Learning with Structured Representations for Negation Scope Extraction

Hao Li and **Wei Lu**

Singapore University of Technology and Design
8 Somapah Road, Singapore, 487372
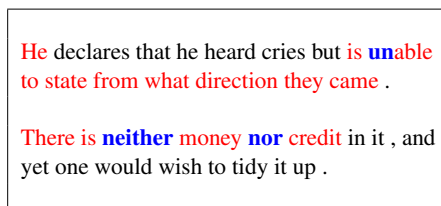`hao_li@mymail.sutd.edu.sg`, `luwei@sutd.edu.sg`

Abstract

We report an empirical study on the task of negation scope extraction given the negation cue. Our key observation is that certain useful information such as features related to negation cue, long distance dependencies as well as some latent structural information can be exploited for such a task. We design approaches based on conditional random fields (CRF), semi-Markov CRF, as well as latent-variable CRF models to capture such information. Extensive experiments on several standard datasets demonstrate that our approaches are able to achieve better results than existing approaches reported in the literature.

1 Introduction

Negation is an important linguistic phenomenon (Morante and Sporleder, 2012), which reverts the assertion associated with a proposition. Broadly speaking, the part of the sentence being negated is called *negation scope* (Huddleston et al., 2002). Automatic negation scope detection is a vital but challenging task that has various applications in areas such as text mining (Szarvas et al., 2008), and sentiment analysis (Wiegand et al., 2010; Councill et al., 2010). Negation scope detection task commonly involves a negation cue which can be one of the following 3 types – either a single word (e.g., *not*), affixes (e.g., *im-, -less*) or multiple words (e.g., *no longer*) expressing negation. Figure 1 presents two real examples for such a task, where the first example involves discontinuous negation scope of an affix cue. The second example shows a discontinuous negation cue and its corresponding discontinuous negation scope.

Most existing approaches tackled the negation scope detection problem from a boundary detec-

He declares that he heard cries but is **un**able to state from what direction they came .

There is **neither** money **nor** credit in it , and yet one would wish to tidy it up .

Figure 1: Two examples with negation cues in bold blue and negation scope in red.

tion perspective, aiming to identify whether each word token in the sentence belongs to the negation scope or not. To perform sequence labeling, various approaches have been proposed based on models such as support vector machines (SVMs) (with heuristic rules) (Read et al., 2012; de Albornoz et al., 2012; Packard et al., 2014), conditional random fields (CRF) (Lapponi et al., 2012; Chowdhury and Mahbub, 2012; White, 2012; Zou et al., 2015) and neural networks (Fancellu et al., 2016; Qian et al., 2016). These models typically either make use of external resources for extracting complex syntax and grammar features, or are based on neural architectures such as long short-term memory networks (LSTM) and convolutional neural networks (CNNs) to extract automatic features.

We observe that there are some useful features that can be explicitly and implicitly captured and modelled in the learning process for negation scope extraction. We use the term partial scope to refer to a continuous text span that is part of discontinuous scope, and use the term gap to refer to the text span between two pieces of partial scope.

From the first example in Figure 1 we can observe that, with the negation cue as a prefix in a word, the partial scope *before*, *after* and *in the middle* of the negation cue differ in terms of composition of words and their associated syntactic roles in the sentence. Furthermore, the type of cue
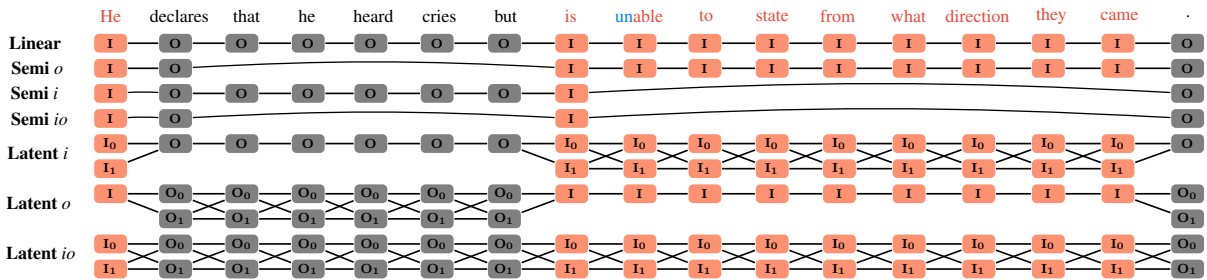
Figure 2: Label assignments of model variants for the first example mentioned in Figure 1.

