Negation-Instance Based Evaluation of End-to-End Negation Resolution

Elizaveta Sineva[1,2], **Stefan Grünewald**[1,2], **Annemarie Friedrich**[1], **Jonas Kuhn**[2]
[1]Bosch Center for Artificial Intelligence, Renningen, Germany
[2]Institut für Maschinelle Sprachverarbeitung, University of Stuttgart
elisavetas2106@gmail.com
stefan.gruenewald|annemarie.friedrich@de.bosch.com
jonas.kuhn@ims.uni-stuttgart.de

Abstract

In this paper, we revisit the task of negation resolution, which includes the subtasks of cue detection (e.g. "not", "never") and scope resolution. In the context of previous shared tasks, a variety of evaluation metrics have been proposed. Subsequent works usually use different subsets of these, including variations and custom implementations, rendering meaningful comparisons between systems difficult. Examining the problem both from a linguistic perspective and from a downstream viewpoint, we here argue for a negation-instance based approach to evaluating negation resolution. Our proposed metrics correspond to expectations over per-instance scores and hence are intuitively interpretable. To render research comparable and to foster future work, we provide results for a set of current state-of-the-art systems for negation resolution on three English corpora, and make our implementation of the evaluation scripts publicly available.

Test Sentences (gold scope)	System A TP/FP/FN	System B TP/FP/FN
(1) If **not**, I 'll have to do with you.	0 / 4 / 0 ✗	0 / 0 / 0 ✓
(2) He made **no** remark, but the matter remained in his thoughts.	1 / 0 / 2 ✗	3 / 0 / 0 ✓
(3) Well, Mrs. Warren, I can see that you have any particular cause for concern, **nor** do I understand why I, whose time is of some value, should interfere in the matter.	16 / 0 / 0 ✓	10 / 2 / 6 (✓)
Scope Tokens P/R:	81.0/89.5	86.7/68.4
F1:	**85.0**	76.5
Instance-Based P/R:	66.7/77.8	94.4/87.5
F1:	71.8	**90.8**

Figure 1: **Different scoring metrics for negation scope resolution predictions.** ✓/✗ = our judgment of system correctness taking into account linguistic criteria. Parentheses indicate partial correctness.

1 Introduction

Negation is a complex semantic phenomenon in natural language that "transforms an expression into another expression whose meaning is in some way opposed to the original" (Morante and Blanco, 2021). It occurs frequently, with the proportion of sentences with negation in English corpora ranging between 9 and 32% (Jiménez-Zafra et al., 2020b). Natural Language Processing (NLP) applications that may benefit from negation resolution include sentiment analysis (Wiegand et al., 2010; Moore and Barnes, 2021) and information extraction. Negation is also still a challenge in machine translation (Fancellu and Webber, 2015; Bentivogli et al., 2016; Hossain et al., 2020a) and natural language inference (Hossain et al., 2020c; Geiger et al., 2020).

Negation Resolution (Morante and Blanco, 2021) refers to the task of automatically retrieving the elements of a sentence that are affected by the negation introduced by a *cue*. The cue's *scope* is "the part of the meaning that is negated" (Huddleston and Pullum, 2002). The task is difficult due to the multitude of ways in which negation may be expressed. Despite having been a continuously active research area especially since two shared tasks (Farkas et al., 2010; Morante and Blanco, 2012), building robust computational models is far from being a solved task, in part due to a lack of annotation standards (Jiménez-Zafra et al., 2020b).

Negation resolution has traditionally been addressed by heavily relying on syntactic parses (e.g. Sanchez Graillet and Poesio, 2007; Sohn et al., 2012; Mehrabi et al., 2015). Recently, end-to-end neural approaches to modeling negation resolution (Fancellu et al., 2016, 2018; Khandelwal and Sawant, 2020; Kurtz et al., 2020) have claimed

Proceedings of the 25th Conference on Computational Natural Language Learning (CoNLL), pages 528–543
November 10–11, 2021.

superior performance.

A significant problem in the field of negation resolution is that due to the large variety of possible evaluation setups and metrics, and the use of standard scoring scripts vs. custom implementations, it is not obvious how to meaningfully compare existing approaches and benchmark new models. Differences in evaluation setup include, for example, whether gold cues are provided for scope resolution, and which subset and variations of evaluation metrics are used. In the CoNLL 2010 Shared Task on detecting hedges and resolving their scope (Farkas et al., 2010), only exact cue and scope matches were counted. The *SEM 2012 Shared Task on negation resolution (Morante and Blanco, 2012) proposed additional less strict metrics based on token-level matching. The organizers of both tasks explicitly state that their evaluation is intended as a starting point and that further work on defining a scope evaluation measure that better captures the impact of partial matches is necessary.

In this paper, we take a step back and revisit the evaluation of negation resolution in order to provide a more unified picture of the relative performance of different systems. As a result, we propose a negation-instance based evaluation framework that defines intuitively interpretable and linguistically motivated metrics, facilitating a graded scoring of cue and scope matching. As illustrated by Figure 1, a gradual judgment of a system's scope resolution capability may be misleading if simply computing F1 over tokens in gold or predicted scopes. In (1), system A marked a non-existent scope, which may be detrimental to a user's trust in the system. In (2), A did not mark the main event, i.e., extracting a correct logical representation would be impossible. In (3), despite getting a bad recall score, B's output captures all relevant arguments and complements of the negated proposition headed by "understand."

The core of our proposed method is to normalize precision and recall scores per instance and compute an expectation for these instance-wise match scores treating all instances equally during score aggregation, following the insight that failing on short or long negation scopes may be equally detrimental. We apply our metrics to a set of recent neural models for negation resolution, including NegBERT, tagging-based, and dependency-parsing based approaches, providing a competitive range of baselines for future work to compare with.

Our **contributions** are as follows. (1) We provide a concise and formal overview of existing evaluation metrics with the aim of facilitating a principled comparison of approaches to negation resolution. (2) We propose a linguistically motivated and intuitively interpretable negation-instance based scope resolution scoring framework. (3) Using our (and previously proposed) metrics, we conduct a reproducibility study on negation resolution, reporting performance scores of a variety of relevant baseline systems in a uniform experimental setup.[1] (4) As a side result, we show that a modern transformer-based reimplementation of the tagging-based system by Fancellu et al. (2016) achieves the best negation resolution performance under most circumstances.

2 Related Work

We here introduce the linguistic terminology and concepts used in this paper, and briefly survey related work in computational linguistics.

2.1 Linguistic Background

A negation **cue** signals to the listener or reader that the inverse of something is referred to. In the ConanDoyle-neg (CD-neg) annotation scheme (Morante et al., 2011), a negation cue may be a single word such as "not," a multi-word expression such as "no more," or a negation affix such as "im" (e.g., in "imprecise"). The **scope** is the part of the sentence that is "affected" by the negation signaled by the cue (Huddleston and Pullum, 2002).

Logically, negation is an operator taking a proposition, which typically correspond to (sub-)clauses (e.g., $like(m, p)$ for "Mary likes pizza"), converting the corresponding assertion to an assertation stating that something is not the case ($\neg like(m, p)$ for "Mary does **not** like pizza"). However, in natural language, depending on the embedding context as well as pragmatic presuppositions, it is not always the case that negation operators convert something to its logical complement (Horn, 2010; Blanco and Moldovan, 2011b). For example, "she is not unhappy" does not mean "she is happy."

In CD-neg, the aim of the annotation is to make explicit which event (process, or state) is affected by the change of polarity (Morante et al., 2011; Morante and Daelemans, 2012). The word referring to the event is marked as *Event*, but only if the

[1] Our code is available at github.com/boschresearch/negation_resolution_evaluation_coll2021.

event is factual. Thus, in cases such as "He may not know the answer," no event is annotated. To avoid terminological confusion, in this paper, we ignore *Event* annotations and call "know" in the example above the *main predicate* of the negation.

