

Learning to Detect Negation with ‘Not’ in Medical Texts

Ilya M. Goldin
Intelligent Systems Program
Learning Research & Development Center
University of Pittsburgh
Pittsburgh, PA 15260
goldin@pitt.edu

Wendy W. Chapman
Center for Biomedical Informatics
University of Pittsburgh
Pittsburgh, PA 15260
chapman@cbmi.upmc.edu

Abstract

While state of the art techniques can address the problem of automatically detecting negated medical observations, negation using the word ‘not’ presents a harder problem than other kinds of negation. We apply machine learning techniques to distinguish sentences where the word ‘not’ does and does not negate a medical observation. Our corpus contains hospital reports such as progress notes and emergency room notes. We use two different machine learning algorithms, Naive Bayes and Decision Trees, and both achieve significant improvement over the baseline. We also analyze the data and the classifiers’ behavior and output to learn more about the problem and the usefulness of various features in our feature vector.

1 Introduction

With the advent of the world wide web and the increase in computing capabilities, huge amounts of textual data have become electronically available - web pages, digital libraries [1], newspaper article archives, and hospital information systems contain billions of documents that prevent feasible human extraction or retrieval of information. NLP techniques, both statistical and symbolic, have been applied to automated data mining of text. The National Library of Medicine has created information extraction tools for biomedical knowledge bases and for Medline abstracts in particular [2, 3, 4, 5, 6]. Medical text mining techniques are being used for, among other things, extracting biomedical knowledge from journal articles [7, 8], indexing teaching files [9], identifying patients of interest for research studies [10, 11], creating vocabularies from text [12, 13, 14, 15], enhancing online searches with

patient information [16, 17], and encoding phrases into standard vocabularies [18].

Techniques for automatically indexing knowledge in the literature may also be applied to textual hospital records, which contain patient-specific clinical observations and test results. However, accurate indexing of clinical observations in dictated records, such as emergency department or radiology reports, particularly requires an understanding of the relation of the condition to an individual patient. For example, temporal modeling is important for determining whether a clinical condition existed in a patient at the current hospital visit or whether the condition occurred in the patient’s past history. Indexing clinical observations in medical reports also entails recognizing whether the condition is attributed to the patient. For instance, a condition may be mentioned hypothetically in an emergency department report (as in “If the patient experiences shortness of breath, she should return”). Because of the frequency of negated observations in clinical reports [19], a particularly important element of indexing clinical conditions in the reports is determining whether the condition is present or absent in the patient.

Natural language processing (NLP) systems created for the medical domain consider whether a clinical observation is negated in the text [20, 21, 22, 23]. Algorithms for determining negation in most NLP systems are entwined within the NLP system’s syntactic processor. Recently, medical informatics researchers have developed negation algorithms that stand alone and can be incorporated into indexing applications [24, 25]. Chapman et al. [25, 19] describe a regular expression-based algorithm, NegEx, for detecting negated observations in medical reports. NegEx relies on a list of phrases that indicate the absence of a clinical condition (negation phrases) to

negate the conditions within a window of six words or phrases. For example, in the sentence “The patient denies chest pain,” the term ‘chest pain’ is negated by the negation phrase ‘denies.’ NegEx currently contains a list of 183 negation phrases that trigger negation of terms in the Unified Medical Language System (UMLS) [4] representing clinical conditions [26]. NegEx is an attempt to create a simple algorithm for negation that researchers without access to full NLP systems can implement.

In a previous study, NegEx performed overall with 78% recall (sensitivity) and 84% precision (positive predictive value) [25]. When the negation phrase is ‘not,’ precision drops to 58.33% [25] or 63% [19]. The work described in this paper extends NegEx’s ability to accurately determine whether a clinical observation within the six-word window of the negation phrase ‘not’ should actually be negated. We use existing machine learning algorithms, Naive Bayes (NB) and Decision Trees (DT), to learn when ‘not’ correctly predicts negation in an observation and discuss the suitability of using machine learning techniques for this purpose.