as we mentioned earlier may also reveal crucial information for this task. Moreover, two pieces of partial scope separated by a gap might have some long distance dependencies. For instance, in the first sentence of Figure 1, "*He*" as the first partial scope is the subject phrase of the token "*is*" which is the first word of the second partial scope with a long gap in between.

Similarly, the second example shows that a discontinuous negation cue involves multiple words, *neither* and *nor*, which shows the importance of cue features and some long distance dependencies among text spans.

Furthermore, besides explicit features that we are able to define, we believe there exist some implicit linguistic patterns within the scope for a given negation cue. While it is possible to manually design linguistic features to extract such patterns, approaches that can automatically capture such implicit patterns in a domain and language independent manner can be more attractive. How to design models that can effectively capture such features mentioned above remains a research question to be answered.

In the paper, we design different models to capture such useful features based on the above motivations, and report our empirical findings through extensive experiments. We release our code at http://statnlp.org/research/st.

2 Approaches

Based on the observations described earlier, we aim to capture three types of features by designing different models based on CRF.

Negation Cue

The linear CRF (Lafferty et al., 2001) model (which we refer to as **Linear** in this paper), as illustrated in Figure 2 , is used to capture negation cue related features. The probability of predicting a possible output $\mathbf{y}$, the label sequence capturing negation scope, given an input sentence $\mathbf{x}$ is:

$$p(\mathbf{y}|\mathbf{x}) = \frac{\exp\left(\mathbf{w}^T\mathbf{f}(\mathbf{x}, \mathbf{y})\right)}{\sum_{\mathbf{y}' \in \mathcal{Y}(\mathbf{x})} \exp(\mathbf{w}^T\mathbf{f}(\mathbf{x}, \mathbf{y}'))}$$

where $\mathbf{f}(\mathbf{x}, \mathbf{y})$ is a feature function defined over the $(\mathbf{x}, \mathbf{y})$ pair. The negation cue related features mainly involve cue type, position of the cue, as well as relative positions of each partial scope. For example, the cue type refers to the string form of the cue, which could be a single word, an affix (prefix or suffix), or a multi-word expression.

We follow a standard approach to assign tags to words. Specifically, **O** and **I** are used to indicate whether a word appears outside or inside the negation scope respectively.

Long Distance Dependencies

We use the semi-CRF (Sarawagi and Cohen, 2004) model (referred to as **Semi**) to capture long distance dependencies. The semi-CRF is an extension to the linear-CRF. The difference is that the output $\mathbf{y}$ may not be a sequence of individual words. Rather, it is now a sequence of spans, where each span consists of one or more words. The semi-CRF model is more expressive than the linear-CRF model as such a model is able to capture features that are defined at the span level, allowing longer-range dependencies to be captured.

Since the **Semi** approach is capable of modeling a span (which can be a gap or partial scope), we are able to model the features between two separate text spans. We propose three variants for the **Semi** model, as illustrated in Figure 2, to capture different types of long distance features. The **Semi** *i* model regards a piece of partial scope as a span in order to capture long distance dependencies between two gaps. The **Semi** *o* model treats a gap as a span to capture long distance dependencies between two pieces of partial scope. The **Semi** *io*

model regards both partial scope and gaps as spans to capture both types of long distance dependencies mentioned above.