The scope is annotated as the longest relevant part of the sentence, i.e., as the main predicate referring to the negated event and all its arguments and complements. In contrast to BioScope (Szarvas et al., 2008), CD-neg includes the subject, but not the cue in the scope. In constituent negation, the negation marker is attached to the object as in "Mary came to the lecture with **no** books." Still, the negation scopes over the entire sentence and is marked accordingly in CD-neg to achieve representational equality with the sentence "Mary did **not** come to the lecture with books." A constituent-negated subject receives the same treatment.

One element of the scope is singled out as the negation's **focus**, i.e., the part that is *intended* to be interpreted as false (Huddleston and Pullum, 2002). Detecting the focus usually requires leveraging phonetic cues as in "Your kids don't *hate* school" vs. "*Your* kids don't hate school" (Blanco and Moldovan, 2011b). Correct identification of a negation's focus is key to natural language understanding. However, to date, no corpora annotating both negation scopes and focus exist. For some ideas on integrating focus identification into our proposed evaluation framework, see Sec. 6.

2.2 Automatic Negation Resolution

Computational work on negation resolution is generally based on small- to medium-scale corpora, which in addition are often not very compatible due to differences in the employed annotation schemes and underlying tokenization. Jiménez-Zafra et al. (2020b) provide a comprehensive survey of datasets annotated for negation.

As the problem of negation resolution is closely tied to syntax, there are many works leveraging syntactic information, using rules over syntactic structures to resolve negation or speculation scopes (e.g., Velldal et al., 2012; Packard et al., 2014; McKenna and Steedman, 2020) or training systems with explicit or learned syntactic features (Read et al., 2012; Lapponi et al., 2012; Enger et al., 2017; Ren et al., 2018; Jiménez-Zafra et al., 2020a).

Li et al. (2010) frame scope resolution as a shallow semantic parsing task backed up by syntactic parses. In the neural age, Kurtz et al. (2020) frame negation resolution as a dependency parsing task. Several works using neural networks (e.g., Fancellu et al., 2016, 2018; Lazib et al., 2020) train BiLSTMs, or syntactically structured BiLSTMs or GCNs. Qian et al. (2016) propose a CNN-based architecture combined with some path/position information. Recently, a range of papers has explored BERT-based models for negation resolution (Khandelwal and Sawant, 2020; Khandelwal and Britto, 2020; Britto and Khandelwal, 2020; Shaitarova and Rinaldi, 2021). Further related work includes datasets annotated for focus (Blanco and Moldovan, 2011a; Altuna et al., 2017), and the computational modeling thereof (e.g., Hossain et al., 2020b). In addition, there is a growing body of work addressing negation within the context of neural language models and commonsense reasoning using them (e.g., Hossain et al., 2020c; Geiger et al., 2020; Hosseini et al., 2021; Jiang et al., 2021).

3 Evaluation Metrics and Settings

In this section, we first give an overview of the various evaluation metrics used in shared tasks and research publications. We then propose a framework for negation resolution focusing on instance-level metrics. Negation resolution as a standalone NLP task is usually split into two sub-tasks, cue detection and scope resolution, where the latter depends on the former.[2] Hence, scope resolution may be evaluated in two **settings**, with either *gold cue* information being given or in an end-to-end manner where systems also have to *predict cues*.

We first introduce some notations with the aim of a unified presentation of metrics. A **negation instance** is a tuple (c, s) consisting of a set of cue tokens $c = (c^1, c^2, ..., c^C)$ and a set of scope tokens $s = (s^1, s^2, ..., s^S)$. Punctuation may be excluded from these sets by definition. In the case of affix negation, the affix is treated as a separate token if annotated or predicted as a cue. I_g is the set of gold standard, I_p the set of predicted negation instances.

3.1 Metrics used in Shared Tasks

Previously proposed metrics for negation resolution can be divided into metrics requiring exact cue matches and metrics requiring partial cue matches (perhaps confusingly called *No Cue Match*).

Cue Detection. In this step, gold standard and predicted cue annotations are matched to each other.

[2] Morante and Blanco (2012) also use *scope resolution* to refer to the entire task of negation resolution.

When matching the gold standard cue c_g and the predicted cue c_p, we can either require an exact match ($c_g = c_p$) or a partial match ($c_g \cap c_p \neq \varnothing$).

The CoNLL 2010 Shared Task on detecting hedges and their scope in text (Farkas et al., 2010) required exact cue matches, while the *SEM 2012 Shared Task (Morante and Blanco, 2012) employed two metrics, one counting only exact and one counting also partial cue matches as true positives. For these metrics, each negation instance (called *scope* in these works) counts as an instance, which can either be evaluated as a true positive (TP), false positive (FP), or false negative (FN). Precision is then computed as TP/(TP+FP), recall is TP/(TP+FN), and F1 is used to summarize the scores.

Scope Resolution. The CoNLL 2010 Task employed only a single strict metric requiring exact cue and scope matches, then computing negation-instance level precision, recall and F1. This metric is intuitive but (as the authors concede) maybe a bit too strict. For example, the metric gives little insight when trying to identify which to two imperfect systems is slightly better at scope resolution, or when trying to evaluate whether a system under- or overpredicts the extents of scopes. In addition, as pointed out by Morante and Blanco (2012), the metric penalizes partially matched negation instances more than missed instances, as these cases count both as an FN and an FP.

As a remedy, the *SEM 2012 Shared Task evaluation employs a suite of scores. Their **Scope-Level Cue Match (SCM)** metric requires an exact cue match, while the **Scope-Level No Cue Match** metric only requires a partial cue match (but exact matching of the scope). In both cases, partial matches are counted only as FNs; however, this results in the problem that TP+FP (the denominator in computing precision) does not correspond to the number of system predictions. Hence, a second version called **"B"** is employed that uses the number of system predictions as the denominator.

For giving credit to partial scope matches, a so-called **Scope Tokens (ST)** metric is used, which computes precision, recall, and F1 for tokens belonging to the scope of a negation instance in the gold data vs. system output. Notably, one token can be counted several times if it belongs to more than one scope.[3] However, each cue must belong to exactly one negation instance. Formally, the scores are computed as follows:[4]

$$P_{tok} = \frac{\sum_{(c_g,s_g)\in I_g,(c_p,s_p)\in I_p,c_g=c_p} |s_g \cap s_p|}{\sum_{(c_p,s_p)\in I_p} |s_p|}$$

$$R_{tok} = \frac{\sum_{(c_g,s_g)\in I_g,(c_p,s_p)\in I_p,c_g=c_p} |s_g \cap s_p|}{\sum_{(c_g,s_g)\in I_g} |s_g|}$$

The *SEM 2012 task employs additional metrics such as an F1 over **Negated Events** (independently of cues and scopes), as well as **Global Negation** which requires cue, scope and event to be correct. Finally, they also report the percentage of correct negation sentences (CNS).[5]

3.2 Motivation for Instance-based Scoring

From the perspective of a linguist, it matters to correctly resolve the scope pertaining to a negation cue, as this is a prerequisite for modeling the semantics of the sentence, e.g., using predicate logic. For a machine comprehension downstream task such as sentiment analysis, it matters that a truth-theoretic interpretation of the sentence would come out correctly, or that the parts of the sentence that the model bases its decision on are interpreted using the correct polarity. NLP tasks such as semantic parsing and event extraction care about factors similar to the predicate logic view. In typical relation extraction setups, events are only indicated implicitly via relations between participants; here, it is of high relevance that a system includes the negated proposition's arguments in the scope.

Both the linguistic and the downstream task views desire that all parts of the negated proposition(s) are detected (high recall), but no more than these (high precision). Ideally, a user of a system could obtain its precision and recall in terms of (a) correctly identifying negation cues and (b) correctly identifying the negated propositions including arguments and complements by indicating the corresponding spans in the surface text. We

[3] This is not stated explicitly in Morante and Blanco (2012), see original evaluation script: https://www.clips.uantwerpen.be/sem2012-st-neg/data.html.