2 Methods

2.1 Classifiers

For this comparison we selected two classifiers, NB and DT. Both are well-established, supervised learning methods for classifying discrete data. One is a statistical approach (NB), and the other symbolic (DT). As a baseline method, we apply the default rule in NegEx, which is to negate any UMLS term within the six-word window following ‘not.’ The discussion below is a minimal overview of the classifiers, which are explained elsewhere. [27, 28]

NB is based on the Bayesian probability formula, which calculates the conditional probability of one event given another. In machine learning, this is used to calculate the probability of a classification hypothesis (from a set of such hypotheses) given a vector of features and their values describing a single instance in the data set. This method is called naive because it makes the assumption that the instances are conditionally independent of each other.

DT learns rules, expressed as “conjunctions of constraints on the attribute values of instances. Each path from the tree root to a leaf corresponds to a conjunction of attribute tests, and the tree itself to a disjunction of these conjunctions.” [27]

2.2 Data Set

The data set contained 207 randomly selected sentences from the Chapman et al. corpus [19] in which a UMLS term would be negated by NegEx with the negation phrase ‘not.’ The corpus from which the sentences were selected contained reports from the MARS (Medical Archival System) repository at the University of Pittsburgh Medical Center and included ten report types, such as progress notes, chest radiograph reports, and emergency room notes.

Every sentence was described according to our feature vector, comprising linguistic information about the sentence. We designed the feature vector by analyzing a small independent sample of sentences in which NegEx inaccurately negated a UMLS term with the negation phrase ‘not’ (Table 1). We identified some easily computable linguistic traits that may serve as clues that ‘not’ should not negate the UMLS term in its scope. For example, the fact that the UMLS term is within a prepositional phrase may indicate that the term should not be negated, as in the sentence “This is not the source of the infection” in which ‘infection’ is a UMLS term that should not be negated. Features were devised for ease of future automation of value assignment.

For lack of complete automation at the time of the study, the authors semi-manually assigned feature values in the following way. First, the data set was parsed by an automated shallow parser (Link Grammar Parser, v4.2, <http://www.link.cs.cmu.edu/link>). Next, the authors collaboratively derived the feature values from the parser’s output. Once the authors agreed on methods for deriving the values, they individually assigned values to the features.

As can be seen from table 1, there is great similarity in how `PP between` and `UMLS in PP` are distributed, and in fact in every instance those features were tagged identically. To avoid (positive or negative) bias for NB, which could interpret two features with identical values as a single, doubly-weighted feature, we arbitrarily eliminated `PP between` from the data set. Whether these features may be distinguishable meaningfully and informatively could be answered empirically, given more data.

The gold standard for training and testing was created by a physician who read all the sentences in the data set. For every UMLS term in the data set, the physician recorded a binary judgment – whether the finding Described by the term was both related to the

Feature	Description	D, U ^a	Most frequent values
UMLS term	The UMLS term itself.	120, 90	pain (20), fracture (7), chest pain, drive, episodes, nausea (6), tachypneic (5), diarrhea, fever, fevers, seizure, vomiting (4), eating, fall, numbness, shortness of breath, tenderness (3)
not construction	Words immediately surrounding 'not'.	16, 5	does not (62), did not (46), has not (32), is not (30), not (10), was not (6), not to (5), had not (4), do not (3), could not, should not (2)
phrase after not	The grammatical phrase following 'not'.	5, 1	vp (176), adjp (15), np (11), pp (4)
num phras btwn	The number of grammatical phrases between 'not' and the UMLS term, excluding S and SBAR.	6, 0	1 (79), 2 (49), 3 (41), 0 (31), 4 (5), 5 (2)
PP between	Was there a prepositional phrase between 'not' and the UMLS term?	2, 0	n (139), y (68)
preposition	If a PP is present, what is the preposition?	13, 4	nil (139), of (33), with (11), for (7), to, in, on (3), by, around (2)
UMLS in PP	Does the UMLS term occur inside a prepositional phrase?	2, 0	n (139), y (68)
determiner	Is there a determiner inside the immediate phrase that contains the UMLS term, or inside its 'parent' phrase, but not preceding the 'not'?	9, 2	nil (101), any (69), the (14), a (10), h ^b (5), an (4), this (2)
parser's POS	What is the part of speech of the UMLS term (as assigned by the parser)?	4, 0	noun (165), verb (24), adjective (13), -ing verb (5)
in/out WHNP	Is the UMLS term in the scope of a WH phrase (like 'where' or 'when')?	2, 0	n (205), y (2)
answer	Does the 'not' of interest actually negate the UMLS phrase?	2, 0	negate (149), not negate (58)

^a'Distinct' values are like types (as opposed to tokens); 'Unique' values occur exactly once

^bThis is a collapsed category for determiners 'his' and 'her'.

