit J Clin Pharmaco* 2011, **73**:674–84.
11. Hahn U, Buyko E, Landefeld R, M ¨ hlhausen M, Poprat M, Tomanek K, Wermter J: **An overview of JCoRe, the JULIE lab UIMA component repository**. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*. Marrakech, Morocco: 2008:1–7.
12. Buyko E, Wermter J, Poprat M, Hahn U: **Automatically Adapting an NLP Core Engine to the Biology Domain**. In *Proceedings of the Joint BioLINKBio-Ontologies Meeting*. Fortaleza, Brasil: 2006:2–5.
13. Berwick RC, Abney SP, Tenny C: *Principle-based parsing: computation and psycholinguistics*. Springer; 1991.

14. Sang E, Buchholz S: **Introduction to the CoNLL-2000 shared task: Chunking**. In *Proceedings of CoNLL-2000 and LLL-2000*. Lisbon, Portugal: 2000:127–32.

15. Cunningham H: **GATE, a General Architecture for Text Engineering**. *Comput Human* 2002, **36**:223–254.

16. Tsuruoka Y: **Bidirectional inference with the easiest-first strategy for tagging sequence data**. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*. Vancouver, British Columbia, Canada: 2005:467–474.

17. Carpenter B: **LingPipe for 99.99 % Recall of Gene Mentions**. In *Proceedings of the Second BioCreative Challenge*. Madrid, Spain: 2007:2–4.

18. Aronson AR: **Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program.** In *Proceedings of the AMIA Symposium*. Philadelphia, USA: 2001, **pp**:17–21.

19. Kudo T, Matsumoto Y: **YamCha: Yet another multipurpose chunk annotator**. *http://www. chasen. org/~ tAKu/software/yamcha* 2005.

20. Tateisi Y, Yakushiji A, Ohta T, Tsujii J: **Syntax annotation for the GENIA corpus**. In *Proceedings of the Companion Volume of the Second International Joint Conference on Natural Language Processing IJCNLP05*. Jeju Island, Korea: 2005:222–7.

21. Kulick S, Bies A, Liberman M, Mandel M, Mcdonald R, Palmer M, Schein A, Ungar L, Winters S, White P: **Integrated Annotation for Biomedical Information Extraction**. In *Human Language Technology conference / North American chapter of the Association for Computational Linguistics annual meetings (HLT/NAACL)*. Boston, MA, USA: 2004.

22. Wermter J, Fluck J, Stroetgen J, Gei ßler S, Hahn U: **Recognizing noun phrases in biomedical text: An evaluation of lab prototypes and commercial chunkers**. In *Proceedings of the 1st International Symposium on Semantic Mining in Biomedicine*. Hinxton, England: 2005:25–33.

23. Zhou GD, Su J: **Named entity recognition using an HMM-based chunk tagger**. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. Morristown, NJ, USA: 2001:473–80.

24. Rinaldi F, Schneider G, Kaljurand K, Hess M, Andronis C, Konstandi O, Persidis A: **Mining of relations between proteins over biomedical scientific literature using a deep-linguistic approach**. *Artif Intell Med* 2007, **39**:127–36.

25. Tjong Kim Sang EF, De Meulder F: **Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition**. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*. Edmonton, Alberta: :142–7.

26. Jentzsch A, Zhao J, Hassanzadeh O, Cheung K-H, Samwald M, Andersson B: **Linking open drug data**. In *Triplification Challenge of the International Conference on Semantic Systems*. Graz, Austria: 2009:3–6.

27. Hirschman L, Yeh A, Blaschke C, Valencia A: **Overview of BioCreAtIvE: critical assessment of information extraction for biology**. *BMC Bioinform* 2005, **6**:S1.

28. Smith L, Tanabe LK, Ando RJN, Kuo C-J, Chung I-F: **Overview of BioCreative II gene mention recognition**. *Genome Biology* 2008, **9**:S2.

29. Leitner F, Mardis SA, Krallinger M, Cesareni G, Hirschman LA, Valencia A: **An overview of BioCreative II. 5**. *Comput Biolog Bioinform* 2010, **7**:385–99.

30. Kim J-D, Ohta T, Pyysalo S, Kano Y, Tsujii J: **Overview of BioNLP'09 shared task on event extraction**. In *Proceedings of the Workshop on BioNLP Shared Task*. Boulder, USA: 2009:1–9.

31. Kim JD, Pyysalo S, Ohta T, Bossy R, Nguyen N, Tsujii J: **Overview of BioNLP shared task 2011**. In *Proceedings of the BioNLP Shared Task 2011 Workshop*. 2011:1–6.

32. Heinze DT, Morsch ML, Potter BC, Sheffer RE: **Medical i2b2 NLP smoking challenge: the A-Life system architecture and methodology**. *J Am Med Inform Assoc* 2008, **15**:40–3.

33. Uzuner O: **Second i2b2 workshop on natural language processing challenges for clinical records.** In *AMIA Symposium*. Washington D.C, USA: 2008:1252.

34. Uzuner O, South BR, Shen S, Duvall SL: **2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text.** *J Am Med Inform Assoc* 2011, **18**:552–6.

35. Kim J-D, Ohta T, Tsuruoka Y, Tateisi Y, Collier N: **Introduction to the bio-entity recognition task at JNLPBA**. In *Proceedings of the international joint workshop on natural language processing in biomedicine and its applications (JNLPBA)*. Stroudsburg, PA, USA: 2004:70–5.

36. Balog K, Serdyukov P, Vries AP de: *Overview of the TREC 2010 entity track*. DTIC Document; 2010.

37. Voorhees EM, Tong RM: **Overview of the TREC 2011 medical records track**. In

Proceedings of the twentieth Text REtrieval Conference (TREC). Gaithersburg, USA: 2011.

38. Rebholz-Schuhmann D, Yepes AJ, van Mulligen EM, Kang N, Kors J, Milward D, Corbett P, Buyko E, Beisswanger E, Hahn U: **The CALBC Silver Standard Corpus - Harmonizing multiple semantic annotations in a large biomedical corpus**. *J Bioinform Comput Biol* 2010, **8**:163.

39. Hirschman L, Colosimo M, Morgan A, Yeh AC-P: **Overview of BioCreAtIvE task 1B: normalized gene lists**. *BMC Bioinform* 2005, **6 Suppl 1**:S11.

40. Morgan AA, Lu Z, Wang X, Cohen AM, Fluck J, Ruch P, Divoli A, Fundel K, Leaman R, Hakenberg J, Sun C, Liu HH, Torres R, Krauthammer M, Lau WW, Liu H, Hsu CN, Schuemie M, Cohen KB, Hirschman LC-P: **Overview of BioCreative II gene normalization**. *Genome Biol* 2008, **9 Suppl 2**:S3.

41. Lu Z, Kao HY, Wei CH, Huang M, Liu J, Kuo CJ, Hsu CN, Tsai RTH, Dai HJ, Okazaki N: **The gene normalization task in BioCreative III**. *BMC Bioinform* 2011, **12**:S2.

42. Bada M, Hunter LE, Eckert M, Palmer M: **An overview of the CRAFT concept annotation guidelines**. In *Proceedings of the Fourth Linguistic Annotation Workshop*. Uppsala, Sweden: 2010:207–11.

43. Polikar R: **Ensemble based systems in decision making**. *IEEE Circuits Syst Mag* 2006, **6**:21–45.

44. Breiman L: **Bagging Predictors**. *Mach learn* 1996, **24**:123–40 ST – Bagging Predictors.

45. Freund Y, Schapire RE: **A decision-theoretic generalization of on-line learning and an application to boosting**. *J comput system sciences* 1997, **55**:119–139.

46. Rätsch G, Onoda T, Müller KR: **Soft margins for AdaBoost**. *Mach learn* 2001, **42**:287–320.

47. Ting KM, Witten IH: **Issues in stacked generalization**. *arXiv preprint arXiv:1105.5466* 2011.

48. Jordan MI, Jacobs RA: **Hierarchical mixtures of experts and the EM algorithm**. *Neural computation* 1994, **6**:181–214.

49. Meystre SM, Savova GK, Kipper-Schuler KC, Hurdle JF: **Extracting information from textual documents in the electronic health record: a review of recent research.** *Yearb Med Inform* 2008:128–44.

50. Krauthammer M, Nenadic G: **Term identification in the biomedical literature.** *J Biomed Inform* 2004, **37**:512–26.

51. Shatkay H: **Hairpins in bookstacks: information retrieval from biomedical text.** *Brief Bioinform* 2005, **6**:222–38.

52. Harris MA, Clark J, Ireland A, Lomax J, Ashburner M, Foulger R, Eilbeck K, Lewis S, Marshall B, Mungall C: **The Gene Ontology (GO) database and informatics resource**. *Nucleic Acids Res* 2004, **32**:258–61.

53. Maglott D, Ostell J, Pruitt KD, Tatusova T: **Entrez Gene: gene-centered information at NCBI**. *Nucleic Acids Res* 2007, **35**:26–31.

54. Bodenreider O: **The Unified Medical Language System (UMLS): integrating biomedical terminology.** *Nucleic Acids Res* 2004, **32**:267–70.

55. Leaman R, Miller C, Gonzalez G: **Enabling recognition of diseases in biomedical text with machine learning: corpus and benchmark**. In *Proceedings of the 3rd International Symposium on Languages in Biology and Medicine (LBM)*. Jeju Island, South Korea: 2009:82–9.

56. Thompson P, Iqbal SA, McNaught J, Ananiadou S: **Construction of an annotated corpus to support biomedical information extraction**. *BMC Bioinform* 2009, **10**:349.

57. Shah NH, Bhatia N, Jonquet C, Rubin D, Chiang AP, Musen MAC-P: **Comparison of concept recognizers for building the Open Biomedical Annotator.** *BMC Bioinform* 2009, **10**:S14.

58. Mutalik PG, Deshpande A, Nadkarni PM: **Use of general-purpose negation detection to augment concept indexing of medical documents**. *J Am Med Inform Assoc* 2001, **8**:598–609.

59. Schuemie MJ, Jelier R, Kors JA: **Peregrine: lightweight gene name normalization by dictionary lookup**. In *Proceedings of the BioCreAtIvE II Workshop*. Madrid, Spain: 2007:131–3.

60. Rebholz-Schuhmann D, Arregui M, Gaudan S, Kirsch H, Jimeno A: **Text processing through Web services: calling Whatizit**. *Bioinformatics* 2008, **24**:296–8.

61. Bundschus M, Dejori M, Stetter M, Tresp V, Kriegel HP: **Extraction of semantic biomedical relations from text using conditional random fields**. *BMC Bioinform* 2008, **9**:207.

62. Bunescu R, Ge R, Kate RJ, Marcotte EM, Mooney RJ, Ramani AK, Wong YW: **Comparative experiments on learning information extractors for proteins and their interactions**. *Artif Intell Med* 2005, **33**:139–156.

63. Saric J, Jensen LJ, Ouzounova R, Rojas I, Bork P: **Extraction of regulatory gene/protein networks from Medline.** *Bioinformatics* 2006, **22**:645–50.

64. Rinaldi F, Schneider G, Kaljurand K, Hess M, Andronis C, Konstandi O, Persidis A: **Mining of relations between proteins over biomedical scientific literature using a deep-linguistic approach.** *Artif Intell Med* 2007, **39**:127–136.

65. Pyysalo S, Airola A, Heimonen J, Björne J, Ginter F, Salakoski T: **Comparative analysis of five protein-protein interaction corpora**. *BMC Bioinform* 2008, **9**:S6.

66. Lee LC, Horn F, Cohen FE: **Automatic extraction of protein point mutations using a graph bigram association**. *PLoS comput biology* 2007, **3**:e16.

67. Chang DT-H, Weng Y-Z, Lin J-H, Hwang M-J, Oyang Y-J: **Protemot: prediction of protein binding sites with automatically extracted geometrical templates**. *Nucleic acids research* 2006, **34**:W303–W309.

68. Chun HW, Tsuruoka Y, Kim JD, Shiba R, Nagata N, Hishiki T, Tsujii J: **Extraction of gene-disease relations from Medline using domain dictionaries and machine learning**. In *Pac Symp Biocomput*. Rockville, USA: 2006, **11**:4–15.

69. Theobald M, Shah N, Shrager J: **Extraction of conditional probabilities of the relationships between drugs, diseases, and genes from pubmed guided by relationships in pharmgkb**. *Summit translat bioinform* 2009:124.

70. Gurulingappa H, Rajput AM, Roberts A, Fluck J, Hofmann-Apitius M, Toldo L: **Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports**. *J Biomed Inform* 2012, **45**:885–92.

71. Bui Q-C, Katrenko S, Sloot PMA: **A hybrid approach to extract protein-protein interactions.** *Bioinformatics* 2011, **27**:259–265.

72. Simpson MS, Demner-fushman D: **Biomedical text mining: a survey of recent progress**. In *Mining Text Data*. Springer; 2012:465–517.

73. Buyko E, Beisswanger E, Hahn U: **The extraction of pharmacogenetic and pharmacogenomic relations–a case study using pharmgkb**. In *Pac Symp Biocomput*. Hawaii, USA: 2012, **376**:376–87.

74. Rindflesch TC, Fiszman M: **The interaction of domain knowledge and linguistic**

structure in natural language processing: interpreting hypernymic propositions in biomedical text. *J Biomed Inform* 2003, **36**:462–477.

75. Gurulingappa H, Rajput AM, Toldo L: **Extraction of Adverse Drug Effects from Medical Case Reports**. *J Biomed Semantics* 2012, **3**:15.

76. Krallinger M, Leitner F, Rodriguez-Penagos C, Valencia A: **Overview of the protein-protein interaction annotation extraction task of BioCreative II**. *Genome Biol* 2008, **9**:S4.

77. Krallinger M, Vazquez M, Leitner F, Salgado D, Chatr-aryamontri A, Winter A, Perfetto L, Briganti L, Licata L, Iannuccelli M: **The protein-protein interaction tasks of BioCreative III: classification/ranking of articles and linking bio-ontology concepts to full text**. *BMC Bioinform* 2011, **12**:S3.

78. Rindflesch TC, Tanabe L, Weinstein JN, Hunter L: **EDGAR: extraction of drugs, genes and relations from the biomedical literature**. In *Pac Symp Biocomput*. NIH Public Access; 2000:517.

79. Roberts A, Gaizauskas R, Hepple M, Davis N, Demetriou G, Guo Y, Kola JS, Roberts I, Setzer A, Tapuria A: **The CLEF corpus: semantic annotation of clinical text**. In *AMIA Annual Symposium Proceedings*. 2007, **2007**:625.

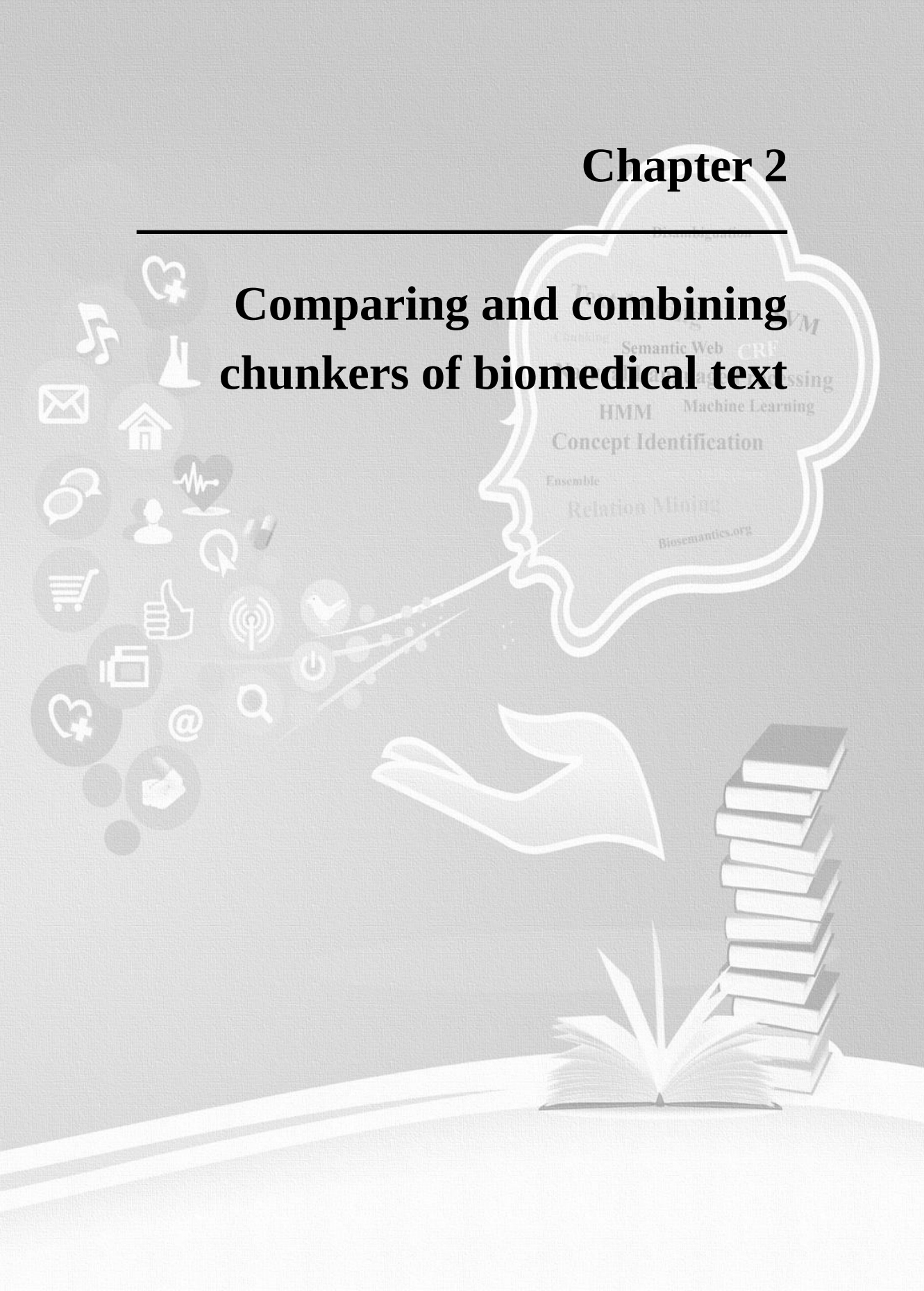
80. Thorn CF, Klein TE, Altman RB: **Pharmacogenomics and bioinformatics: PharmGKB.** *Pharmacogenomics* 2010, **11**:501–505.

81. van Mulligen EM, Fourrier-Reglat A, Gurwitz D, Molokhia M, Nieto A, Trifiro G, Kors JA, Furlong LI: **The EU-ADR corpus: Annotated drugs, diseases, targets, and their relationships**. *J Biomed Inform* 2012, **45**:879–84.

82. Baumgartner WA, Lu Z, Johnson HL, Caporaso JG, Paquette J, Lindemann A, White EK, Medvedeva O, Cohen KB, Hunter L: **Concept recognition for extracting protein interaction relations from biomedical text**. *Genome Biol* 2008, **9**:S9.

Chapter 2

Comparing and combining chunkers of biomedical text



ABSTRACT

Text chunking is an essential pre-processing step in information extraction systems. No comparative studies of chunking systems, including sentence splitting, tokenization and part-of-speech tagging, are available for the biomedical domain. We compared the usability (ease of integration, speed, trainability) and performance of six state-of-the-art chunkers for the biomedical domain, and combined the chunker results in order to improve chunking performance.

We investigated six frequently used chunkers: GATE chunker, Genia Tagger, Lingpipe, MetaMap, OpenNLP, and Yamcha. All chunkers were integrated into the Unstructured Information Management Architecture framework. The GENIA Treebank corpus was used for training and testing. Performance was assessed for noun-phrase and verb-phrase chunking.

For both noun-phrase chunking and verb-phrase chunking, OpenNLP performed best (F-scores 89.7% and 95.7%, respectively), but differences with Genia Tagger and Yamcha were small. With respect to usability, Lingpipe and OpenNLP scored best. When combining the results of the chunkers by a simple voting scheme, the F-score of the combined system improved by 3.1 percentage point for noun phrases and 0.6 percentage point for verb phrases as compared to the best single chunker. Changing the voting threshold offered a simple way to obtain a system with high precision (and moderate recall) or high recall (and moderate precision).

This study is the first to compare the performance of the whole chunking pipeline, and to combine different existing chunking systems. Several chunkers showed good performance, but OpenNLP scored best both in performance and usability. The combination of chunker results by a simple voting scheme can further improve performance and allows for different precision-recall settings.

INTRODUCTION

Chunking (also called shallow parsing or chunk parsing) is a natural language processing technique that attempts to provide some machine understanding of the structure of a sentence, but without parsing it fully into a parsed tree form [1]. It splits text into groups of words that constitute a grammatical unit, like noun phrase (NP), verb phrase (VP), or preposition phrase (PP). For example, a chunker may annotate the sentence "The production of human immunodeficiency virus type 1 was followed in the U937 promonocytic cell line." with the following information:

[NP The production] [PP of] [NP human immunodeficiency virus type 1] [VP was followed] [PP in] [NP the U937 promonocytic cell line].

The concept of text chunking was introduced by Steven Abney in 1991 [1]. As a good approach to parsing he proposed to start with finding correlated chunks of words. In 1995, Ramshaw and Marcus used a machine learning method for chunking annotation [2]. Their work has inspired many others to proceed along the same lines. Until now, there are mainly two approaches for chunking annotations. One is a rule-based approach, in which the chunker consists of a set of regular expression statements. Rule-based systems are relatively easy to develop, do not need a training corpus, but are difficult to adapt to a new domain. The other approach is a statistical one, which uses statistical machine learning methods. Statistical systems are easy to reuse in a new domain, but need a large training corpus.

Typically, Natural language Processing (NLP) applications make use of a "pipeline" of text processing components in order to extract information from text (see figure 1). The performance of an NLP application depends on the performance of each sub-component in the pipeline. Errors made by an "upstream" component will propagate to the "downstream" components and thus negatively impact the performance of the NLP application. Chunking performance always has a direct impact on the NLP system which uses a chunking system as a sub-component. For instance, in [3] it is shown that many concept annotation errors are caused by wrong chunking annotations. In [4], it is stated that for concept annotation in clinical records only complete chunks are considered correct.

Figure 1. Natural language processing pipeline.

Chunking annotations are based on the annotation information of sentence, token, and part-of-speech (POS). The focus of chunkers for the biomedical domain is mainly on the annotation of noun phrases and verb phrases. Noun phrases are important for the recognition and identification of biomedical entities, such as diseases and genes [5, 6]. Patterns of noun phrases and verb phrases can be used for mining relations between biomedical entities [7].

There is a general paucity of data on the performance of text chunkers in the biomedical domain. During the last decade, the performance of different chunking system has been assessed in two comparative evaluation efforts. The first was the shared task in CoNLL-2000 [8], in which the Wall Street Journal (WSJ) corpus was annotated by eleven different chunkers. The six best performing systems had F-scores between 91% and 93%. Interestingly, the three top-ranking systems combined the results of different chunk taggers by majority or weighted voting [9-11], but none of the individual chunkers that were combined, are publicly available. Five years later, a study presented at SMBM-2005 evaluated the performance of four general-purpose chunkers [12]. These chunkers were trained on the Penn TreeBank corpus and were tested for noun-phrase chunking in a biomedical corpus. The results in terms of F-score ranged between 85% and 89%.

Most chunkers in these evaluation studies were not specifically trained for the biomedical domain or are no longer available, and several more recent biomedically-oriented systems were not included. Also, in these evaluation studies the gold standard sentence splitting, tokenization, and POS annotations were used, which is different from what is applicable in a real-life environment. In addition, the GENIA corpus [13], a publicly available biomedical corpus, has recently been greatly expanded, which permits a more comprehensive performance assessment in the biomedical domain. In addition to the traditional focus on noun phrases, the recognition of verb phrases should also be taken into consideration as both can be important in mining biomedical relationships. Finally, the combination of currently available chunkers has not yet been tested as a means to improve chunking performance. A simple voting scheme with different voting thresholds may offer a way to configure precision and recall of a combined system.

In this study we compare six chunkers trained and evaluated on the GENIA corpus. Both the recognition of noun phrases and verb phrases is included in the evaluation. Also usability issues, such as ease of integration and trainability, are compared. We combine the results of the chunkers to improve chunking performance and show how a simple voting scheme can be used to balance precision and recall of a combined system.

METHODS

Chunking software

Six well-known, publicly available and actively maintained chunkers were selected for comparison. All chunkers were downloaded directly from their official websites, and are briefly described below. Table 1 gives an overview of the chunker characteristics.

1. GATE chunker (http://gate.ac.uk): GATE, the General Architecture for Text Engineering [14], is a framework for developing and deploying software components for natural language processing, and can be compared to UIMA. GATE contains many default plug-ins, including the Noun Phrase Chunker, a Java implementation of the Ramshaw and Marcus BaseNP chunker [15]. This chunker inserts brackets marking noun phrases in text using POS tags generated by the GATE plug-in ANNIE.

2. Genia Tagger (http://www-tsujii.is.s.u-tokyo.ac.jp/ GENIA/tagger): Genia Tagger is a combination of a POS tagger, chunker, and named entity recognition tool. It has been developed for processing biomedical texts, in particular Medline abstracts [16]. The Genia Tagger utilizes an algorithm that is based on a maximum entropy model. The models provided with the Genia Tagger are based on the WSJ, GENIA, and PennBioIE corpora. It is not possible to use other corpora for training a model.

3. Lingpipe (http://alias-i.com/lingpipe): Lingpipe is a suite of Java libraries for natural language processing, including POS tagging, named entity recognition, spelling correction, etc. The Lingpipe chunker supports rule-based, dictionary-based, and statistical chunking. We chose to use the statistical chunker, which is based on a hidden markov model, because according to the Lingpipe website, the statistical chunker is the most accurate one. The architecture of Lingpipe makes it easy to integrate with other systems, such as UIMA. Lingpipe supports a training mode and a variety of precompiled models for different domains.

4. MetaMap SPECIALIST (http://mmtx.nlm.nih.gov): MetaMap is a highly configurable program developed by the National Library of Medicine to identify concepts from the UMLS Metathesaurus in biomedical text [6]. Based on the SPECIALIST minimal

commitment parser texts are split into chunks and identified as a concept. The SPECIALIST parser is based on the notion of a special set of so-called barrier words that indicate boundaries between phrases [17]. These barrier words make it possible to run MetaMap without a training model. MetaMap Transfer (MMTx) is a distributable version of MetaMap written in Java.

5. OpenNLP (http://opennlp.sourceforge.net): OpenNLP is an organizational center for open source projects related to natural language processing. The OpenNLP chunker is based on a maximum entropy model [18]. An OpenNLP UIMA wrapper has been developed by JULIE Lab (http://www.julielab.de). The wrapper divides the OpenNLP package into small modules that perform sentence detection, tokenization, POS tagging, chunking, named entity recognition, etc, which makes it easy to configure the pipeline for different purposes. The chunker supports a training mode, and two precompiled models based on the PennBioIE and GENIA corpora are provided.

6. Yamcha (http://chasen.org/~taku/software/yamcha): Yamcha is a generic, customizable, and open source text chunker oriented toward many natural language processing tasks, such as POS tagging, named entity recognition, and text chunking. The Yamcha chunker is based on a support vector machines algorithm. Yamcha can be trained and has been integrated in a variety of applications.

Characteristics	GATE chunker	Genia Tagger	Lingpipe	MetaMap	OpenNLP	Yamcha
Used version	5.0	3.0.1	3.8	2008 v2	2.1	0.33
Release year	2008	2007	2009	2009	2008	2005
Method[a]	TBL	MEM	HMM	BW	MEM	SVM
Coding language	Java	C++	Java	Java	Java	C++
Integration	Easy	Medium	Easy	Medium	Easy	Medium
Training mode	No	No	Yes	No	Yes	Yes
License	GPL	GPL-like	Multi[b]	GPL-like	LGPL	LGPL
Speed (ms/abstract)	230	303	165	337	237	283

Table 1. Chunker characteristics.

[a]TBL=Transformation-Based Learning; MEM=Maximum Entropy Model; HMM=Hidden Markov Model; BW=Barrier Words; SVM=Support Vector Machine.
[b]Multi: Lingpipe provides four license versions, including a free version.

Corpus

There are only a few publicly available corpora that incorporate chunk annotations. We used the GENIA Treebank corpus [13]. The latest version of the GENIA Treebank corpus was released in 2009 and consists of 1,999 Medline abstracts selected from a query using the MeSH terms "human", "blood cells", and "transcription factors". The corpus has been annotated with various levels of linguistic and semantic information, such as sentence, tokenization, POS tagging, chunk annotation, and term and event information. The corpus includes 18,541 sentences, 510,239 tokens, and 258,380 chunk annotations, of which 139,624 are noun phrases and 38,603 are verb phrases.