Implicit Patterns

The latent variable CRF model, denoted as **Latent**, is used to model implicit patterns. The probability of predicting a possible output $\mathbf{y}$, which is the label sequence capturing negation scope information, given an input sentence $\mathbf{x}$ is defined as:

$$p(\mathbf{y}|\mathbf{x}) = \frac{\sum_{\mathbf{h}} \exp\left(\mathbf{w}^T \mathbf{f}(\mathbf{x}, \mathbf{y}, \mathbf{h})\right)}{\sum_{\mathbf{y}' \in \mathcal{Y}(\mathbf{x})} \sum_{\mathbf{h}'} \exp(\mathbf{w}^T \mathbf{f}(\mathbf{x}, \mathbf{y}', \mathbf{h}'))}$$

where $\mathbf{h}$ is a latent variable encoding the implicit pattern.

We believe such latent pattern information can be learned from data without any linguistic guidance. The **Latent** model is capable of capturing this type of implicit signals. For example, as illustrated as **Latent** *io* in Figure 2, each position has $\mathbf{O_0}$, $\mathbf{O_1}$, $\mathbf{I_0}$ and $\mathbf{I_1}$ as latent tags. This way, we can construct features of forms such as "*He/*$\mathbf{O}_i$-*declares/*$\mathbf{I}_j$" that capture the underlying interactions between the words and the latent tag patterns.

In order to investigate the relation between latent variables and tags, we proposed another two latent models. The **Latent** *i* only considers latent variables on $\mathbf{I}$ tags (for partial scope), while **Latent** *o* only takes latent variables on $\mathbf{O}$ tags (for gaps).

3 Experimental Setup

CDS-CO	Train	Dev	Test
#Sentence	847	144	235
#Instance	983	173	264

Table 1: Statistics of the *CDS-CO* corpus.

We mainly conducted our experiments on the *CDS-CO* corpus released from the *SEM2012 shared task (Morante and Blanco, 2012). The negation cue and corresponding negation scope are annotated. For each word token, the corresponding POS tag and the syntax tree information are provided. If the sentence contains multiple negation cues, each of them is annotated separately. The corpus statistics is listed in Table 1. During training and testing, following prior works (Fancellu et al., 2016), only instances with at least one negation cue will be selected. For the sentence containing multiple negation cues, we create as many copies as the number of instances,

System	Token-Level			Scope-Level (Exact Scope Match)					
	P.	R.	F_1	$P_A.$	$R_A.$	F_{1A}	$P_B.$	$R_B.$	F_{1B}
Read et al. (2012)	-	-	-	98.8	64.3	77.9	-	-	-
Packard et al. (2014)	86.1	**90.4**	88.2	98.8	65.5	78.7	-	-	-
Fancellu et al. (2016)	92.6	85.1	88.7	99.4	63.9	77.8	-	-	-
Linear (-c -r)	84.7	73.9	78.6	99.2	49.4	65.6	51.5	49.4	50.4
Linear (-r)	90.6	78.4	84.1	100	60.6	75.5	61.4	60.6	61.0
Linear (-c)	91.0	78.9	84.5	99.3	56.6	72.1	60.0	56.6	58.3
Linear	94.4	82.6	88.1	100	67.9	80.9	69.3	67.9	68.6
Semi *i*	95.0	84.1	89.2	100	67.5	80.6	69.4	67.5	68.4
Semi *o*	94.0	85.3	**89.4**	100	69.1	81.7	**71.1**	69.1	**70.1**
Semi *io*	94.5	84.1	89.0	100	68.3	81.2	70.3	68.3	69.3
Latent *i*	94.4	83.4	88.6	99.4	67.9	80.7	69.6	67.9	68.7
Latent *o*	90.4	83.9	87.1	99.4	65.5	78.9	66.3	65.5	65.9
Latent *io*	**94.8**	83.2	88.6	100	**69.5**	**82.0**	70.6	**69.5**	70.0

Table 2: Main results on *CDS-CO* data

each of which has only one negation cue and its corresponding negation scope.

The L_2 regularization hyper-parameter λ is set to 0.1 based on the development set. We conduct evaluations of negation scope extraction based on metrics at token-level evaluations and scope-level evaluations. There are two versions of evaluation metrics, referred to as version A and version B[1], defined at the scope-level that can be used to measure the performance according to *SEM2012 shared task (Morante and Blanco, 2012).

