Improving Feature Extraction for Pathology Reports with Precise Negation Scope Detection

Olga Zamaraeva
University of Washington
Department of Linguistics
olzama@uw.edu

Kristen Howell
University of Washington
Department of Linguistics
kphowell@uw.edu

Adam Rhine
University of Washington
Department of Linguistics
amrhine@uw.edu

Abstract

We use a broad coverage, linguistically precise English Resource Grammar (ERG) to detect negation scope in sentences taken from pathology reports. We show that incorporating this information in feature extraction has a positive effect on classification of the reports with respect to cancer laterality compared with NegEx, a commonly used tool for negation detection. We analyze the differences between NegEx and ERG results on our dataset and how these differences indicate some directions for future work.

Title and Abstract in Russian

К вопросу о применении формальных грамматик в посторении признаковых векторов для классификации отчетов о заключениях патологоанатомов

Мы предлагаем способ обогащения векторов признаков (*feature vectors*) лингвистичекой структурой. Для внешней оценки эксперимента (*extrinsic evaluation*) мы смотрим на результаты автоматической классификации отчетов о заключениях патологоанатомов (*pathology reports*, такие как заключение о результатах гистологического исследования и др.). В качестве лингвистической структуры мы выбрали сферу действия отрицания (*scope of negation*), так как известно, что отрицание в тексте может влиять на качество классификации медицинских заключений. В эксперименте мы используем имплементацию формальной грамматики английского языка (English Resource Grammar) для определения сферы действия отрицания на уровне предложения. Сначала текст отчета делится на предложения. Из каждого предложения при помощи базы данных UMLS мы выбираем слова и фразы – признаки. Те признаки, которые оказываются в сфере действия отрицания, мы заменяем на специальные признаки. Например, в предложении *No tumor in left breast*, если сфера отрицания определена достаточно широко, признак *left breast* будет заменен на признак *NEG:left breast*. Все признаки, полученные из одного отчета, собираются в один вектор (признаковый вектор данного отчета). Обогащенные таким образом признаковые векторы затем используются для классификации отчетов на группы по расположению первичного очага (в правой или левой части легкого или молочной железы). Мы сравниваем воздействие нашей системы построения признаковых векторов на точность классификации с популярной системой NegEx, которая использует для определения сферы действия отрицания ряд эвристик. Результаты и анализ ошибок позволяют увидеть, что формальная грамматика, основанная на глубоком синтаксическом анализе, действует точнее, чем система NegEx. Система NegEx может ошибочно предположить сферу действия отрицания в плохо структурированном тексте (например, в блоке текста, неправильно разбитом на предложения), а формальная грамматика в этом случае не выдаст никакого ответа и тем самым избежит ошибки. Мы делаем вывод, что использование формальных грамматик для подобных задач может быть целесообразно.

Proceedings of the 27th International Conference on Computational Linguistics, pages 3564–3575
Santa Fe, New Mexico, USA, August 20-26, 2018.

1 Introduction

Detecting negation scope remains a challenge in clinical notes information extraction (see e.g. Morante and Daelemans 2009 for an overview of existing approaches for negation detection and related issues). Perhaps the most extensively used negation detection tool is NegEx (Chapman et al., 2001), a regular expressions-based tool, with 696 citations on Google Scholar,[1] which is part of such well-known systems as cTAKES (Savova et al., 2010). One reason NegEx is so popular is that it is rule based and thus easily adapted to any English dataset. In general, rule based methods are important for domains where data cannot be easily shared and so there is no guarantee an existing machine learning approach can be successfully used, such as the clinical notes domain. That said, NegEx identifies negated tokens from surface strings using a set of domain-specific heuristics which it applies directly to the surface strings rather than to some syntactic or semantic representations of those strings. Such an approach is unlikely to capture the rules of English negation in their generality. In contrast, linguistically motivated implemented formal grammars[2] map surface strings to abstract syntactic and semantic representations and model language rules more accurately and robustly, in a domain-independent fashion.

For this paper, we classified a set of pathology reports diagnosing the cancer as located in the right or in the left part of the body, an important data element used in clinical operations and research for paired organ sites. The contribution of the paper is in incorporating precise negation scope information obtained with an implemented broad-coverage grammar of English into the feature extraction procedure. The experiment yielded incremental though consistent improvement when compared to NegEx, showing fairly clear evidence that incorporating robust domain-independent linguistic structure into feature extraction can be helpful for classification of this kind.