[4] The notation under the sum symbol is to be read as "for each element in I_g, identify at most one element in I_p for which the criterion $c_g = c_p$ holds." Replacing the requirement $c_g = c_p$ in the formulas below with $c_g \cap c_p \neq \varnothing$ results in a version using partial cue matches. However, we argue that this (a) results in technical difficulties for how to integrate partial matches, rending scores less intuitive, and (b) from an downstream point of view, it is crucial to know the full cue. Consider, for instance, the implications of "There should be **no** problems" vs. "There should be **no more** problems."

[5] Fancellu et al. (2017, 2018) also report PCS, "the proportion of negation scopes that we fully and exactly match in the test corpus," which should be the recall of the B-version of SCM.

argue that without any prior information on the type of negation or the average scope length in the application domain given, a score computed based on existing annotated gold standard data should reflect an expectation of how well a system would perform on a random instance.

It is without question that a metric able to identify gradual differences between systems and configurations is invaluable for research and development. The *SEM 2012 Scope Tokens F1 metric, however, effectively weights instances by their scope lengths, with longer scopes contributing more to the overall scores. We here argue that this is not desirable. If a system gets a fair amount of long-scope cases as in example (3) in Figure 1 right, the system will obtain a high overall score. A system that is good at recognizing the exact extents of short scopes (which may not be trivial, especially in long sentences), will not perform on par when using such a metric. From an application point of view, predicting a scope where none exists such as in example (1) in Figure 1 may be quite detrimental to a user's trust in the system, yet, this would have little impact on the system's score.

3.3 Negation-Instance Based Scoring (NIS)

In this section, we propose a new flexible scoring framework for negation resolution. The aim of the scope match metric(s) is to summarize how well a negation resolution system performs overall in terms of being precise and capturing all relevant instances (recall), giving partial credit for scope resolution. The final scores can be intuitively interpreted as the expectation how a system would perform on a random unseen instance. For each pair of negation instances whose cues match exactly, the scope match scoring functions f_P and f_R compute a precision and recall score, respectively. These scores must each range between 0 and 1.

$$P_{inst} = \sum_{(c_p,s_p)\in I_p,(c_g,s_g)\in I_g,c_g=c_p} \frac{1}{|I_p|} f_P(s_g, s_p)$$

$$R_{inst} = \sum_{(c_g,s_g)\in I_g,(c_p,s_p)\in I_p,c_g=c_p} \frac{1}{|I_g|} f_R(s_g, s_p)$$

The formulas above can be interpreted as summing the scores of all gold standard / predicted instances for which a match could be found. In other words, predicted instances for which no cue match has been found in the gold data contribute to the sum with a precision score of 0. Similarly, in the case of recall, gold instances for which no match has been found implicitly contribute with a score of 0.

In our standard metric, giving credit to partial matches, we compute token-level scope matching scores as follows:

$$f_{P,tok}(s_g, s_p) = \begin{cases} \frac{|s_g \cap s_p|}{|s_p|} & \text{if } |s_p| > 0 \\ 1 & \text{else} \end{cases}$$

$$f_{R,tok}(s_g, s_p) = \begin{cases} \frac{|s_g \cap s_p|}{|s_g|} & \text{if } |s_g| > 0 \\ 1 & \text{else} \end{cases}$$

In our formulation above, we weight the match scores returned by f_P or f_r by $\frac{1}{|I_p|}$ or $\frac{1}{|I_g|}$, respectively. In other words, our final **negation-instance scores** (**NIS$_{\text{tok}}$**) correspond to precision and recall scores that are the expectations of the instance-level scores when weighting each instance equally, and thus allow a somewhat intuitive interpretation.

In the strictest case requiring exact scope match, the scope matching functions can be defined as:

$$f_{P,ex}(s_g, s_p) = f_{R,ex}(s_g, s_p) = \begin{cases} 1 & \text{if } s_p = s_g \\ 0 & \text{else} \end{cases}$$

In this case, our metric (**NIS$_{\text{ex}}$**) would correspond to **SCM-B**, the *SEM 2012 B-style Scope-Level F1 (and to the CoNLL 2010 metric).

When comparing the *SEM 2012 Scope Tokens metric to NIS$_{\text{tok}}$, for computing precision, the weighting term $\frac{1}{|I_p|}$ is replaced with $\frac{|s_p|}{Z}$ ($\frac{|s_g|}{Z}$ in the case of recall). Z is the sum of all predicted (gold standard) scope tokens (see also Appendix A). Hence, longer scopes have a higher impact, and the resulting scores are not interpretable as expectations for a random unseen instance. As we have argued above, this may not reflect a system's capacity of negation scope resolution in an ideal way.

4 Modeling

In this section, we explain the negation resolution models that we compare in our experiments in Sec. 5. Due to the large number of negation resolution systems, often with no published code, it is infeasible to provide unified evaluation scores for all prior work. Instead, to provide competitive baselines for future work to compare with, we chose a wide range of neural architectures inspired by previous work and re-implement them. For comparability reasons, we base them all on the same robust transformer-based language model.

Token representation. Our token embedding backbone is the transformer-based XLM-R-large

language model (Conneau et al., 2020), as well as the corresponding word-piece tokenizer. To obtain contextualized word embeddings, we take a weighted sum of the internal states corresponding to the first word piece for each token. The coefficients of this weighted sum are learned during training, employing layer dropout (see Kondratyuk and Straka, 2019). Transformer weights are fine-tuned during training. To determine the effect of injecting implicit syntactic knowledge into the system, in addition to using the default pre-trained XLM-R model, we also run experiments on an XLM-R-**synt** model that was previously fine-tuned on the task of Universal Dependencies parsing on the EWT treebank (Silveira et al., 2014).

4.1 Sequence-Tagging Based Approaches

Tagging pipelines first identify negation cues, and then for each cue, identify its scope using a second tagger. The **NegBERT** system (Khandelwal and Sawant, 2020; Britto and Khandelwal, 2020), which we run using XLM-R, modifies the input for the second step, adding artificial tokens to indicate cues. In addition, we implement a **BiLSTM-Tagger** following the architecture proposed by Fancellu et al. (2016), but using XLM-R as the underlying language model.[6] Cues are predicted using a single linear layer with softmax on top of XLM-R. Scopes are predicted by feeding, for each negation instance, the XLM-R embeddings of the sentences concatenated with a *cue/notcue* embedding to a single-layer BiLSTM and once again using a token-wise linear+softmax layer for classification.

4.2 Dependency-Parsing Based Approaches

This class of models frame negation resolution as a dependency parsing (**DP**) task, as proposed by Kurtz et al. (2020), predicting cues and scopes in a single step by encoding negation instance annotations as dependency trees. The systems we present here differ w.r.t. this encoding (see Figure 2). In the **direct mapping** (Kurtz et al., 2020), cue tokens are modeled as dependents of the artificial root token, and scope and event tokens are attached via a dependency link to all cues they belong to. In addition, we propose a **nested mapping** in which in the cases of embedded scopes, there is only one link from the outer scope's cue to the inner scope's cue, and all other scope tokens are only linked to their corresponding nearest cue. We build these models using the graph-based dependency parser STEPS (Grünewald et al., 2020).[7]

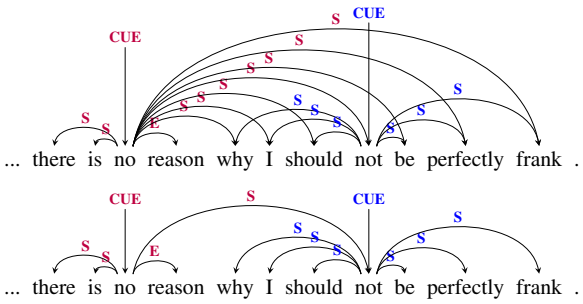

Figure 2: **Direct** (upper) vs. **nested** (lower) encoding.

5 Experiments

We here report our results for end-to-end negation resolution including cue detection and scope resolution. In all of our evaluations, we ignore punctuation tokens. In addition to the models explained in Sec. 4, we report results for a punctuation baseline **(Punct-BL)** that uses gold cues and tags everything between the cue and the next punctuation marker as the scope.