Moreover, to understand the model robustness, we also conducted additional experiments on Bio-Scope (Szarvas et al., 2008) and CNeSP (Zou et al., 2015).

4 Results and Discussion

4.1 Main Results

The main results on the *CDS-CO* corpus are shown in Table 2. $P_A.$, $R_A.$ and $F_{1A}.$ are precision, recall and F_1 measure under version A, while $P_B.$, $R_B.$, $F_{1B}.$ are for version B. Note that none of the prior works reported results under version B. Moreover, *c* refers to the cue type features, *r* refers to relative position of partial scope with respect to the cue.

We focus on **Linear** models first, where **Linear** (-c -r) is the baseline without features *c* and *r* for comparisons. By adding *c*, the **Linear** (-r) model improves the performance by 9.9 and 10.6 in terms of F_1 scores for both versions of evaluation methods at the scope level respectively. By adding *r* solely, the **Linear** (-c) model increases the performance by 6.5 and 7.9 on F_1 scores of both versions. By adding both *c* and *r*, the **Linear** model increases the performance by 15.3 and 18.2 on F_1 scores at the scope level, outperforming previous

---

[1] The official evaluation contains both two versions. We explain the differences between two versions of evaluation in the supplementary material.

This person is alone and <u>can</u> **not** be approached by letter without a breach of that absolute secrecy .

He has been there for ten days, and **neither** Mr. Warren , **nor** I , **nor** <u>the girl has once set eyes upon him.</u>

Figure 3: Examples showing incorrect instances in the **Linear** model but correct in **Semi** *o* model (The incorrect predictions by Linear model are underlined).

works. These improvements demonstrate the importance of the negation cue features.

Compared with the **Linear** model, **Semi** models achieve better results. Specifically, **Semi** *o* model achieves the best result on F_{1B} at the scope level among all the models and achieves the highest result on F_1 at the token level.

The **Latent** *io* model outperforms all the other models in terms of F_{1A} at scope level and a competitive result in terms of F_{1B}.

4.2 Analysis

By analyzing predictions that are incorrect in **Linear** model that are correct in the **Semi** models, we have some interesting observations explaining why **Semi** models work better. The first type of observation is that the **Semi** models tend to predict more correct scope tokens, which improves results at scope level and token level. The second type is that **Semi** models recover some missing remote partial scope, which shows the importance of capturing long distance dependencies. For instance, in the first example of Figure 3, the **Semi** models recover the subject phrase "*This person*" as the first partial scope. The third type happens on discontinuous cues as well as multiple short gaps as shown in the second example in Figure 3. The **Linear** model fails to predict "*Mr. Waren ,*" and "*I ,*" as two pieces of partial scope between three cue words which are also gaps. These observations indicate that **Semi** models are capable of capturing long distance features and can correct some wrong predictions made by the **Linear** model.

Similarly, by analyzing predictions that are incorrect in **Linear** model that are correct in the **Latent** models, we observe that **Latent** models tend to make more accurate predictions. We found that there is only 1 incorrect prediction from the **Latent** *io* that is corrected by the **Linear** model. This indicates that the **Latent** *io* model is able to fix er-

System	Abstract			Full Paper			Clinical		
	F_{1T}	F_{1A}	PCS	F_{1T}	F_{1A}	PCS	F_{1T}	F_{1A}	PCS
Li et al. (2010)	-	-	81.8	-	-	**64.0**	-	-	89.8
Velldal et al. (2012)	-	74.4	-	-	70.2	-	-	90.7	-
Zou et al. (2013)	-	-	76.9	-	-	61.2	-	-	85.3
Qian et al. (2016)	89.9	-	77.1	**83.5**	-	55.3	94.4	-	89.7
Linear	90.3	90.3	82.3	80.8	74.0	58.8	96.4	96.6	93.3
Semi *io*	**92.1**	**91.3**	**84.1**	83.1	**75.1**	60.1	**97.5**	97.1	**94.4**
Latent *io*	91.5	90.8	83.2	79.5	71.0	55.1	97.3	97.0	94.1

Table 3: Results on BioScope datasets.

rors for the **Linear** model without producing other wrong predictions. This analysis implies that the **Latent** models are able to capture some latent patterns to some extent. The performance of the **Latent** *o* model is lower than the performance of **Latent** *io* and **Latent** *i*, indicating that latent variables on tag **I** captures more information.

