Implementation and Evaluation of Four Different Methods of Negation Detection

Sergey Goryachev, M.S.[1], Margarita Sordo, Ph.D.[2], Qing T. Zeng, Ph.D.[3], Long Ngo, Ph.D.[4]
[1-3] DSG, Brigham and Women's Hospital, Harvard Medical School, Boston, MA, USA
[4] Harvard Medical School, Boston, MA, USA

Abstract

Negation status identification for findings or diagnoses is an important medical data mining problem. Negative qualifier assigned to a medical condition may indicate the absence of the condition, so the ability to reliably identify the negation status of medical concepts affects the quality of results produced by the indexing and search tools.

Searching for the best negation algorithm to use in our negation module for the suite of NLP tools, we modified two existing regular expression-based algorithms in an attempt to improve their performance, and created two classification-based methods. The classification-based algorithms were trained on 1745 discharge reports from a Boston-based hospital. The algorithms were evaluated on 100 randomly-chosen outpatient reports from two different Boston-based hospitals, and the results were compared to the gold standard created by two independent human reviewers.

The regular expression and syntactic processing-based algorithms appeared to have better agreement (Kappa = 0.77 to 0.79) with the human reviewers than the classification-based algorithms (Kappa = 0.57 to 0.75). The accuracy of regular expressions methods (91.9-92.3%) was also higher than that of classification based methods (83.5-89.9%).

Based on our results, we have selected NegEx algorithm for our negation module.

1. Introduction

In recent years, a number of Natural Language Processing (NLP) applications have been developed to extract clinical information from medical records [1-5]. The most common types of information extracted are diagnoses or findings. Depending on the context where these concepts are found, they may be considered negated or questionable. Identifying the negation status of a finding is as important as identifying the finding itself. For example, a finding occurring in a negated context may indicate the absence of some medical condition. Search tools looking for documents containing a particular finding may return irrelevant results if they do not take the negation into account.

Several methods for negation status identification have been developed in the recent years to determine whether a finding is negative or positive. Chapman and colleagues developed NegEx[2], a regular expression-based approach that defines a fairly extensive list of negation phrases that appear before or after a finding. If a negation phrase appears within *n* words of a finding, then it is considered to be negated.

Being generally effective, the NegEx regular expression-based approach is somewhat simplistic in locating the negative findings. The NegExpander [1] algorithm, developed by Aronow and colleagues, uses syntactic processing techniques to identify noun phrases or conjunctive phrases that define negation boundaries.

More recently, the machine-learning methods have also been utilized as an alternative to manually extracting negation patterns. Averbuch and colleagues developed an algorithm that automatically learns the negative context patterns in medical narratives using the information gain calculation technique [4].

Following our objective to add the best-performing negation algorithm implementation to a suite of NLP tools we have been developing for the I2B2 (Informatics for Integrating Biology & the Bedside) project, we implemented and modified the Chapman and Aronow algorithms. We also trained Naïve Bayes and Support Vector Machines (SVM) classifiers for negation detection on a set of manually annotated discharge summaries. We tested and compared these four negation detection methods using a sample of outpatient notes.

2. Methods

We have adapted two existing negation algorithms: NegEx, described in section 2.1, and NegExpander, described in section 2.2. We have also trained two machine learning algorithm-based classifiers, a Naïve Bayes and a SVM (section 2.3).

2.1. NegEx Algorithm

The NegEx negation algorithm developed by Chapman et al. works as follows: the input to NegEx is a sentence with identified UMLS terms determined to belong to finding or diagnosis semantic types. The output of NegEx is the negation status assigned to each of the UMLS terms in the sentence: negated, possible or actual. The algorithm uses the following regular expressions triggered by three types of negation phrases:

```
<pre-UMLS negation phrase> {0-5 tokens} <UMLS term>
```

and

```
<UMLS term> {0-5 tokens} <post-UMLS negation phrase>
```

The three types of negation phrases in these expressions are *pre-UMLS*, *post-UMLS* and *pseudo* negation phrases. Pre-UMLS phrases occur before the term they negate, while the post-UMLS phrases occur after the term they negate. Pseudo negation phrases resemble negation phrases but are not reliable indicators of negation; they are used to limit the negation scope. The token can be a word or UMLS term; punctuation is not considered a token. All UMLS terms inside of the 0-5 tokens window are assigned the negation status depending on the nature of the negation phrase: negated or possible.