After summarizing related work (section 2), we present our experiment (section 3). The structure of the experiment is as follows. Each pathology report in the dataset (section 3.1) is represented by a feature vector constructed from a Unified Medical Language System (UMLS)-based feature set, which is obtained as described in section 3.2 and then augmented by what we call *negated features* (section 3.3). To help the reader, we introduce negated features right here first: In any sentence, a lexical feature which was selected as highly informative for the training dataset (e.g. *right*, or *malignancy*) could actually correspond to a negated concept. Assume the feature *malignancy* was selected by a feature selection algorithm and will now be used in a classification task on a test dataset. Compare two hypothetical single-sentence reports from this test dataset: *A malignancy was identified* and *No malignancy was identified*. For the second report, which contains the feature *malignancy* within the scope of negation, we want to exclude that feature when we turn this sentence into a vector. Otherwise, if we were to include *malignancy*, we might misclassify the report as positive.[3] Furthermore, a negated concept (which we will represent as *NEG:malignancy* in this case) may itself be an informative feature. We identify these negated features in the text of the reports using NegEx (as baseline) and the English Resource Grammar (ERG; Flickinger, 2000, 2011), as described in 3.3. Then we train a number of classifiers on the training portion of the dataset represented as feature vectors to discriminate between the classes *left* and *right*; we then evaluate on the test portion of the data. Since our dataset was not annotated for negation, we are performing an extrinsic evaluation of negation detection by showing the effect that using the grammar for feature extraction had on classification. This extrinsic evaluation can later be complemented by an intrinsic one.[4] We present the numeric results (section 4) which show small but consistent improvement of our approach over NegEx and conclude with a detailed analysis of the differences between the two systems, which helps us identify directions for future work.[5]

[1] Accessed June 5, 2018.

[2] Not to be confused with domain specific grammars, common in natural language processing.

[3] Admittedly if the training data contained lots of negated examples, the feature would not come up as one of the best for positive classification. However, negated and non-negated concepts may be unbalanced in the data, and furthermore, machine learning-based classification here is just one scenario.

[4] As discussed in the related work section, the effect of finding negation scope using a precision grammar of English has been examined by MacKinlay et al. (2012) on the sentence-level.

[5] To protect possible traces of sensitive data, the code will be available upon request.

2 Related Work

Both rule-based and machine learning approaches exist for detecting negation scope. Morante and Daelemans (2009) is an example of a machine learning approach. Their system consists of two classification tasks performed with supervised machine learning methods: finding negation tokens and classifying tokens as the first or last (or neither) token within the scope of negation. Morante and Daelemans (2009) report high scores on the BioScope dataset (Vincze et al., 2008). However, machine learning approaches like this one require sufficient annotated training data not always available for clinical tasks.

Sohn et al. (2012) use dependency parses in their rule-based system, noting that complex sentence structure is difficult to learn automatically. A similar approach is taken by Mehrabi et al. (2015). In both of these experiments, a set of rules was created specifically for the purposes of the project, and therefore the systems cannot necessarily be directly extended to work on other datasets. Furthermore, ad hoc rules are likely to be incomplete with respect to the variety of ways to express negation and the complexity of syntactic structures over which negation can scope.

Packard et al. (2014) used the English Resource Grammar (ERG; Flickinger, 2000, 2011) for negation scope detection in literary texts, to outperform the state-of-the-art at the time. In the biomedical domain, the ERG was used by MacKinlay et al. (2012). In one task described in their paper (Task 3), MacKinlay et al. (2012) constructed feature vectors for specific "events", which were already identified by annotations in their training data. They used the Minimal Recursion Semantics (MRS; Copestake et al., 2005) and Robust Minimal Recursion Semantics (RMRS; Copestake, 2003) representations produced by the ERG to identify speculation and negation over those events. Their results for negation performed best in the BIONLP 2009 shared task.

Our study differs from MacKinlay et al. (2012) in one key way: whereas they look only at events of interest (which they assume to be already identified in the data), we look at the entire text of a given pathology report, without relying on sentence-level annotation of any kind. In other words, we hypothesize that the usefulness of incorporating precise grammatical information about negation scope generalizes to the document level. Other differences include that MacKinlay et al. (2012) looked at scientific biomedical literature, while we use clinical reports as our data.

3 Experiment

The main contribution of the paper is the use of a broad coverage grammar (the ERG) for feature extraction on the level of a complete pathology report. We describe the dataset in section 3.1. The feature extraction technique is explained in detail in sections 3.3-3.5. To evaluate, we perform feature selection (section 3.2) and use four popular ensemble classifiers (section 4.1).

3.1 Data

The dataset (Table 1) comes from the Surveillance, Epidemiology and End Results (SEER) Program[6] and is a subset of 4000 randomly selected, de-identified annotated pathology reports, representing 4 registries and more than 70 pathology labs. This subset was used in a graduate level course[7] on Natural Language Processing in Cancer Informatics at the University of Washington .[8] It is annotated for a variety of things, including whether the primary tumor was found in the right or left part of the body (such as right vs. left breast or lung). It is not annotated for negation. A small number of reports were classified to have both right and left sites affected; we exclude them from the experiment in order to have a balanced dataset.

We relied primarily on the NLTK sentence tokenizer (Bird et al., 2009) to split pathology note sections into individual sentence tokens. Subtopic headers (identified by their line-ending colon punctuation) were extracted into separate sentence tokens, and mid-sentence line breaks were removed from sentence tokens. The dataset was divided into training and test randomly, using a 66% / 33% split (two thirds used for the training).

[6] https://seer.cancer.gov/

[7] LING575, Winter 2017

[8] Unfortunately, this particular dataset is not available for public use for reasons outside of authors' control. Applying the system to a public dataset remains part of future work.

	Total	LAT	RIGHT	LEFT
Training	581	446	234	212
Test	293	233	122	111

Table 1: Dataset size. LAT are reports gold-annotated for right or left laterality.