Datasets. We conduct our experiments on three corpora from a variety of domains. In our experiment, we focus on the CD-neg (**CD**) dataset, which has been shown to be most challenging among the English negation corpora (Fancellu et al., 2017). The corpus comprises seven literary texts corresponding to 5,520 sentences with 1,432 negation instances. CD annotates "neither...nor" as a multiword cue with a single scope. However, from a semantic point of view, we interpret "Neither Mary nor Sam like pizza" as $\neg like(m,p) \wedge \neg like(s,p)$, which actually suggests annotating two separate instances with overlapping scopes.[8] We noticed that the majority of cases in which systems got multiword cue detection wrong were like this, detecting only part of the cue but resolving the scope correctly. Rather than punishing systems for this, we decide to re-annotate the dataset accordingly, fixing a total of 10 cases.

In addition, we conduct experiments on the **BioScope** corpus (Szarvas et al., 2008) using the abstracts subset (11,871 sentences), as well as the

[6] Fancellu et al. (2016) use task-specific learned or word2vec embeddings (Mikolov et al., 2013).

[7] https://github.com/boschresearch/steps-parser

[8] The original annotation may be seen as corresponding to an (equivalent) formalization of $\neg\,(like(m,p) \vee like(s,p))$.

System	Cues-B F_1	SCM F_1	SCM-B (NIS_{ex}) F_1	ST P	ST R	ST F_1	NIS_{tok} P	NIS_{tok} R	NIS_{tok} F_1
Punct-BL	*100.0*	*17.0*	*9.9*	*89.8*	*62.0*	*73.3*	*93.3*	*58.8*	*72.1*
NegBERT	91.1	78.8	**70.1**	86.1	**87.8**	86.9	83.9	**88.7**	86.3
BiLSTM-Tagger	92.3	**79.6**	70.0	**90.8**	85.4	**88.0**	**90.3**	86.3	**88.2**
DP-direct	92.8	73.2	61.4	85.4	87.4	86.4	84.6	87.7	86.1
DP-nested	93.3	72.8	60.6	86.6	83.0	84.8	85.6	85.4	85.5
DP-direct-synt	92.8	79.1	69.2	87.4	86.5	86.9	86.2	88.3	87.2
DP-nested-synt	**93.4**	79.2	69.2	88.0	84.6	86.2	87.2	86.6	86.9

Table 1: **Comparison of Evaluation Scores** on when training and testing on **CD-neg**. For cues, we report the *SEM 2012 B version for exact cue matching **(Cues-B)**. **SCM** and **SCM-B** refer to the standard and B versions of Scope Level F1 of *SEM 2012, respectively; **ST** refers to the Scope Tokens metric. Underlined: Comparison NegBERT vs. DP-direct-synt.

SFU Review corpus (Taboada et al., 2006; Konstantinova et al., 2012), which comprises 400 reviews in eight different domains. The BioScope and SFU datasets are also annotated for speculation; our experiments only use the negation annotations. More details on all corpora are given in Table 9 in Appendix C.

5.1 Experimental Setup

We use the official training-dev-test split for CD-neg. For BioScope and SFU, we create our own 80-10-10 splits for these datasets. For more information, see Appendix C. To tokenize BioScope, we use NLTK (Loper and Bird, 2002) with custom rules for punctuation and URLs.

Our models are implemented using PyTorch (Paszke et al., 2019) and the Huggingface Transformers library (Wolf et al., 2020). Training is performed on a single nVidia Tesla V100 GPU. We use a unified set of hyperparameters for the underlying XLM-R language model, but different hyperparameters for the parser/tagger layers on top. For a detailed description, see Appendix B.

5.2 Results

Table 1 shows the results for our set of competitive neural systems on CD-neg. For cue detection, all systems perform similarly well, with the parsing-based approach using the nested representation and a syntactically fine-tuned XLM-R (DP-nested-synt) having a slight advantage. We ran the NegBERT system for end-to-end negation resolution using the original code, but our evaluation scripts.[9] In this unified evaluation setup, in the SCM metrics, we can only see that the DP models based on standard XLM-R underperform; all other systems perform roughly similarly. The less strict ST and NIS_{tok} metrics reveal different precision-recall trade-offs for the direct vs. the nested encoding. The BiLSTM-Tagger turns out to be the most accurate model for scope resolution, reflected similarly in the ST and the NIS_{tok} scores. Our architecture similar to the one of Fancellu et al. (2018) seems to outperform NegBERT, which adds artificial tokens to indicate cues.[10]

As expected to some extent, the NIS_{tok} and ST scores are similar in general. Both scores identify the BiLSTM-Tagger as the most precise system and NegBERT as having the highest recall, with the Tagger achieving the best F_1. However, while the bold-facing in Table 1 indicates similar patterns for the top systems, the ranking of the other systems differs when comparing ST and NIS_{tok} scores. For example, ST assigns the same summary statistic (F_1) to NegBERT and DP-direct-synt, while in terms of NIS_{tok}, the F_1 of DP-direct-synt is more than 1 point higher. Comparing ST and NIS_{tok} scores, we can see that NIS_{tok} generally assigns higher recall, but slightly lower precision to systems such as NegBERT or DP; the BiLSTM-Tagger's precision drops less. Hence, the ST scores for models such as NegBERT or DP slightly underestimate recall because the systems failed more often on longer instances; and in turn they slightly over-estimate precision, e.g., because wrongly predicted instances often have short scopes. We here argue that while monitoring the system on several metrics is generally a good idea, NIS_{tok} constitutes a more realistic gradual end-to-end evaluation met-

[9] Khandelwal and Sawant (2020) report results for scope resolution that appear to be based on gold cues.

[10] We ran the scope tagger (trained on gold cues) on the output of the best cue tagger run as chosen by dev set performance. Note that while this reflects a real-life development setup, the DP models predict cues and scopes in one run. Likewise, NegBERT scores were produced by running the system off-the-shelf without optimizing the cue tagger separately.

	System	Cues-B F_1	NIS_{ex} F_1	ST F_1	NIS_{tok} P	NIS_{tok} R	NIS_{tok} F_1
BioScope	*Punct-BL*	*100.0*	*43.2*	*65.7*	*77.0*	*69.7*	*73.5*
	NegBERT	90.5	79.5	84.5	83.3	91.5	87.2
	BiLSTM-Tagger	94.3	**83.0**	88.6	91.7	91.7	91.7
	DP-nested-synt	**95.4**	82.7	**90.4**	**92.7**	**92.1**	**92.4**
SFU	*Punct-BL*	*100.0*	*45.9*	*66.5*	*75.5*	*87.4*	*81.0*
	NegBERT	81.9	69.5	71.0	67.9	86.3	75.9
	BiLSTM-Tagger	**86.7**	**73.2**	**77.9**	**74.9**	**89.8**	**81.7**
	DP-nested-synt	86.5	71.1	77.1	73.7	88.9	80.5

Table 2: **In-Domain Comparison of Systems** on the BioScope and SFU datasets.

System	Train	Test	Cues-B F_1	NIS_{ex} F_1	ST F_1	NIS_{tok} F_1
NegBERT	Bio	CD	**65.2**	8.9	**55.2**	**51.1**
BiLSTM-Tagger			61.6	**10.1**	46.0	46.8
DP-nested-synt			64.8	9.2	51.8	47.9
NegBERT	SFU	CD	**69.3**	9.7	56.7	53.6
BiLSTM-Tagger			68.8	**10.1**	**58.0**	**53.8**
DP-nested-synt			68.1	9.6	55.2	51.6
NegBERT	CD	Bio	60.6	**19.0**	49.3	46.2
BiLSTM-Tagger			64.3	17.7	**56.7**	**54.3**
DP-nested-synt			**65.3**	18.7	54.7	50.9
NegBERT	SFU	Bio	78.7	55.5	63.2	71.0
BiLSTM-Tagger			81.0	53.4	65.9	72.8
DP-nested-synt			**82.0**	**57.2**	**67.2**	**74.0**
NegBERT	CD	SFU	51.3	10.0	37.2	37.5
BiLSTM-Tagger			**50.2**	**9.0**	**37.5**	**37.8**
DP-nested-synt			50.3	9.0	36.2	36.5
NegBERT	Bio	SFU	63.4	48.8	53.7	58.6
BiLSTM-Tagger			**64.5**	**53.1**	**57.0**	**61.4**
DP-nested-synt			56.9	40.1	47.5	51.5

Table 3: **Cross-Domain Comparison of Systems** between CD-neg, BioScope, and SFU.

ric for negation resolution systems, and should be adopted as the main summary statistic by subsequent works or shared tasks.

