Negation Detection in Automated Medical Applications

A Survey

Stefan Gindl

Vienna University of Technology
Institute of Software Technology & Interactive Systems

Asgaard-TR-2006-1

October 2006

Author: Stefan Gindl

Contact: Vienna University of Technology
Institute of Software Technology & Interactive Systems

Favoritenstrasse 9-11/188
A-1040 Vienna
Austria, Europe

Abstract

In medical reports patient data is mostly stored in narrative language, which is the spoken or written input from the responsible physicians. To allow computer processing it is necessary to translate this natural language into a format a computer is able to understand and deal with. In this wide area of translation efforts the so called "Negation Detection", i.e. the detection of negated phrases, which do not deliver supplementary information, has an essential importance, since for automatic systems the negation detection can work like a filter system, filtering irrelevant information from important knowledge pieces. It is necessary to decide whether a given medical phenomenon must be taken into account or if it can be ignored, because it is not present at the patient of interest.

This survey examines already existing negation detection procedures and compares their accuracy. Furthermore relevant background information is provided in order to get familiar with the vocabulary used in this special field.

1 Introduction

A large part of medical information is only available in natural language, which makes an algorithmic processing with computers a very difficult task. So to say, this information is "locked up" in files and databases in a format not suitable for automated processing [Hripcsak et al., 1995]. Great efforts have been made to transcribe this information into a format a computer can work with. Despite of this the processing of negated terms is still an open topic. Negation detection demands a lot of knowledge about language itself to correctly identify negated words or terms in free text. Endeavour on this topic may help to develop computer assisted treatment strategies to improve patient care.

Negation detection contributes to an information reduction when searching in databases. Search results can be minimized to the entities searched for, otherwise even negated entities would be returned in queries. This report contains already existing algorithms for the task of negation detection. It shows, how possible approaches can look like. Further works on this topic will apply negation detection procedures in the field of clinical guidelines.

2 Motivation

Negation is an important part of inter-human communication. It can be used to invert concepts and to show refusal of opinions. Sometimes persons in life-threatening situations use the simple word "no" to express their helplessness. In the developement of a person's psyche and character a "no" implies that this person has developed his or her own will and is able to express this. A person who never says "no" will never be asked to give his or her view on a problem, since the answer is already known. The concept of negation is therefore a universal concept in the most languages if not in all languages.

The existence of negations in language has been philosophised by man since more than 2000 years. Even Aristotle played with this peculiarities and tried to find a classification of the different types of negation in natural language and found four classes of negation. He came to the result that negation can be splitted into four types which he named as correlation (e.g., double vs. half), contrariety (e.g., good vs. bad), privation (e.g., blind vs. sighted) and contradiction (e.g., he sits vs. he does not sit) [Horn, 1989].

The right understanding of a negation demands a competent knowledge of the used language, since even one word can completely change the sense of the statement. The statement can then be inverted, weakened or amplified. The following simple example by Horn [Horn, 1989] shows this effects in negated sentences (as cited in [Mutalik et al., 2001]):

1. I'm not tired.
2. I'm not a bit tired. (which equals "I'm not at all tired.")
3. I'm not a little tired. (which equals "I'm quite tired.")

Even this example shows the complexity of a negation in natural spoken or written language. In the first sentence the negation is very clear and one cannot be confused in its meaning. The following two sentences generate subtle problems, because one of them states the complete absence of tiredness ("I'm not a bit tired."), whereas the other sentence describes a condition of exhaustion ("I'm not a little tired."). A german reader who is not very familiar to the english language can get problems with phrases when they are written down. The second and third sentence seem to be equal and the lack of mimic art or a tonal amplification leaves the reader confused. An english native speaker will in contrast to a non-native speaker have no problems with the right understanding of the semantic of this sentences.

A search for negations in natural language is therefore a difficult task, but the search for them in a medical scope can be easier. Medical language is much more restricted than narrative speech, a physician will not use stilistic elements like double negation extensively to write reports or patients histories. Thanks to this fact a negation detection in a medical scope is supposed to be less difficult because of this restrictions.

3 Negation Detection in Medical Scope

Although the subject of negation has already been discussed by the ancient Greek there have been little achievements of an automated negation understanding and processing. Specifically in the field of medicine there are only few studies concerning this topic. The two most important studies relating to an automated negation detection in medical documents have been carried out by the University of Pittsburgh and the Yale University School of Medicine. Furthermore, the University of Sofia in Bulgaria has performed a study for the identifying of negations for the Bulgarian language, which is by now a merely researched field in this language.

In order to fully understand the explanation of used negation detecting techniques, the knowledge of some important medical technologies will be necessary. The following chapter gives an insight into used technologies, which are relevant for the comprehension of the performed studies.

3.1 Background

For the automated processing of medical texts systematized terminology collections are applied to index biomedical concepts. A brief introduction into the used terminologies is given in the following section.

3.1.1 SNOMED – Systematized Nomenclature of Medicine – Cinical Terms®

SNOMED® represents a systematized medical terminology. It is a "comprehensive and precise clinical reference terminology that provides unsurpassed clinical content and expressivity for clinical documentation and reporting." [SNOMED].

SNOMED® was developed by the College of American Pathologists (CAP). The CAP had by now worked with the ancestor of SNOMED®, SNOP (Systematized Nomenclature of Pathology). In the year 1974 SNOP was extended with the help of Dr. Roger Coté in order to contain vocabulary beyond the scope of the pathologic special field and SNOMED® was born.

In 1977 SNOMED® was turned from its printed representation into an electronic tool, created to facilitate the use for physicians, researchers, and healthcare professionals.

With SNOMED-RT (Reference Terminology) a new generation of SNOMED® was introduced in January 1999. SNOMED-RT was created to better fulfill the needs in the fields of health states, disease states, pathophysiology, treatments and outcome.

