

複数人議論における発話間の関係を対象とした関係分類

姫野拓未

九州工業大学大学院情報工学府
t_himeno@pluto.ai.kyutech.ac.jp

嶋田和孝

九州工業大学情報工学部
shimada@pluto.ai.kyutech.ac.jp

1 はじめに

研究室や会社では、新しい研究のアイデアや経営戦略を打ち出すために会議が日々行われている。そこで行われた会議内容を会議に参加していない人に共有するためには、議事録が不可欠である。会議に参加していない人が議事録を読み返すことは議論全体の論点を把握することには適しているが、会議で結論に達するまでの流れを把握することは難しい。そこで、会議の議事録を議論構造として表すことができれば、発話間の関係を捉えることができ、結論に達するまでの議論の流れや議論全体の論点を素早く把握することができる。また、その発話間の関係の役割を捉えることができれば、発話の意図を捉えることができ、議事録理解の支援ができると考えられる。

文章を構造化するタスクの一つとして議論マイニングがある [1]。議論マイニングとは、例えば小論文のような文章を入力文として、主張を表している文と、その主張を支持したり反論したりする文を自動的に判別するタスクである。他にも、文書要約 [2] や自動採点 [3]、論文執筆支援 [4] など数多くの自然言語処理タスクで取り込まれている。一般に議論マイニングは要素分類、要素抽出、関係抽出、関係分類の4つのサブタスクから構成されている。

我々は以前の研究において、複数人議論を対象に議論マイニングのサブタスクの中でも議論構造の構築のために重要なタスクである発話間の関係抽出に焦点を当てて研究を行った [5]。関係抽出は議論構造に関連する文や節から成り立つ2つの論理要素に対して関係の有無を推定するタスクである。発話間の関係抽出を行うことによって、議論構造において議論の流れを把握を支援できる。しかし、以前の研究だけでは議論構造における発話間の関係の役割を捉えることができない。そこで本研究では、複数人議論を対象に関係抽出後のタスクである発話間の関係分類に焦点を当てる。関係分類は論理要素のペア

を対象に、主張に対する支持や反論のような関係を持つ役割のラベルを割り当てるタスクである。例えば Stab らは学生が書いたエッセイを対象に関係分類を取り組んでいる [6]。Stab らはエッセイの特徴を捉えた素性を作成し、論理要素の関係分類を行っている。対象としている関係では主張を支持する関係と反論する関係の2種類のみである。一方で、本研究で対象とする複数人議論では主張を支持・反論する関係だけでなく、選択肢のある質問から回答を選択する関係などエッセイには存在しない関係が多く存在する。

本研究では、複数人議論を対象に Self-Attention を用いて発話内容に着目したモデルを構築し関係分類を行う。本研究の貢献は次の2点である。

- 様々な言語の分散表現や Self-Attention を用いて関係分類モデルを構築した。
- BERT [7] に基づく関係分類結果より本研究で作成したモデルの方が高精度であることを確認した。

2 データセット

本研究では複数人議論コーパスの一つである AMI corpus を用いる [8]。AMI corpus はあらかじめ議題が与えられている状態で行われるシナリオ会議を収録したコーパスであり、書き起こしデータが公開されている¹⁾。議題の設定として、架空の家電企業に勤める異なる役職の4人の従業員が、市場に出回っている古いデザインのテレビリモコンの代わりとなる新しいテレビリモコンを開発するという議題で計4回行われる。

また本研究では発話間の関係分類タスクのためのデータセットとして、Twente Argument Schema (TAS) を用いる [9]。TAS は AMI corpus のシナリオ会議の議論で生じる議論構造を明らかにするために作成されたアノテーションスキーマである。TAS における