Table 2 compares the various system architectures on the BioScope and SFU data when trained and tested in-domain. As the difference between the DP models was small (see Appendix C), we report only DP-nested-synt. On both datasets, the DP-nested-synt models outperform NegBERT. On BioScope, the parsing-based approach clearly outperforms NegBERT and the BiLSTM-Tagger; on SFU, the BiLSTM-Tagger performs best.

Finally, we also give results for cross-domain performance. Overall, the BiLSTM-Tagger seems most robust. However, the NegBERT system performs close to or better than the BiLSTM-Tagger when moving from another to the ConanDoyle-neg dataset, and the DP-nested-synt model has an advantage when moving from SFU to BioScope. In sum, particularly cross-domain negation resolution is still far from being solved. We hope that our linguistically motivated evaluation framework can aid the development of more robust negation resolution systems.

6 Discussion

A general problem with most existing text corpora annotated for negation is that annotations are only created on the surface level. However, in the words of Blanco and Moldovan (2011a), "Negation does not stand on its own, to be useful, it should be added as part of another existing knowledge representation." Our negation-instance based framework was detailed above such that it can directly be applied to existing widely used negation corpora. However, the precision and recall scoring functions f_P and f_R can easily be designed in other ways. For instance, if leveraging a dependency parse of the sentence, argument structure could be approximated by taking as the set of elements in a scope not the tokens, but only the dependents of the main predicate to which the negation cue attaches. In this way, relative clauses as in (3) in Figure 1 would have zero impact. Thinking ahead, a truly linguistically motivated scoring function could even weight components by their importance of being detected as belonging to a scope, e.g., missing a restrictive relative clause could be penalized more than missing a non-restrictive one.

If we had a dataset marking cues, scopes and foci, our scoring framework could assign a high weight for detecting the focus correctly. Similarly, detecting the main predicates correctly could be incorporated into the score. We here decided against including event annotations as they are only marked on factual events and hence, in our opinion, should be evaluated as a separate task (as was done in most metrics in previous shared tasks).

7 Conclusion and Outlook

The aim of this paper is to provide a concise reference of evaluation metrics and setups for negation resolution, making it easier for NLP researchers and developers to enter the research area. Our core contribution is to detail the linguistic motivation for employing a new instance-based approach to evaluating the performance of end-to-end negation

resolution systems, giving credit to partial scope matches but relying on exact cue matches. We argue that this metric is well-motivated and intuitively interpretable and should hence be adopted by future studies or shared tasks. In addition, our experimental study, comparing a set of recent neural architectures on a similar basis, will serve as a reference for future work.

Besides implementing a variety of linguistically motivated extensions with the aim of deeper system analyses using our framework as suggested above, an important next step is to evaluate the suite of models used in this paper on further datasets in languages other than English (e.g., Zou et al., 2015; Liu et al., 2018; Jiménez-Zafra et al., 2018).

References

Begoña Altuna, Anne-Lyse Minard, and Manuela Speranza. 2017. The scope and focus of negation: A complete annotation framework for Italian. In *Proceedings of the Workshop Computational Semantics Beyond Events and Roles*, pages 34–42, Valencia, Spain. Association for Computational Linguistics.

Luisa Bentivogli, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. 2016. Neural versus phrase-based machine translation quality: a case study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 257–267, Austin, Texas. Association for Computational Linguistics.

Eduardo Blanco and Dan Moldovan. 2011a. Semantic representation of negation using focus detection. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 581–589, Portland, Oregon, USA. Association for Computational Linguistics.

Eduardo Blanco and Dan Moldovan. 2011b. Some issues on detecting negation from text. In *Twenty-Fourth International FLAIRS Conference*.

Benita Kathleen Britto and Aditya Khandelwal. 2020. Resolving the scope of speculation and negation using transformer-based architectures. *arXiv preprint arXiv:2001.02885*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Timothy Dozat and Christopher D. Manning. 2018. Simpler but more accurate semantic dependency parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 484–490, Melbourne, Australia. Association for Computational Linguistics.

Martine Enger, Erik Velldal, and Lilja Øvrelid. 2017. An open-source tool for negation detection: a maximum-margin approach. In *Proceedings of the Workshop Computational Semantics Beyond Events and Roles*, pages 64–69, Valencia, Spain. Association for Computational Linguistics.

Federico Fancellu, Adam Lopez, and Bonnie Webber. 2016. Neural networks for negation scope detection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 495–504, Berlin, Germany. Association for Computational Linguistics.

Federico Fancellu, Adam Lopez, Bonnie Webber, and Hangfeng He. 2017. Detecting negation scope is easy, except when it isn't. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 58–63, Valencia, Spain. Association for Computational Linguistics.

Federico Fancellu, Adam Lopez, and Bonnie L. Webber. 2018. Neural networks for cross-lingual negation scope detection. *CoRR*, abs/1810.02156.

Federico Fancellu and Bonnie Webber. 2015. Translating negation: Induction, search and model errors. In *Proceedings of the Ninth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 21–29, Denver, Colorado, USA. Association for Computational Linguistics.

Richárd Farkas, Veronika Vincze, György Móra, János Csirik, and György Szarvas. 2010. The CoNLL-2010 shared task: Learning to detect hedges and their scope in natural language text. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning – Shared Task*, pages 1–12, Uppsala, Sweden. Association for Computational Linguistics.

Atticus Geiger, Kyle Richardson, and Christopher Potts. 2020. Neural natural language inference models partially embed theories of lexical entailment and negation. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 163–173, Online. Association for Computational Linguistics.

Stefan Grünewald, Annemarie Friedrich, and Jonas Kuhn. 2020. Graph-based universal dependency parsing in the age of the transformer: What works, and what doesn't. *CoRR*, abs/2010.12699.

Laurence R. Horn, editor. 2010. *The Expression of Negation*. De Gruyter Mouton.

Md Mosharaf Hossain, Antonios Anastasopoulos, Eduardo Blanco, and Alexis Palmer. 2020a. It's not a non-issue: Negation as a source of error in machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3869–3885, Online. Association for Computational Linguistics.

Md Mosharaf Hossain, Kathleen Hamilton, Alexis Palmer, and Eduardo Blanco. 2020b. Predicting the focus of negation: Model and error analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8389–8401, Online. Association for Computational Linguistics.

Md Mosharaf Hossain, Venelin Kovatchev, Pranoy Dutta, Tiffany Kao, Elizabeth Wei, and Eduardo Blanco. 2020c. An analysis of natural language inference benchmarks through the lens of negation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9106–9118, Online. Association for Computational Linguistics.

Arian Hosseini, Siva Reddy, Dzmitry Bahdanau, R Devon Hjelm, Alessandro Sordoni, and Aaron Courville. 2021. Understanding by understanding not: Modeling negation in language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1301–1312, Online. Association for Computational Linguistics.

Rodney Huddleston and Geoffrey K. Pullum. 2002. *The Cambridge Grammar of the English Language*. Cambridge University Press.

Liwei Jiang, Antoine Bosselut, Chandra Bhagavatula, and Yejin Choi. 2021. "I'm not mad": Commonsense implications of negation and contradiction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4380–4397, Online. Association for Computational Linguistics.