Table 1: The feature vector and the distribution of data. For each feature, all values occurring more than once are listed, with the exception of `UMLS term`, for which only values occurring more than twice are listed.

current visit and negated. Only one human expert rater was used in creating the gold standard.

2.3 Experimental Design

The feature vectors for every sentence in the data set were processed by the two classifiers (DT and NB) and the baseline with ten repetitions of 10-fold cross-validation to assure statistical reliability. (The statistics reported for each measure are averages over all the runs.)

All features were treated as nominal (as opposed to numeric), including `num phras btwn`, which was 'discretized' into 6 classes corresponding to each of

the possible values.

The classifiers and baseline we used are part of WEKA (v3.2.2, <http://www.cs.waikato.ac.nz/ml/weka>). All classifiers were run with their default WEKA settings. Specifically, DT used a confidence threshold for pruning of 0.25, employed both subtree replacement and subtree raising, and required a minimum of 2 instances per leaf.

The algorithms were compared using the standard information retrieval metrics of precision (number of true positives divided by the sum of true positives and false positives) and recall (number of true positives divided by the sum of true positives and false negatives) [29]. These metrics are regularly used to com-

pare classifiers [30].

We also report accuracy (percent correct) as well as the F-measure, which combines recall and precision into a single number, [29]. The F-measure can be weighted to give higher importance to either precision or recall via the coefficient β . We weighed precision and recall equally ($\beta = 1$).

3 Results

Metric	Baseline	DT	NB
Precision	0.72	0.81 ^a	0.88 ^{ab}
Recall	1.00	0.99 ^b	0.93 ^a
F-measure	0.84	0.89 ^a	0.90 ^a
Pct Correct	72.00	82.76 ^a	85.07 ^{ab}

^aNB or DT vs baseline significantly different

^bNB vs DT significantly different

Table 2: Classifier performance. All comparisons are two-tailed t-tests at $p < 0.01$.

The data set was comprised of sentences in which a negation was triggered by NegEx with the negation phrase ‘not.’ Therefore, the baseline always has perfect recall. The purpose of this study was to increase NegEx’s precision of negation with ‘not.’ As shown in Table 2, DT and NB both had significantly higher precision than the baseline. Recall dropped significantly with NB but remained the same as baseline with DT. Both classifiers had higher F-measures than the baseline, and there was no significant difference between F-measures for the two classifiers.

To examine how the classifiers increased precision of negation, we will refer to the best possible decision tree, i.e., the one learned on the entire data set (Figure 1). We cannot judge the performance of this tree since we used the entire data set to generate it, but the tree should perform better than the average-quality tree measured in table 2, because it was learned on a larger data set.

The tree has a depth of two with the `determiner` feature at the root and one child node (under `determiner = ‘a’`) based on the `num phras between feature`. The choice of `determiner` as the top level feature coincides with the fact that the majority (80%) of UMLS terms are nouns, and most findings in medical reports are expressed as nouns.

All occurrences of `determiner = ‘the’` corresponded to medical observations that were not actually negated. This rule articulates an intuitively under-

standable concept: if the medical finding is referred to using ‘the’, i.e. the definite article, then the finding is probably observed to be present, e.g., “examination could not be performed due to the *aphasia*”, “he does not articulate the *pain*”.

```

determiner = nil: negate (101 / 27)
determiner = the: not negate (14)
determiner = any: negate (69 / 5)
determiner = a
├── num phras btwn = 0: not negate (2)
│   ├── num phras btwn = 1: negate (5)
│   ├── num phras btwn = 2: not negate (2)
│   ├── num phras btwn = 3: negate (1)
│   ├── num phras btwn = 4: negate (0)
│   └── num phras btwn = 5: negate (0)
├── determiner = an: negate (4)
├── determiner = h: not negate (5)
├── determiner = this: not negate (2)
├── determiner = these: not negate (1)
└── determiner = some: negate (1)

```

Figure 1: Decision Tree. Legend: feature: class (correctly classified / any incorrectly classified).