The dataset partially motivated the choice of right vs. left laterality as our target annotation: On the one hand, classifying documents for the location of the primary tumor is a useful task that could be performed by a machine; on the other, it was one of the gold annotations occurring most often in the reports. Filtering all reports for this annotation yielded a fairly large and balanced subset of documents (Table 1).

3.2 Features

We represented each report as a feature vector as follows. Features were extracted sentence by sentence and then aggregated into one set for the entire report. First we extracted all concepts for each report (as described in section 3.2.1), and then selected 44 features using recursive feature elimination with Random Forest[9] on 33% of the training dataset set and supplementing that set symmetrically, e.g. adding 'left upper lobe' where we only had 'right upper lobe'. The resulting features were: 'right breast', 'right lower lobe', 'right lung', 'right upper lobe','left breast', 'left lower lobe', 'left lung', 'left upper lobe', 'left', 'right', 'rul', 'lul', 'rll', 'lll', 'identified', 'carcinoma', 'adenocarcinoma', 'malignancy', 'malignant', 'tumor', 'in situ', 'invasive', 'infiltration', 'atypic', 'atypia', 'margin', 'rt breast', 'lt breast', 'rt lung', 'lt lung', 'rt upper lobe','lt upper lobe', 'rt lower lobe', 'lt lower lobe', 'metastasis', 'metastatic', 'found', 'suspicious', 'evidence', 'masses', 'tissue', 'specimen', 'determined', 'ruled out'. We supplement the set manually because one third of the training set used to select features was not sufficient to produce a list of features that robustly extended to the training set. For example, some of the training reports did not have the feature *right upper lobe* but we noticed that they have *left upper lobe* instead and decided to apply a symmetric manual extension to the feature set across the board. This way we hope to obtain informative features while not overfitting to the training dataset.

3.2.1 Basic Features

We constructed basic feature vectors for each report by using MetaMap Lite to recognize concepts from the UMLS term database (Aronson, 2001). Each sentence was parsed to find the longest matching UMLS concepts and phrases, using the default MetaMap Lite setting for both stop words and POS tagging. A report feature vector is the set of all the features listed above in section 3.2 that were extracted by MetaMap for each sentence in this report. We only count each feature once per report.

3.3 Negation Scope and Negated Features

We compare two different methods of extracting negated features and their effect on classification: NegEx (Chapman et al., 2001) which we consider our baseline, and the English Resource Grammar (ERG; Flickinger, 2000, 2011).[10] If a concept was marked as negated by the tool under consideration, we remove the original feature from the feature list extracted from the sentence and replace it with the negated feature (e.g. *NEG:tumor*, for a sentence like *No tumor identified*). Of course, if the same feature was detected but not marked as negated in a different sentence in the same report, the report will still have the non-negated version of the feature in its vector.

Table 2 shows the total number of feature tokens,[11] for all sentences in the dataset, and separately the number of feature tokens added by NegEx and the ERG. The ERG adds fewer features, which means it detects negation in fewer occurrences than NegEx. Later in the paper, we discuss that while sometimes

[9] We took all the top scoring features which looked meaningful, namely stopping once we saw the word 'Dr.'.

[10] Version 1214.

[11] We use the classic token/type distinction to talk about the features. For example, the sentence *Signs of malignancy in left upper lobe; no signs of malignancy in right upper lobe* contains two tokens of the feature type *lobe* and just one token of the feature type *right upper lobe*.

	total	+NegEx	+ERG
total	25159	3214	3002

Table 2: Feature tokens (features occurrences in each sentence).

it is detrimental to the evaluation results, other times features added by NegEx turn out to be noise, and skipping them is actually more precise and beneficial.

3.4 NegEx

NegEx (Chapman et al., 2001) takes text (assuming that the whole text is one sentence) as input and returns a list of concepts that it considers negated in that text. It attempts to determine what the scope of negation is, but sometimes makes mistakes. For example, in the sentence *No chest pain, no shortness of breath, and no abdominal pain*, the concept *chest pain* is affirmed (not negated) by NegEx.[12] (We will see a similar error hurting NegEx's performance in section 4.3.3.) While any specific error can be fixed, it can be difficult to fix such issues robustly and not cause regressions without relying on a general set of linguistically motivated language rules. When this method of negation detection was applied to feature extraction, the list of feature types was appended with *NEG:in-situ*, *NEG:left breast*, *NEG:left*, *NEG:identified*, *NEG:carcinoma*, *NEG:malignancy*, *NEG:margin*, *NEG:tumor*, *NEG:right*, *NEG:invasive*.