In the same year in March a fusion of SNOMED-RT and the United Kingdom National Health Service (NHS) Clinical Terms Version 3 (former known as the Read Codes) was performed. SNOMED-CT® (Clinical Terms), the currently most comprehensive vocabulary for medical terms, evolved from this merging.

In SNOMED-CT® medical information like diseases is defined by concepts. These concepts are refined by several hierarchies or axes, for example, "disease", "finding site" or "procedure". In order to identify concepts they are provided with a concept ID, which is a simple numerical value. Existing relations between concepts are stored too.

The information provided for a given concept is composed of a unique numerical code, a unique name (the so called "Fully Specified Name") and further descriptions like synonyms. Moreover, information about the concepts location in the hierarchy and relations between other concepts are recorded.

Important facts of SNOMED-CT® [SNOMED]:

- Its included Semantic Net contains over 300,000 medical concepts and the relationships among them
- More than 7 million relationships are available
- The concepts cover several fields like diagnosis, drug definition, findings, procedures, anatomy etc.
- Multiple axes and hierarchies
- Includes three main hierarchies (finding, disease, procedure) and 15 supporting hierarchies
- A description logic is used for concept representation

The picture on the following page shows a graphical representation of the underlying architecture:

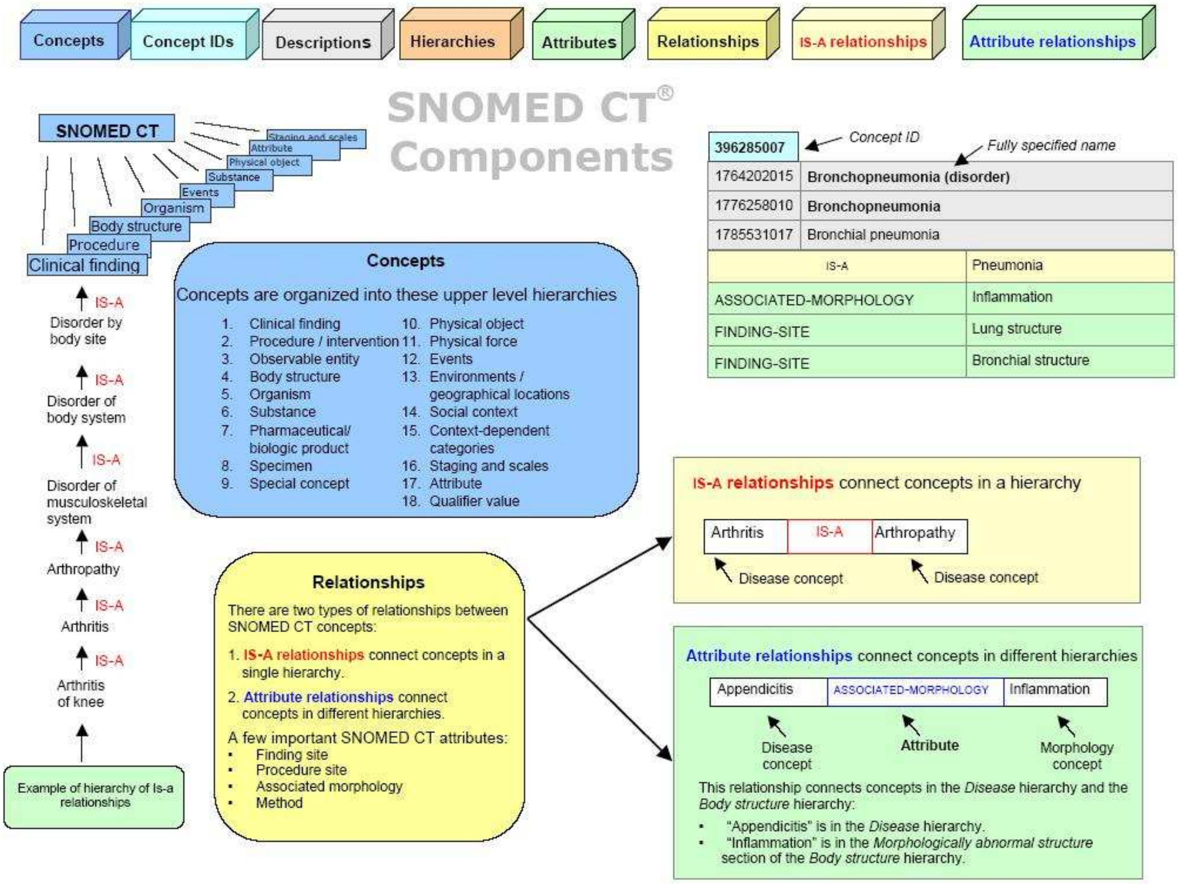

Figure 1 Illustration of the SNOMED-CT® components [SNOMED]

3.1.2 International Classification of Diseases and Related Health Problems (ICD)

The ICD is released by the World Health Organization (WHO) and emerged from a catalogue of causes of death which was introduced by the French physician Jaques Bertillon in the year 1893. The American Public Health Association (APHA) recommended the adoption of this catalogue for the improvement of medical treatment in the USA, Canada and Mexico. In order to keep the register up to date the APHA recommended a revision every ten years, which was accomplished by the Mixed Commissions, a group consisting of the International Statistical Institute and the Health Organization of the League of Nations until 1948.

At that time the WHO took was committed the function of revising the ICD, which happened one year later with the 6th revision.

In 1977 the WHO published the 9th revision of the ICD (ICD-9)consisting of three volumes, the first two of them containing diagnosis codes, the third covering procedure codes.

From the ICD-9 emerged the ICD-9-CM (Clinical Modifications), which is used for the calculation of hospital treatment costs in order to facilitate the account with concerned insurance companies.