We decided to modify the original NegEx configuration after testing it on 100 discharge summary reports using the 2004 AA UMLS database, because the initial testing results were unsatisfactory. We found that many valid findings/diagnoses were omitted by the algorithm because their semantic types were not determined to be findings/diagnoses. Besides, the original NegEx list of irrelevant terms did not cover the new terms in UMLS 2004 AA. The authors of NegEx define a UMLS term as irrelevant if it belongs to the finding or diagnosis semantic type, but is not an actual finding or diagnosis that can be negated, for example, *history*. To address these problems, we extended the list of semantic types; specifically, we included two new semantic types: T191 (Neoplastic Process) and T046 (Pathologic Function). Also, we added 650 new terms to the list of irrelevant terms, such as *disease*, *syndrome*, *family history*, and *complaint*. We re-used the original list of negation phrases from the latest version of NegEx without modification.

2.2. NegExpander Algorithm

NegExpander is the negation algorithm developed by Aronow et al. This algorithm is part of the InQuery information retrieval system developed by the University of Massachusetts at Amherst. NegExpander solves the problem of identifying negated UMLS terms by constructing conjunctive phrases that define the negation boundaries. Conjunctive phrase is a group of noun phrases connected with conjunctions such as "*and*," "*or*" and "*,*."

The input to the algorithm is a sentence with identified UMLS terms and part-of-speech (POS) tags. NegExpander first searches for conjunctive phrases inside of a sentence. Then the algorithm searches for negation phrases inside the conjunctive phrases. If at least one negation phrase is found, the negation is expanded to all UMLS terms inside of a conjunctive phrase.

Following a test run on 100 discharge summary reports, the important fact we observed was that not all negation phrases reside inside of conjunctive phrases. For example, such a strong negation indicators as *denies* and *declines* are verbs and thus can not be a part of any noun or conjunctive phrases that should be negated. Chapman shows that *denied* and *denies* negation phrases are among the top 14 negation indicators that account for 15% of all negations alone [6]. We decided to take into account the negation phrases outside of the conjunctive phrases, and distinguish between pre- and post-UMLS negation phrases outside of the conjunctive phrases. The notion of pre- and post-UMLS phrases was borrowed from the NegEx algorithm.

As a result, the extended algorithm first applies the negation status to all UMLS terms inside a conjunctive phrase, if a negation phrase is found inside of a conjunctive phrase. Second, the algorithm searches for conjunction phrases that match the following regular expressions:

```
<pre-UMLS negation phrase> {0-2 words} <conjunctive phrase>
```

or

```
<conjunctive phrase> {0-2 words} <post-UMLS negation phrase>.
```

Finally, the algorithm reverses the negation status for all UMLS terms inside the matching conjunctive phrases.

2.3. Machine Learning Classifiers

Inspired by our recent successful use of the machine learning approach to extract the smoking status from medical records [7], we created two classifiers using Weka machine learning software [8]. Both classifiers were first trained on a set of human-annotated sentences taken from 1745 Brigham and Women's Hospital (BWH) discharge reports containing finding/diagnosis terms and their negation status (actual, negated and possible). We later used the trained classification models to detect the negation status of a finding/diagnosis term based on its sentence context.

The training data for the classification was randomly selected from a very large research patient data repository (RPDR). We used the suite of NLP tools that we have developed to extract UMLS terms from each discharge report. The extraction process is shown in the **Figure 1**. Each report was processed using a Tokenizer, Part-of-Speech Tagger (based on the decision list tagging method described in [9]), Sentence Splitter, Noun Phrase Finder (based on the transformational learning method described in [10]), and UMLS Concept Finder. The UMLS Concept Finder utilized a list of 16 semantic types, which was the same as the list we used with NegEx and NegExpander, to identify findings and diagnoses terms. The complete set of the extracted UMLS terms was then passed to the UMLS Concept Filter to eliminate irrelevant UMLS terms.