1) <http://groups.inf.ed.ac.uk/ami/download/>

表 1 Relation Label とその説明

ラベル	説明
Positive	主張の支持
Negative	主張の否定
Uncertain	Positive か Negative か曖昧な関係
Request	より多くの情報を求めたり、もう一度促す関係
Specialization	前の発話を具体化する関係
Elaboration	前の発話を繰り返す関係
Option	選択肢のある質問から回答を選択する関係
Option exclusion	上記の否定
Subject-to	特定の発話が他の発話で真実であるか理解できる関係

議論構造は2つの要素から成り立っている。ある話者の発話全体、また発話の一部、複数の発話から成り立っているノードとそのノード間を結ぶエッジから成り立っている。また、エッジにはノード間どのような役割であるかを示す Relation Label が付与されている。Relation Label の詳細を表 1 に示す。ある主張に対して支持する関係である「Positive」や、より多くの情報を求めたりもう一度促したりするような関係である「Request」など計9種類のタグが付与されている。さらに、TAS では一つの会議(ダイアログ)の中である議題が挙げられてから、その議題が結論に達した時やその議題を途中で中断したり放棄したりするまでをディスカッションと定義している。各ディスカッションは一つのダイアログ中に一つまたは複数存在し、各ノードは一つのディスカッションに複数存在する。本研究では各ディスカッション内のノード間を結ぶエッジに付与された Relation Label を分類する実験を行う。

3 手法

本研究で用いたモデルの概要図を図 1 に示す。ディスカッション内で関係がある二つのノードペアをそれぞれ Bi-LSTM を用いて隠れ層を算出する。次に、それぞれの隠れ層に対して Self-Attention を用いて隠れ層を更新する。Self-Attention によって得られた隠れ層を連結し、最終層が作成される。その層を Softmax 関数に入力することによって Relation Label を分類する。

3.1 入力層

AMI corpus のディスカッション内で関係を持つ任意のノードペア $node_i = (w_1, w_2, \dots, w_k), node_j = (w_1, w_2, \dots, w_l)$ を入力とする。ここでの w はノード内の単語を表しており、 k と l はノードの単語長を表している。次に、ノード内の単語を4種類の分散表現の中から1つまたは複数の分散表現を連結してベクトル $V_i = (v_1, v_2, \dots, v_k), V_j = (v_1, v_2, \dots, v_l)$ に変換する。4種類の分散表現は Word2Vec²⁾ (図 1 における W), GLoVe³⁾ (G), ELMo⁴⁾ (E), Transformer⁵⁾ (T) であり、 v_x は w_x の単語ベクトルを表す。Transformer のモデルには bert-base-uncased⁶⁾ を利用する。

3.2 Bi-LSTM

本研究では Bi-LSTM [10] を用いてノード全体のベクトルを獲得する。それぞれのベクトル変換したノードの単語ベクトルを Bi-LSTM で以下のように算出する。

$$\vec{h}_x = \overrightarrow{LSTM}(v_x, \vec{h}_{x-1}) \quad (\text{順方向}) \quad (1)$$

$$\overleftarrow{h}_x = \overleftarrow{LSTM}(v_x, \overleftarrow{h}_{x+1}) \quad (\text{逆方向}) \quad (2)$$

$$h_x = [\vec{h}_x; \overleftarrow{h}_x] \quad (3)$$

h_x は Bi-LSTM モデルの隠れ層のベクトルであり、 $[\cdot; \cdot]$ は二つのベクトルの連結を表す。それぞれの Bi-LSTM から出力された隠れ層全体のベクトル H_i, H_j は以下ようになる。

$$\begin{cases} H_i = (h_1, h_2, \dots, h_k) \\ H_j = (h_1, h_2, \dots, h_l) \end{cases} \quad (4)$$

3.3 Self-Attention

Bi-LSTM の出力層を基に Self-Attention を算出する。Lin ら [11] が提案した Self-Attention を使用し、重みを以下のように算出する。

$$\alpha_x = \text{softmax}(\omega^T \tanh(H_x)) \quad (5)$$

ω は学習可能なパラメータのベクトルである。そして、この重みを利用して以下のようにそれぞれの Bi-LSTM の最終的な隠れ層を算出する。

$$\begin{cases} H_{i\alpha} = \alpha_i H_i \\ H_{j\alpha} = \alpha_j H_j \end{cases} \quad (6)$$