The majority (93%) of instances in which the `determiner = ‘any’` corresponded to medical observations that were actually negated (Table 3). This rule is also pleasantly intuitive: the pattern *not...any...medical observation* resonates with the negation of the medical observation, e.g., “she is not having any *pain* now”, “his other four doses of okt-3 were not associated with any *side effects* after he was affectively premedicated”, “she underwent a barium swallow which did not demonstrate any organ or axial *volvulus*”.

When a `determiner` did not precede the UMLS term (i.e., `determiner = ‘nil’`), no additional rules were learned from the feature vector and the default action is to negate the term as NegEx would have done. As expected, the precision of negation for this subset of instances was the same as the precision of the baseline for the entire data set (73% and 72%, respectively).

4 Discussion

Compared with previous work by Chapman et al. [25, 19], both classifiers showed improvement in precision of negation with ‘not’ over the baseline NegEx. The

tradeoff of increased precision was a decrease in recall, but the decrease was quite small. The F-measures of NB and DT classifiers were not significantly different, which is consistent with findings by Wilcox [31] that different classification techniques do not account for differences in performance as much as differences in the vector processed by the techniques.

Det	Instance
any	she does not give any history of any *weakness muscle* ⇒ Not negated because it refers to patient’s past, not the current visit.
any	the patient does not have any other sources of *infection* ⇒ Here, not refers to ‘sources’, not ‘infection’.
any	the pain did not start around any *cocaine* use ⇒ Not negated because ‘cocaine’ is not a finding, although ‘cocaine use’ could be.
nil	the patient was instructed not to *drive* for the next six months ⇒ Not negated, because ‘drive’ is a UMLS term in a different dictionary sense, i.e. in the sense of “energy, push, or aggressiveness” ^a .
nil	does not give a history of previous *joint pain*
nil	qqqq notes that it was not as bad as previous *headaches* ⇒ Not negated because of reference to patient’s past, not the current visit.
nil	the patient had not eaten all day felt *lightheaded* and then collapsed ⇒ Noise due to removal of punctuation in preprocessing. ‘felt lightheaded’ is a separate phrase, out of the scope of ‘not’.

^adrive, n., 7. The American Heritage Dictionary of the English Language, 4th ed. Boston: Houghton Mifflin, 2000. <http://www.bartleby.com/61/.5/31/2002>.

Table 3: Examples of DT’s false positives. Some extraneous context stripped.

The findings of this study could be summarized into a simple rule that could be easily implemented into NegEx as follows: When negation of a UMLS term is triggered with the negation phrase ‘not,’ if the term is preceded by ‘the’ then do not negate. This rule

would not require any syntactic processing and would increase NegEx’s precision.

We will perform additional studies to try to increase precision of UMLS terms that are not preceded by the determiner ‘the.’ Some ideas for additional features could involve expanding the feature vector to include parameters like keywords or phrases that are highly predictive of either negation or its absence (e.g. ‘associated with’, ‘consistent with’, ‘complain’, ‘history’, ‘clear’). This feature would benefit from a study of *n*-grams in sentences using ‘not’. (Arguably, NB takes such information into account via the probabilities of the highly fragmented UMLS term feature.) Other features could involve some kinds of semantic or other mark-up for UMLS terms.

Decreasing the level of noise in indexing UMLS terms would also boost NegEx’s precision. Specifically, in instances like “he does not take *seizure* medications”, *seizure* is the UMLS term, but it is not being used in this sentence as a finding [32]. We could help address this in two ways. First, if the UMLS term is not the head of the phrase containing it, then the term should probably not be negated. Second, if the UMLS term is being used as a different part of speech from its UMLS definition, it should probably not be negated. (See the *drive* problem in figure 3). This latter technique would not address the case where the UMLS term is a homonym of a more frequently used word with the same part of speech.