Training and testing of the chunkers

For the chunkers that do not provide a training mode, we used the default settings (GATE chunker, MetaMap) or a pretrained model that is provided by the developers (Genia Tagger). Although OpenNLP can be trained, we also chose to use a pretrained model supplied by the developers. Both the pretrained models had been learned on a subset of 500 abstracts from the GENIA corpus. The same subset of abstracts was used to train the two other chunkers, Lingpipe and Yamcha, both of which provide a training mode but not a pretrained model. For testing, we used the GENIA corpus after excluding the 500 abstracts that had been used for training.

Evaluation pipeline

We used the Unstructured Information Management Architecture (UIMA) framework [19] to integrate the chunking software and assess the performance of the different chunkers.

The workflow of the integrated chunking UIMA framework includes five parts. First, the UIMA Collection Reader reads the texts and gold standard annotations of the GENIA corpus. Subsequently, each of the chunkers is activated to parse the text corpus and annotate them with chunk tags. Since GATE chunker, Lingpipe and OpenNLP already have a UIMA wrapper, they were easily integrated with UIMA. For Genia Tagger, MetaMap and Yamcha, we developed a java process to communicate between these chunkers and UIMA. The annotation results of the chunkers are fed back into UIMA pipeline. Subsequently, the Stopword Filter removes all stopwords from chunks (see next section) and aligns the start and stop position of the phrases. Finally, the Annotation Comparator calculates the precision, recall, and F-score for each chunker.

Performance evaluation

We used a stopword filter followed by exact matching to compare the gold standard annotations with the chunker annotations. To reduce the effect of insignificant differences between chunks, stopwords from the stopword list in PubMed (http://www.medparse.com/umlsstop.htm) and punctuation marks were removed from both the gold standard and the chunker annotations. Subsequently, phrases were compared by exact match, similar to the procedure followed in the CoNLL-2000 task. A phrase was counted as true positive if the gold standard and chunker annotations were identical, i.e., both annotations had the same start and end location in the corpus. A phrase annotated by the gold standard was counted as false negative if the chunker did not match it exactly; a phrase annotated by the chunker was counted as false positive if it did not exactly match the gold standard. Two phrases that overlapped but did not match exactly were thus counted as a false-positive one and a false-negative one. The performance of the chunkers was evaluated in terms of precision, recall, and F-score. These measures are commonly used to quantify the performance of NLP systems and were also used in CoNLL-2000 and SMBM-2005, facilitating performance comparison across studies.

Combination of chunker results

We combined the results of the different chunkers by a simple voting scheme. For each phrase annotated by a chunker, the number of chunkers that exactly matched the phrase was counted. If the count was larger or equal than a preset voting threshold, the phrase was considered to be annotated by the combined system, otherwise it was not annotated. The voting threshold was varied between one and the maximum number of chunkers (six for noun phrases, five for verb phrases).

RESULTS

Performance of the chunkers

Chunker	Noun phrases			Verb phrases		
	Precision	Recall	F-score	Precision	Recall	F-score
GATE chunker	73.2	78.9	76.0	n.a	n.a	n.a
Genia Tagger	88.0	90.0	88.9	95.0	95.5	95.2
Lingpipe	83.0	86.0	84.5	90.3	90.2	90.3
MetaMap	80.8	87.1	83.8	74.4	83.1	78.5
OpenNLP	89.4	90.0	89.7	96.2	95.2	95.7
Yamcha	87.4	88.8	88.1	94.7	93.6	94.1

Table 2. Performance of six chunkers on the GENIA corpus.

Table 2 shows the performance of the chunkers on the 1,499 GENIA abstracts that had not been used for training. GATE chunker was not evaluated for verb-phrase recognition since it does not recognize verb phrases. For noun phrases, the best performing chunker is OpenNLP (F-score 89.7%), with Genia Tagger and Yamcha performing slightly less (F-scores 88.9% and 88.1%, respectively). For verb phrases, OpenNLP also performed best (F-score 95.7%), followed by Genia Tagger and Yamcha (F-scores 95.2% and 94.1%, respectively).

Error analysis

For each chunker, 100 noun phrase errors and 100 verb phrase errors were randomly selected and manually classified into different error types (table 3). This error categorization has previously been used by Wermter et al. [12]. Both for noun and verb phrases, the majority of errors are due to coordination errors or incorrectly chunked parenthesized items. Most chunkers combine noun phrases that are separated by “and” or “or” into one noun phrase, which explains why the number of false negatives in the coordination category is about twice the number of false positives. For verb phrases, coordination errors are mostly made by Genia Tagger, OpenNLP, and MetaMap. Genia Tagger and OpenNLP often erroneously combine adjacent verb phrases into one verb phrase, whereas MetaMap splits a long verb phrase into smaller phrases. For example, in the sentence “IL-4 gene expression is tightly controlled at the level of transcription”, MetaMap annotates “is” as a be-verb, “tightly” as an adverbial phrase, and “controlled” as a verb phrase, whereas other chunkers annotate “is tightly controlled” as a verb phrase.

	Noun-phrase error type[a]							Verb-phrase error type[a]					
Chunker	Coor	Par	Verb	Adv	Adj	Noun	Total	Coor	Verb	Adv	Adj	Noun	Total
GATE chunker	5/10[b]	29/16	3/1	1/1	5/6	14/9	57/43	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.
Genia Tagger	8/20	32/12	2/3	1/1	4/7	5/5	52/48	18/35	23/9	4/4	1/5	1/0	47/53
Lingpipe	8/17	16/16	4/2	2/1	9/8	9/8	48/52	0/0	13/50	8/8	13/0	8/0	42/58
MetaMap	2/7	28/13	6/4	4/3	11/11	4/7	55/45	29/14	13/13	10/1	9/1	10/0	71/29
OpenNLP	12/24	11/7	4/4	1/2	7/11	9/8	44/56	11/21	25/25	0/0	5/11	1/1	42/58
Yamcha	13/31	9/17	1/1	3/2	5/8	5/5	36/64	1/1	38/42	1/1	1/10	5/0	46/54

Table 3. Number of false-positive and false-negative errors for different error types in noun-phrase and verb-phrase recognition.

[a] See http://www.biosemantics.org/index.php?page=chunk for a description of the error types and examples.
[b] 5 false positives/10 false negatives.

To test the impact of errors on sentence splitting, tokenization, and POS-tagging, we reran three of the chunkers (OpenNLP, Yamcha, and GATE) using the gold standard sentence, token, and POS annotations in the GENIA corpus. The other three chunkers had no option to use this gold standard information as their input and were not tested. For noun phrases, the F-score was 91.6% for OpenNLP (an improvement of 1.9 percentage point, cf. table 2), 89.0% for Yamcha (increase 0.9), and 78.2% for GATE (increase 2.2). For verb phrases, the F-score was 95.8% for OpenNLP (increase 0.1) and 94.5% for Yamcha (increase 0.4).

Usability

Apart from chunker performance, other considerations also play a role when a chunker has to be integrated with other components in a natural language processing system. These include coding language, ease of integration (documentation, well-defined APIs, technical support), execution speed, and possibility to train the chunker. Support of a training mode may be particularly important if the chunker has to be applied in an application field for which it was not trained before. Table 1 gives an overview of the different characteristics of the six chunkers. We assessed different aspects of usability by a point-scoring system, as shown in table 4. Overall, Lingpipe and OpenNLP score highest. Both are easy to integrate in other systems, provide detailed technical

information, are fast, and support a training mode. Lingpipe is the only system that provides technical support.

	GATE chunker	Genia Tagger	Lingpipe	MetaMap	OpenNLP	Yamcha
Recent release[a]	1	0	1	1	1	0
Technical support	0	0	1	0	0	0
Speed[b]	1	0	1	0	1	0
Supports training mode	0	0	1	0	1	1
Easily integrated with UIMA[c]	1	0	1	0	1	0
Documentation[d]	1	0	1	1	1	0
Annotates both NPs and VPs	0	1	1	1	1	1
Total score	4	1	7	3	6	2

Table 4. Usability assessment scores of the different chunkers.

[a] Most recent version released less than 2 years ago.

[b] Annotation speed is faster than the median speed of the chunkers (260 ms/abstract).

[c] Has UIMA interface or can be integrated into UIMA within 100 lines of code.

[d] Has comprehensive user manual and detailed API documentation.

Performance of the combined annotations

Table 5 shows the performance of the combined chunker annotations for different voting thresholds. The highest F-scores are obtained with a voting threshold of three: 92.8% for noun phrases and 96.3% for verb phrases. As compared to the best single chunker, OpenNLP, these results are 3.1 percentage point higher for noun phrases and 0.6 percentage point for verb phrases. Lowering the voting threshold improves recall and decreases precision, whereas a higher threshold improves precision and decreases recall. The highest recalls, for a voting threshold of one, are 98.5% for noun phrases and 99.5% for verb phrases (with moderate precisions of 59.9% and 72.7%, respectively). The highest precisions are 98.4% for noun phrases (threshold six) and 99.1% for verb phrases (threshold five), with recalls of 58.2% and 74.1%, respectively.

Voting threshold[a]	Noun phrases			Verb phrases		
	Precision	Recall	F-score	Precision	Recall	F-score
1	59.9	98.5	74.5	72.7	99.5	84.0
2	82.4	96.7	89.0	91.3	98.0	94.5
3	91.8	93.8	92.8	96.7	95.9	96.3
4	94.6	90.5	92.5	98.3	89.9	93.9
5	96.8	79.9	87.5	99.1	74.1	84.8
6	98.4	58.2	73.1	n.a	n.a	n.a

Table 5. Performance of the combined chunker annotations on the GENIA corpus for different voting thresholds.

[a] The minimum number of chunkers that have to agree on an annotation for it to be accepted as an annotation of the combined system.

DISCUSSION

To our knowledge, this is the first study to compare the performance of the whole chunking pipeline for both noun- and verb-phrase chunking in the biomedical domain, and the first that combines multiple existing systems to improve chunking performance. Our results indicate that OpenNLP performs best for both noun-phrase recognition and verb-phrase recognition, followed by Genia Tagger and Yamcha. The other three chunkers performed less well. The reason that MetaMap showed low performance on verb phrases is because it always splits a long verb phrase into smaller phrases. The other chunkers recognize verb phrases much better than noun phrases. One explanation is that verb phrases are typically less complex than noun phrases. For example, verb phrases do not include parenthesized or bracketed elements, which proved an important source of error for noun-phrase recognition. Also, most verb phrases contain only one word.

The two chunking systems in our comparison that use maximum entropy models, OpenNLP and Genia Tagger, showed the highest performance, both for noun phrases and for verb phrases. However, we should be cautious to suggest that these models generally perform better than the other methods because only a small number of systems were compared, because the difference with the next best performing chunker (Yamcha), which is SVM-based, is not large, and because there are factors other than the chunking method that may impact performance.

The precision and recall of noun-phrase chunking in our study is lower than the results reported by CoNLL-2000 [8], Wermter et al. [12], and Buyko et al. [18]). There are several possible reasons: (1) CoNLL-2000 used the WSJ corpus for training and testing. This corpus is likely easier to annotate than a biomedical corpus; (2) Wermter et al. [12] used the WSJ corpus for training and 200 abstracts from the GENIA beta version corpus for testing, and Buyko et al. [18] used 500 GENIA abstracts for training and testing. We used a large set of 1,499 GENIA abstracts for testing, which may have yielded a more precise performance estimate; (3) Previous studies used the gold standard sentence, token and POS annotations for testing chunking performance, whereas in this study those annotations were generated by the chunkers themselves. Indeed, when we tested OpenNLP and Yamcha using the gold standard annotations, similar results were obtained as previously reported.

Differences between the gold standard sentence, token, and POS annotations, or the annotations generated by the systems themselves, do not appear to have a major impact on chunking performance. For noun phrases, differences in F-scores for OpenNLP, Yamcha, and GATE varied between 0.9 and 2.2 percentage point. For verb phrases, differences were less than 0.5 percentage point. The usability of a chunker is not only determined by its performance, but also depends on its ease of integration with other systems, execution speed, and trainability. Lingpipe and OpenNLP scored high on each of these points.

We used a simple voting scheme to combine the annotations of the different chunkers. The combined annotations performed better than the best single chunker, both for noun phrases (F-scores 89.7% vs. 92.8%) and for verb phrases (F-scores 95.7% vs. 96.3%). The relatively small performance improvement for verb phrases may be explained by the fact that the performance for verb-phrase recognition is already quite high for most chunkers, which makes further improvement more difficult. Another reason may have to do with the number of chunkers that are used to create the combined annotations: five chunkers for verb phrases, six chunkers for noun phrases. Interestingly, when we left out one chunker when generating the combined noun phrases, performance always decreased, varying between 92.0% (OpenNLP left out) and 92.6% (GATE left out). This suggests that each chunker, even the worst performing, contributes to the improved performance of the combined annotations. One may speculate whether the combination of more than six chunkers would further improve the results.

Several previous studies also combined the annotations of different chunkers, using a variety of (weighted) voting techniques [9-11, 20, 21]. Contrary to our approach, however, in these studies only one chunking method was used to train different chunkers for various sets of input features. The difference in performance between the individual

chunkers was always small, and although in all studies the combined system performed better than the best single system, the difference in F-score was never larger than 0.8 percentage point. We used a simple voting scheme to combine the output of multiple existing chunkers and found a much larger performance improvement for noun phrases. The greater heterogeneity of the chunkers in our study appears to be advantageous in improving chunking performance.

Whether the chunking systems that we evaluated are good enough for application in practical NLP tasks is still an open question. Clearly, there seems to be room for improvement, in particular for noun-phrase detection. A simple alternative to improving the performance of individual chunkers, is to use the combined annotations of several chunking systems. We showed that a combined system for noun-phrase recognition performs substantially better than the best single system. Another consideration is that the system with the highest F-score is not necessarily always the best. Some NLP tasks require high precision, even if this implies moderate recall and F-score, whereas other tasks require high recall. The individual chunkers show some limited variability in their performance figures (cf. table 2), but none of the precision or recall values is really high. Our combination approach offers the possibility to vary the combined system across a large range of precision and recall by varying the voting threshold. Thus, the performance of a combined system can easily be tuned to best meet specific requirements.

Whether the current chunking results are good enough is also determined by the impact of different chunking errors on the performance of the whole information extraction pipeline. For example, it may well be that splitting or joining a verb phrase is less important than missing or inserting a noun or verb phrase when it comes to information extraction. It would be interesting to investigate the impact of chunking errors on real NLP tasks.

Few studies have compared the performance of chunkers on different test corpora. In the study by Wermter et al. [12], a significant performance loss (in the order of 4 percentage points) was observed when various chunking systems trained on the WSJ corpus were tested on a small subset of the GENIA corpus. Campbell et al. [22] argued that general-purpose NLP tools cannot be readily applied to the analysis of medical narratives, although this has been questioned for POS taggers [23]. There is some evidence that a system trained in one domain performs equally well when retrained in another domain. POS taggers trained on clinical text achieved similar performance as the same taggers trained on Penn Treebank [24, 25]. Buyko et al. [18] compared the performances of the NLP components of OpenNLP, including the chunker, when trained on the WSJ corpus and on two biomedical corpora (GENIA and PennBioIE). They conclude that the performance figures from the newspaper domain are comparable with those from the

biology domain. This weakly suggests that our results can be generalized to other domains provided that systems are properly retrained on domain-specific corpora.

CONCLUSION

OpenNLP performed best on both noun-phrase and verb-phrase recognition, closely followed by Genia Tagger and Yamcha. With respect to usability, Lingpipe and OpenNLP scored best. Combination of the annotations of the different chunkers by a simple voting scheme is a straightforward way to improve chunking performance, and allows to balance precision and recall of the combined system.

Error type	Example
Noun phrases	
Coordination/enumeration (Coo)	Corpus: *[Hormonal interactions]$_{NP}$ and [glucocorticoid receptors]$_{NP}$ in patients…*
	Chunker: *[Hormonal interactions and glucocorticoid receptors]$_{NP}$ in patients…*
Parenthesized/bracketed items (Par)	Corpus: *…incubated with [RA]$_{NP}$ ([10(-7) M]$_{NP}$) or DMSO solvent*
	Chunker: *…incubated with [RA]$_{NP}$ ([10]$_{NP}$([-7]$_{NP}$)[M]$_{NP}$) or DMSO solvent*
Verbs (Ver)	Corpus: *…the cell cycle has been identified and linked at [varying degrees]$_{NP}$*
	Chunker: *…the cell cycle has been identified and linked at [varying]$_{VP}$ [degrees]$_{NP}$*
Adverbs (Adv)	Corpus: *…in which only [extremely low levels]$_{NP}$ of HIV-1 expression are detected*
	Chunker: *…in which only extremely [low levels]$_{NP}$ of HIV-1 expression are detected*
Adjectives (Adj)	Corpus: *…and has been shown to have [immunomodulatory activities]$_{NP}$ in vivo.*
	Chunker: *…and has been shown to have immunomodulatory [activities]$_{NP}$ in vivo.*
Nouns (Nou)	Corpus: *The footprinted binding site is homologous to the [consensus AP1 motif]$_{NP}$*
	Chunker: *The footprinted binding site is homologous to the [consensus]$_{NP}$ [AP1 motif]$_{NP}$*
Verb phrases	
Coordination/enumeration (Coo)	Corpus: *TCF-1 alpha, [originally identified]$_{VP}$ and [purified]$_{VP}$ through its…*
	Chunker: *TCF-1 alpha, [originally identified and purified]$_{VP}$ through its…*
Verbs (Ver)	Corpus: *…the cell cycle has been identified and linked at [varying degrees]$_{NP}$*
	Chunker: *…the cell cycle has been identified and linked at [varying]$_{VP}$ [degrees]$_{NP}$*
Adverbs (Adv)	Corpus: *Constructs were [stably transfected]$_{VP}$ into murine erythroleukemia…*
	Chunker: *Constructs were [stably]$_{VP}$ [transfected]$_{VP}$ into murine erythroleukemia…*
Adjectives (Adj)	Corpus: *…TAD-B remains [unphosphorylated]$_{ADJP}$ by protein from…*
	Chunker: *…TAD-B remains [unphosphorylated]$_{VP}$ by protein from…*
Nouns (Nou)	Corpus: *Brief exposure to [fludarabine]$_{NP}$ [led to]$_{VP}$ a sustained loss of STAT1…*
	Chunker: *Brief exposure to [fludarabine]$_{VP}$ [led to]$_{VP}$ a sustained loss of STAT1…*

Appendix 1. Categorization of noun phrase and verb phrase error types

REFERENCES

1. Abney SP. Parsing by chunks. In: Berwick RC, Abney SP, Tenny C, editors. **Principle-based parsing: computation and psycholinguistics.** *Dordrecht: Kluwer Academic Publ; 1991. p. 257-78.*

2. Ramshaw LA, Marcus MP. Text chunking using transformation-based learning. **In: Proceedings of the Third ACL Workshop on Very Large Corpora**. *Cambridge, USA; 1995. p. 82-94.*

3. Pratt W, Yetisgen-Yildiz M. **A study of biomedical concept identification: MetaMap vs. people.** *In: Proceedings of the AMIA Annual Symposium. Washington; 2003. p. 529-33.*

4. I2b2. 2010 i2b2/VA Challenge Evaluation. Concept Annotation Guidelines. cited 2010 October 6. Available from: https://www.i2b2.org/NLP/Relations/assets/Concept%20Annotation%20Guideline.pdf.

5. Zhou GD, Su J. **Named entity recognition using an HMM-based chunk tagger.** *In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. Morristown, USA; 2001. p. 473-80.*

6. Aronson AR. **Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program.** *In: Proceedings of the AMIA Symposium. Washington; 2001. p. 17-21.*

7. Rinaldi F, Schneider G, Kaljurand K, Hess M, Andronis C, Konstandi O, Persidis A. **Mining of relations between proteins over biomedical scientific literature using a deep-linguistic approach.** *Artif Intell Med 2007;39:127-36.*

8. Sang EF, Buchholz S. **Introduction to the CoNLL-2000 shared task: chunking.** *In: Proceedings of CoNLL-2000 and LLL-2000. Lisbon, Portugal; 2000. p. 127-32.*

9. Kudoh T, Matsumoto Y. **Use of support vector learning for chunk identification.** *In: Proceedings of CoNLL-2000 and LLL-2000. Lisbon, Portugal; 2000. p. 142-4.*

10. Sang EF. **Text chunking by system combination.** *In: Proceedings of CoNLL-2000 and LLL-2000. Lisbon, Portugal; 2000. p. 151-3.*

11. Van Halteren H. **Chunking with WPDV models.** *In: Proceedings of CoNLL-2000 and LLL-2000. Lisbon, Portugal; 2000. p. 154-6.*

12. Wermter J, Fluck J, Stroetgen J, Geißler S, Hahn U. **Recognizing noun phrases in biomedical text: an evaluation of lab prototypes and commercial chunkers.** *In:*

SMBM 2005 - Proceedings of the 1st International Symposium on Semantic Mining in Biomedicine. Hinxton, England; 2005. p. 25-33.

13. Tateisi Y, Yakushiji A, Ohta T, Tsujii J. **Syntax annotation for the GENIA corpus. In: Proceedings of the IJCNLP.** *Jeju Island, Korea; 2005. p. 222-7.*

14. Cunningham H. GATE, **a general architecture for text engineering.** *Comput Hum 2002;36:223-54.*

15. Cunningham H, Maynard D, Bontcheva K, Tablan V, Ursu C, Dimitrov M, Dowman M, Aswani N, Roberts I, Li Y. **Developing language processing components with GATE version 5 (a user guide).** *University of Sheffield, 2009.*

16. Tsuruoka Y. **Bidirectional inference with the easiest-first strategy for tagging sequence data.** *In: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing. Vancouver, Canada; 2005. p. 467-74.*

17. Tersmette KWF, Scott AF, Moore GW, Matheson NW, Miller RE. **Barrier word method for detecting molecular biology multiple word terms.** *In: Proceedings of the 12th Annual Symposium on Computer Applications in Medical Care. Washington, USA; 1988. p. 207-21.*

18. Buyko E, Wermter J, Poprat M, Hahn U. **Automatically adapting an NLP core engine to the biology domain.** *In: Proceedings of the Joint BioLINKBio-Ontologies Meeting. Fortaleza, Brasil; 2006. p. 65-8.*

19. Ferrucci D, Lally A. **UIMA: an architectural approach to unstructured information processing in the corporate research environment.** *Natural Language Engineering 2004;10:327-48.*

20. Kudo T, Matsumoto Y. **Chunking with support vector machines.** *In: the North American Chapter of the Association for Computational Linguistics. Pittsburgh, PA, USA; 2001. p. 1-8.*

21. Sang EF. **Memory-based shallow parsing.** *J Mach Learn Res 2002;2:559-94.*

22. Campbell DA, Johnson SB. **Comparing syntactic complexity in medical and non-medical corpora.** *In: Proceedings of the AMIA Symposium. Washington, USA; 2001. p. 90-2.*

23. Wermter J, Hahn U. **Really, is medical sublanguage that different? Experimental counter-evidence from tagging medical and newspaper corpora.**

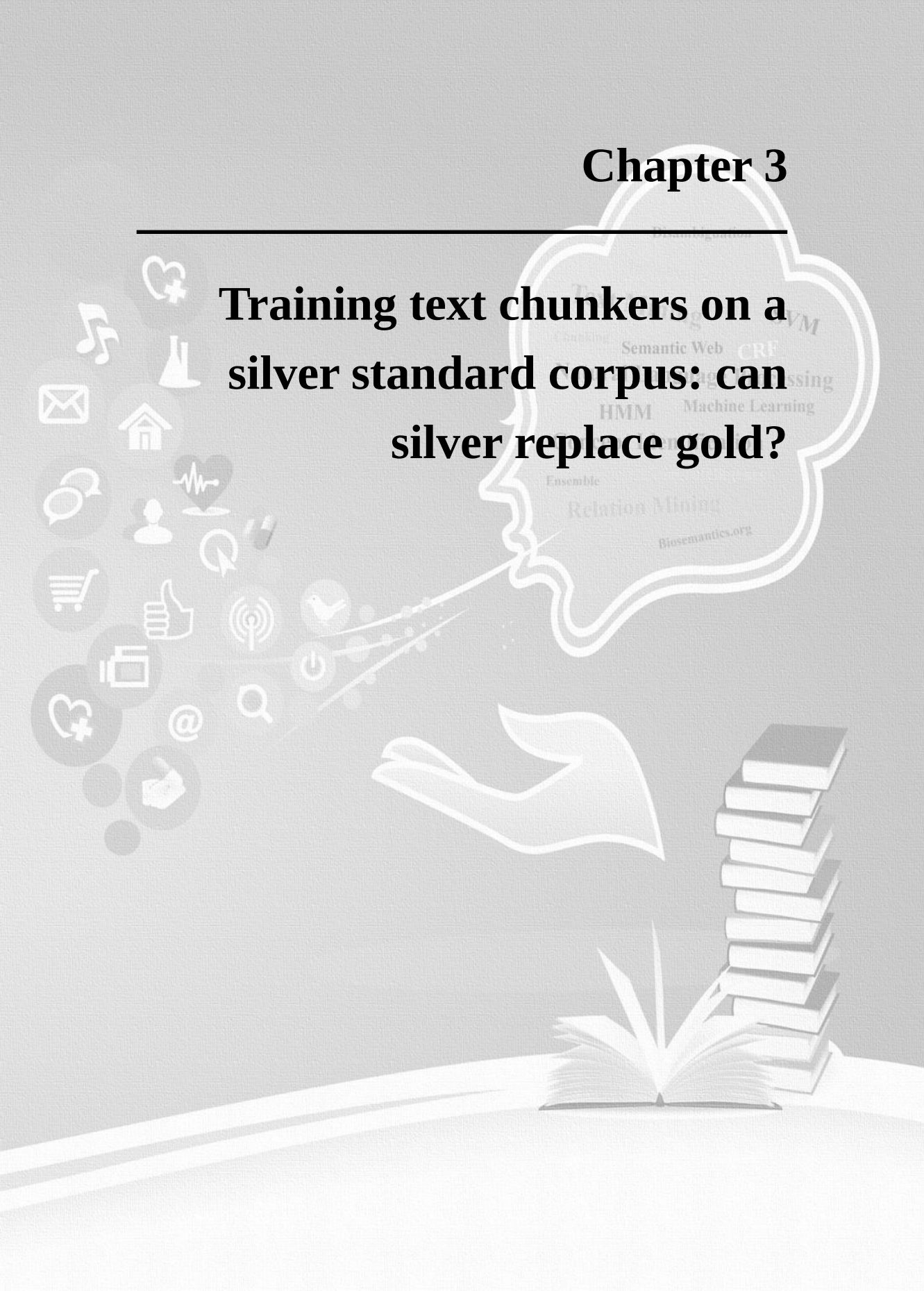
In: Medinfo 2004 - Proceedings of the 11th World Congress on Medical Informatics. San Francisco, USA; 2004. p. 560-4.

24. Pakhomov SV, Coden A, Chute CG. **Developing a corpus of clinical notes manually annotated for part-of-speech.** *Int J Med Inform 2006;75:418-29.*

25. Liu K, Chapman W, Hwa R, Crowley RS. **Heuristic sample selection to minimize reference standard training set for a part-of-speech tagger.** *J Am Med Inform Assoc 2007;14:641-50.*

Chapter 3

Training text chunkers on a silver standard corpus: can silver replace gold?



ABSTRACT

Background

To train chunkers in recognizing noun phrases and verb phrases in biomedical text, an annotated corpus is required. The creation of gold standard corpora (GSCs), however, is expensive and time-consuming. GSCs therefore tend to be small and to focus on specific subdomains, which limits their usefulness. We investigated the use of a silver standard corpus (SSC) that is automatically generated by combining the outputs of multiple chunking systems. We explored two use scenarios: one in which chunkers are trained on an SSC in a new domain for which a GSC is not available, and one in which chunkers are trained on an available, although small GSC but supplemented with an SSC.

Results

We have tested the two scenarios using three chunkers, Lingpipe, OpenNLP, and Yamcha, and two different corpora, GENIA and PennBioIE. For the first scenario, we showed that the systems trained for noun-phrase recognition on the SSC in one domain performed 2.7-3.1 percentage points better in terms of F-score than the systems trained on the GSC in another domain, and only 0.2-0.8 percentage points less than when they were trained on a GSC in the same domain as the SSC. When the outputs of the chunkers were combined, the combined system showed little improvement when using the SSC. For the second scenario, the systems trained on a GSC supplemented with an SSC performed considerably better than systems that were trained on the GSC alone, especially when the GSC was small. For example, training the chunkers on a GSC consisting of only 10 abstracts but supplemented with an SSC yielded similar performance as training them on a GSC of 100-250 abstracts. The combined system even performed better than any of the individual chunkers trained on a GSC of 500 abstracts.