2) <http://code.google.com/archive/p/word2vec/>

3) <https://nlp.stanford.edu/projects/glove/>

4) <https://allennlp.org/elmo>

5) <https://github.com/huggingface/transformers>

6) <https://huggingface.co/bert-base-uncased>

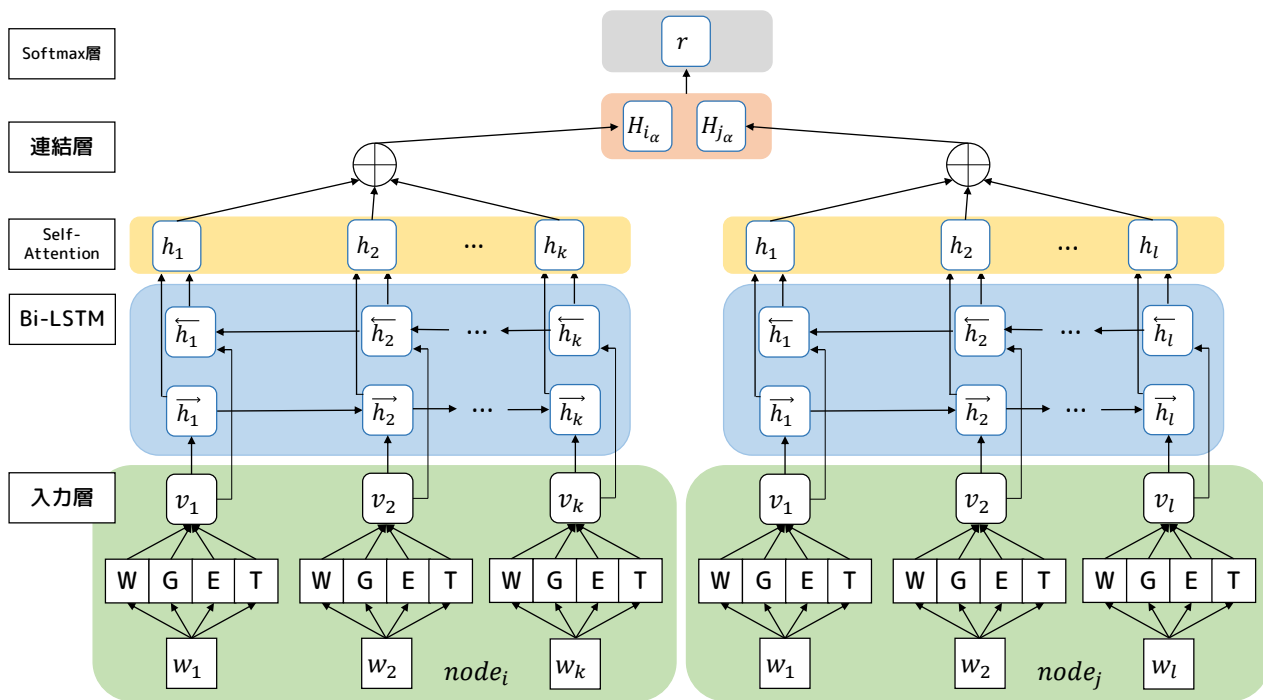


図1 Bi-LSTMを用いたモデルの概要図

3.4 出力層

それぞれの Bi-LSTM の出力層を連結してクラスラベルに対応するインデックスを出力する。まず、それぞれの Self-Attention の出力層を以下のように連結する。

$$H^* = [H_{i_\alpha}; H_{j_\alpha}] \quad (7)$$

連結した層を用いて、クラスラベル r を以下のように算出する。

$$\hat{y}_r = \text{softmax}(WH^* + b) \quad (8)$$

$$r = \arg \max_{r \in R} \hat{y}_r \quad (9)$$

W, b は学習可能なパラメータであり、 \hat{y}_r はクラスラベルに対応する確率値、 R は Relation Label である。

3.5 訓練

モデルの訓練について説明する。損失関数には NLLoss を利用し、最適化手法には Adam [12] を利用する。学習率は 0.001 に設定し、バッチサイズは 128、隠れ層の次元は 300 に設定する。