The currently used versions are the 1999 published 10th revision (ICD-10) and the ICD-9-CM.

It provides alphanumerical codes for the identification of diseases or abnormal findings in humans. The ICD contains 22 different fields, each having a unique code. Diseases or findings within a special field are then identified by this field's code and an additional numeric code specific for the finding. For example, the occurence of neoplasms is coded with the letters C and D, malign neoplasm have the range from C00 to C97 applied. C30 to C39 stands for malign neoplasms of the respiratory and intrathoracic organs. The value C34 is used for malign neoplasms of bronchus and lung. To conclude, the diagnosis "lung cancer in the upper lobe" would be codified by the alphanumeric value C34.1.

3.1.3 Unified Medical Language System (UMLS)

The UMLS is a controlled collection of vocabularies with its internal structure designed to map between them. The system was initially designed in 1986 by the National Library of Medicine, USA.

As UMLS is a collection of various vocabularies, it allows conversion of medical terms from one vocabulary into another. This fact is described in [Humphreys et al., 1993] as follows:

> *"The UMLS approach assumes continuing diversity in the formats and vocabularies of different information sources and in the language employed by different elements of the biomedical community. It is not an attempt to build a single standard biomedical vocabulary."*

The structure of the UMLS contains three main knowledge sources:

1. **The Metathesaurus®:** It represents the base of the UMLS, containing 5 million concept names identifying more then 1 million biomedical concepts. These names come from over 100 controlled medical vocabularies such as SNOMED CT or the ICD-9-CM. Metathesaurus is intended to enable the work with the different vocabularies and provides facilities for the information exchange between different medical databases. The scope of it is as wide as the underlying used source vocabularies. A software program can receive information from Metathesaurus, users can pose inquiries and they can obtain help for a conversion of a vocabulary to the existing uniform vocabularies.
2. **The Semantic Network:** Semantic networks are generally the representation of a given knowledge base [McCray, 1989]. The network contains nodes to

represent semantic concepts and links for the representation of relations between the concepts. For example, there are nodes for the representation of organisms, biologic function or chemicals. For UMLS the network is established to facilitate the use of the system. It categories the concepts in the UMLS Metathesaurus and offers relations between concepts.

3. **The SPECIALIST Lexicon:** It lexicon provides the lexical information for the SPECIALIST Natural Language Processing System. It can be used as a general English lexicon expanded by a medical vocabulary. The tools were developed to deal with the high variability of words in the natural language. The identification of inflected words (i.e., treat: treats – treated – treating) can be accomplished with these.

The UMLS is an ontology – a system describing instances, concepts, attributes, and relations. But other than common ontologies it has a two-level structure:

1. the Metathesaurus, a unified collection of many different medical terminologies, a compilation of terms, concepts, relationships, and associated information;
2. the Semantic Network containing semantic types (one may think of semantic types as high-level concepts, i.e., broad categories), organized in a hierarchy of IS-A links.

Lee and Geller [Lee & Geller, 2006] call such a structure a Terminological Knowledge Base (TKB):

> *"... any structure that consists of (1) a semantic network of semantic types; (2) a thesaurus of concepts; and (3) assignments of every concept to at least one semantic type"*

3.2 Evaluation

The accuracy of an implemented algorithm needs to be analysed with approved statistical parameters, in order to allow a comparison with different, already existing algorithms. The value of a statistical parameter is also a statement about the efficiency and precision of an algorithm. High values signify a high quality of the used procedure. Although this seems to be a simple decision strategy for the usage of an algorithm, one has to be careful with the outcome of statistical analyses. Ignorance or the intentional abuse of these statistical parameters can very easily lead to a wrong decision for the choice of an algorithm. In a strict medical context adulterations are used to enable the launching of a new medication which has not been tested carefully enough before.

For the evaluation mainly four statistical parameters are used, which are listed below.

3.2.1 Sensitivity

The sensitivity is a statistical parameter and very often used in medicine. It provides information about the ability of a test to identify individuals having a special disease (an individual having a given disease is called "positive", e.g., "HIV-positive", meaning that the person carries the HI-virus). In other words, the sensitivity evaluates the probability of detecting a statistical unit to be positive, if this unit is really positive. The better the test, the more of the individuals having a given disease are identified. In the context of negation detection a high sensitivity means, that a large percentage of occuring negations is recognised by the algorithm.

The formula for this statistical parameter is relatively simple. As sensitivity reflects the accuracy of finding positive test results (i.e. the presence of a disease, the presence of a negation), the number of the found true positive statistical units is divided by all existing

positive units in a basic population (when talking of sensitivity in biological circumstances) or in an analysed text (in the case of negation detection):

$$Sensitivity = \frac{\sum TruePositives}{\sum TruePositives + \sum FalseNegatives}$$

In the context of negation detection, the attributes "positive" and "negative" must not be confused. In this case, a concept is true positive, when it is negated.

3.2.2 Specificity

This is the counterpart of the sensitivity, measuring the accuracy of finding true negative individuals in a medical context or finding not negated concepts in negation detection. The specificity is the probability of a test resulting negative, if the examined statistical unit is negative in actual fact. It is calculated by dividing the number of true negative statistical units by all existing negative units in the basic population:

$$Specificity = \frac{\sum TrueNegatives}{\sum TrueNegatives + \sum FalsePositives}$$

The existence of an HI-virus is tested with the so called ELISA test (Enzyme-linked Immunosorbent Assay). This test comes with both sensitivity and specifity about 99.5 %, other sources speak of 99.9 %. Such high values for this two statistical parameters indicate, that the test is of an excellent quality.

3.2.3 Positive Predictive Value (PPV)

The positive predictive value provides information about the probability of having a given disease, when a test result is positive. A medical test should have a high PPV, otherwise a patient or his examining physician could never be sure, if the searched disease is present in fact.