3.5 ERG

3.5.1 Overview

The English Resource Grammar (ERG; Flickinger, 2000, 2011) is a large, broad-coverage precision grammar of English.[13] The term 'precision' means that it encodes syntactic and lexical rules in a form that aims for linguistic adequacy and generality, in this case using the Head Driven Phrase Structure Grammar theory of syntax (HPSG; Pollard and Sag, 1994). The grammar maps surface strings to syntactic as well as semantic representations which are licensed by lexical and phrase structure rules. The ERG is supported by the DELPH-IN research consortium, along with a variety of tools for parsing, generation, and representing syntactic and semantic structures.[14] It is the largest HPSG-based grammar, but grammars of different sizes exist for other languages.[15]

3.5.2 Minimal Recursion Semantics

The ERG produces semantic representations compositionally in the format of Minimal Recursion Semantics (Copestake et al., 2005), which we then use to extract negated features. Figure 1 shows a syntactic structure (left) and an MRS (right) for the fragment sentence *No evidence of malignancy*. The syntactic representation is a familiar constituency tree. The flat MRS representation is a bag of quantifiers, relations, and predicates associated with handles (such as *h13*), which can be identified with each other or not, allowing for scope underspecification.[16] In this example, *evidence* is analyzed is being in the scope of the *no* quantifier. The word *malignancy* is analyzed as the first argument of *evidence* (through the indefinite quantifier, unexpressed on the surface level). The predicate of this sentence is *unknown*, reflecting that it is a fragment.

A dependency MRS for the sentence *A tumor was not identified* is shown in Figure 2. DMRS representations (Copestake, 2009; Copestake et al., 2016) are essentially a simplified version of MRS where variables and their relationships are expressed as links. DMRS representations consist of elementary predications (EPs) corresponding to surface tokens, such as the EP for negation, as well as abstract EPs contributed by grammatical rules. The EPs are connected by links, forming a semantic dependency

[12] We used the version downloaded from `https://github.com/chapmanbe/negex`.

[13] Demo: `http://erg.delph-in.net/logon`.

[14] To follow up on the earlier example of NegEx making a mistake with the sentence *No chest pain, no shortness of breath, and no abdominal pain*, we verified that the ERG returns a syntactic structure which has all three nouns in scope of negation.

[15] See e.g. `http://www.delph-in.net/wiki/index.php/Grammars`.

[16] See Copestake et al. (2005) for a discussion and examples of scope underspecification and other issues related to MRS.

graph.[17] DMRS representations are related to MRS (a DMRS can be created from an MRS deterministically) but are easier to look at and to work with, and for our experiment we used the DMRS representations.

One advantage of semantic representations like MRS or DMRS is that they abstract away from the syntactic notions of subjects and objects, normalizing active/passive pairs to the same representation. This is illustrated in Figure 2, where *tumor* is the ARG2 (the theme) of *identify*, despite being the grammatical subject of a passive sentence. Another advantage is a systematic treatment of scope, which is discussed in the next section.

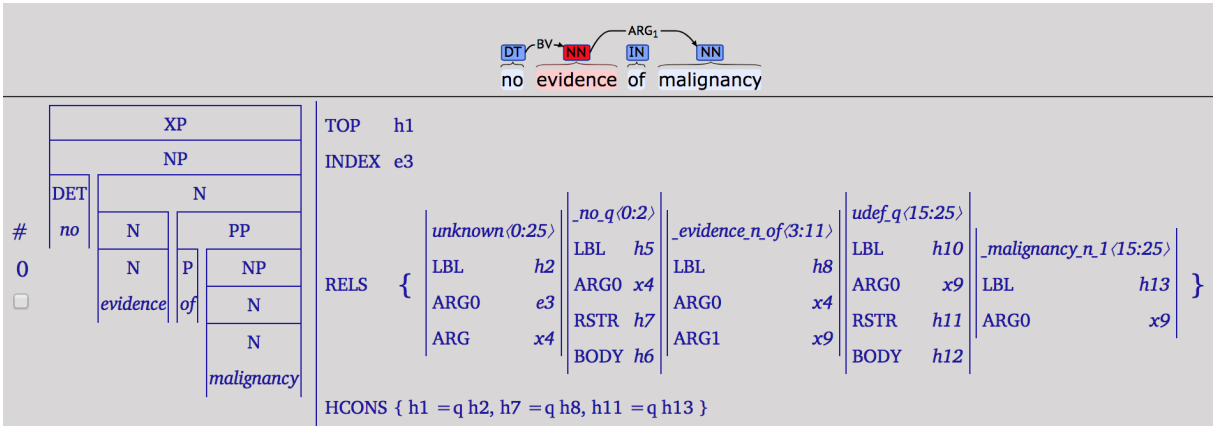

Figure 1: The ERG's syntactic parse and the MRS of the fragment sentence: *No evidence of malignancy*

3.5.3 Extracting Features with the ERG

We parsed every sentence in each report with the ERG loaded into the ACE parser (Crysmann and Packard, 2012).[18] ACE has a default parse ranking procedure trained on a treebank (Toutanova et al., 2005), and we selected the top ranked parse. The coverage of the parser was 22858/28788 sentences (79%). It is not surprising that not all sentences are parsed, given the highly fragmented nature of the reports, where newlines are often used to separate different parts of the same sentence. Statistical parsers always return a parse; however, we will show that when striving for precision, no parse can be better than a nonsensical parse.[19] Once we parsed each sentence using the ERG, we crawled the dependency (DMRS) representations to extract negated concepts. We extracted two types of negation: predicate negation (the negation of events) and quantifier negation (the negation of entities).