Conclusions

We conclude that an SSC can be a viable alternative for or a supplement to a GSC when training chunkers in a biomedical domain. A combined system only shows improvement if the SSC is used to supplement a GSC. Whether the approach is applicable to other systems in a natural-language processing pipeline has to be further investigated.

BACKGROUND

Chunking is a natural language processing technique that splits text into groups of words that constitute a grammatical unit, e.g., a noun phrase or a verb phrase. It is an important processing step in systems that try to automatically extract information from text. Most chunkers are based on machine learning methods and require a text corpus annotated with chunks for training the system. The creation of a gold standard corpus (GSC) is tedious and expensive: annotation guidelines have to be established, domain experts must be trained, the annotation process is time-consuming, and annotation disagreements have to be resolved. As a consequence, GSCs in the biomedical domain are generally small and focus on specific subdomains, which limit their usefulness.

In this study we investigate an alternative, automatic approach to create an annotated corpus. We have shown before that a system combining the outputs of various chunkers performs better than each of the individual chunkers. Here we postulate that the annotations of such a combined system on a given corpus can be taken as a reference standard, establishing a "silver standard corpus" (SSC).

To test the practical value of this approach, we explore two use scenarios of such an SSC. In the first scenario, a chunker has to be trained for a biomedical subdomain for which a GSC is not available. Rather than creating a new GSC, we generate an SSC for the new domain and train the chunker on the SSC. In the second scenario, a GSC from the domain of interest is available but its size is small and a chunker trained on it gives suboptimal performance. Rather than expanding the GSC, we supplement the GSC with an SSC from the same domain and train the chunker on the combined GSC and SSC to improve chunker performance.

Related work

During the past decade, much research has been devoted to systems that combine different classifiers, also called multiple classifier systems or ensemble-based systems [1]. The general idea is that the combined wisdom of multiple classifiers reduces the risk of errors, and indeed it has been shown many times that a combined system performs better than the best individual classifier. Multiple classifier systems have been applied in many domains, including biomedical text mining and information extraction. For instance, Smith et al. [2] combined the results of 19 systems for gene mention recognition, and found that the combined system outperformed the best individual system by 3.5 percentage points in terms of F-score. Kim et al. [3] combined eight systems for event extraction and showed that the performance of the combined system increased by 4

percentage points as compared to the best individual system. We previously combined six publicly available text chunkers using a simple voting approach [4]. The F-score of the combined system improved by 3.1 percentage points for noun-phrase recognition and 0.6 percentage point for verb-phrase recognition as compared to the best single chunker.

The notion that a combination of systems can be used to create a "silver standard" corpus has been explored in the CALBC (Collaborative Annotation of a Large Biomedical Corpus) project [5]. Through CALBC, the natural-language processing community has been invited to annotate a very large biomedical corpus with a variety of named-entity recognition systems. The combined annotations of multiple systems may provide a valuable resource for system development and evaluation, and the automatically generated creation of an SSC would allow corpora of unprecedented size. In a very recent study, Chowdhury and Lavelli compared a gene recognition system trained on an initial version of the CALBC SSC against the system trained on the BioCreative GSC [6]. The system trained on the SSC performed considerably worse than when trained on the GSC, but the authors propose several ways to automatically improve the quality of the SSC and are of the opinion that, in the absence of a GSC, a system trained on the SSC could be useful in the semi-automatic construction of a GSC.

METHODS

Chunking systems

To generate a silver standard, we used five well-known and publicly available chunkers: GATE chunker 5.0 [7], Lingpipe 3.8 [8], MetaMap 2008v2 [9], OpenNLP 2.1 [10], and Yamcha 0.33 [11]. Three of these chunkers are trainable (Lingpipe, OpenNLP, Yamcha), the other two do not have a training option. Sentence splitting, tokenization, and part-of-speech tagging were included in our chunking pipeline, either as integral part of the chunkers (Yamcha, Lingpipe) or as separate components (OpenNLP). We used the gold-standard sentence, token, and part-of-speech annotations for training, but did not use this information in creating the SSC or evaluating the trained models: the input of the annotation pipeline consisted of plain abstracts, the output were chunking annotations. All chunkers annotate noun phrases and verb phrases, except for GATE which only generates noun phrases. More information on characteristics and performance of these chunkers can be found in our previous comparative study of chunkers [4], which also included Genia Tagger. Since Genia Tagger comes with a fixed pre-trained model based on the corpora that we use in this study, it could bias the results of our experiments and was not included. All chunkers were used with default parameter settings.

Corpora

There are only a few publicly available corpora in the biomedical domain that incorporate chunk annotations. We used the GENIA Treebank corpus [12] and the PennBioIE corpus [13].

The GENIA corpus [12] has been developed at the University of Tokyo. The 1.0 version of the corpus was released in 2009 and consists of 1,999 Medline abstracts selected from a query using the MeSH terms "human", "blood cells", and "transcription factors". The corpus has been annotated with various levels of linguistic and semantic information, such as sentence splitting, tokenization, part-of-speech tagging, chunking annotation, and term-event information. For chunker training, we selected a subset of 500 abstracts that constituted a previous version of the GENIA corpus [12].

The PennBioIE Treebank corpus [13] has been developed at the University of Pennsylvania. The 0.9 version of the corpus was released in 2004 and includes the CYP and Oncology corpora of the Linguistic Data Consortium. The CYP corpus consists of 324 Medline abstracts on the inhibition of cytochrome P450 enzymes. The Oncology corpus consists of 318 Medline abstracts on cancer and molecular genetics. The corpus has been tokenized and annotated with paragraph, sentence, part-of-speech tagging, chunking annotation, and biomedical named-entity types.

Creation of the silver standard

We used a simple voting scheme to generate silver standard annotations from the annotations produced by the different chunkers. For each phrase identified by a chunker, the number of chunkers that gave exactly matching annotations was counted. If the count was larger than or equal to a preset voting threshold, the phrase was considered a silver standard annotation, otherwise it was not. In all our experiments, we used a voting threshold of three out of five chunkers for noun phrases, and a threshold of two out of four for verb phrases (GATE only generates noun phrases). These thresholds gave uniformly the best results in terms of F-score when the silver standard annotations of the training data were evaluated against the gold standard. The Unstructured Information Management Architecture (UIMA) framework [14] was used to integrate all chunking systems and combine their result.

Silver standard as alternative for gold standard

To test whether an SSC could serve as a substitute for a GSC, we compared the performance of chunkers trained on silver standard annotations of the abstracts in the PennBioIE corpus with the performance of the chunkers trained on the gold standard annotations of the same corpus. To create the SSC, the trainable chunkers (Lingpipe, OpenNLP, Yamcha) were trained on the gold standard annotations of 500 abstracts of the GENIA corpus. The chunkers then annotated the PennBioIE corpus and the annotations of all chunkers were combined to yield the silver standard. Subsequently, Lingpipe, OpenNLP, and Yamcha were trained on the PennBioIE SSC and on the PennBioIE GSC, using 10-fold cross-validation. In the cross-validation procedure for the SSC, the annotations of the abstracts in each test fold were taken from the GSC. Thus, the performance of chunkers trained on either SSC or GSC was always tested on the GSC.

Silver standard as supplement of gold standard

To test whether an SSC would have additional value as a supplement for a given GSC, we compared the performance of chunkers trained on a subset of the GENIA GSC with the performance of the chunkers trained on the same subset supplemented with an SSC. Specifically, subsets of 10, 25, 50, 100, and 250 abstracts were selected from the initial GENIA training set of 500 abstracts, each subset being contained in the next larger one. Lingpipe, OpenNLP, and Yamcha were trained on the gold standard annotations of each subset and the total set, and tested on the 1,499 GENIA abstracts that were not used for training. For each subset, the chunkers trained on that subset were subsequently used to create an SSC of the abstracts in the set of 500 abstracts that were not part of the subset, i.e., for the GSC subset of 10 abstracts, the SSC consisted of the remaining 490 abstracts; for the subset of 25 abstracts, the SSC consisted of 475 abstracts; etc. The GSC and corresponding SSC (together always totaling 500 abstracts) were then used to train the chunkers. Their performance was tested again on the 1,499 GENIA abstracts not used for training. The above experiment was repeated 10 times, each time starting with a different randomly selected subset of 10 abstracts. The reported results are the averaged F-scores of the 10 experiments.

Performance evaluation

The chunker and silver standard annotations were compared with the gold standard annotations by exact matching, similar to the procedure followed in CoNLL-2000 [15]. An annotation was counted as true positive if it was identical to the gold standard annotation, i.e., both annotations had the same start and end location in the corpus. A

phrase annotated by the gold standard was counted as false negative if the system did not render it exactly; a phrase annotated by a system was counted as false positive if it did not exactly match the gold standard. Performance of the chunkers and silver standard was evaluated in terms of precision, recall, and F-score.

To reduce the effect of insignificant differences between chunks, words from the stopwords list in PubMed [16] and punctuation remarks were removed before matching if they appeared at the start or the end of a phrase. For instance, "[the protein's binding site on the DNA molecule]NP is..." is considered the same annotation as "the [protein's binding site on the DNA molecule]NP is...", and "the medicine [often causes]VP…" is considered the same as "the medicine often [causes]VP…".

RESULTS

Silver standard as alternative for gold standard

Table 1 shows the performance of the three trainable chunkers and the combined system on the PennBioIE GSC when trained on three different corpora: GENIA GSC, PennBioIE SSC, or PennBioIE GSC. GATE and MetaMap could not be trained and when tested on the PennBioIE GSC had F-scores of 78.2% (MetaMap) and 72.8% (GATE) for noun phrases, and 77.7% (MetaMap) for verb phrases. Clearly, the trainable chunkers perform better if they are trained on the PennBioIE SSC than on the GENIA GSC. The increase in F-scores varies between 1.7 and 3.1 percentage points for noun phrases and between 1.0 and 3.3 percentage points for verb phrases. Although performance further increases when training on PennBioIE GSC instead of PennBioIE SSC, differences are not large: 0.2 to 0.8 percentage point for noun phrases, 0.3 to 1.7 percentage point for verb phrases. OpenNLP consistently shows the best performance both for noun and verb phrases. The combined system performs better than any of the individual chunkers, including GATE and MetaMap which proved to have F-scores lower than each of the three trainable chunkers, in agreement with our previous findings [4]. The largest improvement of the combined system is seen when the individual chunkers are trained on the GENIA GSC. Remarkably, the performance difference between the combined systems based on GENIA GSC and PennBioIE SSC is only small (0.2 percentage point). To test the consistency of this result, we redid the experiment with interchanged corpora, i.e., GENIA GSC was used for training the chunkers and generating the SSC, and PennBioIE GSC was used for testing. The F-score of the combined system by using GENIA SSC for training was 0.5 (noun phrases) and 0.4 (verb phrases) percentage point better than the F-score of the combined system by using PennBioIE GSC for training, which is comparable with the results of the initial experiment.

System	Training set for noun phrases			Training set for verb phrases		
	GENIA GSC	PennBioIE SSC	PennBioIE GSC	GENIA GSC	PennBioIE SSC	PennBioIE GSC
Lingpipe	75.8%	78.5%	78.7%	90.6%	91.6%	91.9%
OpenNLP	80.8%	83.9%	84.7%	90.7%	93.2%	94.8%
Yamcha	80.1%	83.2%	84.0%	89.5%	92.8%	94.2%
Combined	84.3%	84.5%	87.2%	93.7%	93.9%	95.5%

Table 1. Performance (F-score) of chunkers and their combination when trained for noun-phrase and verb-phrase recognition on different training sets. All systems are tested on the PennBioIE corpus.

Silver standard as supplement of gold standard

Table 2 shows the performances of chunkers and the combined system when trained on GSCs of varying sizes and on the GSCs supplemented with an SSC. For all sizes of the GSC, the systems trained on a combination of GSC and SSC always perform better than the systems trained on the GSC alone. Clearly, the improvement is largest for small sizes of the GSC, leveling off with increasing size. The performance obtained with a small set of GSC abstracts combined with an SSC is comparable to a larger GSC set without SSC. For instance, each system trained on a GSC of only 10 abstracts supplemented with the SSC performs better than the system trained on a GSC of 100 abstracts alone; For larger GSC sizes, the performance of OpenNLP or Yamcha trained on 100 or 250 GSC abstracts plus the SSC is within 1 percentage point of the performance of the system trained on the next larger size of the GSC alone (250 and 500 abstracts, respectively).

	Lingpipe		OpenNLP		Yamcha		Combined	
GSC size	GSC	GSC+ SSC	GSC	GSC+ SSC	GSC	GSC+ SSC	GSC	GSC+ SSC
Noun phrases								
10	65.8%	80.8%	83.0%	87.9%	82.7%	85.6%	86.8%	90.7%
25	72.2%	81.1%	85.7%	88.3%	84.3%	86.0%	87.9%	90.9%
50	76.8%	81.3%	87.5%	88.6%	85.4%	86.2%	88.9%	91.2%
100	78.2%	81.9%	87.9%	88.9%	85.6%	86.6%	89.3%	91.5%
250	82.4%	82.8%	88.3%	89.3%	86.7%	87.2%	90.6%	92.0%
500	84.5%	n.a	89.7%	n.a	88.1%	n.a	92.8%	n.a
Verb phrases								
10	64.1%	86.9%	84.3%	93.6%	86.2%	92.5%	91.3%	94.6%
25	73.8%	87.3%	88.8%	94.0%	89.7%	92.9%	93.0%	94.9%
50	79.2%	87.6%	92.1%	94.4%	91.7%	93.1%	94.4%	95.5%
100	83.6%	87.9%	93.6%	94.7%	92.3%	93.4%	95.4%	95.8%
250	88.3%	88.7%	95.0%	95.3%	93.8%	93.9%	95.8%	96.0%
500	90.3%	n.a	95.7%	n.a	94.1%	n.a	96.3%	n.a

DISCUSSION

We have investigated the use of an SSC as a substitute or a supplement of a GSC for training chunkers in the biomedical domain. The SSC as a substitute for a GSC corresponds with a use scenario in which a chunker created for one subdomain has to be adapted to another, where a GSC for the new domain is not available. We have shown that a system trained on an SSC for the new domain performs considerably better than if that system is trained on the GSC of another subdomain, and only slightly worse (<1 percentage point) than if the system was trained on a GSC for the new domain. In the second use scenario, we supplemented a (small) GSC with an SSC for the same domain as the GSC. The addition of the SSC always improved the chunker performance, particularly if the size of the initial GSC was small.

Our results on the practical value of an SSC are different from those that were recently reported by Chowdhury and Lavelli [6]. They found a considerable drop in performance of a gene recognition system trained on the CALBC SSC as compared to the system trained on the BioCreative GSC, and also noticed that the system trained on a combination of SSC and GSC performed worse than on the GSC only. There may be several reasons for these differences. One is that the SSC that we used for training the chunkers was evaluated against the GSC of the same subdomain, whereas in the other study the domains from which the CALBC SSC and the BioCreative GSC are taken, are more divergent. Another possible reason is that the quality of the CALBC SSC is simply not good enough, which may be related to the difficulty of the CALBC task. Named entity recognition is generally considered more difficult than chunking, having to deal with increased complexities in boundary recognition, disambiguation, and spelling variation of entities. Clearly, the better a silver standard will approach a gold standard for the domain of interest, the better the performance of systems trained on an SSC. It should be noted that the performance of the silver standard compared with the gold standard in our study is far from perfect: the PennBioIE SSC has an F-score of 84.5% for noun phrases and 93.9% for verb phrases. Performance figures of the CALBC SSC against GSCs for named-entity recognition are not yet available, but we presume that they will be much lower. However, despite the differences between an SSC and GSC, chunking systems trained on these corpora showed remarkably similar performances. It is still an open question how an SSC of lower (or higher) quality affects the performance of a system trained on the SSC.

We used a simple voting approach to create an SSC. More sophisticated voting methods exist, such as weighted voting [17] or Borda count [18], but these methods require information about the confidence or rank of the chunks, information that is not available for the chunkers in this study. We also tested a combined system based on the output of the three trainable chunkers instead of all five chunkers. When trained on GENIA GSC and tested on PennBioIE GSC, the F-score of the combined system dropped to 82.1% for noun phrases and 91.9% for verb phrases. Since this performance is considerably lower than that of the combined system based on all chunkers, we did not further pursue the use of an SSC based on the three trainable chunkers only.

We used exact matching in performance assessment of the chunkers and creation of the SSC. By removing stopwords before matching we tried to remove “uninformative” words that should not play a role in determining whether phrases are the same, similar to other studies (e.g., [19, 20]). Our main consideration to remove stopwords was that chunking is usually an intermediate step in the information extraction pipeline, and whether an unimportant word (e.g., “the” at the start of a noun phrase) is detected or not,

is unlikely to affect subsequent processing steps (e.g., named entity recognition). Stopword removal can be seen as a relaxation of the strict matching requirement. When systems trained on GENIA GSC were tested on PennBioIE GSC but without removing stopwords, performances dropped by 3.7-5.5 percentage points for noun phrases and 3.6-6.3 percentage points for verb phrases. This shows that chunkers may considerably differ with the gold standard with respect to the annotation of stopwords. We did not want to further relax the matching criterion, e.g., by allowing partially matching boundaries, first because this would produce matches between phrases that differ in other than uninformative words (and thus should be considered different), and second because it is not obvious how partially matching phrases should be combined in a single phrase for inclusion in the SSC.

Since the creation of an SSC is automatic, its size can be very large. For different text-processing applications, increasing amounts of data for training classifiers have been shown to improve classifier performance [21-23]. Use of an SSC may be beneficial in mitigating the "paucity-of-data" problem [21].

The combination of systems always performed better than any of the individual systems, but performance increase of the combined system was larger when the individual systems were trained on GENIA or PennBioIE GSCs than when they were trained on the PennBioIE SSC (cf. Table 1). A possible explanation for this phenomenon is that the SSC incorporates results from the chunkers that are subsequently trained on it. As a consequence, the diversity of the chunkers trained on the SSC may be less than those trained on the GSCs. Indeed, when we pairwise determined the F-score between two chunkers trained on GENIA GSC and PennBioIE GSC, the average score was 78.2% and 80.2%, respectively, in comparison to 87.4% for PennBioIE SSC (without stopword removal these figures were 72.6%, 73.9%, and 82.5%, respectively). This indicates better agreement between the chunkers (less diversity) for the SSC. Since annotation diversity is generally considered a key factor for the improvement seen by ensemble systems (4), it may be expected that the combined chunker system shows a smaller increase of performance when based on the SSC than on the GSCs.

We showed that chunkers can obtain almost similar performances whether trained on an SSC or a GSC, but this does not mean that we can dispose of GSCs altogether. Obviously, to create the SSC we need trained chunkers, and thus a GSC for their initially training. We explored the use of a GSC from another, but related, domain than the domain of interest. Alternatively, we supplemented a GSC with an SSC in the same domain of interest. Using this approach, good results can be achieved with remarkably small-sized GSCs. Our experiments indicated that a GSC consisting of only 10 or 25 abstracts but expanded with an SSC yields similar performances as a GSC of 100 or 250 abstracts.

Practically, these results suggest that the time and effort spent in creating a GSC of sufficient size may be much reduced.

We have tested two use scenarios of an SSC in the field of text chunking, but the proposed approach is general and could be used in any field in which GSCs are needed to train classifiers. Further investigations will have to reveal how the quality of an SSC affects classifier performance and whether the use of SSCs in other application areas is equally advantageous as their use in text chunking.

CONCLUSIONS

We have shown that an automatically created SSC can be a viable alternative for or a supplement to a GSC when training chunkers in a biomedical domain. A combined system only shows improvement if the SSC is used to supplement a GSC. Our results suggest that the time and effort spent in creating a GSC of sufficient size may be much reduced. Whether the approach is applicable to other systems in a natural-language processing pipeline has to be further investigated.

REFERENCES

1. Polikar R: **Ensemble based systems in decision making**. *IEEE Circuit Syst Mag* 2006, **6**:21-45.

2. Smith L, Tanabe LK, Ando RJ, Kuo CJ, Chung IF, Hsu CN, Lin YS, Klinger R, Friedrich CM, Ganchev K, *et al*: **Overview of BioCreative II gene mention recognition.** *Genome Biol* 2008, **9**(Suppl 2):S2.

3. Kim J, Ohta T, Pyysalo S, Kano Y, Tsujii J: **Overview of BioNLP'09 shared task on event extraction**. In *Proceedings of the Workshop on BioNLP: Shared Task; Boulder*. 2009:1-9.

4. Kang N, van Mulligen EM, Kors JA: **Comparing and combining chunkers of biomedical text.** *J Biomed Inform* 2011, **44**:354-360.

5. Rebholz-Schuhmann D, Yepes AJ, van Mulligen EM, Kang N, Kors J, Milward D, Corbett P, Hahn U: **The CALBC silver standard corpus - harmonizing multiple semantic annotations in a large biomedical corpus**. In *Proceedings of the Third International Symposium on Languages in Biology and Medicine; Jeju Island, South Korea*. 2009:64-72.

6. Chowdhury MFM, Lavelli A: **Assessing the practical usability of an automatically annotated corpus**. In *Proceedings of the Fifth Linguistic Annotation Workshop; Portland*. 2011:101-109.

7. Cunningham H: **GATE, a general architecture for text engineering**. *Comput Humanities* 2002, **36**:223-254.

8. Carpenter B: **LingPipe for 99.99% recall of gene mentions**. In *Proceedings of the Second BioCreative Challenge Evaluation Workshop; Valencia*. 2007:307-309.

9. Aronson AR: **Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program**. In *Proceedings of the AMIA Symposium; Washington DC*. 2001:17-21.

10. Buyko E, Wermter J, Poprat M, Hahn U: **Automatically adapting an NLP core engine to the biology domain**. In *Proceedings of the Joint BioLINK-Bio-Ontologies Meeting; Fortaleza*. 2006:65-68.

11. Kudo T, Matsumoto Y: **Chunking with support vector machines**. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies; Pittsburgh*. 2001:1-8.

12. Tateisi Y, Yakushiji A, Ohta T, Tsujii J: **Syntax Annotation for the GENIA corpus**.

In *Proceedings of the Second International Joint Conference on Natural Language Processing; Jeju Island, South Korea*. 2005:222-227.

13. Kulick S, Bies A, Liberman M, Mandel M, McDonald R, Palmer M, Schein A, Ungar L, Winters S, White P: **Integrated annotation for biomedical information extraction**. In *Human Language Technology conference/North American Chapter of the Association for Computational Linguistics Annual Meeting; Boston*. 2004:61-68.

14. Ferrucci D, Lally A: **UIMA: an architectural approach to unstructured information processing in the corporate research environment**. *Nat Lang Eng* 2004, **10**:327-348.

15. Sang E, Buchholz S: **Introduction to the CoNLL-2000 shared task: chunking**. In *Proceedings of CoNLL-2000 and LLL-2000; Lisbon*. 2000:127-132.

16. **PubMed stopword list** [http://www.nlm.nih.gov/bsd/disted/pubmedtutorial/020_170.html].

17. Littlestone N, Warmuth MK: **The weighted majority algorithm**. *Inform Comput* 1994, **108**:212-261.

18. Van Erp M, Schomaker L: **Variants of the borda count method for combining ranked classifier hypotheses**. In *Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition; Amsterdam*. 2000:443-452.

19. Seki K, Mostafa J: **An application of text categorization methods to gene ontology annotation**. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and Development in Information Retrieval; Salvador, Brazil*. 2005:138-145.

20. Boyack KW, Newman D, Duhon RJ, Klavans R, Patek M, Biberstine JR, Schijvenaars B, Skupin A, Ma N, Börner K: **Clustering more than two million biomedical publications: comparing the accuracies of nine text-based similarity approaches**. *PLoS One* 2011, **6**:e18029.

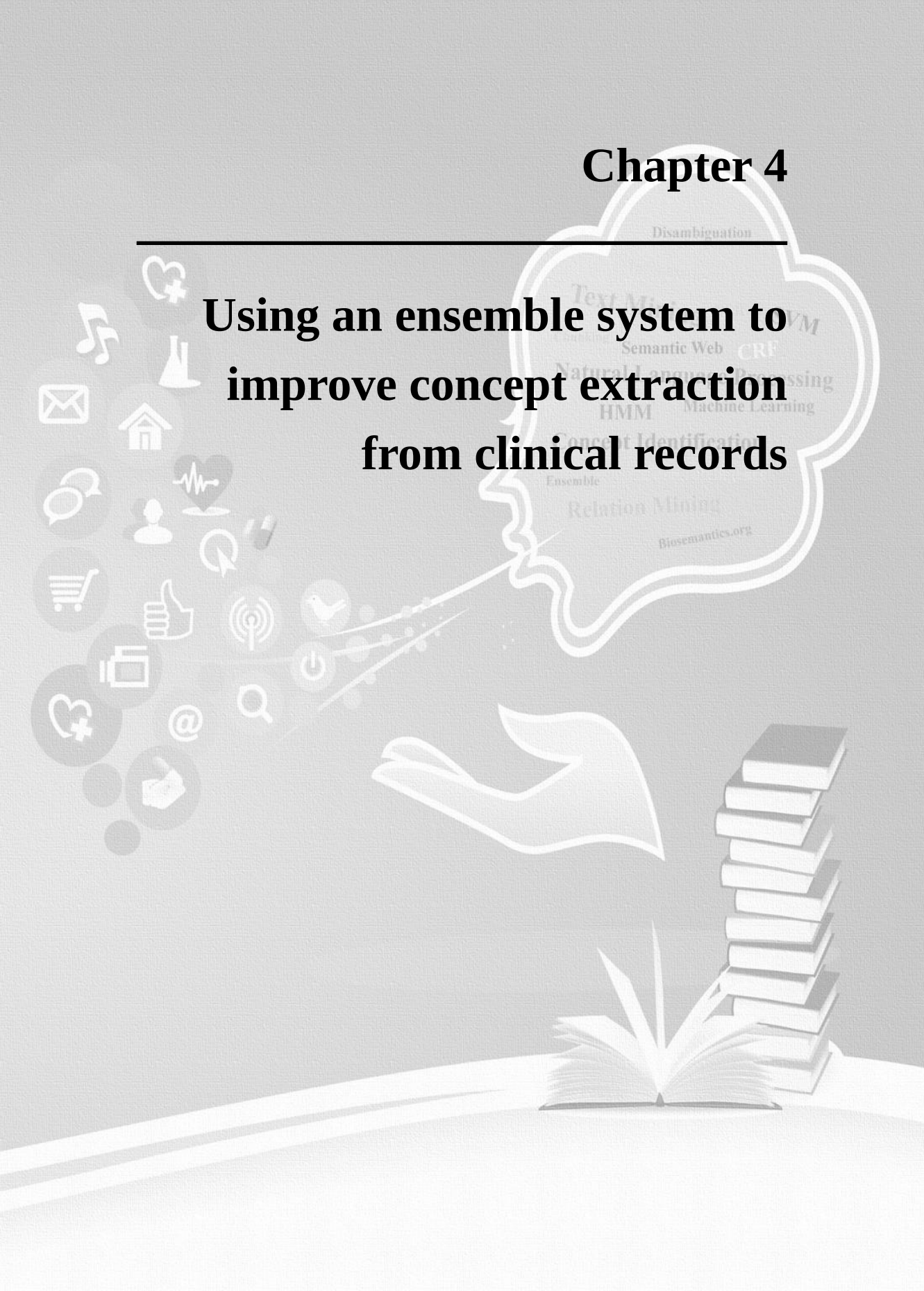
21. Banko M, Brill E: **Mitigating the paucity-of-data problem: exploring the effect of training corpus size on classifier performance for natural language processing**. In *Proceedings of the First International Conference on Human Language Technology Research; San Diego*. 2001:1-5.

22. Yarowsky D, Florian R: **Evaluating sense disambiguation across diverse parameter spaces**. *Nat Lang Eng* 2002, **8**:293-310.

23. Surdeanu M, Turmo J, Comelles E: **Named entity recognition from spontaneous open-domain speech**. In *Annual Conference of the International Speech Communication Association; Lisbon*. 2005:3433-3436.

Chapter 4

Using an ensemble system to improve concept extraction from clinical records



ABSTRACT

Recognition of medical concepts is a basic step in information extraction from clinical records. We wished to improve on the performance of a variety of concept recognition systems by combining their individual results.

We selected two dictionary-based systems and five statistical-based systems that were trained to annotate medical problems, tests, and treatments in clinical records. Manually annotated clinical records for training and testing were made available through the 2010 i2b2/VA (Informatics for Integrating Biology and the Bedside) challenge. Results of individual systems were combined by a simple voting scheme. The statistical systems were trained on a set of 349 records. Performance (precision, recall, F-score) was assessed on a test set of 477 records, using varying voting thresholds.