For the calculation of this value a simple formula can be used. It is the number of true positive results divided by all positive result (false positives are included):

$$PPV = \frac{\sum TruePositives}{\sum TruePositives + \sum FalsePositives}$$

The positive predictive value for the ELISA test is 91 %. This means, that of 100 persons with a positive test result, in fact nine of them do not carry the HI-virus, so they are mistakenly considered to be positive.

3.2.4 Negative Predictive Value (NPV)

This value is the counterpart of the positive predictive value. It gives information on the probability of the lack of a given feature, if a test resul is negative.

The formula is similar to the formula of the PPV:

$$NPV = \frac{\sum TrueNegatives}{\sum TrueNegatives + \sum FalseNegatives}$$

The ELISA test has a negative predictive value of 99.999 %.

3.2.5 Correlation between Statistical Parameters

Sensitivity and specifity are strongly related: the knowledge of only one of these two statistical parameters is not a very meaningful measurement of the accuracy of a testing procedure. For example, the sensitivity can easily be pushed to 100 % by regarding each and every test result as positive. Only when combined with the specificity (which would in this case normally be surprisingly low) the sensitivity becomes a suitable measurement.

Both PPV and NPV are dependent on the so called prevalence, which is either the total number of occuring positive objects in a statistical universe or the ratio of the number of occuring positive objects at a specific time and the total number of objects in that universe at that time.When the prevalence is high it is easy to gain a good PPV, vice versa it is easy to obtain a good NPV if the prevalence is low.

3.3 A Simple Algorithm for Identifying Negated Findings and Diseases in Discharge Summaries – NegEx

This study carried out by the working group around Wendy Chapman proceeded on the assumption that a relatively simple detection algorithm can produce usable results [Chapman et al., 2001]. They designed an algorithm called NegEx and applied it to discharge summaries to find out, if findings and diseases were negated by the physicians. The found results were compared to the results of a former used baseline algorithm.

3.3.1 Preprocessing of Selected Documents

The input of both algorithms was a preprocessed text sequence, the output of the both is the decision if a phrase is negated in a sentence.

Preprocessing was done in three steps:

1. **Only one sentence per line:** this leads to an information loss when more then one sentence is negated by one and the same negation.
2. **Removal of punctuation:** syntactic information represented by punctuation was extracted.
3. **Indexing of findings and diseases:** in each sentence medical terms were replaced by their UMLS-representation. Because of the large extent of UMLS the indexed terms were delimited to the scope of "Finding", "Disease or Syndrome" and "Mental or Behavioral Dysfunction".

3.3.2 Description of the Algorithms

For the realization of the study two algorithms were used. A simple baseline algorithm was compared with the more sophisticated NegEx algorithm.

The baseline algorithm was an algorithm used by the so called IPS system to identify patient subgroups and was created by the University of Pittsburgh. It searches for six negation phrases and marks all UMLS terms from the negation phrase to the end of the sentence.

NegEx was designed on the basis of this simple algorithm but was extended by 29 negation phrases to a total amount of 35 detectable negations. The phrases were searched by manually reading 1050 reports. In comparison with the baseline algorithm NegEx uses a more complex way of extraction. While the baseline algorithm negates everything from the occurence of the negation until the end of the sentence, NegEx differs between two basic negation types, where one group comprises pseudo-negations which only change the meaning of a term (e.g., "gram-negative") or which are double negatives (e.g., "not ruled out"). The second group covers the real negations and these are extracted by NegEx. In this group a differentiation between the position of the negation phrase in relation to the negated phrase is carried out. In the first type the negationen phrase is positioned before the UMLS term:

<negation phrase> * <UMLS term>

In the second type the negation phrase is positioned after the UMLS term:

<UMLS term> * <negation phrase>

The asterisk stands in this context for up to five tokens which can be either words or UMLS terms. With this method negated UMLS terms can be detected even if they do not stand next to the negation phrase. Moreover, a more distinct extraction than with the baseline algorithm is possible.

3.3.3 Performance Analysis

A performance analysis was carried out which came to the result, that both algorithms marked negation with practicable precision. The baseline algorithm had its strength in sensitivity and negative predictive value, NegEx performed better in specifity and negative predictive value. Table 1 shows the detailled results of each algorithm:

	Baseline algorithm (%)	NegEx (%)
Sensitivity	88.27	77.84
Specificity	85.27	94.51
PPV	68.42	84.49
NPV	93.01	91.73

Table 1 Comparison of the performance of the baseline algorithm and NegEx.

These results follow from the different designs of the two algorithms. The baseline algorithm, extracting everything from a found negation to the end of the sentence were the negation was found, results in a high sensitivity, but this also leads to an extraction of words which are not really negated. This fact lowers the positive predictive value.

NegEx with its higher precision results in a very high specificity, however, the disadvantage of its design is, that words, which are actually negated, are not extracted because of their spatial distance to the negation phrase. This loss in sensitivity could be removed by a widening of the scope, within words are extracted, yet would this lead to a loss of specificity.

3.3.4 Applications of NegEx

In [Meystre & Haug, 2005] an implementation of NegEx serves for the identification of negations too. In this work a system was created to allow automated processing of problem lists in medical reports. As they are important components of medical records they need to be as accurate and up to date as possible. In reality, the hand written problem lists are often incomplete and therefore there is a need for simple processing routines, so that the problem lists gain in accuracy.

In order to identify medical concepts a mapping of the free text to the UMLS by MetaMap [Aronson, 2001], an element of the SPECIALIST lexicon, is carried out. As MetaMap does not support negation detection a following implementation of a detection algorithm was required and the authors chose the NegEx algorithm. The authors have not carried out an efficiency analysis of their implementation of NegEx but relied on the values given in the study by Chapman and colleagues [Chapman et al., 2001].