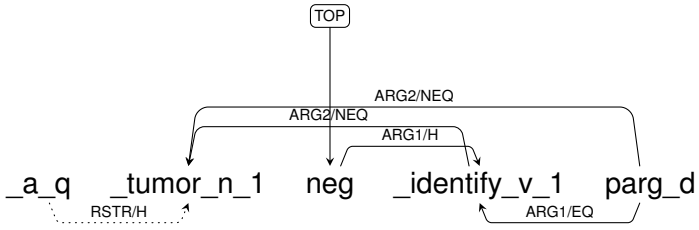

Figure 2: A semantic dependency graph (DMRS) produced by the ERG for: *A tumor was not identified.*

Predicate negation is represented with the EP *neg* (the semantic contribution of *not*) which scopes over the predications contributed by the verb (*neg*'s ARG1) and the verb's own arguments. From the DMRS we extract the verb as well as its argument labeled ARG2. This label refers to the argument receiving the action of the verb, or the semantic theme. For example, in *Skin: invasive carcinoma does not invade dermis or epidermis*, we extract the verb *invade* and its ARG2 which is the the coordinated

[17] The DMRS graphs were produced using `https://github.com/delph-in/delphin-viz`.

[18] Version 0.9.24.

[19] That said, a more sophisticated tokenization technique, which would reconstruct more full sentences, would improve ACE's coverage.

noun phrase *dermis or epidermis*.[20] Extracting the ARG2 yields the same result in passive sentences, such as *Additional distinct breast masses or lesions are not grossly identified*, where the 'things not identified' are the *additional distinct breast masses or lesions*.

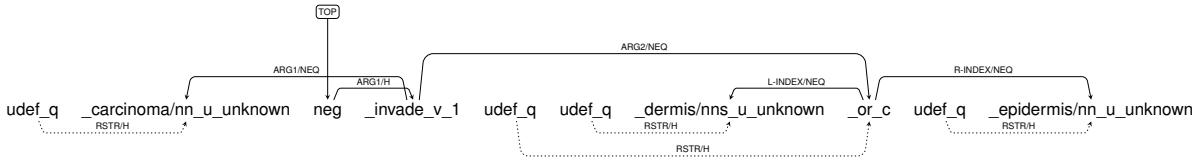

Figure 3: DMRS for the sentence *Carcinoma does not invade dermis or epidermis*

Quantifier negation is the negation of a noun phrase by the quantifier *no*, as in *There is no evidence of vasculitis, granulomatous inflammation, or neoplasm*. First we identify the predications restricted by the quantifier *no*. Here the restricted predication is the noun phrase headed by *evidence*, which itself contains a prepositional phrase with coordinated noun phrases. We identify each concept in the noun phrase and mark it as negated.

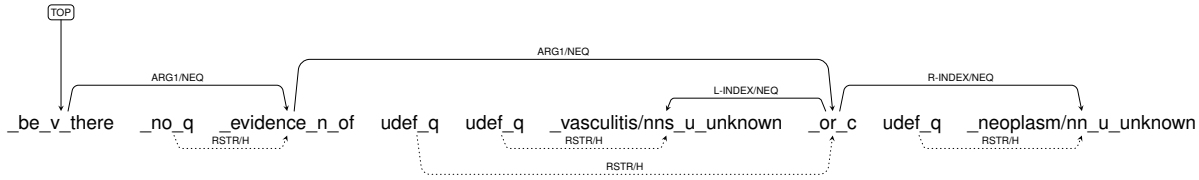

Figure 4: DMRS for the sentence *There is no evidence of vasculitis or neoplasm*

When this negation detection method is applied to feature extraction, the original list of features is appended by *NEG:tumor*, *NEG:identified*, *NEG:carcinoma*, *NEG:malignancy*, *NEG:margin*, *NEG:in situ*, *NEG:evidence*. Compared to NegEx, this is fewer feature types as well as fewer feature tokens added. As we will see, this may lead to higher precision in classification. On the other hand, the lack of highly informative concepts related directly to *left* and *right* (such as *left lung*) indicates we could get better results if we supplied the ERG with more data that it can successfully parse. There are certainly sentences in the data in which the concepts like this are within the scope of negation. As we will see, NegEx sometimes gets it right and other times it does not. The reason the ERG does not detect them is usually that it was not supplied a well-formed sentence due to tokenization issues.

4 Results and Error Analysis

4.1 Evaluation by Classification

To test our feature extraction technique, we used four popular ensemble algorithms which generally perform well on small datasets, taking advantage of the scikit-learn Python library (Pedregosa et al., 2011). The parameters were chosen by cross-validation on a small development set.

classifier	code	cite	# estimators	learn. rate	max. depth	other
AdaBoost	AB	Freund and Schapire (1995)	50	1	-	SAMME.R
Random Forest	RF	Breiman (2001)	100	-	20	-
Gradient Boost	GB	Friedman (2001)	100	0.1	1	-
Voting classifier	VC	Pedregosa et al. (2011)	-	-	-	hard, no weights

Table 3: Classifiers used for evaluation

4.2 Numeric Results

Table 4 shows the results. The first column is the ensemble classifier code (4.1); *right* and *left* are laterality classes. Incorporating information about negated concepts is beneficial in many cases; the