4 実験

本節では、4.1 節において 2 節で述べた複数人議論コーパスを用いた実験設定について説明する。4.2 節において、分類結果について説明する。

4.1 実験設定

本研究で用いた実験データの詳細について説明する。実験データは、AMI corpus のダイアログ 92 対話から作成された 219 個のディスカッションを用いて実験を行った。訓練データ、開発データ、評価データに AMI corpus をダイアログ単位で 84 対話、4 対話、4 対話と分割した。また、それぞれのディスカッションの数は 201 個、13 個、12 個、関係分類で用いるノードペアの数は 4001 個、241 個、242 個となっている。データ分布の詳細を表 2 に示す。表 2 より、各データにおいて「Positive」の数が多く、偏りがあるデータ分布となっている。

本研究の比較対象として、BERT [7] を baseline として利用する。BERT は大規模なテキストコーパスを用いて事前学習を行った後、タスクごとに fine-tuning する汎用言語モデルである。

4.2 実験結果

表 3 に各モデルの分類結果を示す。評価値を適合率 (Precision)、再現率 (Recall)、F1 値 (F-score) で評価した。表 3 では利用した分散表現を示した Bi-LSTM のみのモデルとそのモデルに対して Self-Attention を組み込んだモデル (各行の “+ Self-Attention”) それぞれで評価している。表 3 より、分散表現のみのモデルより Self-Attention を用いたモデルの分類結果が良

表2 データ分布の詳細

	ダイアログ	ディスカッション	Relation Label								
			Pos	Neg	Unc	Req	Spe	Ela	Opt	Opt ex	Sub-to
訓練データ	84	201	1846	373	219	195	113	156	517	10	219
開発データ	4	13	103	15	13	12	2	57	26	2	11
評価データ	4	12	125	26	11	6	7	34	25	0	8

表3 発話間の関係分類を行った実験結果

Model	Precision	Recall	F-score
<i>Bi-LSTM_W</i>	0.48	0.51	0.49
+ Self-Attention	0.51	0.52	0.51
<i>Bi-LSTM_G</i>	0.53	0.55	0.54
+ Self-Attention	0.55	0.55	0.54
<i>Bi-LSTM_E</i>	0.57	0.60	0.58
+ Self-Attention	0.60	0.59	0.59
<i>Bi-LSTM_T</i>	0.53	0.54	0.54
+ Self-Attention	0.61	0.62	0.61
<i>Bi-LSTM_{W+G+E+T}</i>	0.56	0.59	0.57
+ Self-Attention	0.58	0.59	0.57
<i>Bi-LSTM_{E+T}</i>	0.60	0.60	0.59
+ Self-Attention	0.62	0.62	0.61
BERT (baseline) [7]	0.59	0.61	0.59

いことが確認できた。このことより、Self-Attentionが複数人議論の関係分類においても有効であることが確認できる。また特に、Transformerを分散表現に使用しSelf-Attentionを利用したモデルと、ELMoとTransformerを分散表現に使用しSelf-Attentionを利用したモデルは、fine-tuningしたBERTよりF1値が高いことを確認できた。