3.4 Use of Methods for the Parsing Process of Formal Computer Languages – NegFinder

In [Mutalik et al., 2001] a different approach to negation detection was chosen. Here, a lexical scanner and a parser were applied, which are methods used to develop tools for the processing of (formal) computer languages. This procedure resulted in considerable values of sensitivity and specifity.

To achieve these results, a system of several components was used like a pipeline. Each component represents a step to the final result, which is the accentuation of negated phrases in the processed text.

3.4.1 The Components Pipeline

The interesting text file undergoes four steps of operation. The result is a modified version of the original with all negations marked up, so the found negations are easier to be verified by a human reader.

The pipeline consists of the following components:

1. **Concept finding:** This step is carried out to find UMLS concepts in the document. If such a concept is found, some information about it is stored in the output. This information consists of the phrase that matches the concept, its length, the position in the text, and the UMLS concept.
2. **Input transformation:** In the second step the information of the first step is combined with the original document. Each phrase that represents a UMLS concept is replaced by its UMLS concept ID.

3. **Lexing/Parsing:** The output of the former steps is passed to a lexical scanner (lexer). This lexical scanner contains knowledge of a large amount of possible negation signals. Moreover, the lexer is able to classify found negation. It is able to decide, if a given negation signal normally precedes or succeeds its negated phrase, or if it generally negates more then one concept. After this, the parser receives the classified sentences as single tokens. It applies its grammar rules to connect the negation signals with the phrases they negate. When this step is finished, a document is available that contains the information about found negations.
4. **Verification:** The original document is marked up using different colours for different types of negation, the negation signal themselves are marked with a special colour. Negation signals get the color blue, a negated concept is marked in red, combined concepts can be seen in orange and text phrases not representing medical concepts are highlighted in magenta.

3.4.2 Concept Finding

The algorithm used for the identification of concepts tries to match phrases up to five words. If this is not possible, it splits the phrases into subsections and performs another matching attempt.

For the purpose of concept matching the authors use the IBM Intelligent Miner for Text, which allows linguistic analysis and is able to identify special terms like names or abbreviations[1]. The medical vocabulary used comes from the UMLS Metathesaurus, which is stored on a MS SQL Server.

3.4.3 Input Transformation

In this processing step recognised medical concepts are replaced by their UMLS-IDs. The output of this step is a file containing numbers and textual sequences, very difficult to read for a human except he/she knows the concepts associated with the ID. This output file is then passed to the lexical scanner.

3.4.4 The Lexical Scanner

The lexer is used for the identification of occuring negation signals. Moreover, it is able to determine the scope of the affected terms. For the purpose of finding negation signals more than 60 words or patterns are recorded, which enable the lexer to mark up negations.

Thanks to its implementation the lexer is able to answer following questions considering a negation:

[1] http://www.searchtools.com/index.html, accessed 18.09.06

1. Is the negation signal a preceding or succeeding one? The word "no" normally negates terms following it, whereas in the case of "not present" the negated terms stands before the negation signal.
2. Is only one concept or more than one concept negated? "No", for example, is able to negate more than one following term.
3. The lexer is also able to decide in negations of more than one concept, if the negation signal normally precedes or succeeds the negated phrases. For this purpose, it uses a difference between the conjunction operators "or" and "and". "Or" normally indicates a preceding negation signal ("no murmurs, rubs or gallops"), whereas "and" signifies a succeeding signal ("murmurs, rubs and gallops are absent").

Above this, the lexer is also able to tell the parser about the scope of a negation. For this purpose it uses a vocabulary of terminating terms. This terms indicate, that the end of the negated phrases is reached and that a new part of the sentence begins. For example, many prepositions terminate a negated sequence as a general rule. In the sentence "no complications during surgery" only the concept of complications is negated. Another example for terminating terms are relative pronouns, such as "which" or "that. In the note "There was absence of temperature sense which supports spinothalamic involvement" the negation of the concept "temperature sense" just indicates the occurence of "spinothalamic involvement".

When the lexer has finished its decision finding process, it passes the indexed phrases as single tokens to the parser.

3.4.5 The Parser

The parser used in this study works on the tokens it receives from the lexical scanner. Information about negation terminators or the scope in which a negation signal negates is provided by the lexical scanner. Because there is no need for an exhaustive analysis of the syntax, the grammar used by the parser is much simpler than normally used for the processing of natural language.

The actual job of the parser is to finally decide about the characteristics of a negation. It decides, where a negation starts or ends, how many words are concerned by the negation signal and if the negated phrases precedes or succeeds the negation operator. The accomplishment of this job is strongly dependent on the information it receives from the lexer. If there is no information about negation terminators, it considers the end of the negation three words after the negation operator. The number of exactly three words for the scope of a negation was given by an analysis of the sensitivity and specifity reached with a scope of more, or less, than three words.

3.4.6 Evaluation

The evaluation of the algorithm was splitted into two parts. In one part, the designers of NegFinder validated marked up documents, in the other part an independent observer was engaged to do this.

In the first of these two mentioned parts, the NegFinder was applied to a test set of 60 medical documents. The output of this procedure was then validated by the three authors. For this, they read through the documents, in which the negations were already marked up. They counted as well negations the NegFinder had missed (false-negatives) as negations the algorithms had marked by mistake (false-positives). Moreover, information about incorrectly marked negations was collected.

For the other part of the evaluation an independent human observer read through ten medical documents and tried to allocate negations exhaustively. After this, the NegFinder was applied to this set and the results were compared.