The combined annotation system achieved a best F-score of 82.2% (recall 81.2%, precision 83.3%) on the test set, a score that ranks third among 22 participants in the i2b2/VA concept annotation task. The ensemble system had better precision and recall than any of the individual systems, yielding an F-score that is 4.6 percentage point higher than the best single system. Changing the voting threshold offered a simple way to obtain a system with high precision (and moderate recall) or one with high recall (and moderate precision).

The ensemble-based approach is straightforward and allows the balancing of precision versus recall of the combined system. The ensemble system is freely available and can easily be extended, integrated in other systems, and retrained.

INTRODUCTION

Automated extraction of information from unstructured text in clinical records is a burgeoning field of research, with applications in clinical decision support, diagnostic coding of diseases, adverse event detection, and clinical text mining, among others [1, 2]. The recognition of named entities or concepts, such as medical problems, tests, and treatments, is a basic initial step in information extraction from clinical records. Many different methods for named entity recognition in the biomedical field have been developed, but no single method has yet been shown to generally perform best.

In this study, we leveraged the performance of a number of systems that recognize medical concepts in clinical records by combining the output of the individual systems. Apart from performance improvement, our approach should allow for easy adjustment of the balance between precision and recall of the ensemble system, which could be fitted for tasks that require either a high precision or a high recall. We tested our approach by partaking in the concept extraction task of the 2010 i2b2/VA (Informatics for Integrating Biology and the Bedside) challenge on clinical records [3]. The ensemble system, called ACCCA (A Combined Clinical Concept Annotator), is available as a web service or can be downloaded (http://www.biosemantics.org/ACCCA_WEB).

BACKGROUND

A number of systems have specifically been developed for information extraction from clinical records, e.g., HITEx [4], MedLEE [5], cTAKES [6], MPLUS [7], MEDSYNDIKATE [8], and BioTeKS [9]. These systems have been applied to many different tasks, e.g., detection of adverse events in medical records of hospitalized patients [10], extraction of family history from discharge summaries [11], and detection of signs of pneumonia in radiology reports [12], to name a few. Although these systems generally perform very well, many contain rule-based components that are not easily trained and may require considerable effort to adjust to the task at hand. Also, many of the systems are not publicly available, or only under a license construction.

In addition to these clinical record processing systems, there are numerous other tools that were originally designed for named-entity recognition (NER) in biomedical literature, but also have been applied to clinical records. A well-known example is MetaMap [13], a program that identifies concepts from the Unified Medical Language System (UMLS) Metathesaurus [14] in biomedical text. MetaMap is a dictionary-based system and cannot be automatically trained, but many other systems, such as Lingpipe [15] or tools from the OpenNLP suite [16], are based on a statistical model and can be

trained for a particular task if an appropriate training set is available.

Several recent reports [17-23] describe a number of systems that were used in the 2010 i2b2/VA challenge. These systems utilize a variety of machine learning classifiers or are based on existing NER tools that were retrained for the i2b2 concept annotation task. Most systems operate on large feature sets, derived from the text itself or from external sources, such as UMLS. Several systems combine the statistical model with postprocessing rules for error correction and disambiguation [17, 19, 22]. An overview of all systems that participated in the i2b2 challenge can be found in [3].

For our ensemble system, we selected five statistical and two dictionary-based NER systems. The statistical systems include ABNER [24], Lingpipe [15], OpenNLP Chunker [25], JNET [26], and StanfordNer [27]. They are all publicly available, can easily be trained, and utilize a variety of statistical models. The dictionary-based systems are MetaMap [13] and Peregrine [28], a concept-recognition tool developed in our institute. Both systems could easily be adapted to the i2b2 challenge task.

Ensemble systems, also called multiple classifier systems, combine the output of different classifiers, and have been shown to perform better than the best individual classifier [29]. They have been applied in many different fields, including that of biomedical text processing. For example, Smith et al. [30] combined the results of 19 systems for gene mention recognition in the BioCreative II corpus. They found that the combined system outperformed the best individual system by 3.5 percentage points in terms of F-score. In a study by Baumgartner et al. [31] with the same data set, the results of three systems for gene name recognition were combined. The combined system had an F-score that outperformed the best single system by 3.4 percentage point. In the same study six gene tagging systems were combined using a voting threshold of 1 in order to maximize recall. Kim et al. [32] combined eight systems for event extraction related to protein biology and showed that the performance of the combined system increased by 4 percentage points with respect to the best individual system. We previously combined six publicly available text chunkers using a simple voting approach [33]. The F-score of the combined system improved by 3.1 percentage points for noun-phrase recognition and by 0.6 percentage point for verb-phrase recognition as compared to the best single chunker.

Although ensemble systems appear to work well in various biomedical domains, it has not yet been investigated whether the approach is also effective for concept recognition in clinical records, which is regarded as more difficult than concept recognition in scientific literature [1]. The outcome is uncertain because there is still much unclarity about what the characteristics of individual systems should be for an ensemble system to

work. System diversity is generally acknowledged to be an important factor [29, 34], but although many measures have been proposed to quantify diversity, studies that correlated diversity measures with system performance have shown inconclusive results [35].

METHODS

Clinical records

The clinical records used in this research were provided by the i2b2 National Center for Biomedical Computing. The data consisted of discharge summaries from Partners Healthcare and Beth Israel Deaconess Medical Center, and discharge summaries and progress notes from the University of Pittsburgh Medical Center. All records had been manually annotated for three types of concepts or named entities (medical problems, tests, and treatments), according to guidelines provided by the i2b2/VA challenge organizers. A total number of 18550 medical problems, 12899 tests, and 13560 treatments were annotated. An example annotation is: "The patient had [increasing dyspnea]$_{\text{PROBLEM}}$ on exertion, he had [a bronchoalveolar lavage]$_{\text{TREATMENT}}$ performed, and [CBC]$_{\text{TEST}}$ was unremarkable." A set of 349 records was made available for training, and an additional set of 477 records was released for testing.

Concept annotation systems

We selected seven annotation systems that reflect a variety of approaches to the recognition of named entities. Two of them were dictionary-based systems, the other five were statistical named-entity recognizers and chunkers. All of them were downloaded directly from their official websites, except our locally developed Peregrine [28]. For all systems their default configurations and parameters were used for both training and testing, and no attempt was made to optimize their performance. The following systems were used.

1. ABNER (A Biomedical Named Entity Recognizer) (http://pages.cs.wisc.edu/~bsettles/abner/) is a software tool for text analysis in molecular biology [24]. The core of the system is a statistical machine learning system using a linear-chain conditional random field (CRF) model [36] with a variety of orthographic and contextual features. We used version 1.5, released in 2005.

2. Lingpipe (http://alias-i.com/lingpipe) is a suite of Java libraries for natural language processing, including part-of-speech (POS) tagging, named entity recognition, spelling correction, etc [15]. The Lingpipe chunker supports rule-based, dictionary-based, and

statistical chunking. We used the statistical chunker based on a hidden Markov model [37], which according to the Lingpipe website is the most accurate one. The version we used is 3.8, released in 2009.

3. MetaMap (http://mmtx.nlm.nih.gov/) is a dictionary-based system to identify concepts from the UMLS Metathesaurus in biomedical text [13]. Based on a minimal commitment parser texts are split into chunks, in which concepts are identified. MetaMap cannot be trained. MetaMap Transfer (MMTx) is a distributable version of MetaMap written in Java. We used the 2010 version. UMLS concepts identified by MetaMap were mapped to the three concept types in the i2b2/VA task following guidelines from the challenge organizers [3].

4. OpenNLP Chunker (http://opennlp.sourceforge.net) is made available by OpenNLP, an organizational center for open source projects related to natural language processing. An Unstructured Information Management Architecture (UIMA) [38] wrapper for OpenNLP has been developed by JULIE Lab (http://www.julielab.de) [25]. The wrapper divides the OpenNLP package into small modules that perform sentence detection, tokenization, POS tagging, chunking (OpenNLP Chunker), etc., which makes it easy to configure the pipeline for different purposes. OpenNLP Chunker is based on a maximum entropy model [39]. We used version 2.1, released in 2008.

5. JNET (JULIE Lab Named Entity Tagger) (http://www.julielab.de/) is a generic and configurable named entity recognizer [26]. The comprehensive feature set allows to employ JNET for most domains and entity types. JNET uses a CRF model. The version we used is 2.3, released in 2008.

6. Peregrine is a dictionary-based concept recognition and identification tool, developed at the Erasmus University Medical Center (http://biosemantics.org). Peregrine includes a number of disambiguation rules to improve performance [28]. For the i2b2/VA challenge, we used the UMLS 2009 Metathesaurus filtered for relevant semantic types, in combination with chunking annotations to improve precision. The UMLS semantic types were mapped to the clinical concept types as specified in the i2b2/VA challenge guideline.

7. StanfordNer (http://nlp.stanford.edu/software/CRF-NER.shtml) is a named entity recognizer developed by the Stanford Natural Language Processing Group [27]. It is based on a linear chain CRF model. We used version 1.1, released in 2009.

Training of systems

All five statistical systems were trained on the 349 records of the i2b2/VA training corpus.

The i2b2 record annotations were easily converted to the required input format for each of the systems. All systems used tokens and contextual information as their input features. Some systems (OpenNLP Chunker and JNET) also needed part-of-speech (POS) information. We used the OpenNLP POS module to generate the POS tags, and integrated them in the training records. As MetaMap and Peregrine are dictionary-based systems, training was not needed.

Processing and evaluation pipeline

All systems were integrated in the UIMA framework, which was easily accomplished since the systems either had UIMA components or a webservice interface. The 477 records in the i2b2/VA test set were read by the UIMA Collection Reader and annotated by each of the seven systems. Subsequently, for each record the annotation results of the systems were combined into a combined annotation, output in the i2b2/VA annotation format, and submitted for evaluation in the i2b2/VA challenge. Precision, recall, and F-score of the individual annotation systems and the ensemble system were computed for two boundary matching strategies: exact matching (both the start and the end of the system annotation must match the reference annotation), and inexact matching (at least one of the annotation boundaries must match). Exact and inexact matching was done both with and without the requirement that the concept types (problem, test, treatment) of the annotations should match.

Combination of annotations

We used a simple voting scheme to combine the results of the different systems. For each annotation by a system, the number of systems that exactly matched that annotation was counted. If the count was larger or equal than a preset voting threshold, the annotation was considered to be confirmed by the combined system, otherwise it was discarded. In case systems made two different but overlapping annotations that both qualified for the voting threshold, the annotation that was supported by the largest number of systems was selected and the other one discarded. If the number of systems that supported each overlapping annotation was the same, one annotation was randomly selected.

RESULTS

Performance of individual and combined annotation systems

The performance of the seven individual annotation systems on the test set, as well as

the performance of the combined system with and without incorporating MetaMap, are given in Table 1. The voting thresholds for the two combined systems were based on the thresholds that gave the highest F-score on the training set: 3 for the system with MetaMap, 2 for the system without. In terms of F-score, the three trainable named-entity recognizers (ABNER, JNET, and StanfordNer) performed best, the dictionary-based systems (Peregrine and MetaMap) performed worst. Since performance of the combined system increased, however slightly, when MetaMap was excluded, we chose to do our further analyses based on the ensemble system without MetaMap.

Annotation system	Recall	Precision	F-score
ABNER	69.3	79.6	74.1
JNET	76.5	78.8	77.6
Lingpipe	74.0	73.1	73.5
Metamap	21.2	22.5	21.8
OpenNLP Chunker	63.3	78.4	70.0
Peregrine	40.0	56.5	46.8
StanfordNer	72.3	82.0	76.8
Combined system (with MetaMap)	81.0	83.1	82.0
Combined system (without MetaMap)	81.2	83.3	82.2

Table 1. Performance of the annotation systems on the i2b2 test set.

Recall, precision, and F-score of this system are all higher than those of any individual system. The improvement in F-score of the combined system as compared to the best performing single system, JNET, is 4.6 percentage points. The F-score of this ensemble system ranked third among the performances of 22 different systems that participated in the i2b2/VA concept annotation task. When we derived the combined annotation from five rather than six individual systems, the F-score of the combined system decreased by 0.1 percentage point (leaving out Peregrine) to 0.9 percentage point (leaving out JNET). When we stepwise varied the number of systems in the ensemble system from six to three, at each step removing the system with the smallest performance contribution to the ensemble, performance decreased from 82.2% (six systems) to 82.1% (five systems, Peregrine removed), 81.1% (four systems, OpenNLP Chunker removed), and 79.7%

(three systems, Lingpipe removed).

To measure the diversity between the systems, we pairwise determined the F-score and assumed that the higher the F-score, the lower the diversity (an F-score of 100% indicating perfect agreement, i.e., no diversity). The F-scores ranged from 47.2% (MetaMap and JNET) to 82.1% (JNET and StanfordNer). The averaged diversity between the dictionary-based systems and the statistical systems (57.5%) was much higher than the diversity between MetaMap and Peregrine (73.6%), or than the averaged diversity between the five statistical systems (77.3%).

Effect of varying the voting threshold

When we varied the voting threshold from 1 to 6, precision increased from 60.7% to 97.6%, and recall dropped from 84.0% to 19.3% (Figure 1). The F-score was highest for a threshold of 2.

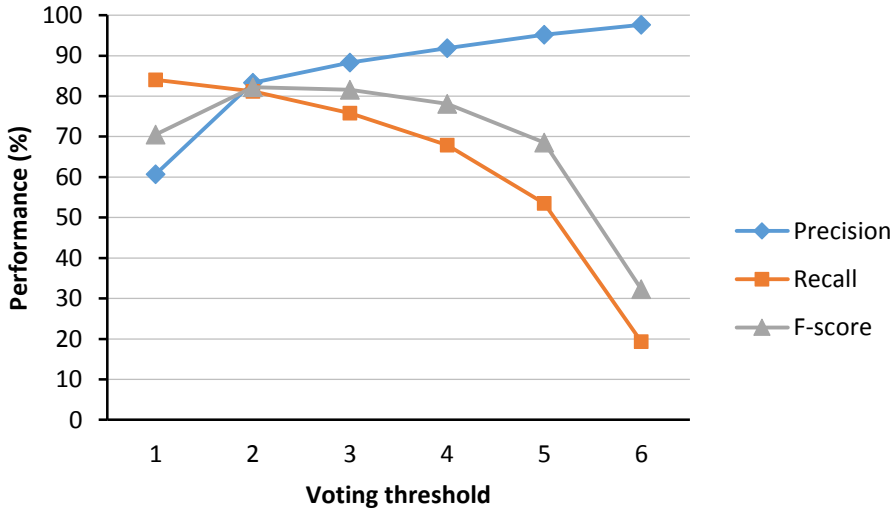


Figure 1. Performance of the ensemble system for varying voting threshold.

Error analysis

Table 2 shows detailed performance results of the combined system with voting threshold 2 for different matching strategies. A number of errors were due to mismatches in concept

type (problem, test, treatment) of the combined annotation and the reference annotation. When the concept types were not required to match, the F-score increased by 2.5 percentage points, to 84.7%. If additionally only one of the boundaries of the combined annotation had to match the reference annotation, the F-score further rose to 92.3%. The remaining errors (false positive and false negative annotations) had a mismatch at both the start and the end of the annotation boundaries. The differences in F-scores of the combined system for each of the three concept types were at most 1.8 percentage point. For the individual systems, these differences were within 3 percentage points.

Boundaries	Concept type	Recall	Precision	F-score
Exact	Same	81.2	83.3	82.2
Exact	Different	83.6	85.8	84.7
Inexact	Same	90.9	90.3	90.6
Inexact	Different	91.2	93.5	92.3

Table 2. Performance of the combined annotation system for different boundary and concept type matching of the annotations against the reference.

In a further analysis, we randomly selected ten clinical records and categorized the errors made by the ensemble system, based on exact match. It was found that 35% of the errors were caused by completely wrong or missed annotations, 20% were caused by overlapping but not exactly matching annotations, 16% by punctuation differences, 15% by coordination handling, and 14% of the errors were caused by wrong concept types. Table 3 shows some examples of these errors. Note that overlap, punctuation, and coordination errors disappear if inexact boundary matching is applied.

Error type	Example
wrong/missed	Reference: *...[new T wave inversion in III]* $_{PROBLEM}$ *in patients...*
	System: *...new T wave inversion in III in [patients]* $_{PROBLEM}$ *...*
overlap	Reference: *He denies [increased urinary frequent]* $_{PROBLEM}$ *...*
	System: *He denies increased [urinary frequent]* $_{PROBLEM}$ *...*
punctuation	Reference: *... For [Pain]* $_{PROBLEM}$ *, Mild (1-3), ...*
	System: *... For [Pain, Mild]* $_{PROBLEM}$ *(1-3), ...*
coordination	Reference: *He denies [increased urinary frequent]* $_{PROBLEM}$ *or [urgency]* $_{PROBLEM}$ *...*
	System: *He denies [increased urinary frequent or urgency]* $_{PROBLEM}$ *...*
concept type	Reference: *Exam remarkable for [b / l carotid bruits]* $_{PROBLEM}$ *...*
	System: *Exam remarkable for [b / l carotid bruits]* $_{TREATMENT}$ *...*

Table 3. Examples of common errors made by the combined annotation system.

DISCUSSION

To our knowledge, this is the first study showing that the recognition of medical concepts in clinical records can be improved considerably by combining the output of different annotation systems through a simple voting scheme. The ensemble system had higher precision, recall, and F-score values than any of the individual systems considered in this study. In terms of F-score, the combined system outperformed the best single system, JNET, by 4.6 percentage point. The system is freely available (http://www.biosemantics.org/ACCCA_WEB), can easily be extended or integrated with other systems, and can be retrained for other tasks.

The statistical systems that were designed for NER (ABNER, JNET, StanfordNer) performed better than the other systems, as might be expected. Remarkably, the two chunkers in our study, Lingpipe and OpenNLP Chunker, which we used here for a concept recognition task, did not lag far behind in performance. The low performance of MetaMap and Peregrine may partly be explained by the use of UMLS, which is not specifically geared towards terms in clinical records. Also, these systems carry out concept identification, a more difficult task than concept recognition [40], and then assign the identified concepts to the three categories at hand, a somewhat roundabout

way of named entity recognition. A better filtering of UMLS terms, possibly expanded with the reference annotations in the i2b2/VA training corpus, would likely improve these systems' results. Another factor that may well have affected the results, is that the statistical systems were specifically trained for the current task, whereas the dictionary-based systems were not.

Contrary to other top-ranking systems in the i2b2/VA challenge [17, 20, 22, 23], we did not try to optimize the parameters of the individual systems in our study, nor did we use more advanced contextual features, e.g., those based on negation or speculation detection, which might have further improved the performance of our systems. It is notable that even with the simple and straightforward approach that we took, the ensemble system ranked among the best-achieving systems in the i2b2 challenge, which shows the practicality and viability of the approach.

Removal of the worst performing system, MetaMap, from the ensemble system slightly increased its performance, but subsequent removal of any of the other individual systems resulted in performance degradation of the ensemble system. This suggests that almost all systems, even a low-performing system like Peregrine, contribute to the high performance of the combined annotation system.

What characteristics of the individual systems make our ensemble system perform well? Classifier diversity is generally considered a necessary condition for performance improvement of ensemble systems [29]. We have tried to achieve diversity by combining different types of classifiers. However, as mentioned above, it is difficult to quantify diversity, and the relationship between classifier diversity and performance of the combined system is not clear [35]. Moreover, the diversity measures proposed in the literature assume that a fixed set of samples is classified [29, 35], but this is not the case for our systems, which recognize varying amounts of concepts. In our approach, we used pair wise F-scores as a measure to quantify diversity between concept recognition systems. Apart from diversity, the accuracy of the classifiers should play a role: clearly, one would like classifiers to agree on a classification if it is correct.

Our results suggest that classifier accuracy correlates better with the performance of the ensemble system than classifier diversity. The two systems with the lowest performance, MetaMap and Peregrine, had the largest diversity with the other systems but hardly added to, or even deteriorated, the performance of the ensemble system. When varying the number of systems in the ensemble, the least performing systems gave the smallest contribution to the ensemble performance. Improvement may be achieved by the addition of other, better performing systems than MetaMap or Peregrine, but considering the flattening F-score curve with increasing number of systems in the ensemble, we suspect

such improvement not to be large.

Similarly to previous studies that used voting approaches in natural language processing tasks [30, 31, 33], we have used a simple voting scheme. Other, more advanced combination methods exist, e.g., weighted majority voting [41], Borda count [42], behavior knowledge space [43], decision templates [44], Bayesian approaches [45], and Dempster-Shafer rule [46], amongst other methods [47]. There is no consensus as to which of them performs best [29], although simple methods such as (weighted) majority approaches have shown consistent performance over a broad spectrum of applications [48, 49]. Other than for the simple voting approach that we used, these combination schemes require additional information, such as ranks or probabilities of the individual classifications, information which is not provided by the systems in our study. It might be possible to assign weights to the individual classifiers based on prior knowledge, and use this information in a weighted voting scheme, but whether this would result in better achievement of the ensemble system is left for future research.

A potential benefit of an ensemble-based system is the possibility to tune the operating characteristics of the system to a specific application. The system with the highest F-score is not under all circumstances the best. Some tasks may require high precision, even if this implies moderate recall and F-score, whereas other tasks require high recall. The individual systems show some variability in their performance figures (cf. Table 1), but none of their precision or recall values is really high. Our combination approach offers the possibility to vary the combined system across a large range of precision and recall values by varying the voting threshold (cf. Figure 1). Thus, the performance of a combined system can easily be tuned to best meet specific requirements.

Our error analysis indicated that a small part of the errors of the combined system can be attributed to a wrongly assigned concept type. Almost half of the remaining errors were due to a mismatch in one of the annotation boundaries. Many of these errors resulted from incorrect handling of coordination or punctuation, which are also common error types for the recognition of noun phrases in biomedical text [33, 50]. The impact of these errors on the performance of a whole information extraction pipeline is still an open question. For example, it may well be that erroneously splitting or joining annotations is less important in terms of information extraction performance than missing or inserting annotations. It would be interesting to learn how different annotation errors affect real clinical record processing tasks.

CONCLUSION

The combination of six existing systems for recognizing medical concepts in clinical records provides substantially better results than any of the individual systems. The ensemble-based approach is straightforward and allows the balancing of precision versus recall of the combined system.

REFERENCES

1. Meystre SM, Savova GK, Kipper-Schuler KC, Hurdle JF. **Extracting information from textual documents in the electronic health record: a review of recent research.** *Yearb Med Inform 2008: 128-44.*

2. Demner-Fushman D, Chapman WW, McDonald CJ. **What can natural language processing do for clinical decision support?** *J Biomed Inform 2009;42: 760-72.*

3. Uzuner O, South BR, Shen S, Duvall SL. **2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text.** *J Am Med Inform Assoc 2011;18: 552-6.*

4. Zeng QT, Goryachev S, Weiss S, Sordo M, Murphy SN, Lazarus R. **Extracting principal diagnosis, co-morbidity and smoking status for asthma research: evaluation of a natural language processing system.** *BMC Med Inform Decis Mak 2006;6: 30.*

5. Friedman C. **Towards a comprehensive medical language processing system: methods and issues.** *Proc AMIA Annu Fall Symp 1997: 595-9.*

6. Savova GK, Masanz JJ, Ogren PV, Zheng J, Sohn S, Kipper-Schuler KC, Chute CG. **Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications.** *J Am Med Inform Assoc 2010;17: 507-13.*

7. Christensen LM, Haug PJ, Fiszman M. **MPLUS: a probabilistic medical language understanding system.** *In: Proceedings of the ACL-02 workshop on Natural language processing in the biomedical domain. Philadelphia, USA; 2002. p. 29-36.*

8. Hahn U, Romacker M, Schulz S. **Creating knowledge repositories from biomedical reports: the MEDSYNDIKATE text mining system.** *Pac Symp Biocomput 2002: 338-49.*

9. Mack R, Mukherjea S, Soffer A, Uramoto N, Brown E, Coden A, Cooper J, Inokuchi A, Iyer B, Mass Y. **Text analytics for life science using the Unstructured Information Management Architecture.** *IBM Syst J 2004;43: 490-515.*

10. Melton GB, Hripcsak G. **Automated detection of adverse events using natural language processing of discharge summaries.** *J Am Med Inform Assoc 2005;12: 448-57.*

11. Goryachev S, Kim H, Zeng-Treitler Q. **Identification and extraction of family history information from clinical reports.** *AMIA Annu Symp Proc 2008: 247-51.*

12. Fiszman M, Chapman WW, Aronsky D, Evans RS, Haug PJ. **Automatic detection**

of acute bacterial pneumonia from chest X-ray reports. *J Am Med Inform Assoc 2000;7: 593-604.*

13. Aronson AR. **Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program.** *Proc AMIA Symp 2001: 17-21.*

14. Bodenreider O. **The Unified Medical Language System (UMLS): integrating biomedical terminology.** *Nucleic Acids Res 2004;32: D267-70.*

15. Carpenter B. **LingPipe for 99.99% recall of gene mentions.** *In: Proceedings of the Second BioCreative Challenge Evaluation Workshop. Valencia, Spain; 2007. p. 307-9.*

16. Buyko E, Wermter J, Poprat M, Hahn U. **Automatically adapting an NLP core engine to the biology domain.** *In: Proceedings of the Joint BioLINK and Bio-Ontologies Meeting. Fortaleza, Brasil; 2006. p. 65-8.*

17. de Bruijn B, Cherry C, Kiritchenko S, Martin J, Zhu X. **Machine-learned solutions for three stages of clinical information extraction: the state of the art at i2b2 2010.** *J Am Med Inform Assoc 2011;18: 557-62.*

18. Roberts K, Harabagiu SM. **A flexible framework for deriving assertions from electronic medical records.** *J Am Med Inform Assoc 2011;18: 568-73.*

19. Patrick JD, Nguyen DH, Wang Y, Li M. **A knowledge discovery and reuse pipeline for information extraction in clinical notes.** *J Am Med Inform Assoc 2011;18: 574-9.*

20. Torii M, Wagholikar K, Liu H. **Using machine learning for concept extraction on clinical documents from multiple data sources.** *J Am Med Inform Assoc 2011;18: 580-7.*

21. Minard AL, Ligozat AL, Ben Abacha A, Bernhard D, Cartoni B, Deléger L, Grau B, Rosset S, Zweigenbaum P, Grouin C. **Hybrid methods for improving information access in clinical documents: concept, assertion, and relation identification.** *J Am Med Inform Assoc 2011;18: 588-93.*

22. Jiang M, Chen Y, Liu M, Rosenbloom ST, Mani S, Denny JC, Xu H. **A study of machine-learning-based approaches to extract clinical entities and their assertions from discharge summaries.** *J Am Med Inform Assoc 2011;18: 601-6.*

23. D'Avolio LW, Nguyen TM, Goryachev S, Fiore LD. **Automated concept-level information extraction to reduce the need for custom software and rules development.** *J Am Med Inform Assoc 2011;18: 607-13.*

24. Settles B. **ABNER: an open source tool for automatically tagging genes, proteins and other entity names in text.** *Bioinformatics 2005;21: 3191-2.*

25. Buyko E, Wermter J, Poprat M, Hahn U. **Automatically adapting an NLP core engine to the biology domain.** *In: Proceedings of the Joint BioLINK and Bio-Ontologies Meeting. Fortaleza, Brasil; 2006. p. 65-8.*

26. Hahn U, Buyko E, Landefeld R, M ¨hlhausen M, Poprat M, Tomanek K, Wermter J. **An overview of JCoRe, the JULIE lab UIMA component repository.** *In: Proceedings of the LREC. Marrakech, Morocco; 2008. p. 1-7.*

27. Finkel JR, Grenager T, Manning C. **Incorporating non-local information into information extraction systems by gibbs sampling.** *In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. Ann Arbor, USA; 2005. p. 363-70.*

28. Schuemie MJ, Jelier R, Kors JA. **Peregrine: lightweight gene name normalization by dictionary lookup.** *In: Proceedings of the BioCreAtIvE II Workshop. Madrid, Spain; 2007. p. 131-3.*

29. Polikar R. **Ensemble based systems in decision making.** *IEEE Circuits Syst Mag 2006;6: 21-45.*

30. Smith L, Tanabe LK, Ando RJ, Kuo CJ, Chung IF, Hsu CN, Lin YS, Klinger R, etc. **Overview of BioCreative II gene mention recognition.** *Genome Biol 2008;9 Suppl 2: S2.*

31. Baumgartner WA, Lu Z, Johnson HL, Caporaso JG, Paquette J, Lindemann A, White EK, Medvedeva O, Cohen KB, Hunter L. **Concept recognition for extracting protein interaction relations from biomedical text.** *Genome Biol 2008;9 Suppl 2: S9.*