Let us focus on the token-level performance of our model. We obtained satisfactory precision scores, but comparatively low recall scores. Meanwhile, at the scope level, our precision scores are comparable to the previous works, but our recall scores are consistently better, indicating our models are capable of successfully recovering more gold scope information from the test data. Our further analysis shows that our models tend to predict negation scope that is significantly shorter than the gold scope for those instances that involve some long negation scope. We find that around 1/3 of the word tokens appearing inside any negation scope come from such instances. These facts make token-level recall of our models comparatively low.

In addition, we inspect the top 200 features with highest feature weights, and we find that around 45% of them are related to POS tags with label transition (the string form concatenating current tag and next tag), indicating POS tag features play an important role in the learning process for our models.

4.3 Experiments on Model Robustness

To understand the robustness of our model, we additionally conducted two sets of experiments.

BioScope

The BioScope corpus (Szarvas et al., 2008) contains three data collections from medical domains: *Abstract*, *Full Paper* and *Clinical*. NLTK (Bird and Loper, 2004) is used to perform tokenization and POS tagging for preprocessing. Following (Morante and Daelemans, 2009; Qian et al., 2016), we perform 10-fold cross validation on *Ab-*

System	Product Review			
	F_{1T}	F_{1A}	F_{1B}	PCS
(Zou et al., 2015)	-	-	-	60.93
Linear	89.60	81.86	69.39	69.39
Semi *io*	**90.78**	83.49	71.69	71.69
Latent *io*	90.60	**83.95**	**72.43**	**72.43**

Table 4: Results on *Product Review* from CNeSP.

stract, whereas the results on *Full Paper* and *Clinical* are obtained by training on the full *Abstract* dataset and testing on *Full Paper* and *Clinical* respectively. The latter experiment can help us understand the robustness of the model by applying the learned model to different types of text within the same domain.

The **Semi** *io* model mostly outperforms the other models. Comparing against all the prior works, our models are able to achieve better results on *Abstract* under both token-level and scope-level F_1 as well as PCS [2]. Moreover, we also obtain significantly higher results in terms of scope-level F_1 on *Full Paper* and *Clinical*, indicating the good robustness of our approaches. Note that the PCS score on *Full Paper* is not as satisfactory as on *Clinical*. This is largely because the model is trained on *Abstract*, but *Full Paper* contains much longer sentences with longer negation scope, which presents a challenge for our model as discussed in the previous sections. On the other hand, the baseline systems (Li et al., 2010; Velldal et al., 2012) adopt features from syntactic trees, which allow them to capture long-distance syntactic dependencies.

CNeSP

To understand how well our model works on another language other than English, we also conducted an experiment on the *Product Review* collection from the CNeSP corpus (Zou et al., 2015). We used Jieba (Sun, 2012) and Stanford tagger (Toutanova and Manning, 2000) to perform Chinese word segmentation and POS tagging. Following the data splitting scheme described in (Zou et al., 2015), we performed 10-fold cross-validation and the results are shown in Table 4. Our model obtains a significantly higher PCS score than the model reported in (Zou et al., 2015). The results further confirm the robustness of our model, showing it is language independent.

---

[2] PCS is defined as percentage of correct scope which is the same as the recall score.