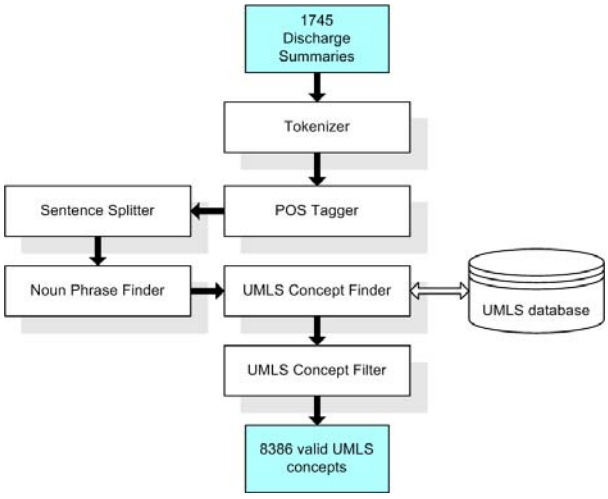

Figure 1. The process of UMLS terms extraction.

As the next step, the negation status of the extracted 8386 terms was manually classified by a human expert as *actual*, *negated* or *possible*. The classification was done solely based on a single sentence context of a term.

Before training the classifiers, we applied the special pre-processing to the sentences with UMLS terms. First, using the list of phrases borrowed from the NegEx algorithm, we identified pre- and post-UMLS negation phrases (e.g. *denies*, *was ruled out*), conditional possibility phrases (e.g. *rule out*) and conjunctions (e.g. *however*) and represented them with the corresponding tags: `[PRE_NEG]`, `[PRE_POS]`, `[POST_NEG]`, `[POST_POS]` and `[CONJ]`. Each finding/diagnosis term was replaced with the `[TERM]` tag. Next we replaced the remaining words with their corresponding POS tags, such as `[VBN]` and `[NN]`. The punctuation symbols remained unchanged.

Using the resulting tagged sentences, we created an attribute record for every finding/diagnosis term. We selected six adjacent tags (negation phrase tags, conditional possibility phrase tags, conjunction tags, POS tags and punctuation symbols only) from each side of a concept. If no tags were found on either side of a concept, `[NULL]` tags were used instead. A sample attribute record is shown below:

```
[VBD] [VBN] [NN] [,] [CONJ] [,] [VBD]
[PRE_NEG] [VBN] [.] [NULL] [NULL]
```

The attributes were selected through trial and error. The simpler bag-of-words approach, for example, did not appear to be promising.

The created set of attribute records along with their corresponding human classifications was supplied to the Weka software to train two different classification models: a Naïve Bayes classifier and a Support Vector Machines classifier.

2.4. Evaluation

To evaluate the performance of the negation algorithms, we randomly selected 100 notes, 50 for BWH and 50 for Massachusetts General Hospital (MGH), from a set of 862,643 outpatient notes. Using the suite of NLP tools, we tokenized the selected outpatient notes, added POS tags and split the notes into sentences. Next, we identified the UMLS terms using the UMLS Concept Finder tool. A total number of 1538 finding/diagnosis terms were identified: 772 in the BWH notes and 766 in the MGH notes. Two human experts (voter 1 and voter 2) judged whether these UMLS term was *negated*, *possible* or *actual* in a single sentence context.

We applied the four negation identification methods to the selected 100 outpatient notes and compared their results against the human judgments. This analysis essentially compared the four negation methods with voter 1 and voter 2's expert assessment

and used Kappa statistics to document the level of agreement between the negation methods and human assessments (if the estimate is 0.8 or above, there is excellent agreement between the algorithm and the voter's assessment, while 0.6 to 0.8 is considered good agreement).

3. Results

Among the 1538 terms, 1071 were positive (69.64%), 430 were negative (27.96%) and 37 were possible (2.41%) according to voter 1 and 1082 were positive (70.35%), 441 were negative (28.67%) and 15 were possible (0.98%) according to voter 2. The accuracy (i.e., rate of correct negation status identification) of the algorithms ranged from 0.84 to 0.92 (**Table 1**).

Table 1. Combined accuracy of four negation algorithms.

	Accuracy		
Algorithm	**Voter 1**	**Voter 2**	**Average**
NegEx	92.2627	91.5475	91.9051
NegExpander	92.6528	91.8756	92.2627
SVM	90.0520	89.7919	89.9220
Naïve Bayes	83.4850	84.9155	84.2003

BWH estimates of Kappa are all higher than MGH ones (**Table 2** and **Table 3**). This indicates heterogeneity across centers, and thus the analysis may be more appropriate when it is separated. A combined analysis of both centers was also done (**Table 4**). In all three analyses, the NegEx and NegExpander algorithms appear to perform better than the SVM and Naïve Bayes ones. The difference in performance between NegEx, NegExpander and SVM, however, is small.