32. Kim J, Ohta T, Pyysalo S, Kano Y, Tsujii J. **Overview of BioNLP'09 shared task on event extraction.** *In: Proceedings of the Workshop on BioNLP: Shared Task. Boulder, USA; 2009. p. 1-9.*

33. Kang N, van Mulligen EM, Kors JA. **Comparing and combining chunkers of biomedical text.** *J Biomed Inform 2011;44: 354-60.*

34. Brown G, Wyatt J, Harris R, Yao X. **Diversity creation methods: a survey and categorisation.** *J Info Fusion 2005;6: 5-20.*

35. Kuncheva LI, Whitaker CJ. **Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy.** *Mach Learn 2003;51: 181-207.*

36. Lafferty J, McCallum A, Pereira F. **Conditional random fields: probabilistic models for segmenting and labeling sequence data.** *In: Proceeding of 18th International Conference on Machine Learning. San Francisco, USA; 2001. p. 282-9.*

37. Rabiner LR. **A tutorial on hidden Markov models and selected applications in speech recognition.** *Proc IEEE 1989;77: 257-86.*

38. Ferrucci D, Lally A. **UIMA: an architectural approach to unstructured information processing in the corporate research environment.** *Nat Lang Eng 2004;10: 327-48.*

39. Berger AL, Pietra VJD, Pietra SAD. **A maximum entropy approach to natural language processing.** *Comput Linguist 1996;22: 39-71.*

40. Krauthammer M, Nenadic G. **Term identification in the biomedical literature.** *J Biomed Inform 2004;37: 512-26.*

41. Littlestone N, Warmuth MK. **The weighted majority algorithm.** *Inform Comput 1994;108: 212-61.*

42. Borda JC. Mémoire sur les élections au scrutin. **Histoire de l'Académie Royale des Sciences**, *Paris 1781.*

43. Raudys Š, Roli F. **The behavior knowledge space fusion method: analysis of generalization error and strategies for performance improvement.** *Multi Class Syst 2003: 55-64.*

44. Kuncheva L, Bezdek JC, Duin RPW. **Decision template for multiple classifer fusion: an experimental comparison.** *Pattern Recogn Lett 2001;34: 228-37.*

45. Altinçay H. **On naive Bayesian fusion of dependent classifiers.** *Pattern Recogn Lett 2005;26: 2463-73.*

46. Ahmadzadeh MR, Petrou M. **Use of Dempster-Shafer theory to combine classifiers which use different class boundaries.** *Pattern Anal Appl 2003;6: 41-6.*

47. Wanas NM, Dara RA, Kamel MS. **Adaptive fusion and co-operative training for classifier ensembles.** *Pattern Recogn 2006;39: 1781-94.*

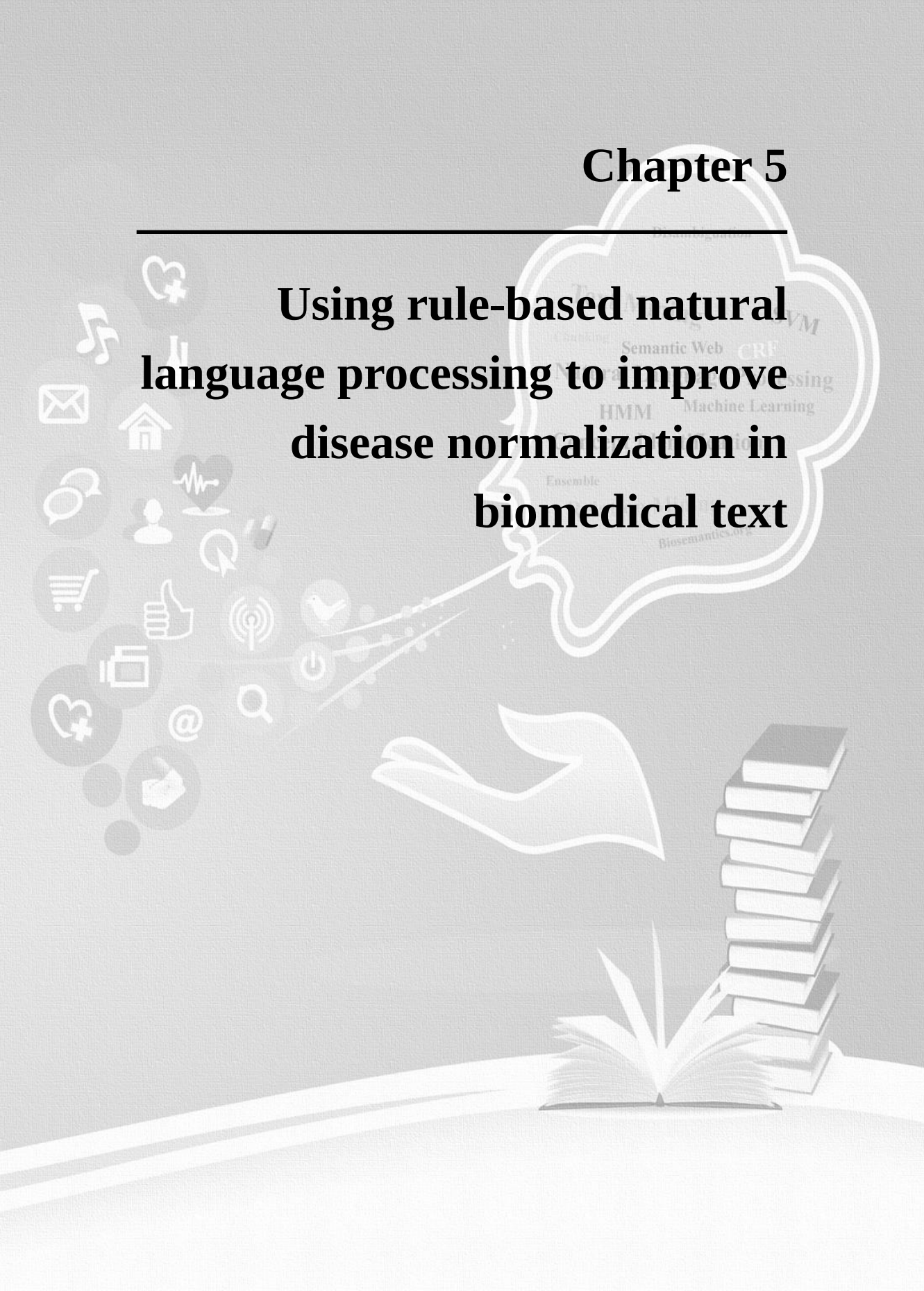
48. Alkoot FM, Kittler J. **Experimental evaluation of expert fusion strategies.** *Pattern Recogn Lett 1999;20: 1361-9.*

49. Kittler J, Hatef M, Duin RPW, Matas J. **On combining classifiers.** *IEEE PAMI 1998;20: 226-39.*

50. Wermter J, Fluck J, Stroetgen J, Geißler S, Hahn U. **Recognizing noun phrases in biomedical text: an evaluation of lab prototypes and commercial chunkers.** *In: Proceedings of the 1st International Symposium on Semantic Mining in Biomedicine. Hinxton, England; 2005. p. 25-33.*

Chapter 5

Using rule-based natural language processing to improve disease normalization in biomedical text



ABSTRACT

Objective

In order for computers to extract useful information from unstructured text, a concept normalization system is needed to link relevant concepts in a text to sources that contain further information about the concept. Popular concept normalization tools in the biomedical field are dictionary-based. In this study we investigate the usefulness of natural language processing (NLP) as an adjunct to dictionary-based concept normalization.

Methods

We compared the performance of two biomedical concept normalization systems, MetaMap and Peregrine, on the Arizona Disease Corpus (AZDC), with and without the use of a rule-based NLP module. Performance was assessed for exact and inexact boundary matching of the system annotations with those of the gold standard and for concept identifier matching.

Results

Without the NLP module, MetaMap and Peregrine attained F-scores of 61.0% and 63.9%, respectively, for exact boundary matching, and 55.1% and 56.9% for concept identifier matching. With the aid of the NLP module, the F-scores of MetaMap and Peregrine improved to 73.3% and 78.0% for boundary matching, and to 66.2% and 69.8% for concept identifier matching. For inexact boundary matching, performances further increased to 85.5% and 85.4%, and to 73.6% and 73.3% for concept identifier matching.

Conclusion

We have shown the added value of NLP for the recognition and normalization of diseases with MetaMap and Peregrine. The NLP module is general and can be applied in combination with any concept normalization system. Whether its use for concept types other than disease is equally advantageous remains to be investigated.

INTRODUCTION

Most biomedical knowledge comes only in unstructured form, such as in scientific articles and reports. The sheer volume of these textual sources requires computer processing to extract usable information. An important step in the information extraction task is the recognition and normalization of relevant concepts in a text [1]. Concept or named-entity recognition aims at finding text strings that refer to entities, and marking each entity with a semantic type, like "gene", "drug", or "disease". Concept normalization goes beyond entity recognition. It assigns a unique identifier to the recognized concept, which links it to a source that contains further information about the concept, such as its definition, its preferred name and synonyms, and its relationships with other concepts.

Much research has been done in concept recognition, but fewer studies addressed the more difficult task of concept normalization. Concept normalization systems are often dictionary-based, i.e., they try to find concept occurrences in a text by matching text strings with concept names and their corresponding identifiers in a dictionary. The dictionary is composed of entries from one or more knowledge sources, such as Gene Ontology [2], Entrez Gene [3], or the Unified Medical Language System (UMLS) [4]. Typically, dictionary-based systems use little or no linguistic information to find concepts, and the potential added value of such information is largely unknown.

In this study, we investigate the usefulness of natural language processing (NLP) techniques to improve biomedical concept normalization. We present a set of rules that utilize NLP information, and show that these rules substantially improve the performance of two concept normalization systems, MetaMap [5] and Peregrine [6], in recognizing and normalizing diseases in biomedical text.

BACKGROUND

Compared to the number of named-entity recognition systems, the number of concept normalization systems in the biomedical field is small. Reported systems include MetaMap [5], Mgrep [7], Negfinder [8], Peregrine [6], and Whatizit [9]. All of these systems use a dictionary to find concepts in text and map them to concept identifiers. Several systems, such as MetaMap, perform some lexical analysis in the normalization process, but part-of-speech (POS) and chunking information are mostly not considered.

Concept normalization is generally considered a more difficult task than concept recognition. This is reflected in the variety of named-entity recognition challenges in the biomedical domain, e.g., BioCreative [10] and BioNLP [11] (recognition of proteins and

genes in scientific literature), and TREC [12] and i2b2 [13] (drugs, diseases, and treatments in electronic patient records), whereas normalization challenges have been few. Substantial work on gene normalization has been done in a series of gene normalization tasks that were part of the BioCreative competitions [14–16]. In BioCreative I and II, the gene normalization task consisted of finding the identifiers of genes and gene products mentioned in sets of abstracts from four model organisms: yeast, fly, mouse (BioCreative I), and human (BioCreative II). In the gene normalization task in BioCreative III, systems had to assign identifiers to all named genes in full-text articles without being limited to a particular organism. For all tasks, unique gene identifiers had to be provided at the document level, rather than for individual gene mentions. The different systems participating in these challenges used a wide variety of methods, including pattern matching, machine learning, and lexical resources lookup [14–16]. Heuristic rules were mostly developed and implemented in an ad-hoc and custom manner. Hybrid use of rule-based and machine learning methods was observed in system descriptions [15,16]. For example, GNAT [17], the top-performing gene normalization system in BioCreative II, combines dictionary matching and machine learning to recognize gene mentions; the machine-learning component of such programs requires a subsequent step to match predicted mentions to identifiers. In the next steps, recognized gene names are validated by means of several dedicated filters to remove false positives, while ambiguous mentions are disambiguated by comparing the current text with several sources of background information existing for each candidate gene. Another example of a high-performance gene normalization system is GeNo [18]. This system also combines dictionary-based and machine-learning based gene name detection, using approximate string matching to link gene mentions with dictionary identifiers. It employs automatic term variant generation and false positive filtering. The system fully relies on publicly available data and resources. GeNo did not participate in BioCreative, but was shown to have a performance on par with GNAT. Another top-ranking system in BioCreative I and II was the proprietary ProMiner gene normalization system [19]. ProMiner employs a strictly dictionary-based approach, relying on well-curated dictionaries and approximate string matching.

Species name recognition is an important subtask in many systems participating in the BioCreative III challenge. Some of the systems employed LINNAEUS, an open source species normalization system [20]. LINNAEUS follows a dictionary-based approach, using a time-efficient implementation of regular expressions for document tagging. Post-processing includes acronym detection, filtering of common words, and disambiguation.

Many algorithms and methods have been proposed to solve common problems encountered in concept recognition and concept normalization tasks [21]. For instance,

to solve the problem of gene mention coordination, multiple conditional random fields (CRFs) and n-gram language models were used [22]. CRFs were also used in another study to decompose complex coordinated entity expressions into constituent conjuncts, to determine the missing elements, and thus to reconstruct explicitly all the single named entity mentions [23]. Rule-based procedures for resolving simple conjunctions of gene mentions have also been used [17,24,25]. Many papers addressed the problem of abbreviation detection and expansion [26–29]. Proposed approaches range from simple rule-based algorithms to sophisticated machine-learning methods. Term variation is another common problem that concept normalization systems have to deal with. Methods that have been proposed include approximate string matching, heuristic pattern matching rules, enhanced dictionaries, etc [30–32]. Finally, a common problem addressed in many systems is the removal of false-positive mentions that result from the recognition stage. One often used approach is to filter out terms that have an ambiguous meaning in common English [33]. but more sophisticated methods (e.g., scoring the similarity between the semantic profile of a concept and the document in which it occurs [18]) have also been proposed.

There are only a few corpora in the biomedical domain that incorporate concept annotations, notably the Arizona Disease Corpus (AZDC) [34], the BioCreative gene normalization corpora [14–16], the Colorado Richly Annotated Full-Text (CRAFT) corpus [35], and the Gene Regulation Event Corpus (GREC) [36]. Among these, AZDC is the only one that includes information about concept boundaries and UMLS concept identifiers, and that is publicly available. Based on the AZDC corpus, very recently the larger NCBI corpus was developed [37], but this corpus only contains annotations of disease mentions, not concept identifiers. Therefore, we used AZDC as the gold standard corpus (GSC) for our experiments.

The AZDC was used before by Leaman et al. [34] to test the performance of one dictionary-based system and two statistical systems (BANNER [34] and JNET [38]). The dictionary-based system yielded an F-score of 62.2%, while BANNER and JNET achieved F-scores of 77.9% and 77.2%, respectively. Chowdhury and Faisal [39] developed another machine-learning based system, BNER, and tested it on the same corpus, achieving an F-score of 81.1%. Both these studies were targeted at concept recognition, not at concept normalization.

METHODS

Corpus

The AZDC has been developed at the Arizona State University. It was released in 2009 [34]. The corpus has been annotated with disease concepts, including UMLS codes, preferred concept names, and start and end points of disease mentions inside the sentences. The whole corpus consists of 2784 sentences, taken from 793 Medline abstracts, and 3455 disease annotations. Annotations have been mapped to a concept unique identifier (CUI) in the UMLS Metathesaurus. Each annotation belongs to one of the following semantic types defined in the UMLS: disease or syndrome, neoplastic process, congenital abnormality, acquired abnormality, experimental model of disease, injury or poisoning, mental or behavioral dysfunction, pathological function, sign or symptom.

We divided the corpus into two parts: one-third of the sentences was used for developing the NLP module, the other two-thirds for testing.

Concept normalization systems

We evaluated two concept normalization systems, MetaMap and Peregrine. Both systems were downloaded from their official websites with default configurations and parameters, and no attempt was made to optimize their performance.

MetaMap (http://metamap.nlm.nih.gov/) is a dictionary-based system for normalizing concepts from the UMLS Metathesaurus in biomedical texts [5]. It makes use of a minimal-commitment parser, which splits texts into chunks in which concepts are identified. MetaMap also performs word-sense disambiguation (WSD). MetaMap is dictionary-based and cannot be trained. MetaMap Transfer (MMTx) is a distributable version of MetaMap written in Java. We used the 2011 version, which includes version 2011AA of the UMLS Metathesaurus.

Peregrine (https://trac.nbic.nl/data-mining/) is a dictionary-based concept recognition and normalization tool, developed at the Erasmus University Medical Center (http://www.biosemantics.org). Peregrine finds concepts by dictionary look-up, and performs WSD [6]. Rewrite and suppression rules are applied to the terms in the dictionary to enhance precision and recall [40]. In our experiments, we used Peregrine with version 2011AB of the UMLS Metathesaurus.

NLP module

The NLP module that we have developed consists of a number of rules that combine the annotations of a concept normalization system with POS and chunking information. We

used the OpenNLP tool suit (http://opennlp.apache.org/) to obtain the necessary POS and chunking information. The OpenNLP tool suit is based on a maximum entropy model. An OpenNLP UIMA wrapper has been developed by JULIE Lab (http://www.julielab.de). The wrapper divides the OpenNLP package into small modules that perform sentence detection, tokenization, POS tagging, and chunking, which makes it easy to configure the pipeline for different purposes [41]. The rules in the NLP module are divided into five submodules, which address specific tasks and are described in the following. A detailed description of the rules is available as Supplementary Material.

1. Coordination. This submodule performs coordination resolution. The approach is straightforward and extends the one described by Baumgartner et al [24]. For instance, in the following sentence from the AZDC: "We calculated age related risks of all, colorectal, endometrial, and ovarian cancers in nt943+3 A--T MSH2 mutation carriers […]", MetaMap and Peregrine both recognize "ovarian cancers" as a concept, but miss "colorectal cancers" and "endometrial cancers". Using POS- and chunking information, this module reformats the coordination phrase and feeds the reformatted text into the concept normalization systems for proper annotation of the concepts.

2. Abbreviation. This submodule combines the abbreviation expansion algorithm of Schwartz and Hearst [26], with POS and chunking information to improve the recognition of abbreviations. We chose the Schwartz and Hearst algorithm because it is very easy to implement and to combine with other rules, and has shown consistently good performance in different studies [24,25]. For an instance of abbreviation errors, in the sentence "Deficiency of aspartylglucosaminidase AGA causes a lysosomal storage disorder Aspartylglucosaminuria AGU", the concept normalization systems annotated "Deficiency of aspartylglucosaminidase" and "Aspartylglucosaminuria" as disease concepts, in agreement with the gold standard annotations, but they did not recognize the abbreviations. Since "AGA" is used as the abbreviation of "aspartylglucosaminidase", an enzyme, it should not be annotated as a disease concept, but "AGU" should be identified as such. This was accomplished by means of a rule that checks whether the last noun in a noun phrase is an abbreviation of all preceding tokens in the noun phrase.

3. Term variation. Dictionary-based systems can only find concepts if the terms by which these concepts are denoted in text are part of the dictionary. Although UMLS covers some term variation, many variations are missing. The submodule in question uses a shallow parsing based approach, similar to Ferrucci et al [42]. It contains a number of rules that adjust noun phrases and feed the adjusted phrase into the concept normalization system again, to check whether it refers to a concept. For instance, if a noun phrase includes a preposition, such as "deficiency of hex A", which is not part of the UMLS, the word order is changed into "hex A deficiency", which is contained in the UMLS.

4. Boundary correction. This submodule contains several rules that correct the start- and end positions of concepts identified by the systems, based on POS- and chunking information. For instance, if the POS of the start- or end token from a concept annotation is a verb, preposition, conjunction, or interjection, it then uses POS information to adjust the concept start- or end position. By applying these rules, an erroneous annotation such as "phenylketonuria Is", contrived from "classical phenylketonuria is an autosomal recessive disease", could be corrected to "phenylketonuria".

5. Filtering. This submodule has two rules that suppress concepts although they had been identified by the system. The first rule removes a concept if the concept annotation in the text has no overlap with a noun phrase because in our experience, most UMLS concepts in biomedical abstracts belong to a noun phrase, or at least overlap with it. The second rule removes a concept if it is part of a concept filter list, a common approach to increase precision as used by many systems in the BioCreative competitions [14–16]. Our list contains 23 generic concepts (e.g., "disease", "abnormality") that were wrongly annotated by Peregrine in the training set.

The rules in the NLP module were developed on the basis of an error analysis of the Peregrine annotations of the training set. The annotations of MetaMap were not used for this development.

Performance evaluation

The annotations of the concept normalization systems were compared with the gold standard annotations by exact and inexact matching, both of the concept boundaries (following the same procedure as in [34,39]) and of the concept identifiers. For exact boundary matching, an annotation was counted as true positive if it was identical to the gold standard annotation, i.e., if both annotations had the same start and end location in the corpus. If a gold-standard annotation was not given, or not rendered exactly by the system, it was counted as false negative; if an annotation found by the system did not exactly match the gold standard, it was counted as false positive. For concept identifier matching, the same rules applied as for exact boundary matching with the additional requirement that for a true positive outcome the concept identifiers had to match; if not, the annotations were counted as false positive as well as false negative. Performance was evaluated in terms of precision, recall, and F-score.

The performance of the systems was also tested by using two methods of inexact boundary matching: one-side boundary matching (i.e., at least one boundary of the system annotation had to match the gold standard annotation) and overlap matching (i.e.,

at least one word of the system annotation had to overlap with the corpus annotation). Inexact concept identifier matching followed the same rules as inexact boundary matching but in addition required the concept identifiers to match.

An error analysis was carried out on a sample of 100 randomly selected errors that were made by each concept normalization system after applying the NLP module. Errors were grouped into five categories, following the task categorization of the five NLP submodules.

Processing pipeline

All systems and the NLP module were integrated in the Unstructured Information Management Architecture (UIMA) framework [43], which was easily accomplished since they either were available as a UIMA component [41] or had a web service interface. A UIMA processing pipeline was implemented, which first read the AZDC test set by the UIMA Collection Reader. Then the test set was annotated by MetaMap and Peregrine. Because the AZDC includes nine UMLS semantic types, only the annotated concepts belonging to these types were considered for evaluation. Subsequently, the NLP submodules post-processed the annotation results. Some rules, such as the coordination rules, not only processed the annotations but also modified the original input sentence. The modified text was then fed into the concept normalization systems for re-annotation. Finally, the system annotations before and after post-processing by the NLP submodules were evaluated separately against the gold standard annotations.

RESULTS

Performance of the concept normalization systems without and with NLP

Table 1 shows the performance of the two concept normalization systems on the AZDC test set. Without the NLP module, MetaMap achieved an F-score of 61.0% for exact boundary matching, and 55.1% for concept identifier matching. The F-scores of Peregrine were 63.9% for exact boundary matching and 56.9% for concept identifier matching. With the aid of the NLP module, the F-scores of MetaMap and Peregrine increased by 12.3 and 14.1 percentage points, respectively, for exact boundary matching, and by 11.1 and 12.9 percentage points for concept identifier matching. Both for MetaMap and Peregrine, there is a larger increase in precision than in recall (table 1).

For inexact, one-side boundary matching, MetaMap and Peregrine, without the NLP module, reached F-scores of 79.5% and 77.7%, respectively (table 1). With the NLP

module, these F-scores increased to 85.5% and 85.4%. The performances for concept identifier matching increased from 65.3% to 73.6% and from 64.9% to 73.3%, respectively. The F-scores for overlap matching showed only minimal improvement (<1 percentage point, data not shown).

	Exact matching						Inexact matching					
	Boundaries			Identifiers			Boundaries			Identifiers		
System	Precision	Recall	F-score	Precision	Recall	F-score	Precison	Recall	F-score	Precison	Recall	F-score
MetaMap	60.9	61.1	61.0	55.0	55.2	55.1	79.3	79.7	79.5	65.1	65.5	65.3
MetaMap + NLP	76.1	70.7	73.3	68.7	63.9	66.2	89.1	82.2	85.5	76.1	71.3	73.6
Peregrine	63.5	64.3	63.9	56.6	57.3	56.9	77.1	78.4	77.7	64.6	65.3	64.9
Peregrine + NLP	82.2	74.2	78.0	73.5	66.4	69.8	89.6	81.6	85.4	76.7	70.2	73.3

Table 1. Performance (in %) of MetaMap and Peregrine, with and without the NLP module, on the AZDC test set for exact and inexact matching of concept boundaries and identifiers.

Performance of NLP submodules

Table 2 shows the incremental performance improvement for the various NLP submodules, based on exact boundary matching. The baseline was the performance of MetaMap and Peregrine without any submodules. The coordination module, abbreviation module, and boundary correction module each contributed between 3.0 to 3.5 percentage points to the betterment of performance. The smallest contribution to raising the F-score was by the filtering module, with the two rules in this module equally contributing to the performance improvement.

NLP submodules	MetaMap			Peregrine		
	Precision	Recall	F-score	Precision	Recall	F-score
Baseline	60.9	61.1	61.0	63.5	64.3	63.9
+Coordination	63.4	64.4	64.0	67.5	67.0	67.2
+Abbreviation	66.7	67.7	67.2	70.7	70.8	70.7
+Term variation	68.9	69.3	69.1	74.3	71.8	73.0
+Boundary correction	73.7	70.7	72.2	78.8	74.2	76.4
+Filtering	76.1	70.7	73.3	82.2	74.2	78.0

Table 2. Performance (in %) of MetaMap and Peregrine with incremental contributions of the NLP submodules on the AZDC test set for exact boundary matching.

Error analysis

We randomly selected 100 errors that MetaMap and Peregrine, together with the NLP module, each made on the test set, and manually classified them into different error types (Table 3). The error profiles of MetaMap and Peregrine were very similar. The majority of errors were due to term variation and boundary errors. For instance, the term "alpha-Gal A deficiency", referring to the concept "alpha-galactosidase deficiency", was not found as the term does not occur in UMLS. Boundary errors mainly occurred because of nested annotations in the gold standard. For example, in "Therefore, we screened eight familial gastric cancer kindreds of British and Irish origin [...]", both "familial gastric cancer" and "gastric cancer" were annotated in the gold standard, whereas the systems only annotated "gastric cancer". Filtering errors were mainly due to concepts that were inconsistently annotated by the gold standard. For example, in "Because of the variable expression of nm23-H1 in different tumors [...]", "tumors" was annotated by the gold standard, but it was not in "In neuroblastoma, higher levels of p19/nm23 [...] were observed in advanced stage tumors compared with limited stage disease". Coordination and abbreviation errors were relatively few. For example, in the sentence "the majority of familial breast/ovarian cancer", the gold standard annotated "familial breast cancer" and "familial ovarian cancer", whereas the coordination module recognized "breast cancer" and "ovarian cancer" but failed to include "familial". An abbreviation error occurred in the sentence "One of five PWS/AS patients analyzed to date has an identifiable, rearranged HERC2 transcript derived from the deletion event." The systems did not annotate "PWS" (Prader-Willi syndrome) and "AS" (Angelman syndrome), which had been defined (and correctly annotated) in a preceding sentence.

System	Coordination	Abbreviation	Term variation	Boundary	Filtering
MetaMap + NLP	12	13	28	24	23
Peregrine + NLP	11	14	28	26	21

Table 3. Distribution across five error types of 100 randomly selected errors of each system on the AZDC test set.

DISCUSSION

We have investigated the use of NLP to improve the performance of two concept normalization systems. By applying a set of post-processing rules that utilize POS and chunking information, the F-scores of MetaMap and Peregrine on AZDC improved by 12.3 and 14.1 percentage points, respectively, for exact boundary matching, and by 11.1 and 12.9 percentage points, respectively, for concept identifier matching. For inexact matching, the improvement was smaller but still in the order of 6-8 percentage points. To our knowledge, this is the first study that assesses the performance of systems in normalizing disease concepts.

Concept recognition performed substantially better than concept normalization, even if the boundaries matched exactly. This may partially be explained by the fact that the systems assigned the wrong CUI to ambiguous terms. However, on closer inspection it turned out that the gold-standard annotators often took into account the context in which a term was used and assigned a more specific CUI than the systems. For instance, in the sentence “A DNA-based test for the HFE gene is commercially available, but its place in the diagnosis of hemochromatosis is still being evaluated.”, the systems assigned the concept “hemochromatosis” (C0018995), whereas the GSC annotated the concept “hereditary hemochromatosis” (C0392514). It should be noted that “hemochromatosis” is not part of the list of terms in the UMLS corresponding with the concept C0392514. Thus, this concept is not even considered by the disambiguation algorithms of the systems. Knowledge-based disambiguation approaches that can take into account the concept relationships defined in the UMLS may be able to solve these disambiguation problems.

Usage of the NLP module gives a larger increase in precision than in recall (cf. Table 1), even though most rules are aimed at finding missed concepts. This can partly be

explained by the filtering submodule, which by its nature can only improve precision, but also some of the other submodules improve precision more than recall. The reason is that if a rule finds a missed concept it often suppresses one or more erroneous concepts that were initially found by the system, thus improving precision.