5 Related Work

The negation scope extraction task has been studied within the NLP community through the BioScope corpus (Szarvas et al., 2008) in biomedical domain, usually together with the negation cue detection task. The negation scope detection task has mostly been regarded as a boundary detection task. Morante et al. (2008) and Morante and Daelemans (2009) tackled the task by building classifiers based on k-nearest neighbors algorithm (Cover and Hart, 1967), SVM (Cortes and Vapnik, 1995) as well as CRF (Lafferty et al., 2001) on each token to determine if it is inside the scope. Li et al. (2010) incorporated more syntactic features such as parse tree information by adopting shallow semantic parsing (Gildea and Palmer, 2002; Punyakanok et al., 2005) for building an SVM classifier. With similar motivation, Apostolova et al. (2011) proposed a rule-based method to extract lexico-syntactic patterns to identify the scope boundaries. To further investigate the syntactic features, Zou et al. (2013) extracted more syntactic information from constituency and dependency trees obtained from parsers to feed into the SVM classifier. Qian et al. (2016) adopted a convolutional neural network based approach (LeCun et al., 1989) to extract position features and syntactic path features encoding the path from the cue to the candidate token along the constituency trees. They also captured relative position information between the words in the scope and the cue as features in their model.

In order to resolve the corpus scarcity issue in different languages for the negation scope extraction task, Zou et al. (2015) constructed a Chinese corpus CNeSP analogous to the BioScope corpus. They again tackled the negation scope extraction task using CRF with rich syntactic features extracted from constituency and dependency trees.

6 Conclusion

We explored several approaches based on CRF to capture some useful features for solving the task of extracting negation scope based on a given negation cue in a sentence. We conducted extensive experiments on a standard dataset, and the results show that our models are able to achieve significantly better results than various previous approaches. We also demonstrated the robustness of our approaches through extensive analysis as well as additional experiments on other datasets.

References

Emilia Apostolova, Noriko Tomuro, and Dina Demner-Fushman. 2011. Automatic extraction of lexico-syntactic patterns for detection of negation and speculation scopes. In *Proc. of ACL*. http://www.aclweb.org/anthology/P11-2049.

Steven Bird and Edward Loper. 2004. Nltk: The natural language toolkit. In *Proc. of ACL*. https://doi.org/10.3115/1219044.1219075.

Md Chowdhury and Faisal Mahbub. 2012. Fbk: Exploiting phrasal and contextual clues for negation scope detection. In *Proc. of *SEM2012*. http://www.aclweb.org/anthology/S12-1045.

Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning* https://link.springer.com/content/pdf//10.1007/BF00994018.pdf.

Isaac G Councill, Ryan McDonald, and Leonid Velikovich. 2010. What's great and what's not: learning to classify the scope of negation for improved sentiment analysis. In *Proc. of NeSp-NLP*. https://aclanthology.info/pdf/W/W10/W10-3110.pdf.

Thomas Cover and Peter Hart. 1967. Nearest neighbor pattern classification. *IEEE transactions on information theory* https://ieeexplore.ieee.org/document/1053964.

Jorge Carrillo de Albornoz, Laura Plaza, Alberto Díaz, and Miguel Ballesteros. 2012. Ucm-i: A rule-based syntactic approach for resolving the scope of negation. In *Proc. of *SEM2012*. http://www.aclweb.org/anthology/S12-1037.

Federico Fancellu, Adam Lopez, and Bonnie L Webber. 2016. Neural networks for negation scope detection. In *Proc. of ACL*. https://doi.org/10.18653/v1/p16-1047.

Daniel Gildea and Martha Palmer. 2002. The necessity of parsing for predicate argument recognition. In *Proc. of ACL*. https://doi.org/10.3115/1073083.1073124.

Rodney Huddleston, Geoffrey K Pullum, et al. 2002. The cambridge grammar of english. *Language. Cambridge: Cambridge University Press* http://www.academia.edu/download/37907813//2001025630.pdf.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of ICML*. http://repository.upenn.edu/cis_papers/159.

Emanuele Lapponi, Erik Velldal, Lilja Øvrelid, and Jonathon Read. 2012. Uio 2: sequence-labeling negation using dependency features. In *Proc. of *SEM2012*. http://www.aclweb.org/anthology/S12-1042.

Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. 1989. Backpropagation applied to handwritten zip code recognition. *Neural Computation* https://doi.org/10.1162/neco.1989.1.4.541.

Junhui Li, Guodong Zhou, Hongling Wang, and Qiaoming Zhu. 2010. Learning the scope of negation via shallow semantic parsing. In *Proc. of COLING*. http://www.aclweb.org/anthology/C10-1076.