Salud María Jiménez-Zafra, Roser Morante, Eduardo Blanco, María Teresa Martín Valdivia, and L. Alfonso Ureña López. 2020a. Detecting negation cues and scopes in Spanish. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6902–6911, Marseille, France. European Language Resources Association.

Salud María Jiménez-Zafra, Roser Morante, Maite Martin, and L. Alfonso Ureña-López. 2018. A review of Spanish corpora annotated with negation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 915–924, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Salud María Jiménez-Zafra, Roser Morante, María Teresa Martín-Valdivia, and L. Alfonso Ureña-López. 2020b. Corpora annotated with negation: An overview. *Computational Linguistics*, 46(1):1–52.

Aditya Khandelwal and Benita Kathleen Britto. 2020. Multitask learning of negation and speculation using transformers. In *Proceedings of the 11th International Workshop on Health Text Mining and Information Analysis*, pages 79–87, Online. Association for Computational Linguistics.

Aditya Khandelwal and Suraj Sawant. 2020. NegBERT: A transfer learning approach for negation detection and scope resolution. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5739–5748, Marseille, France. European Language Resources Association.

Dan Kondratyuk and Milan Straka. 2019. 75 languages, 1 model: Parsing Universal Dependencies universally. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2779–2795, Hong Kong, China. Association for Computational Linguistics.

Natalia Konstantinova, Sheila C.M. de Sousa, Noa P. Cruz, Manuel J. Maña, Maite Taboada, and Ruslan Mitkov. 2012. A review corpus annotated for negation, speculation and their scope. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 3190–3195, Istanbul, Turkey. European Language Resources Association (ELRA).

Robin Kurtz, Stephan Oepen, and Marco Kuhlmann. 2020. End-to-end negation resolution as graph parsing. In *Proceedings of the 16th International Conference on Parsing Technologies and the IWPT 2020 Shared Task on Parsing into Enhanced Universal Dependencies*, pages 14–24, Online. Association for Computational Linguistics.

Emanuele Lapponi, Erik Velldal, Lilja Øvrelid, and Jonathon Read. 2012. UiO 2: Sequence-labeling negation using dependency features. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 319–327, Montréal, Canada. Association for Computational Linguistics.

Lydia Lazib, Bing Qin, Yanyan Zhao, Weinan Zhang, and Ting Liu. 2020. A syntactic path-based hybrid neural network for negation scope detection. *Frontiers of computer science*, 14(1):84–94.

Junhui Li, Guodong Zhou, Hongling Wang, and Qiaoming Zhu. 2010. Learning the scope of negation via shallow semantic parsing. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 671–679, Beijing, China. Coling 2010 Organizing Committee.

Qianchu Liu, Federico Fancellu, and Bonnie Webber. 2018. NegPar: A parallel corpus annotated for negation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Edward Loper and Steven Bird. 2002. NLTK: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, pages 63–70, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Nick McKenna and Mark Steedman. 2020. Learning negation scope from syntactic structure. In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, pages 137–142, Barcelona, Spain (Online). Association for Computational Linguistics.

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

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Andrew Moore and Jeremy Barnes. 2021. Multi-task learning of negation and speculation for targeted sentiment classification. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2838–2869, Online. Association for Computational Linguistics.

Roser Morante and Eduardo Blanco. 2012. *SEM 2012 shared task: Resolving the scope and focus of negation. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 265–274, Montréal, Canada. Association for Computational Linguistics.

Roser Morante and Eduardo Blanco. 2021. Recent advances in processing negation. *Natural Language Engineering*, 27(2):121–130.

Roser Morante and Walter Daelemans. 2012. ConanDoyle-neg: Annotation of negation cues and their scope in conan doyle stories. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 1563–1568, Istanbul, Turkey. European Language Resources Association (ELRA).

Roser Morante, Sarah Schrauwen, and Walter Daelemans. 2011. Annotation of negation cues and their scope: Guidelines v1. *Computational linguistics and psycholinguistics technical report series, CTRS-003*, pages 1–42.

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 *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 69–78, Baltimore, Maryland. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Zhong Qian, Peifeng Li, Qiaoming Zhu, Guodong Zhou, Zhunchen Luo, and Wei Luo. 2016. Speculation and negation scope detection via convolutional neural networks. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 815–825, Austin, Texas. Association for Computational Linguistics.

Jonathon Read, Erik Velldal, Lilja Øvrelid, and Stephan Oepen. 2012. UiO1: Constituent-based discriminative ranking for negation resolution. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 310–318, Montréal, Canada. Association for Computational Linguistics.

Y. Ren, H. Fei, and Q. Peng. 2018. Detecting the scope of negation and speculation in biomedical texts by using recursive neural network. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 739–742, Los Alamitos, CA, USA. IEEE Computer Society.

Olivia Sanchez Graillet and Massimo Poesio. 2007. Negation of protein protein interactions: Analysis and extraction. *Bioinformatics (Oxford, England)*, 23:i424–32.

Anastassia Shaitarova and Fabio Rinaldi. 2021. Negation typology and general representation models for cross-lingual zero-shot negation scope resolution in Russian, French, and Spanish. In *Proceedings of the 2021 Conference of the North American Chapter of*

the Association for Computational Linguistics: Student Research Workshop, pages 15–23, Online. Association for Computational Linguistics.

Natalia Silveira, Timothy Dozat, Marie-Catherine de Marneffe, Samuel Bowman, Miriam Connor, John Bauer, and Chris Manning. 2014. A gold standard dependency corpus for English. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2897–2904, Reykjavik, Iceland. European Language Resources Association (ELRA).

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

György Szarvas, Veronika Vincze, Richárd Farkas, and János Csirik. 2008. The BioScope corpus: annotation for negation, uncertainty and their scope in biomedical texts. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*, pages 38–45, Columbus, Ohio. Association for Computational Linguistics.

Maite Taboada, Caroline Anthony, and Kimberly Voll. 2006. Methods for creating semantic orientation dictionaries. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy. European Language Resources Association (ELRA).

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.

Michael Wiegand, Alexandra Balahur, Benjamin Roth, Dietrich Klakow, and Andrés Montoyo. 2010. A survey on the role of negation in sentiment analysis. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, pages 60–68, Uppsala, Sweden. University of Antwerp.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Bowei Zou, Qiaoming Zhu, and Guodong Zhou. 2015. Negation and speculation identification in Chinese language. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 656–665, Beijing, China. Association for Computational Linguistics.

A Negation-Instance Based vs. Token-Level Weighting

We here explain the correspondence between NIS_{tok} and scope token-level scoring (ST) as employed in the *SEM 2012 task in greater detail. We use precision for our example, recall is computed analogously.

First, let us define the normalizing constant Z as the sum over all predicted scope lengths.

$$Z = \sum_{(c_p,s_p)\in I_p} |s_p|$$

For ST, precision is computed as:

$$P_{tok} = \frac{\sum_{(c_g,s_g)\in I_g,(c_p,s_p)\in I_p,c_g=c_p} |s_g \cap s_p|}{Z}$$

For NIS_{tok}, we compute precision as follows (in this formulation for simplicity disregarding the case that s_p could be empty).

$$P_{inst} = \sum_{(c_p,s_p)\in I_p,(c_g,s_g)\in I_g,c_g=c_p} \frac{1}{|I_p|} \cdot \frac{|s_g \cap s_p|}{|s_p|}$$

If we change the weighting of the scores that each instance contributes to the sum from uniform ($\frac{1}{|I_p|}$) to a weighting scheme that weights instances by their scope length ($\frac{|s_p|}{Z}$), we arrive at the token-level metric:

$$\begin{aligned} P_{tok} &= \sum_{(c_p,s_p)\in I_p,(c_g,s_g)\in I_g,c_g=c_p} \frac{|s_p|}{Z} \cdot \frac{|s_g \cap s_p|}{|s_p|} \\ &= \sum_{(c_p,s_p)\in I_p,(c_g,s_g)\in I_g,c_g=c_p} \frac{1}{Z} \cdot |s_g \cap s_p| \\ &= \frac{1}{Z} \sum_{(c_p,s_p)\in I_p,(c_g,s_g)\in I_g,c_g=c_p} |s_g \cap s_p| \\ &= \frac{\sum_{(c_g,s_g)\in I_g,(c_p,s_p)\in I_p,c_g=c_p} |s_g \cap s_p|}{Z} \end{aligned}$$

which corresponds to the definition of ST above.