[20] An additional trivial step is required to extract *dermis* and *epidermins* as separate concepts

CL	NE					ERG				
	precision		recall		micro-avg F1	precision		recall		micro-avg F1
	right	left	right	left		right	left	right	left	
AB	0.9754	0.9459	0.9520	0.9722	0.9614	**0.9917**	**0.9554**	**0.9600**	**0.9907**	**0.9742**
RF	0.9760	**0.9722**	**0.9760**	0.9722	**0.9742**	**0.9835**	0.9464	0.9520	**0.9815**	0.9657
GB	0.9835	0.9464	0.9520	0.9815	0.9657	**0.9836**	**0.9550**	**0.9600**	0.9815	**0.9700**
VC	0.9756	0.9600	0.9545	0.9722	0.9657	**0.9917**	0.9600	**0.9554**	**0.9907**	**0.9742**

Table 4: Classification results with four different classification algorithms; NegEx vs. ERG.

improvement is small but fairly robust across different ensemble algorithms. The only algorithm where NegEx-detected negation was better than the ERG is Random Forest, and even for Random Forest, the precision is higher with the ERG for one of the classes and the recall is higher for the other.

4.3 Improvement and Error Analysis

Below we will look at specific improvements and regressions, per record. The record ID has been changed for anonymity concerns. We primarily see differences in classification with respect to 7 records; we will call them Records 1-7.

4.3.1 AdaBoost

Improvement: Records 1,2,3. The feature vectors for **Records 1 and 3** do not differ between the ERG and NegEx; Record 1 does not contain any negated features in either case, while Record 3 contains a feature negated by both the ERG and NegEx. This means the improvement is probably due to chance (a tie broken by the classifier in such a way that it shows as an improvement). **Record 2**, however, shows some differences. The features *identified* and *metastatic* were negated by NegEx but not by the ERG. These features and their negated counterparts may have a lot of weight and so an incorrect decision about negation can have impact. It turns out, NegEx uses a fairly big chunk of text with no sentence-final punctuation which happens to start from *Not indentified* to produce multiple negated features. The ERG is not able to parse that chunk of text as a sentence (since it really is not one). In this case, better precision leads to a better result.
Regression: None.

4.3.2 Random Forest

Improvement: Record 1, again can be attributed to chance.
Regression: Records 4, 5, 6. In **Record 4**, the feature *identified* is correctly negated by NegEx but not by the ERG. The source sentence has the form *No X or Y is identified*. The ERG only succeeds in negating X and Y but not the predicate *identified*. We discuss this issue in section 4.4. **Record 5** contains a phrase *negative for malignancy*, for which NegEx has a suitable heuristic. Our ERG algorithm is not sensitive to the word *negative*. We purposefully did not enhance our algorithm with any heuristics since we wanted our approach to be as domain-independent as possible. It makes sense that NegEx beats us here, however one direction for improvement is to map some of NegEx's heuristics to the ERG's semantic relations. NegEx negates multiple features in **Record 6**, including both *right* and *left*, *left breast*, (but not *right breast*) *identified*, *carcinoma*, *tumor*. The ERG does not negate any of them in this case. The correct label for this record's laterality is *right*, so NegEx probably wins by negating many of the features associated with *left*. However, it is hard to say why it chooses to do so; for example, it negates *left breast* in the sentence: *Left breast: According to the ultrasound core biopsy protocol data sheet, mammogram demonstrates a 2.0 cm irregular mass, and correlates as palpated.* This Record does not provide us with any insight with respect to the two systems' different behavior.

4.3.3 Gradient Boosting

Improvement: Record 7. For this record, the difference is that the ERG negates the features *evidence* and *tumor* while NegEx does not. In this case, this leads to a better result. The relevant text in the record is *no evidence of ductal carcinoma in situ*. NegEx detects negation for *ductal carcinoma in situ*, but not for *evidence*.

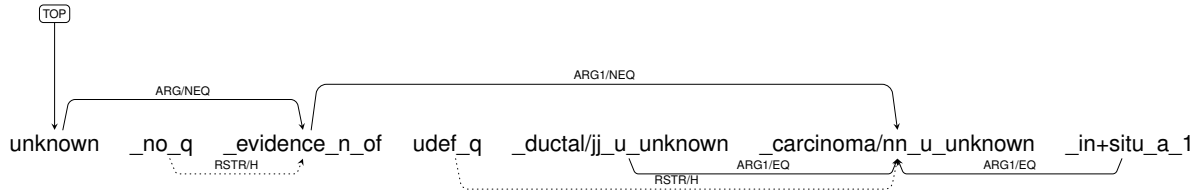

Figure 5: DMRS for *No evidence of ductal carcinoma in situ*

Regression: None.

4.3.4 Voting Classifier

Improvement: Records 1, 2, already analyzed for AdaBoost.
Regression: None.

4.4 Summary

The Improvement and error analysis revealed the following differences between the NegEx and the ERG approaches to feature extraction.