Table 2. Summary of the comparison between four negation algorithms and voters' assessment for Brigham and Women's Hospital (BWH)

	Algorithm	Kappa	Standard Error	95% Confidence Interval
Voter 1	**NegEx**	0.8716	0.0217	0.8291 - 0.9141
	NegExpander	0.8649	0.0223	0.8211 - 0.9087
	SVM	0.8223	0.0239	0.7755 - 0.8692
	Naïve Bayes	0.6308	0.0314	0.5693 - 0.6924
Voter 2	**NegEx**	0.8622	0.0203	0.8223 - 0.9020
	NegExpander	0.8555	0.0209	0.8146 - 0.8964
	SVM	0.8183	0.0227	0.7739 - 0.8627
	Naive Bayes	0.6348	0.0308	0.5745 - 0.6951

Table 3. Summary of the comparison between four negation algorithms and voters' assessment for Massachusetts General Hospital (MGH)

	Algorithm	Kappa	Standard Error	95% Confidence Interval
Voter 1	**NegEx**	0.6810	0.0310	0.6202 - 0.7419
	NegExpander	0.7032	0.0304	0.6435 - 0.7628
	SVM	0.6470	0.0314	0.5854 - 0.7085
	Naïve Bayes	0.5116	0.0338	0.4454 - 0.5778
Voter 2	**NegEx**	0.7016	0.0294	0.6439 - 0.7592
	NegExpander	0.7251	0.0285	0.6692 - 0.7810
	SVM	0.6835	0.0293	0.6260 - 0.7410
	Naïve Bayes	0.6013	0.0321	0.5384 - 0.6643

Table 4. Combined summary of the comparison between four negation algorithms and voters' assessment for BWH and MGH

	Algorithm	Kappa	Standard Error	95% Confidence Interval
Voter 1	**NegEx**	0.7730	0.0197	0.7343 - 0.8117
	NegExpander	0.7811	0.0195	0.7428 - 0.8193
	SVM	0.7317	0.0204	0.6918 - 0.7716
	Naïve Bayes	0.5689	0.0234	0.5231 - 0.6147
Voter 2	**NegEx**	0.7806	0.0183	0.7446 - 0.8165
	NegExpander	0.7890	0.0180	0.7536 - 0.8243
	SVM	0.7498	0.0188	0.7129 - 0.7867
	Naïve Bayes	0.6177	0.0223	0.5739 - 0.6614

Table 5. Sensitivity, specificity, precision and F-measure of four negation algorithms.

	Statistic	BWH	MGH	Average
NegEx	Sensitivity	0.9688	0.9222	0.9455
	Specificity	0.9623	0.9231	0.9427
	Precision	0.8986	0.7981	0.8484
	F-measure	0.9323	0.8557	0.8940
NegExpander	Sensitivity	0.9167	0.8815	0.8991
	Specificity	0.9831	0.9573	0.9702
	Precision	0.9565	0.8942	0.9254
	F-measure	0.9362	0.8878	0.9120
SVM	Sensitivity	0.8649	0.8160	0.8405
	Specificity	0.9715	0.9319	0.9517
	Precision	0.9275	0.8317	0.8796
	F-measure	0.8951	0.8238	0.8595
Naïve Bayes	Sensitivity	0.7403	0.7826	0.7615
	Specificity	0.9291	0.9100	0.9196
	Precision	0.8261	0.7826	0.8044
	F-measure	0.7808	0.7826	0.7817

4. Discussion

We have implemented and tested four negation methods for processing clinical reports. Overall, modified versions of NegEx and NegExpander as well as SVM all showed good agreements with the

human reviewers. There was little difference between the performances of NegEx and NegExpander. Both NegEx and NegExpander did slightly better than SVM. The Naive Bayes method's performance was clearly the worst.

We have also observed that the source of test documents had a pronounced impact on negation detection accuracy. NegEx, NegExpander and SVM had excellent agreement with the human reviewers on the BWH reports, but only good agreements on the MGH ones. This difference may be caused by a wider use of word abbreviations and professional slang, which suggests the difference in personnel training between the two facilities.