次にfine-tuningしたBERTと*Bi-LSTM_{E+T}*の各Relation LabelのF1値の詳細を表4に示す⁷⁾。表4より、fine-tuningしたBERTと比較して*Bi-LSTM_{E+T}*は少数ラベル(例えば事例数が7であるSepなど)であっても少なくともいくつかは正しく分類できるモデルであることを確認できた。しかし、「Uncertain」のようなPositiveかNegativeか曖昧な関係ラベルではfine-tuningしたBERTよりF1値が劣っている。これは本手法がノード自身にSelf-Attentionを付与した後それぞれのノードを連結して分類するモデルであることに起因すると考えられる。つまり、それぞれのノードが独立でもう一つのノードに対してノード

7) Option exclusionは評価データの中に存在しなかったため表から除く。

表4 2つのモデルの詳細なラベルの実験結果の比較

	<i>Bi-LSTM_{E+T}</i>	BERT (baseline) [7]
Pos	0.81	0.80
Neg	0.30	0.30
Unc	0.21	0.37
Req	0.33	0.53
Spe	0.55	0.00
Ela	0.49	0.48
Opt	0.45	0.48
Sub-to	0.22	0.00

ド情報を全く与えていないことが問題となっている。一方でBERTはノードペアを同一のモデルに入力するためそれぞれのノードが相互にノード情報を与え分類することができる。このことから、それぞれのノードに対して相互にノード情報を与えるモデルの構築が今後の課題となる。

5 おわりに

本研究では、複数人議論コーパスを対象に発話間の関係を対象とした関係分類を行った。複数人議論コーパスにはAMI corpusを用いた。次に、発話間の関係分類を行うために複数の分散表現を組み合わせ、Self-Attentionを用いたモデルを構築し関係分類実験を行った。その結果、*Bi-LSTM_{E+T}* + Self-Attentionモデルの推定精度が最も高くなることを確認した。また、fine-tuningしたBERTの分類結果より構築したモデルの分類結果の方がよいことを確認した。さらに、構築したモデルではfine-tuningしたBERTより少数ラベルであっても少なくともいくつかは分類できることを確認した。今後の課題としては、「Uncertain」のような提案したモデルで分類することが難しいラベルに対応するため、ノードの情報を相互に組み込んだモデルを構築することが挙げられる。

謝辞

本研究は科研費20K12110の助成を受けたものです。

参考文献

- [1] Christian Stab and Iryna Gurevych. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, Vol. 43, No. 3, pp. 619–659, 2017.
- [2] Emma Barker and Robert Gaizauskas. Summarizing multi-party argumentative conversations in reader comment on news. In *Proceedings of the Third Workshop on Argument Mining (ArgMining2016)*, pp. 12–20, 2016.
- [3] Debanjan Ghosh, Aquila Khanam, Yubo Han, and Smaranda Muresan. Coarse-grained argumentation features for scoring persuasive essays. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 549–554, 2016.
- [4] Christian Stab and Iryna Gurevych. Training argumentation skills with argumentative writing support. In *Proceeding of SEMDIAL 2017 (SemDial) Workshop on the Semantics and Pragmatics of Dialogue*, pp. 166–167, 2017.
- [5] Takumi Himeno and Kazutaka Shimada. Relation identification using dialogical features in multi-party conversation. In *Proceedings of the 8th International Symposium on Applied Engineering and Sciences*, 2020.
- [6] Christian Stab and Iryna Gurevych. Identifying argumentative discourse structures in persuasive essays. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 46–56, 2014.
- [7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, pp. 1–16, 2018.
- [8] Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, et al. The ami meeting corpus: A pre-announcement. In *Proceedings of International workshop on machine learning for multimodal interaction*, pp. 28–39. Springer, 2005.
- [9] Rutger Rienks, Dirk Heylen, and Erik van der Weijden. Argument diagramming of meeting conversations. In *In Proceedings of Multimodal Multiparty Meeting, Workshop at the 7th International Conference on Multimodal Interfaces*, pp. 85–92, 2005.
- [10] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, Vol. 45, No. 11, pp. 2673–2681, 1997.
- [11] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130*, 2017.
- [12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.