

否定スコープ認識における構文構造利用の再検証

吉田 朝飛¹ 加藤 芳秀² 松原 茂樹^{1,2}

¹ 名古屋大学情報学部 ² 名古屋大学情報連携推進本部

yoshida.asahi.y7@s.mail.nagoya-u.ac.jp

概要

否定スコープ認識とは、自然言語文に現れる否定要素の影響が及ぶ範囲を特定するタスクである。従来手法の多くは否定スコープ認識に構文構造を利用していたが、近年の手法はディープラーニングを利用しており、構文構造を利用せずとも従来手法を上回る性能を達成している。本研究では、否定スコープ認識における構文構造の有用性を再検証する。従来手法で用いられてきた構文解析器を、解析精度の高い最新の構文解析器に置き換えて、否定スコープ認識の性能を評価する。精度の高い構文解析器を利用することにより、従来の手法が最新の手法と同程度の性能を達成できることを確認した。

1 はじめに

否定 (negation) は意味を反転させる言語現象であり、自然言語文において頻繁に出現する。文中の否定を意味する表現、及びその影響範囲を正しく認識することは、自然言語文を理解する上で必要不可欠である。NLP 分野において、否定を自動認識する多くの手法 (例えば [1, 2, 3]) が提案されてきた。

自然言語文に含まれる否定において、否定の意味をもつトークンを**否定要素** (negation cue)、否定要素の影響が及んでいる範囲を**否定スコープ** (negation scope) と呼ぶ。例えば (1) の英文では、not が否定要素であり、not によって意味が反転する He did, go to school が否定スコープである。なお、以下では例文における太字と下線はそれぞれ否定要素と否定スコープを表す。

(1) He did **not** go to school and stayed home.

一般的に、否定認識 (negation detection) は次のサブタスクに分けられる。

- 否定要素検出 (negation cue detection)
- 否定スコープ認識 (negation scope detection / negation scope resolution)

本研究では、英文の否定スコープ認識に取り組む。

否定スコープ認識の研究は、*SEM 2012 Shared Task [4] で否定認識タスクが取り上げられて以来、活発に行われている。*SEM 2012 Shared Task では、構文構造を利用した否定スコープ認識手法 [1] がトークンレベルの評価で最高性能を達成した [4]。一方、近年の研究ではディープラーニングを用いた手法が中心となっており、構文情報を明示的に利用せずとも高い性能を達成している [2, 3]。

ここで注意すべき点は、*SEM2012 Shared Task では、その当時の構文解析器が使用されている¹⁾ことである。現在の構文解析器は当時に比べ、解析精度が大幅に改善されており、それを利用することで、構文構造をベースとした従来手法の性能が向上する可能性がある。構文構造に基づく手法は、ディープラーニングをベースとする手法に比べて解釈性が高く、システムがどのようなメカニズムで否定スコープ認識を行っているかを説明しやすいという利点がある。また、構文ベースの手法が否定スコープの認識に利用する構文情報はドメインに依存して変化しないため、異なるドメインのデータセットに適用することが容易であると期待できる。

そこで本研究では、否定スコープ認識における構文構造の有用性を再検証するために、最新の構文解析器を用いた否定スコープ認識の性能を評価する。実験により、構文に関するヒューリスティックなルールに基づく従来手法が、ディープラーニングに基づく手法と同程度の性能を達成できることを確認した。

2 先行研究

本節では、Read らが提案した、構文構造を用いて否定スコープ認識を行う手法 [1] の概要を説明する。本研究ではこの手法をベースにし、否定スコープ認

1) *SEM2012 Shared Task のデータセットには、構文解析器による構文情報が付与されており、参加者はこの構文情報を利用している。

識における構文構造の有用性を再検証する。

Read らの手法は、否定スコープと構文構造は強い関連があるという考えに基づき、構文木の部分木(句)をスコープと対応づけている。例として、以下の文(2)に対応する図1の構文木を考える。

(2) People **without** a coupon can also enter.