B Hyperparameters

This section describes the hyperparameters used in the systems implemented by us, i.e., the dependency parsers and sequence taggers used for negation resolution.

B.1 XLM-R language model

For the underlying XLM-R language model, we use the same set of hyperparameters in all of our experiments. These are shown in Table 4.

Model	XLM-R-large
Token mask probability	0.15
Layer dropout	0.1
Hidden dropout	0.2
Attention dropout	0.2
Output dropout	0.5

Table 4: Hyperparameter values for the XLM-R language model.

B.2 BiLSTM-Tagger

Cue tagging. For the cue tagging subsystem, we simply use a linear layer with softmax on top of the XLM-R model. The model is then trained using the hyperparameters shown in Table 5.

Optimizer	AdamW
Weight decay	0
Batch size	8
Base learning rate	$2e^{-5}$
LR schedule	Noam
LR warmup	1 epoch

Table 5: Hyperparameter values for cue tagger.

Scope tagging. For the scope tagging subsystem, we add a 1-layer BiLSTM on top of the XLM-R model and then use a linear layer with softmax to classify tokens as part of negation scopes. In this system, we use different learning rates for the XLM-R model vs. the BiLSTM and the classifier. We found that using low learning rates for the entire system causes it to underfit. Furthermore, because the scope tagger is only trained on a smaller number of instances (i.e., only those that actually contain negation instances), we found it beneficial to reduce the batch size compared to the cue tagger. Our final hyperparameters can be found in Table 6.

BiLSTM	
BiLSTM layers	1
BiLSTM hidden size	2×200
BiLSTM dropout	0.0
cue/notcue embedding dim.	128
Optimization	
Optimizer	AdamW
Weight decay	0
Batch size	4
Base learning rate (BiLSTM)	$2e^{-4}$
Base learning rate (XLM-R)	$2e^{-5}$
LR schedule	Noam
LR warmup	1 epoch

Table 6: Hyperparameter values for scope tagger.

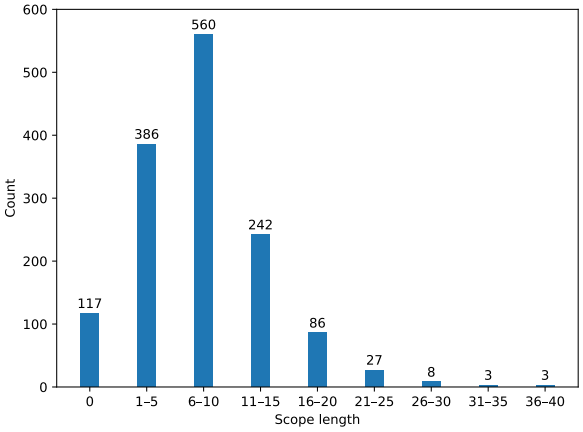

Figure 3: Negation instances by scope length for CD-neg.

B.3 Dependency Parser

Our parser implementation is based on the STEPS parser by Grünewald et al. (2020), which in turn is based on the unfactorized graph parsing approach by Dozat and Manning (2018). Table 7 gives the hyperparameters used for this system. Like the cue tagger, we use only a single learning rate for the entire system (XLM-R model as well as classifier).

Biaffine classifier	
Arc and label scorer dimension	1024
Dropout	0.33
Optimization	
Optimizer	AdamW
Weight decay	0
Batch size	32
Base learning rate	$4e^{-5}$
LR schedule	Noam
LR warmup	1 epoch

Table 7: Hyperparameter values for dependency parsing-based system.

C Additional Tables and Figures

Table 8 contains the full set of experimental results of our study.

Figure 3, Figure 4, and Figure 5 show negation instances by scope lengths for the three datasets used in our experiments.

Table 9 gives details on the datasets annotated for negation used in our study, while Table 10 provides statistics for our data splits. In addition, we will publicly release the **exact splits** (document IDs per dataset) upon publication of our paper.

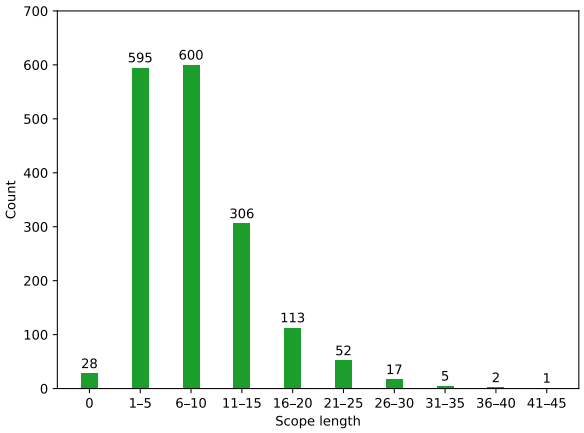

Figure 4: Negation instances by scope length for BioScope abstracts.

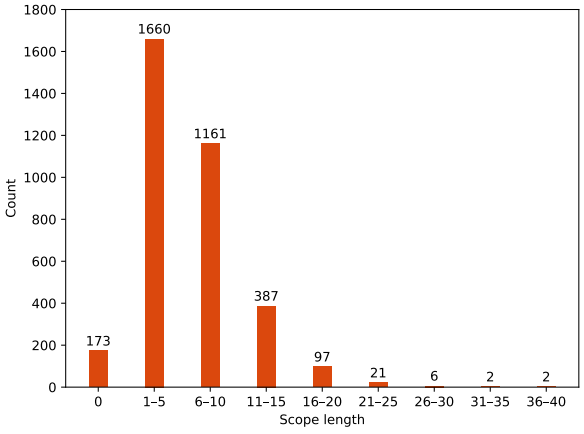

Figure 5: Negation instances by scope length for SFU Review.