4.1 Limitations

This study is limited in several ways. First, the study is specific to a particular negation phrase. However, the findings of the study may be generalizable to other negation phrases. Further research may support the intuitive idea that a finding preceded by the definite article ‘the’ should not be negated by any negation term. Second, we discarded ten sentences from the data set, because the syntactic processor we used to derive the feature vector failed. Failed parsing has implications for any automated system using the output of a syntactic processor. A more robust parser that was created or modified to parse sentences in medical reports would be necessary if we wanted to use the output of the parser for negation. Third, the gold standard was created by a single physician. Reference standards created by multiple physicians are more reliable than those generated by an individual physician [33].

5 Conclusion

We showed how NegEx, a regular expression for “identifying negated findings and diseases in discharge summaries,” could be enhanced through machine learning from a vector of potentially useful linguistic characteristics. We gained new insights into the use of negation using ‘not’ in medical texts that may also be applied to negation with other negation phrases. Moreover, this work could be applicable to entirely different domains, such as law, where it is often crucially important to identify negated observations and statements.

6 Acknowledgments

This project began as an assignment for a class taught by Bruce Buchanan. The authors also thank Jeremy Espino who acted as the gold standard physician, Stefanie Brünighaus, and Kevin Ashley.

References

1. A. T. McCray. Digital library research and application. *Stud Health Technol Inform*, 76:51–62, 2000. 20316446 0926-9630 Journal Article Review Review, Tutorial.
2. A. T. McCray. Extending a natural language parser with umls knowledge. *Proc Annu Symp Comput Appl Med Care*, pages 194–8, 1991. 92223717 0195-4210 Journal Article.
3. A. T. McCray and S. J. Nelson. The representation of meaning in the umls. *Methods Inf Med*, 34(1-2):193–201, 1995. 97227867 0026-1270 Journal Article.
4. A. T. McCray, A. M. Razi, A. K. Bangalore, A. C. Browne, and P. Z. Stavri. The umls knowledge source server: a versatile internet-based research tool. *Proc AMIA Annu Fall Symp*, pages 164–8, 1996. 97103277 1091-8280 Journal Article.
5. A. T. McCray, J. Sponsler, B. Brylawski, and A. Browne. The role of lexical knowledge in biomedical text understanding. In *SCAMC 87*, pages 103–107, 1987.
6. A. T. McCray, S. Srinivasan, and A. C. Browne. Lexical methods for managing variation in biomedical terminologies. *Proc Annu Symp Comput Appl Med Care*, pages 235–9, 1994. 95037246 0195-4210 Journal Article.
7. C. Friedman, P. Kra, H. Yu, M. Krauthammer, and A. Rzhetsky. Genies: a natural-language processing system for the extraction of molecular pathways from journal articles. *Bioinformatics*, 17 Suppl 1:S74–82, 2001. 21365260 1367-4803 Evaluation Studies Journal Article.
8. A. Rzhetsky, T. Koike, S. Kalachikov, S. M. Gomez, M. Krauthammer, S. H. Kaplan, P. Kra, J. J. Russo, and C. Friedman. A knowledge model for analysis and simulation of regulatory networks. *Bioinformatics*, 16(12):1120–8, 2000. 21111532 1367-4803 Journal Article.
9. D. B. Johnson, W. W. Chu, J. D. Dionisio, R. K. Taira, and H. Kangaroo. Creating and indexing teaching files from free-text patient reports. *Proc AMIA Symp*, pages 814–8, 1999. 20032983 1531-605x Journal Article.
10. G. F. Cooper, B. G. Buchanan, M. Kayaalp, M. Saul, and J. K. Vries. Using computer modeling to help identify patient subgroups in clinical data repositories. *Proc AMIA Symp*, pages 180–4, 1998. 99123111 1531-605x Journal Article.
11. W. W. Chapman, G. F. Cooper, P. Hanbury, B. E. Chapman, L. H. Harrison, and M. M. Wagner. Creating a text classifier to detect radiology reports describing mediastinal findings associated with inhalational anthrax and other disorders. *J Am Med Inform Assoc*, 2003. 0 1067-5027 Journal article.
12. H. Liu and C. Friedman. Mining terminological knowledge in large biomedical corpora. *Pac Symp Biocomput*, pages 415–26, 2003. 22490651 Journal Article.
13. H. Yu, V. Hatzivassiloglou, C. Friedman, A. Rzhetsky, and W. J. Wilbur. Automatic extraction of gene and protein synonyms from medline and journal articles. *Proc AMIA Symp*, pages 919–23, 2002. 22352859 1531-605x Evaluation Studies Journal Article.
14. M. Krauthammer, A. Rzhetsky, P. Morozov, and C. Friedman. Using blast for identifying gene and protein names in journal articles. *Gene*, 259(1-2):245–52, 2000. 21100390 0378-1119 Journal Article.