この例において、否定要素 **without** のスコープは People, a coupon である。このスコープは句 NP と対応している。

Read らの手法は、文と否定要素が与えられると、次の手順により否定スコープを特定する。

1. 文を構文解析し、否定要素を含む句をそのスコープの候補とする。
2. 否定スコープの候補である句の中から、ルールベースの手法、もしくは分類器を用いて句を1つ選択する。
3. 選択された句を補正し、スコープを決定する。

手順1では、否定要素(上の例では **without**)が含まれる句(上の例では IN, PP, NP, S)をスコープの候補とする。手順2では、選ばれた候補の中から、スコープに対応する句を1つ選ぶ。Read らはルールベースの手法と分類器を用いる手法を提案しているが、ここでは本研究で利用するルールベースの手法について述べる。用いるルールは、図2に示す Scope resolution heuristics である。Scope resolution heuristics は複数のルールからなり、図の上にあるルールから順番に適用される。例えば、DT//SBAR if SBAR\WHADVP は、否定要素の品詞が DT であり、その祖先に SBAR が存在し、かつその SBAR の子に WHADVP がある場合に適用され、SBAR を否定スコープに対応する句として選択する。いずれのルールも適用されない場合には、default scope が採用される。default scope は、否定要素から左右へとスコープをのぼしていき、文境界または句読点に至るまでの範囲を否定スコープとするものである。

否定スコープと句の対応が単純でない場合もある。図3に、前述した英文(1)の構文木を示す。この文には等位構造が含まれているが、否定要素である **not** のスコープは接続詞 **and**、及び2番目の等位項 **stayed home** には及んでいない。この例が示すように、スコープと句を単純に対応させられない場合がある。手順3の句の補正の処理は、こうした場合に対応するための処理である。句の中に含まれる特

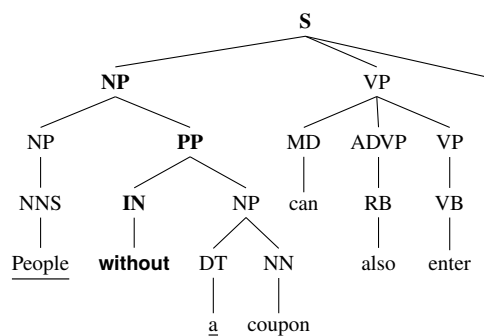


図1 文(2)の句構造木。否定要素 **without** のスコープに対応する句の候補を太字で示している。

```

RB//VP/SBAR if SBAR\WH* (#)
RB//VP/S
RB//S
DT/NP if NP/PP
DT//SBAR if SBAR\WHADVP
DT//S
JJ//ADJP/VP/S if S\VP\VB* [@lemma="be"]
JJ/NP/NP if NP\PP
JJ//NP
UH
IN/PP
NN/NP//S/SBAR if SBAR\WHNP
NN/NP//S
CC/SINV

```

図2 Scope resolution heuristics. 各行が1つのルールに対応し、ルール適用の優先度は上にあるものほど高い。A/B は B が A の親 (parent) であることを、A/B は B が A の祖先 (ancestor) であることを、A\B は B が A の子 (child) であることを示す。(＃) は、本研究で修正を加えるルールである。

定の要素を句から除外するために、以下の処理を行う。

- 句の最初あるいは最後の punctuation (句読点) を除外する。
- 5つのルールからなる Slackening rules を適用し、特定の要素を除外する。
- 不連続なスコープに対応するための Post processing を適用する。具体的には、以下の2つの処理を行う。
 - 等位構造の最後の等位項に否定要素が含まれる場合、それより前の等位項を除外する。
 - 文全体にかかる副詞的要素 (sentential adverb) を除外する。

例えば、文(1)では句として S が選択され、上述の処理により、等位接続詞 (CC) の **and**、CC より後ろにある等位項に含まれる **stayed home**、punctuation であるピリオドが除外される。補正の結果、スコープ

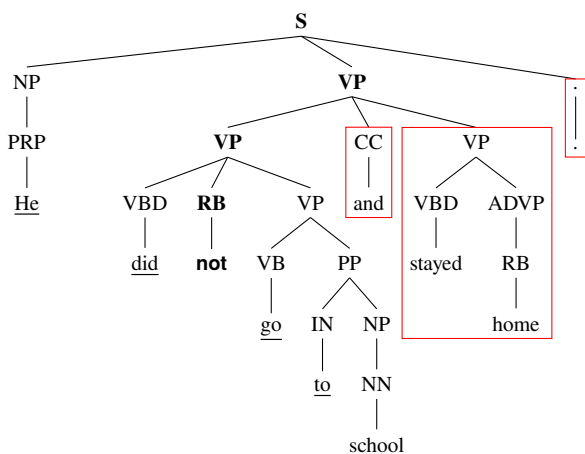


図3 文(1)の句構造木. 赤枠で囲った部分は句の補正によりスコープ候補の句から除外されることを示している.

を正しく認識できる.