1. The two systems are not equally sensitive to **tokenization issues**. Clinical notes contain lots of tabs in place of sentence-final punctuation; that may lead to imprecise tokenization. NegEx does not care about whether its input actually is a sentence, while the ERG does; our ERG algorithm thus produces higher precision results in some cases where NegEx can negate a feature based on a trigger which does not actually belong to the sentence that contains this feature.

2. NegEx employs a variety of **domain-specific heuristics** while our ERG algorithm does not. We would like our approach to not include dataset or domain-specific heuristics, but some of the heuristics could be mapped to the ERG's existing semantic relations such as specific English quantifiers. We could take advantage of using a robust and precise domain-independent resource and apply a heuristic at a higher level, such as parse selection.

3. ACE **parse selection**. The treebanks used to train the ACE parse ranker are not necessarily representative of the data found in pathology reports. Therefore, it was possible for the desired parse to be produced by the ERG, but not be selected. In some cases, the top ranked parse did not exhibit the widest scope interpretation. For example, the top ranked parse for *no masses or previous biopsy sites are identified* only scoped *no* over *masses*, and not the entire coordinated phrase. We also find

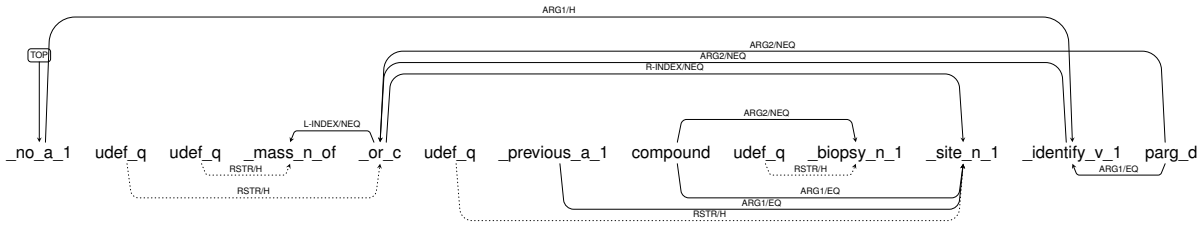

Figure 6: DMRS showing the desired negation scope for *No masses or previous biopsy sites are identified*; this DMRS was not the top-ranked parse

some inconsistency in the top ranked parse for constructions like *associated ductal carcinoma in situ: not identified*. This 'subject: adjective' construction was probably not common in the dataset used to train the parse selection model distributed with the ERG, version 1214, but is frequent in our dataset. In some sentences like the one above (illustrated also by Figure 8), *not* is parsed as a conjunction, or the phrase after the colon is treated as a modifier (effectively treating *identified* as a post-head adjective) rather than as a predicate (negated verb). In these cases, our algorithm, which looks for a *neg* predication and a verb with a theme, fails to negate *identified*. Customized parse

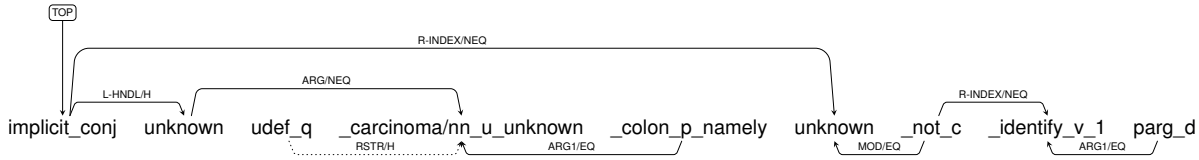

Figure 7: Abridged DMRS showing modifier analysis for *not* associated with some top-ranked parses

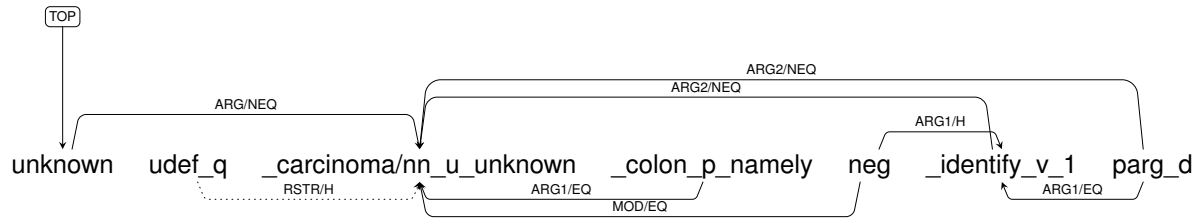

Figure 8: DMRS showing desired, predicate negation analysis for *not*

selection that prefers verbal predicates over modifiers and prefers the widest scope may help extract the features even better and further improve classification.

4. Finally, in some cases it is simply not clear why one approach wins over the other. This includes situations where there is a tie or when there is some difference between the feature vectors produced by the two approaches, but the difference seems to be due to a bug in NegEx (nonetheless leading to NegEx producing a better feature vector). We hope that we will gain more insight once we switch to a larger dataset.