15. M. Krauthammer, P. Kra, I. Iossifov, S. M. Gomez, G. Hripcsak, V. Hatzivassiloglou, C. Friedman, and A. Rzhetsky. Of truth and pathways: chasing bits of information through myriads of articles. *Bioinformatics*, 18 Suppl 1:S249–S257, 2002. 0 1367-4803 Journal article.
16. E. A. Mendonca, J. J. Cimino, and S. B. Johnson. Using narrative reports to support a digital library. *Proc AMIA Symp*, pages 458–62, 2001. 21684226 1531-605x Evaluation Studies Journal Article.
17. E. A. Mendonca, J. J. Cimino, S. B. Johnson, and Y. H. Seol. Accessing heterogeneous sources of evidence to answer clinical questions. *J Biomed Inform*, 34(2):85–98, 2001. 21407373 1532-0464 Journal Article Review Review, Tutorial.
18. Y. A. Lussier, L. Shagina, and C. Friedman. Automating snomed coding using medical language understanding: a feasibility study. *Proc AMIA Symp*, pages 418–22, 2001. 21684218 1531-605x Evaluation Studies Journal Article.
19. W. Chapman, W. Bridewell, P. Hanbury, G. Cooper, and B. Buchanan. Evaluation of negation phrases in narrative clinical reports. In *American Medical Informatics Association Symposium*, pages 105–9, 2001.
20. C. Friedman and G. Hripcsak. Natural language processing and its future in medicine. *Academic Medicine*, 74:890–5, 1999.
21. C. Friedman. A broad-coverage natural language processing system. *Proc AMIA Symp*, pages 270–4, 2000. 21027361 1531-605x Journal Article.
22. L. Christensen, P. J. Haug, and M. Fiszman. Mplus: a probabilistic medical language understanding system. *Proc Workshop on Natural Language Processing in the Biomedical Domain*, pages 29–36, 2002.
23. R. K. Taira and S. G. Soderland. A statistical natural language processor for medical reports. *Proc AMIA Symp*, pages 970–4, 1999. 20033015 1531-605x Journal Article.
24. P. G. Mutalik, A. Deshpande, and P. M. Nadkarni. Use of general-purpose negation detection to augment concept indexing of medical documents: a quantitative study using the umls. *J Am Med Inform Assoc*, 8(6):598–609, 2001. 21547134 1067-5027 Evaluation Studies Journal Article.
25. W. Chapman, W. Bridewell, P. Hanbury, G. Cooper, and B. Buchanan. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of Biomedical Informatics*, 34:301–10, 2001.
26. W. W. Chapman. Negex version 2: A simple algorithm for identifying pertinent negatives in textual medical records, 2003. <http://omega.cbmi.upmc.edu/~chapman/NegEx.html>.
27. T. Mitchell. *Machine Learning*. McGraw Hill, 1997.
28. I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques with Java implementations*. Academic Press, San Diego, 2000.
29. F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.
30. Y. Yang. An evaluation of statistical approaches to text categorization. *Journal of Information Retrieval*, 1(1/2):67–88, 1999.
31. A. Wilcox and G. Hripcsak. Classification algorithms applied to narrative reports. *Proc AMIA Symp*, pages 455–9, 1999. 20032910 1531-605x Journal Article.
32. C. Sneiderman, T. Rindfleisch, and A. Aronson. Finding the findings: identification of findings in medical literature using restricted natural language processing. In *AMIA Annual Fall Symposium*, pages 239–43, 1996.
33. G. Hripcsak and A. Wilcox. Reference standards, judges, and comparison subjects: roles for experts in evaluating system performance. *J Am Med Inform Assoc*, 9(1):1–15, 2002. 21623446 1067-5027 Journal Article.