Although NegEx and NegExpander produced the best testing results, they had some shortcomings. For NegEx, a rigid 5-token window may lead to missing some negated UMLS terms in long lists of terms, or when the negation phrase and a term are separated by five or more words. The algorithm may negate a part of high-level composite concept [3], while the other part may not be negated. The algorithm fails when it encounters a valid negation word that applies to the adjacent verb rather than to the following or preceding indexed terms, e.g. "[UMLS term] **did not** *increase*" [3]. The algorithm does not use any existing knowledge about the sentence, such as the noun phrase boundaries or POS tags to identify the negation boundaries.

Neither the original NegExpander algorithm nor our extended algorithm distinguishes between pre-UMLS and post-UMLS negation phrases *inside* conjunctive phrases. This may result in incorrectly negated UMLS terms preceding the pre-UMLS negation phrases or following the post-UMLS negation phrases inside conjunctive phrases, hence reducing the overall algorithm's specificity.

Another shortcoming of NegExpander is that it doesn't take the conditional possibility phrases such as *rule out* into account. Adding this feature to the algorithm, however, would create a few problems. Assuming we could distinguish between negation phrases and conditional possibility phrases (as they are defined in the NegEx algorithm), we may obtain several negation phrases of different types inside a single conjunctive phrase. In this case, some mechanism, such as a phrase ranking, would have to be added to choose the type of negation (negated or possible) to be applied to the UMLS terms inside the conjunctive phrase. To avoid this complexity, we decided to limit our enhanced algorithm to identifying strictly negative UMLS terms.

The classification-based methods (SVM and Naive Bayes) did not perform as well as NegEx and NegExpander to a large extent because they were trained on BWH discharge summaries and tested on BWH and MGH outpatient notes. (The classifiers performed very well on the training data set when tested through 10-fold cross-validations.) The difference in the training and testing data was intentional, given that negation extraction is a relatively generic task and one cannot expect to train the classifiers for all types of clinical documents.

As a result of this study, we chose to use the NegEx algorithm for our negation finder module. Since the accuracy of NegEx was still not completely satisfactory (~92%), we will continue to explore other methods for improvement.

5. Conclusion

We have implemented and modified two existing regular-expression and syntactic processing-based algorithms and implemented two classification-based algorithms. All four negation algorithms were evaluated on 100 randomly-selected outpatient notes from two Boston-based hospitals. Our results suggest that the regular expression and syntactic processing-based algorithms have better agreements with human reviewers than the classification-based algorithms.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgements

This work is funded by the I2B2 grant number U54 LM008748.

References

1. Aronow, D.B., Feng, F., Croft, W.B., *Ad Hoc Classification of Radiology Reports.* Journal of the American Medical Informatics Association, 1999. **6**(5): p. 393-411.
2. Chapman, W.W., Bridewell, W., Hanbury, P., Cooper, G.F., Buchanan, B.G., *A Simple Algorithm for Identifying Negated Findings and Diseases in Discharge Summaries.* Journal of Biomedical Informatics, 2001. **34**: p. 301-310.
3. Mutalik, P.G., Deshpande, A., Nadkarni, P.M., *Use of General-Purpose Negation Detection to Augment Concept Indexing of Medical Documents.* Journal of the American Medical Informatics Association, 2001. **8**(6): p. 598-609.

4. Averbuch, M., Karson, T., Ben-Ami, B., Maimon, O., Rokach, L. *Context-Sensitive Medical Information Retrieval*. in *Proc. of 11th World Congress on Medical Informatics (MEDINFO-2004)*. 2004. San Francisco, CA: IOS Press.
5. Elkin, P.L., et al., *A controlled trial of automated classification of negation from clinical notes.* BMC Med Inform Decis Mak, 2005. **5**(1): p. 13.
6. Chapman, W.W., Bridewell, W., Hanbury, P., Cooper, G.F., Buchanan, B., *Evaluation of Negation Phrases in Narrative Clinical Reports.* Proc AMIA Symp, 2001: p. 105-109.
7. Zeng, Q.T., et al., *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**(1): p. 30.
8. Witten, I.H., Frank, E., *Data Mining: Practical Machine Learning Tools and Techniques*. 2nd ed. 2005, San Francisco: Morgan Kaufmann.
9. Hepple, M. *Independence and Commitment: Assumptions for Rapid Training and Execution of Rule-based Part-of-Speech Taggers.* in *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL-2000)*. 2000. Hong Kong.
10. Ramshaw, L.A., Marcus, M.P., *Text Chunking using Transformation-Based Learning.* ACL Third Workshop on Very Large Corpora, 1995: p. 82-94.