Peregrine gave a slightly better performance than MetaMap when exact matching was used for evaluation, but for inexact matching the performances were similar, both for concept recognition and for concept identification. The two systems used different UMLS versions (2011AA and 2011AB), but the differences between these versions are very small and unlikely to be the cause of performance differences. Since the NLP module was developed on the basis of the errors made by Peregrine, one might suspect a performance bias in favor of this system in combination with the NLP module. However, when we determined the performance on the training set, the F-score turned out to be only 1.9 percentage point higher than on the test set, indicating hardly any overtraining. For MetaMap, this difference was 1.7 percentage point. With the use of the NLP module, Peregrine and MetaMap showed a comparable gain in performance.

Many of the rules in our NLP module have not been used before in their specific form, but similar such rules have previously been proposed in many different studies. The combination of different types of rules in one system, showing the contribution of each submodule to total performance, and their application to disease normalization, a task which has not been addressed before, is novel in our study. The submodules are general and may be combined, as a whole or separately, with other concept normalization systems.

In developing the NLP module, we manually constructed rules and did not use machine-learning techniques, as did previous studies that used the AZDC for system development and evaluation [34,39]. These machine-learning based systems achieved comparable or slightly better performance for concept recognition as our rule-based systems, but did not address the normalization task. Moreover, we believe our approach offers several benefits. Firstly, machine-learning based systems are often not transparent, whereas the man-made rules are comprehensible. This is likely to ease error detection and correction, incremental rule improvement, and adaptation to other domains. Also, the rules combine input from heterogeneous systems in a very flexible way, which may be more difficult to achieve by machine learning methods that have a fixed knowledge representation model. Finally, machine learning methods require sufficiently large GSCs for training. Although this requirement apparently was met in the case of AZDC, this may not be true for other application areas. We also need a GSC for developing our rules, but the size can be relatively small because human experts also bring in background knowledge that can compensate for scarce data. Finally, although we did not put it to the test, it is conceivable

that machine-learning based concept recognition may still benefit from our NLP module because it may capture patterns not well handled by machine learning. Whether this would also translate into better concept normalization would also depend on the normalization step that needs to follow machine-learning based concept recognition.

The error analysis indicated that about half of the errors that remain after applying the NLP module can be denoted as term variation and filtering errors. While further improvement of the submodules dealing with these errors may be possible, it is more likely that improved dictionaries and disambiguation methods will help to reduce these types of errors. In this respect, further work on the generation of term variants would be useful. We also noticed that shallow parsing sometimes provides insufficient information to resolve errors in complex sentences. Such information may possibly derive from deep parsing, but exploring the usefulness of these techniques for our purposes will be left to future research.

We have shown the added value of the NLP module for the recognition and normalization of diseases with MetaMap and Peregrine. The module is general and can be applied in combination with any concept normalization system. Whether its use for the normalization of other concept types, such as genes or drugs, is equally advantageous still remains to be investigated.

REFERENCES

1 Krauthammer M, Nenadic G. **Term identification in the biomedical literature.** *J Biomed Inform* 2004;**37**:512–26.

2 Harris MA, Clark J, Ireland A, *et al.* **The Gene Ontology (GO) database and informatics resource.** *Nucleic Acids Res* 2004;**32**:258–61.

3 Maglott D, Ostell J, Pruitt KD, *et al.* **Entrez Gene: gene-centered information at NCBI.** *Nucleic Acids Res* 2007;**35**:26–31.

4 Bodenreider O. **The Unified Medical Language System (UMLS): integrating biomedical terminology.** *Nucleic Acids Res* 2004;**32**:267–70.

5 Aronson AR. **Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program.** In: *Proceedings of the AMIA Symposium*. Philadelphia, PA: 2001. 17–21.

6 Schuemie MJ, Jelier R, Kors JA. **Peregrine: lightweight gene name normalization by dictionary lookup.** In: *Proceedings of the BioCreAtIvE II Workshop*. Madrid, Spain: 2007. 131–3.

7 Shah NH, Bhatia N, Jonquet C, *et al.* **Comparison of concept recognizers for building the Open Biomedical Annotator.** *BMC Bioinform* 2009;**10**:S14.

8 Mutalik PG, Deshpande A, Nadkarni PM. **Use of general-purpose negation detection to augment concept indexing of medical documents.** *J Am Med Inform Assoc* 2001;**8**:598–609.

9 Rebholz-Schuhmann D, Arregui M, Gaudan S, *et al.* **Text processing through Web services: calling Whatizit.** *Bioinformatics* 2008;**24**:296–8.

10 Hirschman L, Yeh A, Blaschke C, *et al.* **Overview of BioCreAtIvE: critical assessment of information extraction for biology.** *BMC Bioinform* 2005;**6**:S1.

11 Kim J-D, Ohta T, Pyysalo S, *et al.* **Overview of BioNLP'09 shared task on event extraction.** In: *Proceedings of the Workshop on BioNLP Shared Task*. Boulder, USA: 2009. 1–9.

12 Voorhees EM, Tong RM. **Overview of the TREC 2011 medical records track.** In: *Proceedings of the twentieth Text REtrieval Conference (TREC)*. Gaithersburg, USA: 2011.

13 Uzuner O, South BR, Shen S, *et al.* **2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text.** *J Am Med Inform Assoc* 2011;**18**:552–6.

14 Hirschman L, Colosimo M, Morgan A, *et al.* **Overview of BioCreAtIvE task 1B: normalized gene lists.** *BMC Bioinform* 2005;**6 Suppl 1**:S11.

15 Morgan AA, Lu Z, Wang X, *et al.* **Overview of BioCreative II gene normalization.** *Genome Biol* 2008;**9 Suppl 2**:S3.

16 Lu Z, Kao HY, Wei CH, *et al.* **The gene normalization task in BioCreative III.** *BMC Bioinform* 2011;**12**:S2.

17 Hakenberg J, Plake C, Leaman R, *et al.* **Inter-species normalization of gene mentions with GNAT.** *Bioinformatics* 2008;**24**:126–32.

18 Wermter J, Tomanek K, Hahn U. **High-performance gene name normalization with GeNo.** *Bioinformatics* 2009;**25**:815–21.

19 Hanisch D, Fundel K, Mevissen H-T, *et al.* **ProMiner: rule-based protein and gene entity recognition.** *BMC Bioinform* 2005;**6**:S14.

20 Gerner M, Nenadic G, Bergman CMC-P. **LINNAEUS: a species name identification system for biomedical literature.** *BMC Bioinform* 2010;**11**:85.

21 Dai HJ, Chang YC, Tzong-Han Tsai R, *et al.* **New challenges for biological text-mining in the next decade.** *J Comput Sci Tech* 2010;**25**:169–79.

22 Struble CA, Povinelli RJ, Johnson MT, *et al.* **Combined conditional random fields and n-gram language models for gene mention recognition.** In: *Proceedings of the Second BioCreative Challenge Evaluation Workshop*. Madrid, Spain: 2007. 81–3.

23 Buyko E, Tomanek K, Hahn U. **Resolution of coordination ellipses in biological named entities using conditional random fields.** In: *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics (PACLING)*. Melbourne, Australia: 2007. 163–71.

24 Baumgartner WA, Lu Z, Johnson HL, *et al.* **Concept recognition for extracting protein interaction relations from biomedical text.** *Genome Biol* 2008;**9**:S9.

25 Jimeno-Yepes A, Berlanga-Llavori R, Rebholz-Schuhmann D. **Ontology refinement for improved information retrieval.** *Inform Process Manag* 2010;**46**:426–35.

26 Schwartz Hearst, M. A. AS. **A simple algorithm for identifying abbreviation definitions in biomedical text.** In: *Proceedings of the 8th Pacific Symposium on Biocomputing*. Hawaii, USA: 2003. 451–62.

27 Gaudan S, Kirsch H, Rebholz-Schuhmann D. **Resolving abbreviations to their**

senses in Medline. *Bioinformatics* 2005;**21**:3658–64.

28 Okazaki N, Ananiadou S, Tsujii J. **Building a high-quality sense inventory for improved abbreviation disambiguation.** *Bioinformatics* 2010;**26**:1246–53.

29 Atzeni P, Polticelli F, Toti D. **An automatic identification and resolution system for protein-related abbreviations in scientific papers.** In: *Proceedings of the 9th European conference on Evolutionary computation, machine learning and data mining in bioinformatics*. Torino, Italy: 2011. 27–9.

30 Schuemie MJ, Mons B, Weeber M, *et al.* **Evaluation of techniques for increasing recall in a dictionary approach to gene and protein name identification.** *J Biomed Inform* 2007;**40**:316–24.

31 Tsuruoka Y, McNaught J, Ananiadou S. **Normalizing biomedical terms by minimizing ambiguity and variability.** *BMC Bioinform* 2008;**9**:S2.

32 Ratkovic Z, Golik W, Warnier P. **Event extraction of bacteria biotopes: a knowledge-intensive NLP-based approach**. *BMC Bioinform* 2012;**13**:S8.

33 Wang X, Matthews M. **Distinguishing the species of biomedical named entities for term identification.** *BMC Bioinform* 2008;**9**:S6.

34 Leaman R, Miller C, Gonzalez G. **Enabling recognition of diseases in biomedical text with machine learning: corpus and benchmark.** In: *Proceedings of the 3rd International Symposium on Languages in Biology and Medicine (LBM)*. Jeju Island, South Korea: 2009. 82–9.

35 Bada M, Hunter LE, Eckert M, *et al.* **An overview of the CRAFT concept annotation guidelines.** In: *Proceedings of the Fourth Linguistic Annotation Workshop*. Uppsala, Sweden: 2010. 207–11.

36 Thompson P, Iqbal SA, McNaught J, *et al.* **Construction of an annotated corpus to support biomedical information extraction.** *BMC Bioinform* 2009;**10**:349.

37 Doğan RI, Lu Z. **An improved corpus of disease mentions in PubMed citations.** In: *Proceedings of the 2010 Workshop on Biomedical Natural Language Processing (BioNLP)*. Montreal, Canada: 2012.

38 Hahn U, Buyko E, Landefeld R, *et al.* **An overview of JCoRe, the JULIE lab UIMA component repository.** In: *Proceedings of the Language Resources and Evaluation Conference (LREC)*. Marrakech, Morocco: 2008. 1–7.

39 Chowdhury M, Faisal M. **Disease mention recognition with specific features.** In: *Proceedings of the 2010 Workshop on Biomedical Natural Language Processing*

(BioNLP). Uppsala, Sweden: 2010. 83–90.

40 Hettne KM, Van Mulligen EM, Schuemie MJ, *et al.* **Rewriting and suppressing UMLS terms for improved biomedical term identification.** *J Biomed Semantics* 2010;**1**:1–5.

41 Buyko E, Wermter J, Poprat M, *et al.* **Automatically Adapting an NLP Core Engine to the Biology Domain.** In: *Proceedings of the Joint BioLINKBio-Ontologies Meeting*. 2006. 2–5.

42 Vilares J, Alonso MA, Vilares M. **Extraction of complex index terms in non-English IR: A shallow parsing based approach.** *Inform Process Manag* 2008;**44**:1517–37.

43 Ferrucci D, Lally A. **UIMA: an architectural approach to unstructured information processing in the corporate research environment.** *Nat Lang Eng* 2004;**10**:327–48.

Chapter 6

Knowledge-based extraction of adverse drug events from biomedical text

ABSTRACT

Background

Many biomedical relation extraction systems are machine-learning based and have to be trained on large annotated corpora that are expensive and cumbersome to construct. We developed a knowledge-based relation extraction system that requires minimal training data, and applied the system for the extraction of adverse drug events from biomedical text. The system consists of a concept recognition module that identifies drugs and adverse effects in sentences, and a knowledge-base module that establishes whether a relation exists between the recognized concepts. The knowledge base was filled with information from the Unified Medical Language System. The performance of the system was evaluated on the ADE corpus, consisting of 1644 abstracts with manually annotated adverse drug events. Fifty abstracts were used for training, the remaining abstracts were used for testing.

Results

The knowledge-based system obtained an F-score of 50.5%, which was 34.4 percentage points better than the co-occurrence baseline. Increasing the training set to 400 abstracts improved the F-score to 54.3%. When the system was compared with a machine-learning system, jSRE, on a subset of the sentences in the ADE corpus, our knowledge-based system achieved an F-score of 88.8% which is 7 percentage points better than jSRE trained on 50 abstracts, and still 2 percentage points better than jSRE trained on 90% of the corpus.

Conclusion

A knowledge-based approach can be successfully used to extract adverse drug events from biomedical text without need for a large training set. Whether use of a knowledge base is equally advantageous for other biomedical relation-extraction tasks remains to be investigated.

BACKGROUND

Vast amounts of biomedical information are only offered in unstructured form through scientific publications. It is impossible for researchers or curators of biomedical databases to keep pace with all information in the growing number of papers that are being published.[1, 2] Text-mining systems hold promise for facilitating the time-consuming and expensive manual information extraction process,[3] or for automatically engendering new hypotheses and fresh insights.[4, 5]

In recent years, many systems have been developed for the automatic extraction of biomedical events from text, such as protein-protein interactions and gene-disease relations.[2, 6] Relatively few studies addressed the extraction of drug-related adverse effects, information which is relevant in drug research and development, healthcare, and pharmacovigilance.[7] The reason that this subject has been studied less frequently may in part be explained by the scarcity of large annotated training corpora. Admittedly cumbersome and expensive to construct, these data sets are nonetheless essential to train the machine-learning based classifiers of most current event extraction systems. Relation extraction systems typically perform two tasks: first, they try to recognize the entities of interest, next they determine whether there are relations between the recognized entities. In many previous studies, system performance evaluation was often limited to the second, relation extraction task, and did not consider the performance of the entity recognition task.

In this study, we describe the use of a knowledge base to extract drug-adverse effect relations from biomedical abstracts. The main advantage of our system is that it needs very little training data as compared to machine-learning approaches. Also, we evaluate the performance of the whole relation extraction pipeline, including the entity recognition part.

Related work

To extract biomedical relations from unstructured text a number of approaches have been explored, of which we mention simple co-occurrence, rule-based, and machine-learning based techniques.

The simplest approach is based on the co-occurrence of entities of interest. It assumes that if two entities are mentioned together in the same sentence or abstract, they are probably related. Typically, this approach achieves high recall, but low precision.[8] Since co-occurrence approaches are straightforward and do not involve linguistic analysis, their performance is often taken as a baseline to gauge other methods.[9, 10]

Rule-based techniques are also a popular method for relation extraction. The rules are defined manually using features from the context in which the relations of interest occur. Such features may be prefixes and suffixes of words, part-of-speech (POS) tags, chunking information, etc.[11–13] However, the large amount of name variations and ambiguous terms in the text may cause an accumulation of rules.[5] This approach can increase precision, but often at the cost of significantly lower recall.[14]

Machine-learning approaches automatically build classifiers for relation extraction, using contextual features derived from natural language processing techniques such as shallow parsing, which divides the sentence into chunks,[15, 16] or full dependency parsing, which provides a complete syntactic analysis of sentence structures.[17] The performance of these methods is usually good,[18–20] but they require annotated training sets of sufficient size. Also, processing time may be high.[3]

Hybrid approaches that combine manual and automatic approaches have also become more popular in recent years.[21, 22]

An example of a relation extraction system is JReX, developed by the JULIE lab.[23] JReX uses a support vector machine (SVM) algorithm as its classifier. Originally developed for the extraction of protein-protein interactions, it was later adapted to the domain of pharmacogenomics. Using the PharmGKB database,[24] JReX obtained F-scores in the 80% range for gene-disease, gene-drug, and drug-disease relations.[25] The Semantic Knowledge Representation (SKR) system [26], developed by the National Library of Medicine, provides semantic representations of biomedical text by building on resources currently available at the library. SKR applies two programs, MetaMap[27] and SemRep.[28], both of which utilize information available in the Unified Medical Language System (UMLS).[29] SKR has been used for concept-based query expansion, for identification of anatomical terminology and relations in clinical records, and for mining biomedical texts for drug-disease relations and molecular biology information.[30] Java Simple Relation Extraction (jSRE) is still an other relation extraction tool based on SVM. It has been used for the identification and extraction of drug-related adverse effects from Medline case reports,[31, 32] achieving an F-score of 87% on the ADE corpus.[33] A framework that integrates nine event extraction systems is U-Compare.[34] The U-Compare event meta-service provides an ensemble approach to relation extraction, where the combination of systems may produce a significantly better result than the best individual system included in the ensemble.[34] Hybrid approaches that combine different techniques have also been shown to perform well. Bui et al. [35] proposed a novel, very fast system that combines natural language processing (NLP) techniques with automatically and manually generated rules, and obtained an F-score of 53% on the Genia event corpus,[36] a result that is comparable to other state-of-

the-art event extraction systems.

Most of the existing relation extraction systems use machine-learning algorithms and require an annotated corpus for training. There are several publicly available biomedical text corpora with manually annotated relations, for instance the corpora generated as part of the Biocreative[37–39] and BioNLP[40, 41] challenges, the GENIA event corpus,[36] PharmGKB,[24] and the ADE corpus.[33] Most of these corpora focus on protein-protein interactions or other bio-events, while only two address drug-disease relations (PharmGKB) or drug-adverse effect relations (ADE corpus). As some of the annotations in PharmGKB have been reported to be hypothetical,[42] we chose to use the ADE corpus as the gold standard corpus (GSC) for our experiments.

METHODS

Corpus

The ADE corpus consists of 1644 Medline abstracts with 2972 case reports that were manually annotated and harmonized by three annotators. The selection of the case reports was based on a PubMed query with the MeSH (Medical Subject Headings) terms "drug therapy" and "adverse effect". The corpus contains annotations of 5063 drugs, 5776 conditions (diseases, signs, symptoms), and 6821 relations between drugs and conditions representing clear adverse effect occurrences.[33] Each relation consists of a Medline identifier, the sentence that contains this relation, the text and position of the drug, and the text and position of the adverse effect. Relations were only annotated if they occur in a single sentence. Drugs and conditions were not annotated if they were not part of an adverse event relation. We divided the ADE corpus into two sets: a small training set of 50 randomly selected abstracts, and a test set with the remaining abstracts (Table 1). Contrary to previous studies,[32] we used all sentences in the abstracts, both "positive" sentences that contain at least one relation according to the gold standard, and "negative" sentences that do not contain a relation.

	Training set	Test set	Total
Abstracts	50	1594	1644
Relations	201	6620	6821
Sentences with at least one relation	130	4142	4272
Sentences with no relation	233	7327	7560

Table 1. Number of abstracts, relations, and sentences in the ADE corpus.

Relation extraction system

The relation extraction system consists of two main modules: a concept identification module that identifies drugs and adverse effects, and a knowledge-base module that determines whether an adverse effect relation can be established between the entities that are found. All modules were integrated in the Unstructured Information Management Architecture framework.[43]

We used the Peregrine system (https://trac.nbic.nl/data-mining/) as the basis of our concept identification system. Peregrine is a dictionary-based concept recognition and normalization tool, developed at the Erasmus University Medical Center.[44] It finds concepts by dictionary look-up, performs word-sense disambiguation if necessary, and assigns concept unique identifiers (CUIs). We used Peregrine with a dictionary based on version 2012AA of the UMLS Metathesaurus, only keeping concepts that belong to the semantic groups "Chemicals & Drugs" and "Disorders".[45] Rewrite and suppress rules are applied to the terms in the dictionary to enhance precision and recall.[46] To further improve concept identification, we employed a rule-based NLP module that carries out coordination resolution, abbreviation expansion, term variation, boundary correction, and concept filtering.[47] We previously developed and tested this module for disease identification.[47] The NLP module was not modified for the current task except for the concept filtering, which was adjusted based on our training data.

The knowledge base consists of a graph in which the vertices represent concepts and the edges represent relations between these concepts. The knowledge base is populated with concepts (CUIs) and relations extracted from the UMLS Metathesaurus and the UMLS Semantic Network version 2012AA. Each edge or relation in the knowledge base has a relation type, e.g., "is-a" or "causes". The edges that connect two concepts form a path, with a length equal to the number of edges. The distance between two concepts is defined as the length of the shortest path. Note that there may be multiple shortest paths, but there is only one shortest path length.

For each sentence in the corpus, we determined the distance in the knowledge base between the drugs and adverse effects that were found by the concept identification module. Only if the distance between a drug-adverse effect pair was less than or equal to a distance threshold, a relation was considered present. Based on our training set, a distance threshold of four gave best performance results.

Further reduction of false-positive drug-adverse effect relations was attempted by taking into account the type of the relations in the shortest paths between drugs and adverse

events. In our training set, we counted the number of each relation type in the paths that resulted in false-positive and in true-positive drug-adverse effect relations. If for a relation type the ratio of the false-positive count plus one and the true-positive count plus one was greater than seven, we discarded any path containing that relation type. The value of seven was determined experimentally on the training set as yielding the best performance.

Performance evaluation

In the ADE corpus, drug-adverse effect relations are annotated at the sentence level by specifying the start and end positions of the drug and the adverse effect. We counted a relation found by our system as true positive if the boundaries of the drug and adverse effect exactly matched those of the gold standard. If a gold-standard relation was not found, i.e., if the concept boundaries were not rendered exactly by the system, it was counted as false negative. If a relation was only found by the system, i.e., the concept boundaries did not exactly match the gold standard, it was counted as false positive. Performance was evaluated in terms of precision, recall, and F-score. An error analysis was carried out on a sample of 100 randomly selected errors that were made by our relation extraction system.

RESULTS

Performance of the relation extraction system

Table 2 shows the performance of the Peregrine baseline system on the test set of the ADE corpus, and the incremental contribution for each of the different modules. The baseline system had a high recall but low precision, yielding an F-score of 16.1%. Use of the NLP module more than doubled the F-score. Application of the knowledge base further improved the F-score by 12.6 percentage points. Relation-type filtering increased the F-score by another 4.3 percentage points. Overall, the knowledge-base module decreased recall by 8.1 percentage points, but increased precision by 17.0 percentage points.

System	Precision	Recall	F-score
Baseline	8.9	78.4	16.1
+ NLP module	21.1	82.9	33.6
+ Knowledge base	32.8	78.1	46.2
+ Relation-type filtering	38.1	74.8	50.5

Table 2. Performance (in %) of the baseline relation extraction system and the incremental contribution of different system modules, on the test set of the ADE corpus.

Effect of different distance thresholds in the knowledge base

Table 3 shows the performance of the relation extraction system on the ADE test corpus for different distance thresholds (the maximum allowed length of the shortest path between a drug and an adverse effect) in the knowledge base. The highest F-score of 50.5% is obtained with a distance of four. Lowering the distance threshold increases precision and decreases recall. The highest recall is 76.5% (precision 37.0%) at a threshold of five, the highest precision is 43.2% (recall 1.6%) at a threshold of one.

Threshold	Precision	Recall	F-score
1	43.2	1.6	3.1
2	41.8	15.2	22.3
3	40.6	64.1	49.7
4	38.1	74.8	50.5
5	37.0	76.5	49.9

Table 3. Performance (in %) of the relation extraction system on the test set of the ADE corpus for different distance thresholds in the knowledge base.

Effect of different training set sizes

To assess the effect of increasing amounts of training data on system performance, training sets of 100, 200, and 400 abstracts were selected from the ADE corpus. The abstracts in a training set were a subset of the abstracts in the next larger training set. For each training set, the corresponding test set consisted of the remaining abstracts in the

ADE corpus. Table 4 shows that the performance of the relation extraction system improves with larger amounts of training data, but is leveling off with increasing size. The system obtains an F-score of 54.3% when trained on 400 abstracts, which is an improvement of 3.8 percentage points as compared with the system trained on 50 abstracts The NLP module contributed 1.7 percentage points to this improvement, and the relation-type filter module 2.1 percentage points. The baseline Peregrine module and the knowledge-base module do not require training and thus were not changed.

Abstracts for training	Precision	Recall	F-score
50	38.1	74.8	50.5
100	39.8	75.2	52.1
200	41.1	75.7	53.3
400	42.1	76.3	54.3

Table 4. Performance (in %) of the relation extraction system on the test set of the ADE corpus for different sizes of the training set.

Performance comparison of knowledge based and machine-learning based relation extraction

The ADE corpus has previously been used to develop and evaluate a machine-learning based relation extraction system based on jSRE.[32] To compare the performances of our knowledge-based relation extraction system and a machine learning-based system, we set up the same training and test environment as described by Gurulingappa et al.[32]. Similar to Gurulingappa et al., we removed all relations with nested annotations in the gold standard (e.g., "acute lithium intoxicity", where "lithium" is related to "acute intoxicity"), and only used the positive sentences in the ADE corpus. In [32], the true relations (taken from the gold standard) were supplemented by false relations (taken from co-occurring drugs and conditions that were found by ProMiner,[48] a dictionary-based entity recognition system), in a ratio of 1.26:1. To create a corpus with the same ratio to train and test our system, we took all true relations in which the concepts were found by Peregrine and the NLP module, and randomly added false co-occurrence relations generated by Peregrine and the NLP module, until the ratio of 1.26:1 was reached.

Table 5 shows the performance of our knowledge-base system and the previously reported performance of jSRE.[32] Without any training corpus, i.e., only applying the knowledge base but not the relation-type filtering, which requires training, our system already got an F-score of 88.5%. Additional use of the relation-type filter trained on small

sets of 10 or 50 abstracts, resulted in slightly higher F-scores, which were substantially better than those obtained with jSRE. The best F-score reported for jSRE, when about 90% of the abstracts in the corpus was utilized for training, was 87%.[32]

Training set (abstracts)	Machine learning			Knowledge base		
	Precision	Recall	F-score	Precision	Recall	F-score
0	n/a	n/a	n/a	88.5	88.6	88.5
10	58	6	55	89.1	88.2	88.6
50	79	87	82	91.8	86.1	88.8

Table 5. Performance (in %) of a machine-learning based (jSRE) relation extraction system [32] and the knowledge-based system on a subset of the ADE test corpus (see text).

Error analysis

We randomly selected 100 errors that the system made in our test set, and manually classified them into different error types (Table 6). False-positive errors were mostly due to drugs and adverse effects that were correctly found by the concept identification module, but were wrongly annotated by the knowledge-base module as having a relation. Of the 64 errors of this type, 46 occurred in negative sentences, i.e., sentences that do not contain any drug-adverse effect relation according to the gold standard. For instance, the gold standard did not annotate a relation in "Norethisterone and gestational diabetes", but the system found "norethisterone" as a drug concept, "gestational diabetes" as an adverse effect, and generated a false-positive relation between these two concepts. Eighteen of the 64 errors occurred in positive sentences. For instance, in the sentence "Pneumocystis carinii pneumonia as a complication of methotrexate treatment of asthma", the gold standard annotated a relation between the drug "methotrexate" and the adverse effect "pneumocystis carinii pneumonia", concepts that were also found by the system. However, the system also annotated "asthma" as another adverse effect concept, which generated a false-positive relation between "methotrexate" and "asthma". The second type of false-positive errors was caused by incorrectly found concepts, for which a relation was found in the knowledge base. For instance, in "Drug-induced pemphigus related to angiotensin-converting enzyme inhibitors", the system incorrectly annotated "angiotensin-converting enzyme inhibitors" as a drug, and wrongly established a relation

with “drug-induced pemphigus.” Altogether, false-positive errors accounted for 79% of all errors.

Error type	Number
False-positive relations	
Entities correctly identified, with incorrect relation in the knowledge base	64
Entities incorrectly identified, with a relation in the knowledge base	15
False-negative relations	
Entities correctly identified, but relation filtered out	8
Entities not identified, no relation established	13

Table 6. Error analysis of 100 randomly selected errors on the ADE test set.

False-negative errors were generated because the system missed a concept, or did not find a relation in its knowledge base between two correctly found concepts. An example of the first type of error is the term “TMA” (thrombotic microangiopathy), which the system incorrectly recognized as a drug in the sentence “A case report of a patient with probable cisplatin and bleomycin-induced TMA is presented.” The system then missed the relations between the adverse effect “TMA” and the drugs “cisplatin” and “bleomycin”. The other type of false-negative error is illustrated by the sentence “Encephalopathy and seizures induced by intravesical alum irrigations”, which contains two relations, one between “alum” and “encephalopathy”, the other between “alum” and “seizures”. The concept-recognition module found all three concepts correctly, but the knowledge-base module could not find the relation between “alum” and “seizures”. False-negative errors contributed 21% to the total number of errors.

DISCUSSION

We have investigated the use of NLP and a knowledge base to improve the performance of a system to extract adverse drug events. By applying a set of post-processing rules that utilize POS and chunking information, and exploiting the information contained in the UMLS Metathesaurus and the UMLS Semantic Network, the F-score on the ADE corpus improved by 34.4 percentage points as compared to a simple co-occurrence baseline system. To our knowledge, this is the first study that uses a knowledge base to improve biomedical relation extraction.