To sum it up it can be said, that in the first part already marked documents were controlled by human observers, in the second part the human observer was controlled by Negfinder. This lead to following values for the sensitivity and specifity of the algorithm:

	Human observers read through documents marked up with NegFinder (%)	Human observer and NegFinder worked independently (%)
Sensitivity	95.3	95.7
Specifity	97.7	91.8

Table 2 Differences in sensitivity and specifity in the two different examining methods.

A slight difference in the specifity of the two evaluation procedures can be noticed. This difference, statistically not significant, shows, that reading through already marked up documents creates a bias distorting the true result. On the other hand the difference is so small, that the two evaluation procedures can be treated equivalent.

3.4.7 Applications of NegFinder

The NegFinder algorithm was applied in the work about the knowledge-based MediClass system, which supports the processing of both free text and coded data in electronic medical records (EMR) [Hazlehurst et al., 2005]. MediClass (for "medical classifier") can operate on any EMR which can use the Clinical Document Architecture (CDA) for data representation. CDA is a document structure based on XML to store and transfer clinical documents, which was developed by the Health Layer Seven health care standards organization (the seven stands for the seventh layer of the ISO/OSI Basic Reference Model).

In order to recognize the change of the semantic meaning of concepts in EMR modifiers are searched for which can influence the meaning of a concept regarding it's severity, classification and negation. For negation detection, a small set of negation terms is used for the processing with NegFinder's lexer and parser, since the authors believe that even a little number of detected negation signals lead to reasonable results.

Unfortunately, the authors do not provide statistical values for the reached accuracy of their implementation of NegFinder.

3.5 A Controlled Trial of Automated Classification of Negation from Clinical Notes

The aim of this work was to draw a comparison between the precision of an automated negation detection procedure and a human observer [Elkin et al., 2005]. In the study 41 clinical documents were parsed with the Mayo Vocabulary Server Parsing Engine, which uses the SNOMED CT® terminology to discover medical concepts.

For the analysis of this procedure an independent human observer was employed, who read through the resulting records. Incorrect negations were detected to find out possibilities of improvement of the algorithm.

3.5.1 Method

A set of 41 medical reports was used to evaluate the accuracy of an automated negation detection system. These reports were provided to the Mayo Vocabulary Server as ASCII files containing data in free text. With the Mayo Health Record Parser these text files were splitted into subsections typical for the reports, such as "History", "Physical Examination", "Diagnostic Testing".

This output was then preprocessed to differ between textual fragments and operators like "And", "Or", "Maybe" and so on. Medical concepts in the text were then indexed using SNOMED CT®. After this procedure, the actual negation assignment takes place. For this purpose, a so called "automated negation assignment grammar" is used. This grammar splits free text phrases into four semantic components:

1. Kernel concepts: these contain the semantic basis of the phrase
2. Modifiers: Modifiers change the meaning of kernel concepts in a medical way, e.g. "severity" expresses the level or stage of a disease.
3. Qualifiers: Here a qualifier changes the meaning of a term regarding its temporal or administrative content like "recurrent".
4. Negation qualifiers: these are signals for the occurence of a negation.

Moreover, a collection of negation stopping phrases is used. These phrases signal the end of a negation's scope. For example, in the phrase "The patient denied a history of previous cardiac disease other than palpitations which he experienced while giving representation resultingin syncope", other then limits the scope of the negation signal "denied" to "a history of previous cardiac disease".

3.5.2 Statistical Analysis

To evaluate the quality of this automated negation detection system, a human reviewer had to read through the reports. By the evaluation of the human reader the indexing of SNOMED-CT® turned out to be error-prone. Out of actually 2028 negative concepts, SNOMED-CT® had missed 205 concepts, so these concepts were not passed to the negation assignment module.

All in all, the system provides very good values in accuracy. Table 3 shows the outcome of the statistical analysis:

	Value of the statistical parameter (%)
Sensitivity	97.2
Specificity	98.8
PPV	91.2
NPV	99.6

Table 3 The procedure proposed by [Elkin et al., 2005] reaches high values in all statistical parameter.

3.6 Some Aspects of Negation Processing in Electronic Health Records

This study comes from the university of Sofia and deals with the problem of processing a language different from English, namely Bulgarian [Boytcheva et al., 2005]. For the English language the strongest effort in natural language processing and negation detection has been made, whereas other languages have been treated more like stepchildren. The University of Sofia worked on a system called MEHR (Maintaining Electronically Health Records) for the indexin of medical terms and facts like negations in electronic health records, written in Bulgarian.

In the process of indexing medical terms and tagging negations, an electronic health record (EHR) undergoes five separate steps which are worked off in a serial pipeline. The output of this procedure is a patient's chronicle containing position and scope of found negations and the medical information of this patient.

3.6.1 System Architecture

As mentioned earlier, the system contains of five separate modules, each adding more information to the processed file. Before an EHR is passed to MEHR, it is splitted in the eleven topics typical for bulgarian health records, which are: personal data, anamnesis, status, examinations, consultations, debate, treatment, treatment results, recommendations, working abilities, diagnosis. After this procedure MEHR begins with the analysis of the received file in five steps:

1. **Annotation and chunking:** The A&C module begins with tagging syntactical structures in the received file. It is programmed in Perl and works like a part-of-speech tagger. This module contains a lexicon with about 50 000 words and contains medical vocabulary and terms typical for EHRs.
2. **Post processing module:** The post processing module uses a lexicon with verbs specific for medicine. It also works with verb frames and grammar rules to identify verbal chunks.
3. **Negation treatment module:** In this step the negation detection begins and found negations are marked. A detailled description of the working method of this module is given in the following subsection.
4. **Extractor:** This module uses a database with medicine-specific terms to describe the patient's symptoms and diagnosis.
5. **Filling scenario templates module:** To obtain a correct patient's chronicle this module fills in obligatory and opitional fields of a form mask. It ultimately creates the patient's chronicle.