3 構文構造を用いた手法の再検証

本節では、否定スコープ認識における構文構造の有用性を再検証する方法を説明する. 検証には、2節で説明した Read らの手法 [1] をベースとして用いる. 3.1 節では、本検証で用いる構文解析器について述べる. 3.2 節及び 3.3 節では、本再検証において修正した、Read らの手法の手順 2 と 3 の処理についてそれぞれ説明する.

3.1 構文解析器の変更

*SEM 2012 Shared Task のデータセット [5] には句構造情報も付与されており、構文構造をベースとした従来手法の多くが、この情報を利用して否定スコープ認識を行ってきた.

データセットに付与されている句構造情報は、Charniak and Johnson Reranking Parser [6] を用いて構文解析されたものである. この構文情報には誤りが含まれており、Read らが指摘したようにこの誤りが否定スコープ認識の性能に影響を及ぼす可能性がある. そこで本研究では、より性能の高い構文解析器を使用してデータセットの文を構文解析する. 構文解析には、Berkeley Neural Parser [7, 8] と Attach Juxtapose Parser [9] の 2 種類の構文解析器を使用する. それぞれの構文解析器の性能を表 1 に示す. Penn Treebank での性能は、Reranking Parser, Berkeley Neural Parser, Attach Juxtapose Parser の順に高くなっている.

表 1 使用した句構造解析器の性能

Parser	F ₁ score *
Reranking Parser (2005)	91.02
Berkeley Neural Parser (2018)	95.77
Attach Juxtapose Parser (2020)	96.34

* Penn Treebank Section 23 における性能である.

3.2 Scope Resolution Heuristics の修正

Read らは図 2 の Scope resolution heuristics によりスコープに対応する句を特定する処理を行った. Read らの 1 つ目のルール (図 2 で (#) を付したものは関係詞節を取り出すことを想定したもの) だが考えられるが、このままでは正しく機能しない. 本研究ではこのルールを以下のように修正する.

(3) RB//VP/S/SBAR if SBAR\WHNP

学習データを用いた予備実験により、Read らの 1 つ目のルールをそのまま適用する場合よりも、修正した新たなルールを適用した場合の方が、句と否定スコープの対応率が高いことを確認している.

3.3 補正処理の修正

Read らの手法では 2 節で述べた処理により句の補正を行っているが、本研究では Slackening rules と Post processing の一部に変更を加える.

Slackening rules については、Read らの 5 つのルールに、以下の 1 つのルールを追加する.

- 句の最初にあるカンマで区切られた PP (前置詞句) を除外する.

これは、句の最初にあるカンマで区切られた前置詞句は副詞的に働き、否定スコープに入らないことが多いという学習データの観察に基づいて作られたルールである. 文 (4) は、句の最初にある前置詞句が否定スコープに入らない例である.

(4) In my opinion, he should not go.

Post processing については、2 つ目の処理に修正を加える. Read らは学習データから sentential adverb のリストを作成し、そのリストに含まれる語を除外する処理を行っている. 本研究ではこれを一般化し、「カンマで囲まれた ADVP, INTJ」を sentential adverb とし、カンマと共に除外する. 例えば文 (5)

表2 異なる構文解析器を用いて作成したデータに対する否定スコープ認識の結果

Parser	Scope-level		Token-level			
	Pre.	Rec.	F ₁	Pre.	Rec.	F ₁
Reranking Parser	97.21 (97.14)	69.88 (68.27)	81.31 (80.19)	86.87 (85.48)	93.07 (93.63)	89.86 (89.37)
Berkeley Neural Parser	98.91 (98.88)	72.69 (70.68)	83.80 (82.43)	89.78 (87.88)	92.96 (93.57)	91.34 (90.64)
Attach Juxtapose Parser	98.94 (98.90)	74.70 (72.29)	85.13 (83.53)	90.62 (88.70)	94.68 (95.24)	92.61 (91.85)

* 上段の数値は Read らのルールを 3.2, 3.3 節で述べたように修正した場合の結果, 下段の数値はルールを修正しない場合の結果を表す。

では, カンマで囲まれた ADVP である *however* とその両端のカンマが句から除外される。

(5) I do n't think, however, he is guilty.