The main advantage of our approach as compared to machine-learning approaches is the relatively small set of annotated data required for training. For the ADE corpus, we only used 50 abstracts (3% of the total corpus) to train our system. When we compared our system with a machine-learning system trained on a document set of the same size, our system performed substantially better. Although a machine-learning approach usually performs very well if trained on a sufficiently large training set, the creation of a gold standard corpus (GSC) is tedious and expensive: annotation guidelines have to be established, domain experts must be trained, the annotation process is time-consuming, and annotation disagreements have to be resolved.[49] As a consequence, GSCs in the biomedical domain are generally small and focus on specific subdomains. It should also be noted that even when most of the ADE corpus was used to train the machine-learning system, it did not perform better than our knowledge-based system.

It is difficult to compare the performance of our system with those of the many other relation extraction systems reported in the literature because of the wide variety of relation extraction tasks and evaluation sets. We also evaluated the performance of the whole relation extraction pipeline, whereas other studies often focused on the relation extraction performance under the assumption that the entities involved were correctly recognized.[12, 32, 50–52] Moreover, previous systems were sometimes evaluated on a selected set of abstract sentences. As mentioned earlier, Gurulingappa et al.[32] mainly used positive sentences with at least one relation from the abstracts in the ADE corpus, and did not consider relations with nested entities. Similarly, Buyko et al. only used sentences with at least one gene-disease, gene-drug, or drug-disease relation in the PharmGKB database. Both systems obtained F-scores larger than 80%. In a comparable test setting, our system obtained at least as good results (F-score 89%), but in a more realistic test environment, which included the whole relation extraction pipeline and all sentences of the abstracts, performance dropped considerably (F-score 51%). This can largely be attributed to the additional false-positive relations in the negative sentences of the abstracts, decreasing precision considerably.

Our error analysis indicated that for the majority of errors the entities are correctly identified (72/100), the error being made in the knowledge-base module. To reduce the number of false-negative errors, we plan to extend the knowledge base by including relations mined from other drug-adverse effect databases, such as DailyMed,[53] DBpedia,[54] and DrugBank.[55] False-positive errors generated by the knowledge base may be decreased by including more strict filtering rules on the relation types. We also noted several general concepts, e.g., “patient”, “drug”, and “disease”, that are highly connected. Their removal may improve performance. Finally, we currently took all relation types as equally important and did not consider the plausibility of a path that

connects two concepts. Development of a weighting scheme of different relation types and rules that check the plausibility of the possible paths may be able to better distinguish false from true drug-adverse effect relations.

Our system has some limitations. To establish a potential relation, the knowledge-base module requires concept identifiers as its input. Concept identification is generally considered more difficult than the recognition of named entities, which can serve as the input for machine-learning based relation extraction. Another, related limitation of the current system is that the UMLS Metathesaurus does not provide extensive coverage of genes and proteins. The incorporation of relations from other sources of knowledge, such as UniProt or the databases that are made available through the LODD (Linking Open Drug Data) project, may remedy this drawback.

We have shown that a knowledge-based approach can be used to extract adverse drug events from biomedical text without need for a large training set. Whether use of a knowledge base is equally advantageous for other biomedical relation extraction tasks remains to be investigated.

REFERENCES

1. Jensen LJ, Saric J, Bork P: **Literature mining for the biologist: from information retrieval to biological discovery.** *Nature Reviews Genetics* 2006, **7**:119–129.

2. Zweigenbaum P, Demner-Fushman D, Yu H, Cohen KB: **Frontiers of biomedical text mining: current progress**. *Brief. Bioinform* 2007, **8**:358–75.

3. Simpson MS, Demner-fushman D: **Biomedical text mining: a survey of recent progress**. In *Mining Text Data*. Springer; 2012:465–517.

4. Revere D, Fuller S: **Characterizing biomedical concept relationships**. *Medical Inform* 2005, **8**:183–210.

5. Dai HJ, Chang YC, Tzong-Han Tsai R, Hsu WL: **New challenges for biological text-mining in the next decade**. *J. Comput. Sci. Tech* 2010, **25**:169–79.

6. Cohen AM, Hersh WR: **A survey of current work in biomedical text mining**. *Brief. Bioinform* 2005, **6**:57–71.

7. Krallinger M, Erhardt RAA, Valencia A: **Text-mining approaches in molecular biology and biomedicine**. *Drug Discov Today* 2005, **10**:439–445.

8. Kandula S, Zeng-Treitler Q: **Exploring relations among semantic groups: a comparison of concept co-occurrence in biomedical sources.** *Stud Health Technol Inform* 2010, **160**:995–999.

9. Airola A, Pyysalo S, Björne J, Pahikkala T, Ginter F, Salakoski T: **All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning**. *BMC Bioinform* 2008, **9**:S2.

10. Pyysalo S, Airola A, Heimonen J, Björne J, Ginter F, Salakoski T: **Comparative analysis of five protein-protein interaction corpora**. *BMC Bioinform* 2008, **9**:S6.

11. Jang H, Lim J, Lim J-H, Park S-J, Lee K-C, Park S-H: **Finding the evidence for protein-protein interactions from PubMed abstracts.** *Bioinformatics* 2006, **22**:e220–e226.

12. Rinaldi F, Schneider G, Kaljurand K, Hess M, Andronis C, Konstandi O, Persidis A: **Mining of relations between proteins over biomedical scientific literature using a deep-linguistic approach.** *Artif Intell Med* 2007, **39**:127–136.

13. Fundel K, Küffner R, Zimmer R: **RelEx--relation extraction using dependency parse trees.** *Bioinformatics* 2007, **23**:365–71.

14. Saric J, Jensen LJ, Ouzounova R, Rojas I, Bork P: **Extraction of regulatory**

gene/protein networks from Medline. *Bioinformatics* 2006, **22**:645–50.

15. Kang N, Van Mulligen EM, Kors JA: **Comparing and combining chunkers of biomedical text.** *J Biomed Inform* 2011, **44**:354–60.

16. Huang M, Zhu X, Li M: **A hybrid method for relation extraction from biomedical literature.** *Int J Med Inform* 2006, **75**:443–55.

17. Buchholz S, Marsi E: **CoNLL-X shared task on multilingual dependency parsing**. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*. Association for Computational Linguistics; 2006:149–164.

18. Katrenko S, Adriaans P: **Learning Relations from Biomedical Corpora Using Dependency Tree Levels**. In *KDECB'06 Proceedings of the 1st International Conference on Knowledge Discovery and Emergent Complexity in Bioinformatics*. Ghent, Belgium: 2006, **4366**:61–80.

19. Kim J-H, Mitchell A, Attwood TK, Hilario M: **Learning to extract relations for protein annotation.** *Bioinformatics* 2007, **23**:256–63.

20. Ozg A, Radev DR: **Semi-supervised classification for extracting protein interaction sentences using dependency parsing**. *Comput Linguist* 2007, **1**:228–37.

21. Huang Y, Lowe HJ, Klein D, Cucina RJ: **Improved identification of noun phrases in clinical radiology reports using a high-performance statistical natural language parser augmented with the UMLS specialist lexicon.** *J Am Med Inform Assoc* 2005, **12**:275–285.

22. Demner-Fushman D, Chapman W, McDonald C: **What can natural language processing do for clinical decision support?** *J Biomed Inform* 2009, **42**:760–72.

23. Hahn U, Buyko E, Landefeld R, M ¨ hlhausen M, Poprat M, Tomanek K, Wermter J: **An overview of JCoRe, the JULIE lab UIMA component repository**. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*. Marrakech, Morocco: 2008:1–7.

24. Thorn CF, Klein TE, Altman RB: **Pharmacogenomics and bioinformatics: PharmGKB.** *Pharmacogenomics* 2010, **11**:501–505.

25. Buyko E, Beisswanger E, Hahn U: **The extraction of pharmacogenetic and pharmacogenomic relations–a case study using pharmgkb**. In *Pac Symp Biocomput*. Hawaii, USA: 2012, **376**:376–87.

26. Rindflesch TC, Fiszman M: **The interaction of domain knowledge and linguistic**

structure in natural language processing: interpreting hypernymic propositions in biomedical text. *J Biomed Inform* 2003, **36**:462–477.

27. Aronson AR: **Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program.** In *Proceedings of the AMIA Symposium*. Philadelphia, USA: 2001, **pp**:17–21.

28. Rindflesch T, Fiszman M, Libbus B: **Semantic interpretation for the biomedical research literature**. *Medical Inform* 2005, **8**:399–422.

29. Bodenreider O: **The Unified Medical Language System (UMLS): integrating biomedical terminology.** *Nucleic Acids Res* 2004, **32**:267–70.

30. Rindflesch TC, Aronson AR: **Semantic processing for enhanced access to biomedical knowledge**. In *Real World Semantic Web Applications*. John Wiley & Sons; 2002, **92**:157–72.

31. Gurulingappa H, Fluck J, Hofmann-Apitius M, Toldo L: **Identification of adverse drug event assertive sentences in medical case reports**. In *First International Workshop on Knowledge Discovery and Health Care Management (KD-HCM), European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*. Athens, Greece: 2011:16–27.

32. Gurulingappa H, Rajput AM, Toldo L: **Extraction of Adverse Drug Effects from Medical Case Reports**. *J Biomed Semantics* 2012, **3**:15.

33. Gurulingappa H, Rajput AM, Roberts A, Fluck J, Hofmann-Apitius M, Toldo L: **Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports**. *J Biomed Inform* 2012, **45**:885–92.

34. Kano Y, Baumgartner WA, McCrohon L, Ananiadou S, Cohen KB, Hunter L, Tsujii J: **U-Compare: share and compare text mining tools with UIMA**. *Bioinformatics* 2009, **25**:1997–8.

35. Bui QC, Sloot PMA: **A robust approach to extract biomedical events from literature**. *Bioinformatics* 2012, **28**:2654–61.

36. Tateisi Y, Yakushiji A, Ohta T, Tsujii J: **Syntax annotation for the GENIA corpus**. In *Proceedings of the Companion Volume of the Second International Joint Conference on Natural Language Processing IJCNLP05*. Jeju Island, Korea: 2005:222–7.

37. Krallinger M, Leitner F, Rodriguez-Penagos C, Valencia A: **Overview of the**

protein-protein interaction annotation extraction task of BioCreative II. *Genome Biol* 2008, **9**:S4.

38. Leitner F, Mardis SA, Krallinger M, Cesareni G, Hirschman LA, Valencia A: **An overview of BioCreative II. 5 IEEE/ACM Trans**. *Comput Biolog Bioinform* 2010, **7**:385–99.

39. Krallinger M, Vazquez M, Leitner F, Salgado D, Chatr-aryamontri A, Winter A, Perfetto L, Briganti L, Licata L, Iannuccelli M: **The protein-protein interaction tasks of BioCreative III: classification/ranking of articles and linking bio-ontology concepts to full text**. *BMC Bioinform* 2011, **12**:S3.

40. Kim J-D, Ohta T, Pyysalo S, Kano Y, Tsujii J: **Overview of BioNLP'09 shared task on event extraction**. In *Proceedings of the Workshop on BioNLP Shared Task*. Boulder, USA: 2009:1–9.

41. Kim JD, Pyysalo S, Ohta T, Bossy R, Nguyen N, Tsujii J: **Overview of BioNLP shared task 2011**. In *Proceedings of the BioNLP Shared Task 2011 Workshop*. 2011:1–6.

42. Rinaldi F, Clematide S, Garten Y, Whirl-Carrillo M, Gong L, Hebert JM, Sangkuhl K, Thorn CF, Klein TE, Altman RB: **Using ODIN for a PharmGKB revalidation experiment**. *Database J Biolog Database Cur* 2012, **2012**.

43. Ferrucci D, Lally A: **UIMA: an architectural approach to unstructured information processing in the corporate research environment**. *Nat Lang Eng* 2004, **10**:327–48.

44. Schuemie MJ, Jelier R, Kors JA: **Peregrine: lightweight gene name normalization by dictionary lookup**. In *Proceedings of the BioCreAtIvE II Workshop*. Madrid, Spain: 2007:131–3.

45. Bodenreider O, McCray AT: **Exploring semantic groups through visual approaches**. *J Biomed Inform* 2003, **36**:414–32.

46. Hettne KM, Van Mulligen EM, Schuemie MJ, Schijvenaars BJ, Kors JA: **Rewriting and suppressing UMLS terms for improved biomedical term identification**. *J Biomed Semantics* 2010, **1**:1–5.

47. Kang N, Singh B, Afzal Z, van Mulligen EM, Kors JA: **Using rule-based natural language processing to improve disease normalization in biomedical text**. *J Am Med Inform Assoc* 2012:doi:10.1136/amiajnl–2012–001173.

48. Hanisch D, Fundel K, Mevissen H-T, Zimmer R, Fluck J: **ProMiner: rule-based**

protein and gene entity recognition. *BMC Bioinform* 2005, **6**:S14.

49. Kang N, van Mulligen EM, Kors JA: **Training text chunkers on a silver standard corpus: can silver replace gold?** *BMC Bioinform* 2012, **30**:13–17.

50. Melton GB, Hripcsak G: **Automated detection of adverse events using natural language processing of discharge summaries.** *J Am Med Inform Assoc* 2005, **12**:448–57.

51. Chun HW, Tsuruoka Y, Kim JD, Shiba R, Nagata N, Hishiki T, Tsujii J: **Extraction of gene-disease relations from Medline using domain dictionaries and machine learning**. In *Pac Symp Biocomput*. Rockville, MD, USA: 2006, **11**:4–15.

52. Uzuner O, South BR, Shen S, Duvall SL: **2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text.** *J Am Med Inform Assoc* 2011, **18**:552–6.

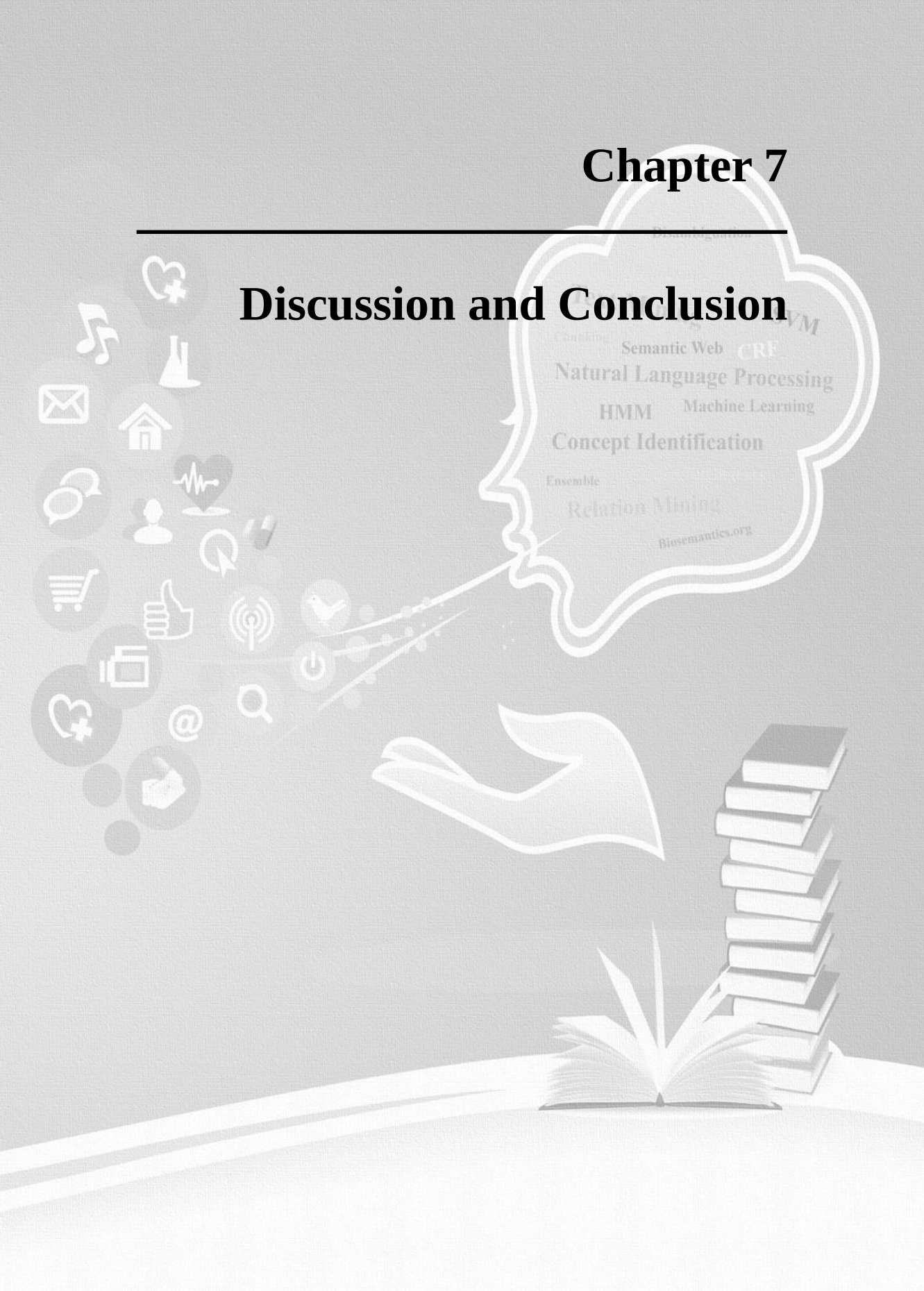
53. Elkin PL, Carter JS, Nabar M, Tuttle M, Lincoln M, Brown SH: **Drug knowledge expressed as computable semantic triples.** *Stud Health Technol Inform* 2011, **166**:38–47.

54. Bizer C, Lehmann J, Kobilarov G, Auer S, Becker C, Cyganiak R, Hellmann S: **DBpedia - A crystallization point for the Web of Data**. *Web Seman Scie Serv Age WWW* 2009, **7**:154–165.

55. Wishart DS, Knox C, Guo AC, Shrivastava S, Hassanali M, Stothard P, Chang Z, Woolsey J: **DrugBank: a comprehensive resource for in silico drug discovery and exploration**. *Nucleic Acids Res* 2006, **34**:668–672.

Chapter 7

Discussion and Conclusion



Using natural language processing (NLP) to improve biomedical concept normalization and relation mining is an important way to improve the performance of biomedical text mining, and it plays a crucial role in automatically gathering facts and evidence necessary for life science [1]. Although many NLP methods have been developed and implemented, this task remains challenging because of the inherent complexity of natural language, the difference between biomedical text mining tasks, and the lack of training data and suitable techniques.

To improve the usage of NLP for biomedical text mining, many aspects need to be studied further, such as the lexical analysis of biomedical text, the disambiguation of concepts, the abbreviation of different concepts, and the availability of training data [2]. Understanding these aspects may bring about better results in NLP methods. As an attempt to contribute to this research field, we have investigated new alternative approaches to a number of relevant NLP facets such as chunking, ensemble system, and a silver standard corpus (SSC) that is automatically generated by combining the output of multiple text mining systems.

Although improving linguistic technologies could still enhance text-mining, the improvements in recent years have been marginal. The question is whether the combination of linguistic technologies with semantic approaches such as a domain knowledge base can improve text-mining. However, this integration of linguistic and semantic methods still faces some challenges, such as how to efficiently and effectively represent human knowledge in a formal computational model, and how to take advantage of semantic techniques and apply them to traditional linguistic text-mining.

SUMMARY OF THE MAIN FINDINGS

This thesis describes the research results of several aspects of using NLP to improve biomedical concept normalization and relation mining. We discuss approaches for several theoretical and practical challenges we encountered during the investigation. We started with a comparison and combination of biomedical chunkers [3], and subsequently investigated the possibility of replacing a gold standard corpus (GSC) with an SSC [4]. After that, we explored the use of an NLP ensemble approach to combine several text mining systems for improving concept extraction from clinical records [5]. Finally, we investigated how NLP can be used to solve the common problems occurred in biomedical concept normalization and relation mining. A summary of the main findings discussed in this thesis can be found below.

In chapter 2, we investigated six frequently used chunkers. With respect to performance

and usability, OpenNLP [6] performed best. When combining the results of the chunkers by means of a simple voting scheme, the F-score of the combined system improved by 3.1 percentage points for noun phrases and 0.6 percentage points for verb phrases as compared to the best single chunker. Changing the voting threshold offers a simple way to modify the system's precision and recall, making it suitable for a number of scenarios that require high precision or high recall.

After comparing and combining the chunkers, we investigated the use of an SSC that has been automatically generated by combining the output of different chunking systems in chapter 3. We explored two scenarios: one in which chunkers are trained on an SSC in a new domain for which a GSC is not available, and one in which chunkers are trained on an available, although small GSC but supplemented with an SSC. From the results of these two scenarios, we conclude that an SSC can be a viable alternative for or a supplement to a GSC when training chunkers in a biomedical domain.

The approach of combining different text-mining systems was explored again in chapter 4, where we selected two dictionary-based systems and five statistical-based systems that were trained to annotate medical problems, tests, and treatments in clinical records. The ensemble system has better precision and recall than any of the individual systems, yielding an F-score that is 4.6 percentage point higher than the best single system. Changing the voting threshold offered a simple way to obtain a system with high precision or high recall. This result shows that the ensemble-based approach is straightforward and allows the balancing of precision versus recall of the combined system.

After investigating the usefulness of NLP as an adjunct to dictionary-based concept normalization, we describe in chapter 5 how with the aid of an NLP module, the F-scores of MetaMap [7] and Peregrine [8] improved by more than 10% for boundary matching, and more than 15% for concept identifier matching. We showed the added value of NLP for the recognition and normalization of diseases with MetaMap and Peregrine. The NLP module is general and might be applied in combination with other biomedical concept normalization systems in a similar domain.

Chapter 6 provides results on the usefulness of a knowledge base and NLP techniques in improving biomedical relation mining. We have shown that the knowledge base could be used to detect the negative sentences and improve the performance of biomedical relation extraction. The performance in a real-life test environment is much lower than in an optimized test environment. The knowledge base module is general and might be applied in combination with any biomedical relation extraction system in a similar domain.

Based on the findings reported in this thesis, we conclude that NLP, the ensemble approach, and the knowledge base could be used to improve the performance of concept normalization and relation mining.

INTERPRETATION OF FINDINGS

In this section we present an overview, discuss the findings as described in chapter 2-6, and provide a detailed answer to the questions in chapter 1. Some important issues in previous chapters are also discussed in this section.

Improve text-mining

Concept or named-entity recognition aims at finding text strings that refer to entities, and marking each entity with a semantic type, such as “gene”, “drug”, or “disease”. Concept normalization goes beyond entity recognition. It assigns a unique identifier to the recognized concept, which links it to a resource that contains further information about the concept, such as its definition, its preferred name, synonyms, and its relationships with other concepts. Most studies described in this thesis used text chunking and concept identification as essential pre-processing steps in relation mining systems. Because of this, high precision chunking and concept identification are required for the high precision relation mining we need.

In chapter 2, we evaluated several chunking systems, compared these systems based on performance, and selected the best chunking system, OpenNLP [6], as our default chunking system for the following studies. OpenNLP obtained an F-score of 89.7% [3], but the combination of different chunking systems obtained an even better result (3.1 percentage point for noun phrases and 0.6 percentage point for verb phrases) than OpenNLP. However, whether this is good enough to be applied in practical NLP tasks is still an open question. There could be room for improvement, in particular for noun-phrase detection.

Whether the current chunking results are good enough for practical use is also determined by the impact of different chunking errors on the performance of the whole information extraction pipeline. For example, it may well be true that splitting or joining a verb phrase is less important than missing or inserting a noun or verb phrase when it comes to information extraction. It would be interesting to investigate the impact of chunking errors on real NLP tasks.

Furthermore, reports [9] showed that the performance of a chunker depends on the

different test corpora for the different domains. For instance, Buyko et al. [6] compared the performance of the NLP components from OpenNLP, including the chunker, when trained on the Wall Street Journal corpus with chunkers trained on two biomedical corpora (GENIA and PennBioIE). They concluded that the performance figures from the newspaper domain are comparable with those from the biology domain. This suggests that our results can be generalized to other domains provided that systems are properly retrained on domain-specific corpora.

In chapter 5 we showed that several NLP modules and rules can be applied to improve disease concept normalization. These rules use syntactic information to solve the common problems that occur in concept normalization tasks [10–12], including coordination, abbreviation, term variation, boundary correction, and filtering.

Many of the rules discussed in chapter 5 have not been used before in their specific form, but similar rules have previously been proposed in many different studies [13–15]. Combining different types of rules into a single system and showing the contribution of each sub-module to the overall performance for disease concept normalization, a task which has not been addressed before, is novel. The sub-modules are general and may be combined, as a whole or separately, with other concept normalization systems in a similar domain. These approaches might also be applied to other text mining tasks, such as knowledge discovery and information retrieval.

We have shown the added value of the NLP module for the recognition and normalization of disease concepts with dictionary-based systems, such as MetaMap [7] and Peregrine [8]. The module is general and might be applied in combination with other concept normalization systems in a similar domain. However, it is still a topic of research to investigate whether these rules will have similar performance improvements when used for the normalization of different concept classes, such as genes or drugs.

Deal with sparse training data

To train a machine learning-based text mining system, a GSC is needed. Although we have shown that an automatically created SSC could be a viable alternative for or a supplement to a GSC when training chunkers in a biomedical domain, it is still not clear if this approach is applicable to other components in an NLP pipeline.

Obviously, the closer a silver standard approaches a gold standard for the domain of interest, the better the performance of systems trained on an SSC will be. It should be noted that the performance of the silver standard compared with the gold standard in our study is far from perfect: the PennBioIE SSC has an F-score of 84.5% for noun phrases

and 93.9% for verb phrases [4]. However, despite the differences between an SSC and a GSC, chunking systems trained on these corpora show remarkably similar performances. It is still an open question as to how an SSC of a lower (or higher) quality affects the performance of a system trained on the SSC.

We have shown that chunkers can obtain almost similar performances whether trained on an SSC or a GSC, but this does not mean that we can dispose of GSCs. Obviously, to create the SSC we need trained chunkers, and thus a GSC for their initial training. However, we can reduce the size of the required GSC. Our experiments indicated that a GSC consisting of only 10 or 25 abstracts but expanded with an SSC yields similar performances to a GSC of 100 or 250 abstracts. Practically, these results suggest that the time and effort spent in creating a sufficiently sized GSC may be much reduced.

In most situations, we have to use a GSC as a standard to evaluate the performance of a text mining system. Due to the fact that the creation of an GSC involves human experts, most GSCs only contain a small homogenous set, and there could be errors introduced by human experts [1, 16, 17]. By using an SSC, we could reduce such errors introduced by humans. Since the creation of an SSC is automatic, it could contain a large heterogeneous set. For different text-processing applications, increasing the amount of data for training a classifier has shown to improve the classifier's performance [18–20]. The use of an SSC may be beneficial in mitigating the "paucity-of-data" problem.

We have tested two scenarios in which an SSC is used in the field of text chunking, but the proposed approach is general and could be used in any field in which GSCs are needed to train classifiers. How the quality of an SSC affects classifier performance is still a topic of research. It is also unclear whether the use of SSCs for other application fields is equally advantageous as their use for text chunking.

The contribution of each sub-module in an ensemble system

The ensemble approach has been shown to be an efficient method of improving the performance of text-mining systems in chapter 2 and chapter 3. For recognizing concepts in clinical records, we also used an ensemble approach. The ensemble system had higher precision, recall, and F-score values than any of the individual systems considered in the study of chapter 4.

Although each sub-module showed a large difference in performance, almost all of them contributed to the performance of the ensemble system. The removal of the worst performing system, MetaMap [7], from the ensemble system slightly increased its performance. The subsequent removal of any of the other individual systems resulted in

a performance degradation of the ensemble system. This suggests that each system, even a low-performing system such as Peregrine [8], contributes to the high performance of the combined annotation system.

Many studies [5, 21–23] have shown that combining different classifiers with different misclassified instances could generate better performance than any individual classifier, which makes up the ensemble system, having the best performance. The reason is that different classifiers yield errors on different instances, and the combination of these classifiers can reduce the overall error to improve the performance of the ensemble system [22].

Our results suggest that classifier accuracy correlates better with the performance of the ensemble system than classifier diversity. When varying the number of systems in the ensemble, the worst performing systems gave the smallest contribution to the ensemble performance, this is similar to what has been reported in [24]. We have tried to achieve diversity by combining different types of classifiers. However, as mentioned in chapter 4, it is difficult to quantify diversity, and the relationship between classifier diversity and the performance of the combined system is not clear [21].