3.6.2 Details of Negation Identification

The authors of the study differ between two kinds of negation, which are common in the bulgarian language: general negations, where a whole statement is negated, and partial negations, in which only one or more parts of a statement are negated.

A general negation in Bulgarian is often indicated by the word "не" (not). With this negation signal an action signalled by a verb is negated, and general negation means, that a verb is negated. Added negation signals like "ни" (nor) or "нито" (neither) are used to repeat the negation or to increase its strength. However, the use of these two negation signals turns the general negation into a partial type. For example, in the sentence "He does not drink." the whole concept of drinking is negated, whereas in the sentence "He does not drink neither wine nor whiskey" the two drinks are negated seperately. Another signal for a negation a verb action, i.e. a general negation, is in Bulgarian the preposition "без" (without) followed by the concerned verb succeeded by the particle "да". MEHR treats both variants in the same way.

In a partial negation a phrase or even an attribute of a verb, but not the verb itself is negated. The preposition "без" is also used in this context, but this time there is no succeeding "да" provided. A single attribute is negated, e.g. "дишане без хрипове" (breathing without crepitating). Another case are the so called inherent negatives. These are verbs having a negative meaning but are presented in a positive form. Examples for them are:

- "няма" (there is not, not exist) as the negation of "нма" (there is, exist). The positive verb differs from the negating verb just by the letter "я", so they are different words, whereas the English only uses "not" in this case to negate an existence.
- "липсвам" (absent) as the negation of "присъствам" (be available).
- "отричам" (deny) as the negation of "признавам" (confess).

The authors also direct their attention to sentences with a more complex structure. Combined elements which then are negated by only one negation signal need a special treatment, i.e. "без хирзутизъм аменорея" (without hirsurtism, amenorea). The same applies for sentences, in which more verbs are negated. In Bulgarian for this purpose a repetition of "не" (not) is used, so that the number of "не" and the number of the negated verbs are the same.

3.6.3 Evaluation

With the proposed procedure an identification rate of negations of about 57 % could be reached. 15 % of the negations could not be identified and 28 % were registered incorrectly. An incorrect match means, that the range of the by a negation signal affected concepts was misunderstood.

3.6.4 Conclusio

As one can see from the description of the five modules, MEHR is more than only a negation recognition system. Although one module is only responsible for the treatment of negations in an EHR, the main goal of the system is more ambitious. The aim is to create a system that automatically is able to produce completely filled chronicles out of electronic health records. Negation detection is just the first step towards this ultimate goal.

3.7 Automatic Mapping Clinical Notes to Medical Terminologies

The study of [Patrick et al, 2006] was carried out to measure the accuracy of medical concept matching in free text medical reports. Additionally, the used algorithm was extended in order to identify negated medical concepts too. The following description of this study focuses primarily on the aspects of negation processing.

The procedure of negation assignment consists of four correlative steps. The input to the systems were clinical notes, which in practice are stored in written free text.

3.7.1 Method

As mentioned before, the processing routine contains four steps, beginning with the preprocessing of the input record:

1. **Preprocessing the input:** The input record undergoes a normalisation process.
2. **Concept matching with SNOMED-CT®:** Medical concepts are replaced by their concept ID.
3. **Detection of negations:** The identification process is explained in detail in subsection 3.7.3.
4. **Qualification and term composition:** This step is used to assign medical concepts that are combined with qualifiers, which in these circumstances are terms that contain extra information about a concept but which do not alter its sense.

3.7.2 Preprocessing

This step has two main tasks to achieve, which are interesting since they help to simplify the input records for a later negation assignment. At first, the input is converted into a basic format. This means, that the file is split into its seperate sentences, all letters are converted to lower case, and differently written phrases are unified (haemocyte vs. hemocyte). The authors of the study call this the normalisation. After this, the record is POS tagged by the GENIA tagger. This part-of-speech tagger is especially designed for the usage in a biomedical scope.

Another important point in this phase of processing is the identification of so called "administrative entities". These entities include information about the dosages of medication, duration of an inhabitation in hospital etc. Quantitative units like "kilogram" are identified as well.

3.7.3 Negation Detection

In this work the same simple algorithm is used as in [Chapman et al., 2001]. For the detection of the negation signal, the same signal list as used in this study is employed, but unlike [Chapman et al., 2001] here a general distinction between two superior types of negation is carried out. On the one hand there are so called pre-coordinated negated phrases, on the other hand there are phrases explicitly negated by a negation signal. Pre-coordinated are negated phrases which are already stored as concepts in SNOMED-CT®, like "no headache", whereas the other type is the classic negation, where a negation term or phrase negates a concept. This second type of negations is treated by an algorithm similar to NegEX, differentiating between phrases that are preceded or succeeded by their negation flag. The scope of a negation signal is also set to five words before or after the signal.

3.7.4 Evaluation

As the algorithm used in this study is the same as in [Chapman et al., 2001], the same statistical values can be applied. Unfortunately, the authors do not deliver any information about an improvement in relation to NegEx thanks to the usage of pre-coordinated phrases.

3.8 Concept Negation in Free Text Components of Vaccine Safety Reports

In the work of [Tolentino et al., 2006] methods of concept matching and integrated negation detection were applied to surveillance systems for the monitoring of adverse events following immunization (AEFI).

AEFI reports contain similar free text components as other clinical narrative reports like radiology reports, physical examinations or discharge summaries. Considering this fact the study was carried out to evaluate results of the application of UMLS mapping and negation detection algorithms in this medical special field.

After the tagging of the AEFI free text records with the UMLS Metathesaurus a rule-based, finite state machine algorithm was applied for the detection of negated concepts. The algorithms negation vocabulary list contained only five negation signal, which were *no*, *neither/nor, ruled out, denies* and *without*.