System	Train	CD Test	Cues-B F_1	SCM F_1	SCM-B (NIS_{ex}) F_1	ST P	ST R	ST F_1	NIS_{tok} P	NIS_{tok} R	NIS_{tok} F_1
Punct-BL	–	*CD*	*100.0*	*17.0*	*9.9*	*89.8*	*62.0*	*73.3*	*93.3*	*58.8*	*72.1*
NegBERT	CD	CD	91.1	78.8	**70.1**	86.1	**87.8**	86.9	83.9	**88.7**	86.3
BiLSTM-Tagger	CD	CD	92.3	**79.6**	70.0	**90.8**	85.4	**88.0**	**90.3**	86.3	**88.2**
DP-direct	CD	CD	92.8	73.2	61.4	85.4	87.4	86.4	84.6	87.7	86.1
DP-nested	CD	CD	93.3	72.8	60.6	86.6	83.0	84.8	85.6	85.4	85.5
DP-direct-synt	CD	CD	92.8	79.1	69.2	87.4	86.5	86.9	86.2	88.3	87.2
DP-nested-synt	CD	CD	**93.4**	79.2	69.2	88.0	84.6	86.2	87.2	86.6	86.9
NegBERT	Bio	CD	65.2	13.0	8.9	75.6	43.7	55.2	71.1	40.0	51.1
BiLSTM-Tagger	Bio	CD	61.6	14.7	10.1	72.1	33.8	46.0	70.5	35.0	46.8
DP-direct	Bio	CD	65.2	13.1	8.9	74.1	36.2	48.5	74.3	35.1	47.6
DP-nested	Bio	CD	63.2	13.6	9.3	79.2	32.9	46.5	77.2	31.9	45.1
DP-direct-synt	Bio	CD	65.6	13.8	9.1	76.5	37.8	50.5	76.4	34.9	47.8
DP-nested-synt	Bio	CD	64.8	13.7	9.2	80.1	38.4	51.8	77.7	34.7	47.9
NegBERT	SFU	CD	69.3	15.2	9.7	74.4	46.1	56.7	70.2	43.6	53.6
BiLSTM-Tagger	SFU	CD	68.8	15.7	10.1	78.3	46.0	58.0	74.0	42.3	53.8
DP-direct	SFU	CD	69.2	14.2	9.1	78.0	42.8	55.3	74.9	40.0	52.1
DP-nested	SFU	CD	68.2	14.0	9.0	77.7	42.4	54.8	74.7	38.6	50.9
DP-direct-synt	SFU	CD	67.5	15.0	9.7	77.7	44.3	56.5	73.1	39.7	51.4
DP-nested-synt	SFU	CD	68.1	15.0	9.6	77.0	43.1	55.2	72.9	39.9	51.6
Punct-BL	–	*Bio*	*100.0*	*59.4*	*43.2*	*67.3*	*64.2*	*65.7*	*77.0*	*69.7*	*73.5*
NegBERT	Bio	Bio	90.5	85.6	79.5	81.0	88.3	84.5	83.3	91.5	87.2
BiLSTM-Tagger	Bio	Bio	94.3	**89.6**	**83.0**	88.9	88.5	88.6	91.7	91.7	91.7
DP-direct	Bio	Bio	**95.4**	83.8	74.3	85.6	89.2	87.3	88.7	92.4	90.5
DP-nested	Bio	Bio	95.3	87.4	79.8	89.1	87.9	88.5	91.2	91.2	91.2
DP-direct-synt	Bio	Bio	95.1	88.2	81.0	89.4	88.9	89.1	91.7	91.7	91.7
DP-nested-synt	Bio	Bio	**95.4**	89.1	82.7	**91.0**	**89.8**	**90.4**	**92.7**	**92.1**	**92.4**
NegBERT	CD	Bio	60.6	24.5	19.0	36.4	77.3	49.3	32.6	80.5	46.2
BiLSTM-Tagger	CD	Bio	64.3	24.6	17.7	45.3	76.1	56.7	41.2	80.1	54.3
DP-direct	CD	Bio	64.4	17.5	12.5	38.0	76.0	50.7	32.1	79.0	45.7
DP-nested	CD	Bio	65.8	18.5	13.1	40.0	78.7	53.0	34.9	78.9	48.4
DP-direct-synt	CD	Bio	64.7	22.2	16.3	40.3	78.0	53.1	34.8	80.3	48.5
DP-nested-synt	CD	Bio	65.3	25.2	18.7	42.2	77.6	54.7	37.2	80.6	50.9
NegBERT	SFU	Bio	78.7	64.7	55.5	73.9	55.4	63.2	76.8	66.1	71.0
BiLSTM-Tagger	SFU	Bio	81.0	64.3	53.4	74.8	58.8	65.9	77.8	68.4	72.8
DP-direct	SFU	Bio	81.6	59.6	48.3	75.4	55.9	64.2	79.2	66.0	72.0
DP-nested	SFU	Bio	82.1	61.1	49.9	77.1	57.4	65.8	80.0	66.8	72.8
DP-direct-synt	SFU	Bio	81.6	66.8	57.1	77.3	59.2	67.0	80.0	68.4	73.7
DP-nested-synt	SFU	Bio	82.0	67.0	57.2	77.8	59.2	67.2	80.1	68.8	74.0
Punct-BL	–	*SFU*	*100.0*	*56.9*	*45.9*	*55.0*	*84.0*	*66.5*	*75.5*	*87.4*	*81.0*
NegBERT	SFU	SFU	81.9	74.2	69.5	63.6	81.0	71.0	67.9	86.3	75.9
BiLSTM-Tagger	SFU	SFU	**86.7**	**78.0**	**73.2**	**71.1**	**86.0**	**77.9**	**74.9**	**89.8**	**81.7**
DP-direct	SFU	SFU	86.1	73.5	66.9	71.2	79.5	75.1	73.2	86.1	79.1
DP-nested	SFU	SFU	85.8	73.0	66.1	70.7	80.1	75.0	70.8	80.2	75.2
DP-direct-synt	SFU	SFU	86.6	74.6	68.5	71.8	80.8	76.0	73.4	87.2	79.6
DP-nested-synt	SFU	SFU	86.5	76.6	71.1	72.2	82.7	77.1	73.7	88.9	80.5
NegBERT	CD	SFU	51.3	14.4	10.0	27.2	58.8	37.2	26.6	63.6	37.5
BiLSTM-Tagger	CD	SFU	50.2	13.4	9.0	28.9	53.6	37.5	28.0	58.0	37.8
DP-direct	CD	SFU	50.6	10.3	6.9	25.9	53.5	34.9	26.4	56.9	36.1
DP-nested	CD	SFU	50.6	11.6	7.8	26.8	53.1	35.6	27.0	56.4	36.5
DP-direct-synt	CD	SFU	50.6	12.3	8.3	27.2	55.1	36.4	26.7	58.8	36.8
DP-nested-synt	CD	SFU	50.3	13.2	9.0	27.2	54.1	36.2	26.6	58.0	36.5
NegBERT	Bio	SFU	63.4	54.0	48.8	60.1	49.7	53.7	63.6	55.4	58.6
BiLSTM-Tagger	Bio	SFU	64.5	56.6	53.1	67.9	49.1	57.0	70.2	54.5	61.4
DP-direct	Bio	SFU	66.6	51.9	45.9	63.2	49.2	55.2	65.1	54.7	59.3
DP-nested	Bio	SFU	65.1	51.2	45.9	64.0	44.7	52.6	66.9	51.2	58.0
DP-direct-synt	Bio	SFU	56.1	48.6	39.4	46.6	47.7	47.0	48.6	53.5	50.8
DP-nested-synt	Bio	SFU	56.9	48.6	40.1	48.2	47.0	47.5	50.7	52.5	51.5

Table 8: Complete experimental results: For cues, we report the *SEM 2012 B version for exact cue matching (**Cues-B**). **SCM** refers to the standard version of Scope Level F1 of *SEM 2012, **SCM-B** to their B-version, which also corresponds to our **NIS_{ex}**. The punctuation baseline *Punct-BL* uses gold cues and predicts scopes as starting from a cue to the next punctuation token.

Dataset	***ConanDoyle-neg***	***BioScope Abstracts***	***SFU Review***
Source	Morante and Daelemans (2012)	Szarvas et al. (2008)	Konstantinova et al. (2012)
Domain	fiction writing	biomedical	review
Sentence #	5,520	11,871	17,263
Negation sentence #	1,227	1,597	3,117
Negation instance #	1,421 (original) / 1,432 (ours)	1,719	3,518
Annotated for speculation	no	yes	yes
Cue is a part of the scope	no	yes	no
Includes discontinuous scopes	yes	no	yes
Includes events	yes	no	no
Annotates negation affixes	yes	rarely, with the whole word as a cue	yes, but with the whole word as a cue
Tokenized	yes	no	yes
File format	CoNLL	XML	XML

Table 9: Overview of datasets annotated for negation used in our study.

Dataset		**Train**	**Dev**	**Test**	**Total**
ConanDoyle-neg	sentence #	3,644	787	1,089	5,520
	sentence %	66%	14.3%	19.7%	100 %
	negation instance #	984	173	264	1,421
	negation sentence #	848	144	235	1,227
	negation sentence %	23.3%	18.3%	21.6%	22.2 %
(reannotated)	negation instance #	987	176	269	1,432
BioScope Abstracts	sentence #	9,500	1,185	1,186	11,871
	sentence %	80%	10%	10%	100%
	negation instance #	1,396	156	167	1,719
	negation sentence #	1,297	148	152	1,597
	negation sentence %	13.7%	12.5%	12.8%	13.5%
SFU Review	sentence #	13,614	1,817	1,800	17,231
	sentence %	79%	11%	10%	100%
	negation instance #	2,835	365	309	3,509
	negation sentence #	2,503	328	276	3,107
	negation sentence %	18.4%	18.1%	15.3%	18%

Table 10: Dataset splits