4 実験

否定スコープ認識における構文構造の有用性を再検証するために, 構文構造に基づく従来手法を異なる構文解析器を用いて作成したデータに適用する実験を実施した。4.1 節で実験の方法を述べ, 4.2 節で実験の結果を述べる。

4.1 実験の方法

実験では, *SEM2012 Shared Task で使用されたデータセット CD-SCO を用いた。Read らの手法を, 3.2 節及び 3.3 節で示した通りに修正した方法で否定スコープ認識を行った。CD-SCO に対して Berkeley Neural Parser と Attach Juxtapose Parser により句構造情報を新たに与え, それぞれの結果を比較した。実験設定は *SEM2012 Shared Task のものと同様であり, 評価指標はスコープレベルの F₁ (Scope-level F₁-measures) とトークンレベルの F₁ (Token-level F₁-measures) である。評価には *SEM2012 Shared Task で配布された評価プログラムを用いた。

4.2 実験結果

異なる構文解析器を使用した場合の否定スコープ認識の結果を表 2 に示す。解析精度の高い構文解析器を用いることにより, スコープレベル, トークンレベルのいずれの基準においても否定スコープ認識の性能が高くなることを確認した。

現時点で最高性能を達成している Khandelwal ら [3] を含め, いくつかの先行研究では, *SEM2012 Shared Task で評価から除外されていた punctuation のトークンを評価に含めている。これらの手法と結

表3 本研究で用いた手法と他の手法との結果比較

Method	Token-level F ₁	
	Punctuation を含む	Punctuation を除外
本研究で用いた手法 *	91.74	92.61
Fancellu ら [2]	88.72	-
Li ら [10]	-	89.4
Khandelwal ら [3]	92.36	-

* Attach Juxtapose Parser を用いて構文解析したもの。

果を比較するため, punctuation のトークンも含めた F₁ も測定した。結果を表 3 に示す。本研究で検証した手法の性能は, punctuation を含めた F₁ において, Khandelwal らの手法とわずか 0.62% の性能差に達している。これは, 高性能の構文解析器を用いることで, 構文に関するヒューリスティックなルールに基づく手法が最高性能のディープラーニング手法に近い性能を達成できることを示している。

5 おわりに

本研究では, 否定スコープ認識タスクにおける構文構造の有用性を再検証した。構文構造をベースとする従来手法が解析精度の高い構文解析器を用いることで, ディープラーニングをベースとした手法に近い性能を達成できることを実験により示した。

予測の精度をより高めるために, ルールベースで行った句を選択する処理に機械学習を導入することが考えられる。

否定スコープ以外にもスコープのある言語現象が存在する。例えば, *all*, *any* などの量量子 (*quantifier*) もスコープがあり, その範囲を正しく認識することは, 量量子を含む自然言語文の意味を理解する上で重要である。本研究の方式を否定以外のスコープ認識に応用させることも今後の課題である。

謝辞

本研究は、一部、科学研究費補助金基盤研究（C）（No. 22K12148）により実施したものである。

参考文献

- [1] Jonathon Read, Erik Velldal, Lilja Øvrelid, and Stephan Oepen. UiO1: Constituent-based discriminative ranking for negation resolution. In ***SEM 2012: The 1st Joint Conference on Lexical and Computational Semantics**, pp. 310–318, Montréal, Canada, 7-8 June 2012. Association for Computational Linguistics.
- [2] Federico Fancellu, Adam Lopez, and Bonnie Webber. Neural networks for negation scope detection. In **Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 495–504, Berlin, Germany, August 2016. Association for Computational Linguistics.
- [3] Aditya Khandelwal and Suraj Sawant. NegBERT: A transfer learning approach for negation detection and scope resolution. In **Proceedings of the 12th Language Resources and Evaluation Conference (LREC'20)**, pp. 5739–5748, Marseille, France, May 2020. European Language Resources Association.
- [4] Roser Morante and Eduardo Blanco. *SEM 2012 shared task: Resolving the scope and focus of negation. In ***SEM 2012: The 1st Joint Conference on Lexical and Computational Semantics**, pp. 265–274, Montréal, Canada, 7-8 June 2012. Association for Computational Linguistics.
- [5] Roser Morante and Walter Daelemans. ConanDoyle-neg: Annotation of negation cues and their scope in conan doyle stories. In **Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'12)**, pp. 1563–1568, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA).
- [6] Eugene Charniak and Mark Johnson. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In **Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)**, pp. 173–180, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.
- [7] Nikita Kitaev and Dan Klein. Constituency parsing with a self-attentive encoder. In **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 2676–2686, Melbourne, Australia, July 2018. Association for Computational Linguistics.
- [8] Nikita Kitaev, Steven Cao, and Dan Klein. Multilingual constituency parsing with self-attention and pre-training, 2018.
- [9] Kaiyu Yang and Jia Deng. Strongly incremental constituency parsing with graph neural networks. In **Neural Information Processing Systems (NeurIPS)**, 2020.
- [10] Hao Li and Wei Lu. Learning with structured representations for negation scope extraction. In **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Pa-**

pers), pp. 533–539. Association for Computational Linguistics, 2018.