Although each sub-module generated different errors, almost half of the errors were due to a mismatch in one of the annotation boundaries. Many of these errors resulted from the incorrect handling of coordination or punctuation, which are also common error types for the recognition of noun phrases in biomedical texts [3, 9]. The impact of these errors on the performance of a whole information extraction pipeline is still a topic of research.

Integrating biomedical knowledge

In chapter 6 we describe an approach to overcome the performance barrier in linguistic text analysis by using a knowledge base to enhance the linguistic analysis with domain semantics. We show in this chapter that the knowledge base (a network that connects biomedical concepts with relations) could be used to improve the performance of biomedical relation extraction. By combining a set of post-processing rules that utilize POS and chunking information with a graph representation of the information contained in the UMLS Metathesaurus and the UMLS Semantic Network, the F-score on the Arizona Disease Corpus improved by 17.5 and 16.9 percentage points. This result shows that prior information contained in a graph database can help relation mining. For relation extraction, the NLP module is mainly used for concept identification, whereas the knowledge base uses the output from the NLP module.

Although in several studies [25, 26] F-scores have been reported up to 90%, the

evaluation setting in many of these studies is typically restricted only to the process of relation mining and therefore does not resemble a real life text mining situation. In a real life relation mining situation there are many essential preceding NLP steps required: sentence splitting, part-of-speech, chunking, named entity recognition, and concept identification. All these preceding steps have an impact on the performance of the relation mining task. When including these preceding NLP steps an F-score of only 50% can be achieved. A second issue was the bias in the evaluation set: only having sentences where a relation is contained and not a general abstract/text from PubMed, which also makes the test environment much easier than the real life situation.

A combination of the knowledge base and the NLP processing can in principle be applied to all different domains that the knowledge base covers. This is an advantage compared with traditional machine learning-based systems, which are often domain dependent because they need domain-specific corpora for training. However, whether our solutions can achieve a similar performance in other domains needs to be investigated.

LIMITATIONS AND FUTURE WORK

In previous chapters, chunking (shallow parsing) information has been used as an important resource for improving concept normalization and relation mining., The highest performance that could be obtained for noun phrase annotation is 89.7% for a single chunker, and 92.8% for a combined chunker [3]. As the performance of a chunker directly impacts the performance of concept normalization and relation mining, it is crucial to improve the chunkers, as was outlined in this thesis by using an ensemble of chunkers. In addition to this ensemble approach, this thesis demonstrated that this approach is beneficial for concept recognition in clinical records [5].

Gold standard corpora are essential for training machine learning-based chunkers and other components in text mining systems. We have shown in this thesis that an automatically created SSC can be a viable alternative for, or supplement to, a GSC when training chunkers in a biomedical domain. However, we did not test this approach in other text mining domains such as named entity recognition, concept normalization, and relation mining. Further investigations will have to reveal how the quality of an SSC affects classifier performances and whether the use of SSCs in other application areas is as equally advantageous as their use in chunking.

For the study of using rule-based NLP to improve disease normalization in biomedical text, some error analysis has been done. The analysis revealed that about half of the errors that remain after applying the NLP module can be denoted as term variation and filtering

errors. While further improvement of the sub-modules dealing with these errors may be possible, it is more likely that improved dictionaries and disambiguation methods will help to reduce these types of errors. In this respect approaches for generating more term variants could be useful. We also noticed that chunking sometimes provides insufficient information to resolve errors in complex sentences. Such information may possibly be derived from deep parsing, but exploring the usefulness of these techniques for our purposes will be left for further research.

Although the knowledge base improves the performance for relation extraction tasks and has several advantages compared with machine learning-based systems, there are still opportunities to further improve the performance. Currently the knowledge base only includes the UMLS Metathesaurus, and no other databases are included. The performance could be improved if the coverage of the knowledge base is expanded. For the UMLS Metathesaurus, it could also be optimized for different tasks, for instance, by trying to remove relations between concepts that do not belong to a clear defined semantic type, or remove relations related to very generic concepts. For the filter module, a simple keyword list has been used to extract the common error relations. Its performance might be improved if the keyword list is extracted using statistical methods.

REFERENCES

1. Dai HJ, Chang YC, Tzong-Han Tsai R, Hsu WL: **New challenges for biological text-mining in the next decade**. *J. Comput. Sci. Tech* 2010, **25**:169–79.

2. Zweigenbaum P, Demner-Fushman D, Yu H, Cohen KB: **Frontiers of biomedical text mining: current progress**. *Brief. Bioinform* 2007, **8**:358–75.

3. Kang N, Van Mulligen EM, Kors JA: **Comparing and combining chunkers of biomedical text.** *J Biomed Inform* 2011, **44**:354–60.

4. Kang N, van Mulligen EM, Kors JA: **Training text chunkers on a silver standard corpus: can silver replace gold?** *BMC Bioinform* 2012, **30**:13–17.

5. Kang N, Afzal Z, Singh B, van Mulligen EM, Kors JA: **Using an ensemble system to improve concept extraction from clinical records**. *J Biomed Inform* 2012.

6. Buyko E, Wermter J, Poprat M, Hahn U: **Automatically Adapting an NLP Core Engine to the Biology Domain**. In *Proceedings of the Joint BioLINKBio-Ontologies Meeting*. Fortaleza, Brasil: 2006:2–5.

7. Aronson AR: **Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program.** In *Proceedings of the AMIA Symposium*. Philadelphia, USA: 2001, **pp**:17–21.

8. Schuemie MJ, Jelier R, Kors JA: **Peregrine: lightweight gene name normalization by dictionary lookup**. In *Proceedings of the BioCreAtIvE II Workshop*. Madrid, Spain: 2007:131–3.

9. Wermter J, Fluck J, Stroetgen J, Geißler S, Hahn U: **Recognizing noun phrases in biomedical text: An evaluation of lab prototypes and commercial chunkers**. In *Proceedings of the 1st International Symposium on Semantic Mining in Biomedicine*. Hinxton, England: 2005:25–33.

10. Hirschman L, Colosimo M, Morgan A, Yeh AC-P: **Overview of BioCreAtIvE task 1B: normalized gene lists**. *BMC Bioinform* 2005, **6 Suppl 1**:S11.

11. Morgan AA, Lu Z, Wang X, Cohen AM, Fluck J, Ruch P, Divoli A, Fundel K, Leaman R, Hakenberg J, Sun C, Liu HH, Torres R, Krauthammer M, Lau WW, Liu H, Hsu CN, Schuemie M, Cohen KB, Hirschman LC-P: **Overview of BioCreative II gene normalization**. *Genome Biol* 2008, **9 Suppl 2**:S3.

12. Lu Z, Kao HY, Wei CH, Huang M, Liu J, Kuo CJ, Hsu CN, Tsai RTH, Dai HJ, Okazaki N: **The gene normalization task in BioCreative III**. *BMC Bioinform* 2011, **12**:S2.

13. Baumgartner WA, Lu Z, Johnson HL, Caporaso JG, Paquette J, Lindemann A, White EK, Medvedeva O, Cohen KB, Hunter L: **Concept recognition for extracting protein interaction relations from biomedical text**. *Genome Biol* 2008, **9**:S9.

14. Schwartz Hearst, M. A. AS: **A simple algorithm for identifying abbreviation definitions in biomedical text**. In *Proceedings of the 8th Pacific Symposium on Biocomputing*. Hawaii, USA: 2003:451–62.

15. Vilares J, Alonso MA, Vilares M: **Extraction of complex index terms in non-English IR: A shallow parsing based approach**. *Inform Process Manag* 2008, **44**:1517–37.

16. Tanabe L, Xie N, Thom LH, Matten W, Wilbur WJ: **GENETAG: a tagged corpus for gene/protein named entity recognition**. *BMC Bioinform* 2005, **6**:S3.

17. Douglass M, Clifford GD, Reisner A, Moody GB, Mark RG: **Computer-assisted de-identification of free text in the MIMIC II database**. In *Comput Cardiology*. 2004:341–4.

18. Banko M, Brill E: **Mitigating the Paucity-of-Data Problem : Exploring the Effect of Training Corpus Size on Classifier Performance for Natural Language Processing**. In *Proceedings of the first international conference on Human language technology research*. Stroudsburg, USA: 2001:1–5.

19. Yarowsky D, Florian R: **Evaluating sense disambiguation across diverse parameter spaces**. *Natural Language Eng* 2002, **8**:293–310.

20. Surdeanu M, Turmo J, Comelles E: **Named entity recognition from spontaneous open-domain speech**. In *Annual Conference of the International Speech Communication Association*. Lisbon, Portugal: 2005:3433–36.

21. Kuncheva LI, Whitaker CJ: **Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy**. *Mach Learn* 2003, **51**:181–207.

22. Polikar R: **Ensemble based systems in decision making**. *Ieee Circuits And Systems Magazine* 2006, **6**:21–45.

23. Wanas NM, Dara RA, Kamel MS: **Adaptive fusion and co-operative training for classifier ensembles**. *Pattern Recogn* 2006, **39**:1781–94.

24. Polikar R: **Ensemble based systems in decision making**. *IEEE Circuits Syst Mag* 2006, **6**:21–45.

25. Gurulingappa H, Rajput AM, Toldo L: **Extraction of Adverse Drug Effects from**

Medical Case Reports. *J Biomed Semantics* 2012, **3**:15.

26. Buyko E, Beisswanger E, Hahn U: **The extraction of pharmacogenetic and pharmacogenomic relations–a case study using pharmgkb**. In *Pac Symp Biocomput*. Hawaii, USA: 2012, **376**:376–87.

SUMMARY OF THE MAIN FINDINGS

This thesis describes the research results of several aspects of using NLP to improve biomedical concept normalization and relation mining. We discuss approaches for several theoretical and practical challenges we encountered during the investigation. We started with a comparison and combination of biomedical chunkers, and subsequently investigated the possibility of replacing a gold standard corpus (GSC) with an SSC. After that, we explored the use of an NLP ensemble approach to combine several text mining systems for improving concept extraction from clinical records. Finally, we investigated how NLP can be used to solve the common problems occurred in biomedical concept normalization and relation mining. A summary of the main findings discussed in this thesis is found below.

In **chapter 2**, we investigated six frequently used chunkers. With respect to performance and usability, OpenNLP performed best. When combining the results of the chunkers by means of a simple voting scheme, the F-score of the combined system improved by 3.1 percentage points for noun phrases and 0.6 percentage points for verb phrases as compared to the best single chunker. Changing the voting threshold offers a simple way to modify the system's precision and recall making it suited for a number of scenarios that require high precision or high recall.

After comparing and combining the chunkers, we investigated the use of a silver standard corpus (SSC) that has been automatically generated by combining the output of different chunking systems in **chapter 3**. We explored two use scenarios: one in which chunkers are trained on an SSC in a new domain for which a GSC is not available, and one in which chunkers are trained on an available, although small GSC but supplemented with an SSC. From the results of these two scenarios, we conclude that an SSC can be a viable alternative for or a supplement to a GSC when training chunkers in a biomedical domain.

The approach to combine different text-mining systems was explored again in **chapter 4**, where we selected two dictionary-based systems and five statistical-based systems that were trained to annotate medical problems, tests, and treatments in clinical records. The ensemble system has a better precision and recall than any of the individual systems, yielding an F-score that is 4.6 percentage point higher than the best single system. Changing the voting threshold offered a simple way to obtain a system with high precision or high recall. This result shows that the ensemble-based approach is straightforward and allows the balancing of precision versus recall of the combined system.

After investigating the usefulness of natural language processing (NLP) as an adjunct to dictionary-based concept normalization, we describe in **chapter 5** that with the aid of the NLP module, the F-scores of MetaMap and Peregrine improved more than 10% for boundary matching, and more than 15% for concept identifier matching. We showed the added value of NLP for the recognition and normalization of diseases with MetaMap and Peregrine. The NLP module is general and can be applied in combination with any concept normalization system.

Chapter 6 provides results on the usefulness of knowledge base and NLP techniques to improve biomedical relation mining. We have shown that the knowledge base could be used to detect the negative sentences and improve the performance of biomedical relation extraction. The performance in a real life test environment is much lower than an optimized test environment. The knowledge base module is general and can be applied in combination with any biomedical relation extraction system.

Based on the findings reported in this thesis, we conclude that NLP, ensemble approach, and knowledge base could be used to improve the performance of concept normalization and relation mining.

SAMENVATTING VAN DE BELANGRIJKSTE BEVINDINGEN

Dit proefschrift beschrijft de onderzoeksresultaten van verschillende NLP benaderingen van normaliseren van biomedische concepten en het vinden van relaties. We beschrijven de aanpak van verschillende theoretische en praktische uitdagingen die we tegen kwamen tijdens het onderzoek. We begonnen met een vergelijk en een combinatie van biomedische chunkers en onderzochten vervolgens de mogelijkheid om een gouden standaard corpus (GSC) te vervangen door een zilver standard corpus (SSC). Daarna hebben we uitgezocht of een NLP ensemble benadering om verschillende text mining systemen te combineren om het extraheren van concepten uit klinische dossiers te verbeteren. Ten slotte hebben we onderzocht hoe NLP gebruikt kan worden om de standaard problemen die onstaan bij het normaliseren van biomedische concepten en het vinden van relaties op te lossen. Een samenvatting van de belangrijkste bevindingen die besproken worden in dit proefschrift vindt u hieronder.

In **hoofdstuk 2** onderzochten we zes vaak gebruikte chunkers. Wanneer gelet wordt op prestatie en bruikbaarheid scoort OpenNLP het best. Wanner de resultaten van de chunkers door middel van een eenvoudig stem schema gecombineerd worden gaat de F-score van de combinatie met 3.1 percentage punten omhoog voor naamwoordelijke zinsdelen en 0.6 percentage punten voor werkwoordelijke zinsdelen vergeleken met de beste chunker. Door het aanpassen van de stem drempel wordt op eenvoudige manier de precision en de recall van het systeem aangepast waardoor het geschikt is voor taken waarin een hoge precisie of een hoge recall is vereist.

Na het vergelijken en combineren van chunkers hebben we het gebruik van een zilver standaard corpus (SSC) dat automatisch gegenereerd werd door het combineren van de resultaten van verschillende chunkers onderzocht in **hoofdstuk 3**. We hebben twee gebruik scneario's onderzocht: één waarin chunkers getrained worden op een SSC in een nieuw toepassingsgebied waar geen GSC beschikbaar is en één waarin chunkers getrained worden op een beschikbaar maar klein GSC dat aangevuld is met een SSC. Uit de resultaten van deze twee scenario's kunnen we concluderen dat een SSC een goed alternatief is om een aanvulling voor een GSC te maken wanneer het gaat om het trainen van chunkers in het biomedische domein.

De bandering om verschillende text mining systemen te combineren is in **hoofdstuk 4** weer onderzocht. Hier selecteerden we twee woordenlijst-gebaseerde systemen en vijf statistische systemen die getrained werden om medische problemen, testen en behandelingen in klinische dossiers te annoteren. Het ensemble systeem heeft een betere precision en recall dan elk van de afzonderlijke systemen, resulterend in een F-score die

4.6 percentage punten hoger is dan het beste systeem. Het veranderen van de stem drempel biedt een eenvoudige manier om hogere precision dan wel hogere recall te krijgen. Het resultaat laat zien dat de ensemble benadering rechttoe is en het mogelijk maakt precision tegen de recall van het gecombineerde systeem uit te wisselen.

Na het onderzoek naar de bruikbaarheid van natuurlijke taal analyse (NLP) als een aanvulling voor het normaliseren van concepten op basis van een woordenlijst beschrijven we in **hoofdstuk 5** dat met behulp van de NLP module the F-scores van MetaMap en Peregrine met meer dan 10% verbeterd werden voor de vaststelling van de woord grenzen en meer dan 15% voor het vinden van de juiste concept identificatie. We toonden de toegevoegde waarde van NLP voor het herkennen en normaliseren van ziekten met MetaMap en Peregrine aan. De NLP module is algemeen en kan in combinatie met elk willekeurig concept normalisatie systeem worden toegepast.

Hoofdstuk 6 laat resultaten zien over de bruikbaarheid van een kennisbank met NLP technieken voor het herkennen van biomedische relaties. We hebben laten zien dat de kennisbank gebruikt kon worden om ontkennende zinnen te herkennen en de prestatie van het extraheren van biomedische relaties te verbeteren. De prestatie in a een test omgeving die de werkelijkheid weerspiegelt is veel lager dan in een geoptimaliseerde test omgeving. De kennisbank is algemeen en kan gebruikt worden in combinatie met elk systeem om biomedische relaties te extraheren.

Op basis van de bevindingen in dit proefschrift concluderen we dat NLP, een ensemble beandering en de kennisbank gebruikt kunnen worden om de prestatie van normalisatie van concepten en het vinden van relaties te verbeteren.

ACKNOWLEDGEMENTS

Looking back over my life during the last four years, a lot of things have happened, especially while working as a PhD student, which was a very special experience for me, and one which I will never forget. It would never have been possible without the help and support of many people. “Thanks”, is such an inadequate word to express my gratitude and appreciation towards all of you.

First of all, I would like to thank my co-promoters and daily supervisors, **Dr. J.A. Kors** and **Dr. E.M. van Mulligen**, for the invaluable and kind guidance and advice. Thank you for giving me the great opportunity to work in the group of Biosemantics, to learn from you, and to have the best environment to develop my scientific career. Jan, you are the most intelligent, attentive, conscientious, patient, and knowledgeable supervisor I have ever met. I will never forget how you spent a lot of time correcting my papers, and give me so much kind advice. Your help was essential in finalizing this thesis. Your broad academic knowledge and rigorous attitude to science impressed me deeply and showed me the best example of how to be a real and good scientist. Erik, you are the most erudite, humorous, kind-hearted, imaginative, and well informed supervisor I have ever met. I am very impressed by your sharp and quick thinking. You always have many creative ideas for our research projects. You guided me through the field of text mining, and taught me how to work precisely, and how to write scientific papers. I will never forget our pleasant journey in US. I cannot express enough gratitude for how you both helped me in so many ways. I was your student and will always be.

I am grateful to all the members of my doctoral committee. Dear **Prof.dr. J. van der Lei**, thank you for giving me the opportunity to work as a PhD student in your department, and the time you spent on guiding me through my research work. **Prof.dr. U. Hahn,** I will never forget your patient answers to my questions about all the systems developed by your group. **Prof.dr. B. Mons, Prof.dr. G.W. Jenster**, **Dr. J.A. Hazelzet**, **Prof.dr. R. Vos**, and **Prof.dr. M.C.J.M. Sturkenboom** I appreciate all the time and energy you have invested in my path towards my promotion. Thanks for your participation.

I would like to thank all my colleagues at the Department of Medical Informatics for making these years so intellectually gratifying. My dear colleagues in the Biosemantics group: **Bharat**, **Zubair**, **Saber**, **Chinh, Kristina, and Martijn**. It is a great pleasure to work in the same office as you. Thanks for sharing so much happiness and sadness with me along the way here. **Peter**, **Herman**, **Macro**, **Erik**, **Reinout**, **Leon**, **Rogier**, **Tiago**, **David**, **Jose**, **Laura**, and **Ferran**, I will never forget our interesting discussions in each meeting. Special thanks to my colleagues: **Desiree**, thank you for your many kind emails and arrangements during my PhD study. **Tineke**, your solicitudes always make me happy

and warm. **Sander**, **Carmen**, **Mees, Preciosa, Hui,** and **Andreas,** thank you for your kind support.

I want to acknowledge and thank all collaborators in the projects I undertook during my PhD. **Dr. D. Rebholz-Schuhmann**, you gave me a deep impression of leading a project effectively. **Ekaterina Buyko**, I appreciate the time you invested in answering so many software questions. **Antonio Jos ́e Jimeno Yepes**, **Chen Li**, **Senay Kafkas**, **Ian Lewin, Elena Beisswanger, Peter Milward, David Milward**, and **Kerstin Hornbostel**, I enjoyed the time spent cooperating with all of you on the CALBC project.

I would like to express my appreciation to the people I met during my short time in US. **Dr. T. Rindflesch**, I admire your rich knowledge of text mining and thank you for answering all the questions related to MetaMap. **Dr. O. Uzuner**, thank you for sharing the i2b2 data. **Dr. S. Jonnalagadda**, I enjoyed the process of exchanging ideas of text mining with you. **Dr. Z. Lu**, thank you for your consolation when I was down.

Furthermore, I would like to thank my previous supervisors during my masters study. **Prof.dr. S.D. Swierstra**, thank you for giving me the opportunity to study as a masters student in your group, and the time you spent on mentoring my thesis. **Dr. L. Holenderski**, I appreciate your guidance of my internship at Philips.

I am so lucky to have had many very kind colleagues and friends during my previous work in the Netherlands. **Mehrzad**, I enjoyed the time we spent together at HP, and hope to have a chance to play billiards with you again. **Johnson**, I know you from your wife **Faridah**, who helped me a lot during my masters study, and when you came to the Netherlands, we became colleagues, what a small world! **Bas**, you were my first Dutch colleague and gave me a very nice impression. **Reinoud** and **Elise**, I am so happy to have received your magazine and read your warm letter every month.

I would like to thank my Chinese colleagues, friends, and families. 商鹏，侯珺，刘凡，温蓓，张凯，刘铭，感谢你们在我最需要帮助的时候给予我的那些支持．刘哲，宝月，亚迪，海波，童苗，赵甜娜，吕鹏，于晓，王甜甜，王永毅，王凯，兰天乐，路狄菲，陈嘉良，刘丽芳，冯静涵，谢谢你们这些年来的陪伴，原本在鹿特丹时我枯燥的生活因为有了你们而变得丰富多彩．王斌，吴雅洁，徐斌，钟丽萍，你们每次晚宴时热情的邀请总会让我感到温暖．

杨天虹，李雪晶，翟宇荣，仓蕾，魏巍，沈开开，Sami，陈嘉俊，赵莹莹，感谢你们曾给我的关怀和帮助。马骏，尹志，曲焕文，崔笑宇，朱津凤，沈丽，于栋，李黎，邬金，留学时那段艰苦并欢乐的时光我永远不会忘记。廖传道，陈传道，谢

晓雨，周歆丹，你们让我深深体会到了主的慈爱。

高大伟，顾海婴，常浩，陈爱华，阚非凡，王蜀燕，李宁，张恩源，青春终将逝去，但那些欢声笑语似乎就发生在昨天。周果宏教授，刘志成教授，感谢你们在我大学时对我的帮助和教导。韩兢，侯静，林琼，程工，刘长春，陈钢，庆幸大学时与你们相识相知。王俊，李蔚，孙建冬，感谢你们在我的第一份工作时曾给予我的指导。特别感谢孙伟，你是我最好的朋友，我在北京的生活因为有你而变得精彩纷呈。

春梅，你陪我度过了在荷兰的 7 年时光。每每想起我们一起度过的那些幸福的日子，我就会热泪盈眶。安息主怀的你，一定能在天堂看到 2 个孩子在这个多彩的世界快乐成长，2 位老人健康生活。

亲爱的爸爸妈妈，你们为了我付出了无限的爱和关怀。无论是我快乐还是悲伤，你们都会用你们的爱来呵护我，鼓励我，安慰我，支持我。我爱你们！

亲爱的大姑姑父，四佰四婶，林燕，林琳，康乐，你们给予过我许多帮助和关怀，让我只身在北京时，也永远能感受到家一样的温暖。

最后，我要感谢李舫，是你给了我新的生活。相信我们一定会用彼此善良，宽容，乐观，慈爱的心，建立一个幸福，温馨，美满，欢乐的家庭。看着礼贤，米乐，泽贤在我们身边无忧无虑，健康快乐的成长。

谨以此书献给我最爱的亲人们。

康宁　　Ning Kang

2013 年 7 月于荷兰　　the Netherlands, July, 2013

LIST OF PUBLICATIONS

----------2013----------

Ning Kang, Bharat Singh, Chinh Bui, Zubair Afzal, Erik M. van Mulligen, and Jan A. Kors. Knowledge-based extraction of adverse drug events from biomedical text. (Submitted)

----------2012----------

Ning Kang, Bharat Singh, Zubair Afzal, Erik M. van Mulligen, and Jan A. Kors. Using rule-based natural language processing to improve disease normalization in biomedical text. Journal of the American Medical Informatics Association, 2012, doi:10.1136/amiajnl-2012-001173

Ning Kang, Erik M. van Mulligen, and Jan A. Kors. Training text chunkers on a silver standard corpus: can silver replace gold?. BMC Bioinformatics, 2012; 13:17

Ning Kang, Zubair Afzal, Bharat Singh, Erik M. van Mulligen, and Jan A. Kors. Using an ensemble system to improve concept extraction from clinical records. Journal of Biomedical Informatics, 2012:45(3):423-8

----------2011----------

Ning Kang, Erik M. van Mulligen, and Jan A. Kors. Comparing and combining chunkers of biomedical text. Journal of Biomedical Informatics, 2011;44:354–60 [5]

Dietrich Rebholz-Schuhmann, Antonio Jimeno, Chen Li, Senay Kafkas, Ian Lewin, Erik van Mulligen, **Ning Kang**, Ekatarina Buyko, Elena Beisswanger, Kerstin Hornbostel, Peter Corbett, David Milward, Alexandre Kouznetsov, Rene Witte, Jonas B. Laurila, Christopher JO Baker, Chen-Ju Kuo, Simone Clematide, Fabio Rinaldi, Richárd Farkas, György Móra, Kazuo Hara, Laura Furlong, Michael Rautschka, Mariana Lara Neves, Alberto Pascual-Morante, Qi Wei, Nigel Collier, Faisal Mahbub Chowdhury, Alberto Lavelli, Rafael Berlanga, Roser Morante, Vincent Van Asch, Walter Daelemans, Jan Kors and Udo Hahn. Assessment of NER solutions against the first and second CALBC Silver Standard Corpus. Journal of Biomedical Semantics 2011, 2(Suppl 5):S11

----------2010----------

Ning Kang, Rogier Barendse, Zubair Afzal, Bharat Singh, Martijn J. Schuemie, Erik M. van Mulligen, and Jan A. Kors. A Concept Annotation System for Clinical Records. Semantic Web Applications and Tools for Life Sciences (SWAT4LS 2010), 2010

Ning Kang, Erik M. van Mulligen, and Jan A. Kors. Chunking of biomedical: a comparative study. AMIA (American Medical Informatics Association) 2010 Symposium Proceedings, 2010 (Distinguished paper award & Journal Eligible paper award)

Ning Kang, Rogier Barendse, Zubair Afzal, Bharat Singh, Martijn J. Schuemie, Erik M. van Mulligen, and Jan A. Kors.Erasmus MC Approaches to the i2b2 Challenge. Fourth i2b2/VA Shared-Task and Workshop Challenges in Natural Language Processing for Clinical Data, 2010 (Top 3 among 22 teams for medical concept annotation task)

Dietrich Rebholz-Schuhmann, Antonio Jos ́e Jimeno Yepes, Erik M. van Mulligen, **Ning Kang**, Jan Kors, Peter Milward, David Corbett, Ekaterina Buyko, Katrin Tomanek, Elena Beisswanger, and Udo Hahn. The CALBC Silver Standard Corpus for biomedical named entities: A study in harmonizing the contributions from four independent named entity taggers. In LREC 2010 – Proceedings of the 7th International Conference on Language Resources and Evaluation, 2010

Rebholz-Schuhmann D, Jimeno-Yepes A, van Mulligen E, **Kang N**, Kors J, Milward D, Corbett P, Buyko E, Beisswanger E, Hahn U, CALBC Silver Standard Corpus, Journal of bioinformatics and computational biology 2010, 8(1):163–79.

Schuemie MJ, **Kang N**, Hekkelman ML, Kors JA: GeneE: gene and protein query expansion with disambiguation. Journal of Bioinformatics, 2010, 26(1):147-48.

ABOUT THE AUTHOR

Ning Kang was born on 21th, Aug, 1977 in Yinchuan, China. He obtained his Bachelor degree in Biomedical engineering from Capital University of Medical Sciences in 2000, and then he worked as a software developer and led a team at Beijing Genomics Institute, Chinese Academy of Science.

In Sep, 2003, he moved to the Netherlands to pursue a Master of Science program in software technology at Department of Information and Computing Sciences, Utrecht University in Utrecht. He obtained his master's degree in June 2005. After that, he worked as a software developer in several companies, such as Philips, HP, etc.

In Dec, 2008 he started to work as a scientific researcher and a PhD candidate at the Biosemantics group, the department of Medical Informatics, Erasmus University Medical Center, Rotterdam. His research was aimed at using natural language processing to improve biomedical concept normalization and relation mining. During his PhD, he was involved in two European Union projects: the CALBC project and the EUADR project. The results obtained from his PhD research were presented in international conferences, and have been submitted or published in peer-reviewed scientific journals. After his PhD, Ning Kang would like to go to industrial companies for his career development.