5 Conclusion

We show as a proof of concept that a precision grammar can help extract potentially informative negated features for a pathology reports classification task, outperforming NegEx across several ensemble classification algorithms. Improvement and error analysis reveals that the ERG, being a *precision* grammar, tends to incorporate fewer noisy features compared to NegEx. Because it looks for a meaningful structure in what it expects to be a well-formed sentence, it will make fewer mistakes in terms of negation scope. At the same time, at this stage we did not go beyond simple predicate negation and did not customize parse selection, which causes our system to make some mistakes that NegEx, which includes multiple heuristics specialized for the domain, does not make. In future work, we will experiment with ways to customize parse selection so that it is better suited for the clinical notes domain, while not needing to change anything in the grammar itself. We will also explore which relations in the ERG may correspond to some of NegEx's heuristics, and this should give us further improvement. Furthermore, in any next stages we will be applying our system to a public dataset such as BioScope (Vincze et al., 2008).

Acknowledgements

This project would not be possible without Emily Silgard from Fred Hutch who provided us with the data and guidance for this project. We thank Michael W. Goodman and Ned Letcher for their delphin-viz tool that made it painless to include the DMRS representations.

References

Alan R Aronson. 2001. Effective mapping of biomedical text to the UMLS metathesaurus: the MetaMap program. In *Proceedings of the AMIA Symposium*, page 17. American Medical Informatics Association.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: Analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Leo Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

Wendy W Chapman, Will Bridewell, Paul Hanbury, Gregory F Cooper, and Bruce G Buchanan. 2001. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of biomedical informatics*, 34(5):301–310.

Ann Copestake. 2003. Report on the design of RMRS. *DeepThought project deliverable*.

Ann Copestake. 2009. Slacker semantics: Why superficiality, dependency and avoidance of commitment can be the right way to go. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1–9. Association for Computational Linguistics.

Ann Copestake, Dan Flickinger, Carl Pollard, and Ivan A Sag. 2005. Minimal recursion semantics: An introduction. *Research on Language and Computation*, 3(2-3):281–332.

Ann Copestake, Guy Emerson, Michael Wayne Goodman, Matic Horvat, Alexander Kuhnle, and Ewa Muszyńska. May 2016. Resources for building applications with dependency minimal recursion semantics. In *Proceedings of 10th International Conference on Language Resources and Evaluation*. URL http://www.lrec-conf.org/proceedings/lrec2016/pdf/634_Paper.pdf.

Berthold Crysmann and Woodley Packard. 2012. Towards efficient HPSG generation for German, a non-configurational language. In *COLING*, pages 695–710.

Dan Flickinger. 2000. On building a more effcient grammar by exploiting types. *Natural Language Engineering*, 6(01):15–28.

Dan Flickinger. 2011. Accuracy vs. robustness in grammar engineering. *Language from a cognitive perspective: Grammar, usage, and processing*, pages 31–50.

Yoav Freund and Robert E Schapire. 1995. A desicion-theoretic generalization of on-line learning and an application to boosting. In *European conference on computational learning theory*, pages 23–37. Springer.

Jerome H Friedman. 2001. Greedy function approximation: A gradient boosting machine. *Annals of statistics*, pages 1189–1232.

Andrew MacKinlay, David Martinez, and Timothy Baldwin. 2012. Detecting modification of biomedical events using a deep parsing approach. *BMC medical informatics and decision making*, 12(1):S4.

Saeed Mehrabi, Anand Krishnan, Sunghwan Sohn, Alexandra M Roch, Heidi Schmidt, Joe Kesterson, Chris Beesley, Paul Dexter, C Max Schmidt, Hongfang Liu, et al. 2015. DEEPEN: A negation detection system for clinical text incorporating dependency relation into negex. *Journal of biomedical informatics*, 54:213–219.

Roser Morante and Walter Daelemans. 2009. A metalearning approach to processing the scope of negation. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, pages 21–29. Association for Computational Linguistics. URL http://aclweb.org/anthology/W09-1105.

Woodley Packard, M. Emily Bender, Jonathon Read, Stephan Oepen, and Rebecca Dridan. 2014. Simple negation scope resolution through deep parsing: A semantic solution to a semantic problem. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 69–78. Association for Computational Linguistics. doi: 10.3115/v1/P14-1007. URL http://aclweb.org/anthology/P14-1007.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830.

Carl Pollard and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. Studies in Contemporary Linguistics. The University of Chicago Press and CSLI Publications, Chicago, IL and Stanford, CA.

Guergana K Savova, James J Masanz, Philip V Ogren, Jiaping Zheng, Sunghwan Sohn, Karin C Kipper-Schuler, and Christopher G Chute. 2010. Mayo clinical text analysis and knowledge extraction system

(cTAKES): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association*, 17(5):507–513.

Sunghwan Sohn, Stephen Wu, and Christopher G Chute. 2012. Dependency parser-based negation detection in clinical narratives. *AMIA Summits on Translational Science proceedings AMIA Summit on Translational Science*, 2012:1–8.

Kristina Toutanova, Christopher D Manning, Dan Flickinger, and Stephan Oepen. 2005. Stochastic HPSG parse disambiguation using the Redwoods corpus. *Research on Language & Computation*, 3 (1):83–105.

Veronika Vincze, György Szarvas, Richárd Farkas, György Móra, and János Csirik. 2008. The BioScope corpus: biomedical texts annotated for uncertainty, negation and their scopes. *BMC bioinformatics*, 9 (11):S9.