Despite the only very little number of used negation signals the algorithm results in rather high values in recall and precision. Recall and precision are parameter used in the special field of information retrieval. They are applied to evaluate the results of a searching process. The recall measures, how many of the relevant documents of a statistical universe could be found by the searching strategy. The precision describes, how many of the found documents were relevant documents and is therefore a measurement for the accuracy of a searching strategy. The detection algorithm used in this study reached a recall of 89 % and a precision of 94 %.

4 Summary

The comparison of the discussed algorithms shows, that even simple implementation strategies can provide respectable results. NegEx with its relatively simple detection strategy has nonetheless good values in sensitivity and specificity. It can be used where a fast implementation is needed.

To achieve better results more sophisticated strategies need to be implemented, as shown in [Mutalik et al., 2001], where a syntactical processing of grammar lead to respectable results. The approach in this study used a more detailed processing strategy of language and is therefore more complicated then the simple NegEx. From this study it can be seen as a humorous secondary aspect, that an automated processing with the presented algorithm has an equal accuracy to a human reader, although the reason for the imperfection of the human is a different than the reason of the algorithm. Whereas an algorithm operates on an incomplete vocabulary and therefore cannot detect every negation, a human reader loose concentration after some time of reading.

In languages different from English the correct recognition of negations seems to be an even more complicated task, as the study on Bulgarian negation shows. In this language the right assignment of the scope of a negation is a great problem for the applied algorithms.

Negation detection in the natural language processing is currently a task, where there are not many research results, which would fully cover the problem. Although there are really good approaches for the job, implemented algorithms are far away from the possibilities of a human, as it is in many scientific fields, where machines take on human tasks. The human reader is limited by his/her abilities in a given language and by the time span of his/her mental concentration, but is able to judge negations on a semantic level.

By now, a full understanding on a semantic level can not be accomplished by automatic systems, but future works on this topic may approach man's possibilities more and more. Perhaps some day the human's capability of language understanding can be combined with the machine's endurance and accuracy.

5 References

[Aronson, 2001]: Alan R. Aronson. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. *Proc. of the AMIA Symposium*, 17--21, 2001.

[Boytcheva et al., 2005]: Svetla Boytcheva, Albena Strupchanska, Elena Paskaleva, Dimitar Tcharaktchiev. Some Aspects of Negation Processing in Electronic Health Records. In *Proc. of International Workshop Language and Speech Infrastructure for Information Access in the Balkan Countries*, pages 1-8, Borovets, Bulgaria, 2005.

[Chapman et al., 2001]: Wendy W. Chapman, Will Bridewell, Paul Hanbury, Gregory F. Cooper, Bruce G. Buchanan. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of Biomedical Informatics*, 34(5):301-310, 2001.

[Elkin et al., 2005]: Peter L. Elkin, Steven H. Brown, Brent A. Bauer, Casey S. Husser, William Carruth, Larry R. Bergstrom, Dietlind L. Wahner-Roedler. A controlled trial of automated classification of negation from clinical notes. *BMC Medical Informatics and Decision Making*, 5(13), 2005.

[Hazlehurst et al., 2005]: B. Hazlehurst, H. R. Frost, D. F. Sittig and V.J. Stevens. Mediclass: A system for detecting and classifying encounter-based clinical events in any electronic medical record. *Journal of the American Medical Informatics Association (JAMIA)*, 12(5):517-529, Sep-Oct 2005.

[Horn, 1989]: Laurence R. Horn. A natural history of negation. *University of Chicago Press*, Chicago, Illinois, June 15 1989.

[Hripcsak et al., 1995]: George Hripscak, Carol Friedman, Philip O. Alderson, William DuMouchel, Stephen B. Johnson, Paul D. Clayton. Unlocking clinical data from narrative reports: a study of natural language processing. *Annals of Internal Medicine*, 122:681-688, 1995

[Humphreys et al., 1993]: B.L. Humphreys & P.L. Schuyler, The Unified Medical Language System: Moving beyond the vocabulary of bibliographic retrieval., in N.C. Broering (ed.), *High-Performance Medical Libraries: advanced information management for the virtual era*, Meckler, Westport, CT, 31--44, , 1993.

[Lee & Geller, 2006]: Yugyung Lee and James Geller. Semantic enrichment for medical ontologies, in *Journal of Biomedical Informatics*, 39(2):209—226, 2006.

[McCray, 1989]: Alexa T. McCray. UMLS Semantic Network. *The 13th annual SCAMC*, 503--507, 1989.

[Meystre & Haug, 2005]: Stéphane Meystre and Peter J. Haug. Automation of a problem list using natural language processing. *BMC Medical Informatics and Decision Making*, 5(1):30, 2005.

[Mutalik et al., 2001]: Pradeep G. Mutalik, Aniruddha Deshpande, Prakash M. Nadkarni. Use of General-purpose Negation Detection to Augment Concept Indexing of Medical Documents: A Quantitative Study using the UMLS. *Journal of the American Medical Informatics Association (JAMIA)*, 8(6):598-609, 2001.

[Patrick et al, 2006]: Jon Patrick, Yefeng Wang, Peter Budd. Automatic Mapping Clinical Notes to Medical Terminologies. In *Proceedings of the 2006 Australasian Language Technology Workshop (ALTW2006)*, pages 75-82, 2006.

[SNOMED]: SNOMED® International: http://www.snomed.org/, September 2006.

[Tolentino et al., 2006]: H. Tolentino, M. Matters, W. Walop, B. Law, W. Tong, F. Liu, P. Fontelo, K. Kohl, and D. Payne. Concept negation in free text components of vaccine safety reports. In *Proc. of the AMIA 2006 Symposium*, page 1122, 2006.