Roser Morante and Eduardo Blanco. 2012. *sem 2012 shared task: Resolving the scope and focus of negation. In *Proc. of *SEM 2012*. http://www.aclweb.org/anthology/S12-1035.

Roser Morante and Walter Daelemans. 2009. A metalearning approach to processing the scope of negation. In *Proc. of CoNLL*. http://www.aclweb.org/anthology/W09-1105.

Roser Morante, Anthony Liekens, and Walter Daelemans. 2008. Learning the scope of negation in biomedical texts. In *Proc. of ACL*. https://doi.org/10.3115/1613715.1613805.

Roser Morante and Caroline Sporleder. 2012. Modality and negation: An introduction to the special issue. *Computational linguistics* 38(2):223–260. https://doi.org/10.1162/COLI_a_00095.

Woodley Packard, Emily M Bender, Jonathon Read, Stephan Oepen, and Rebecca Dridan. 2014. Simple negation scope resolution through deep parsing: A semantic solution to a semantic problem. In *Proc. of ACL*. https://doi.org/10.3115/v1/p14-1007.

Vasin Punyakanok, Dan Roth, and Wen-tau Yih. 2005. The necessity of syntactic parsing for semantic role labeling. In *Proc. of IJCAI*. http://www.ijcai.org/Proceedings/05/Papers/1672.pdf.

Zhong Qian, Peifeng Li, Qiaoming Zhu, Guodong Zhou, Zhunchen Luo, and Wei Luo. 2016. Speculation and negation scope detection via convolutional neural networks. In *Proc. of EMNLP*. https://doi.org/10.18653/v1/d16-1078.

Jonathon Read, Erik Velldal, Lilja Øvrelid, and Stephan Oepen. 2012. Uio 1: Constituent-based discriminative ranking for negation resolution. In *Proc. of *SEM2012*. http://www.aclweb.org/anthology/S12-1041.

Sunita Sarawagi and William W Cohen. 2004. Semi-markov conditional random fields for information extraction. In *Advances in neural information processing systems*. pages 1185–1192. http://papers.nips.cc/paper/2648-semi-markov-conditional-random-fields-for-information-extraction.pdf.

J Sun. 2012. Jieba chinese word segmentation tool. https://github.com/fxsjy/jieba.

György Szarvas, Veronika Vincze, Richárd Farkas, and János Csirik. 2008. The bioscope corpus: annotation for negation, uncertainty and their scope in biomedical texts. In *Proc. of the Workshop on Current Trends in Biomedical Natural Language Processing*. https://aclanthology.info/pdf/W/W08/W08-0606.pdf.

Kristina Toutanova and Christopher D Manning. 2000. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proc. of EMNLP*. https://doi.org/10.3115/1117794.1117802.

Erik Velldal, Lilja Øvrelid, Jonathon Read, and Stephan Oepen. 2012. Speculation and negation: Rules, rankers, and the role of syntax. *Computational linguistics* 38(2):369–410. https://doi.org/10.1162/COLI_a_00126.

James Paul White. 2012. Uwashington: Negation resolution using machine learning methods. In *Proc. of *SEM2012*. http://www.aclweb.org/anthology/S12-1044.

Michael Wiegand, Alexandra Balahur, Benjamin Roth, Dietrich Klakow, and Andrés Montoyo. 2010. A survey on the role of negation in sentiment analysis. In *Proc. of the workshop on negation and speculation in natural language processing*. https://aclanthology.info/pdf/W/W10/W10-3111.pdf.

Bowei Zou, Guodong Zhou, and Qiaoming Zhu. 2013. Tree kernel-based negation and speculation scope detection with structured syntactic parse features. In *Proc. of EMNLP*. http://www.aclweb.org/anthology/D13-1099.

Bowei Zou, Qiaoming Zhu, and Guodong Zhou. 2015. Negation and speculation identification in chinese language. In *Proc. of ACL*. https://doi.org/10.3115/v1/p15-1064.