

複数人議論の構造解析に向けた発話間の関係の有無の推定

姫野 拓未[†] 嶋田 和孝^{††}

[†]九州工業大学 大学院情報工学府

^{††}九州工業大学 大学院情報工学研究院

福岡県飯塚市川津 680-4

E-mail: †{t_himeno,shimada}@pluto.ai.kyutech.ac.jp

あらまし 研究室や会社では、会議が日々行われている。そこで行われた議論内容を会議に参加していない人に共有するには、議事録が不可欠である。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、議論全体の流れや論点を素早く把握することができる。本研究では、複数人議論コーパスを対象に構造解析に向けた発話間の関係の有無を推定するモデルを構築する。まず、複数人議論の特徴を捉えた素性について説明する。そして、機械学習のモデルで素性を用いる場合と用いない場合の推定精度の比較を行い、提案した素性の有用性を確認する。実験の結果、時間情報と発話間の距離の素性が有効な素性であることを確認した。

キーワード 複数人議論, 議論マイニング, 議論構造

Predicting Connection between Utterances in Multi-Party Conversation

Takumi HIMENO[†] and Kazutaka SHIMADA^{††}

[†] Department of Artificial Intelligence, Kyushu Institute of Technology
680-4 Kawazu, Iizuka, Fukuoka 820-8502, JAPAN

^{††} Department of Artificial Intelligence, Kyushu Institute of Technology
680-4 Kawazu, Iizuka, Fukuoka 820-8502, JAPAN

E-mail: †{t_himeno,shimada}@pluto.ai.kyutech.ac.jp

Abstract Conversations are often held in laboratories and companies. To grasp the content of a discussion, a summary is important for people who could not attend the conversation. If the summary is illustrated as an argument structure, it will be helpful to follow the points of the conversation immediately. In this paper, we construct a model that predicts connection between utterances in multi-party conversation. First, we explain the features we design for the task. Next, we report the result of an experiment in which we compared a machine learning method using the feature with one without them. The result showed that time information and distance between utterances were effective.

Key words Multi-Party Conversation, Argument Mining, Argument Structure

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素早く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議論の流れを把握することが難しい。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挙げられる [1]。議論マイニングとは、小論文のような文章の入力文に対して、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。議論マイニングを構成する要素として要素抽出、要素分類、関係抽出、関係分類の4つのサブタスクから成り立っている。要素抽出は、文章中の論理構造に関係する文や節から成り立つ論理要素を抽出するタスクである。また、要素分類は抽出された論理要素に対して主張や前提のような役割のラベルを割り当てるタスクである。さらに、関係抽出は抽出された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。最後に、関係分類は関係抽出で得られた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。

本研究では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実験を行う。ここでの発話間の関係とは、主張を表す発話に対して肯定的な意見や否定的な意見のような反応であったり、質問を表す発話に対して回答を行う発話であったりする発話関係が含まれる。まず、発話間の関係を推定するために用いた複数人議論コーパスについて説明する。次に、そのデータを用いて、機械学習による発話間の関係の有無を推定するモデルの構築、およびそのモデルに用いる素性について述べる。そして構築したモデルを用いて、素性を用いた場合と用いない場合の推定精度の比較や推定結果に対するエラー分析、モデルに用いた各素性の有効性を評価する。

2. 関連研究

議論全体の論点を理解することに関する研究に議事録を自動生成しているものがある。このタスクでは、議論の重要な発話を抽出し、抽出した発話の統合や補完を行うことで、会議内容の要約を生成する手法が存在する [2]。しかし、議論全体の論点を理解することには適しているが、重要文の抽出のみでは議論の結論に至るまでの流れを理解することには適していない。本研究では、議論全体の論点を理解することに加え、結論に達するまでの議論の流れを素早く把握することを考慮する。

また、近年議論マイニングの研究が盛んに行われている。議論マイニングは、文章を構造化することに適しており、文書要約や自動採点、論文執筆支援など数多くの自然言語処理のタスクで取り組まれている [3]~[7]。このタスクの中でも、学生が書いたエッセイに対して議論マイニングに取り組み、要素間の関係抽出に取り組んでいるタスクがある [8]。エッセイの特徴を捉えた素性を作成し、論理要素間の関係の有無の推定を行っている。しかし、対象としている文章が書き言葉であるエッセイは、文章の形が主張の文の直後に前提の文があるように決まった形で形式化されており、自由な発話で形式化されていない複数人議論とは異なる。そこで、本研究では書き言葉で考慮できない時間情報など、複数人議論の特徴を捉えた素性を作成する必要がある。また、要素間の関係の有無を推定する研究として、Pointer Network を用いた関係抽出手法を提案されている [9]。しかし、文章の段落内に関係を持つ要素がある前提で取られ、複数人議論のような議題の移り変わりが不規則な文章全体に対してこの手法は適していない。本研究では、複数人議論の一つの議題を対象に議論マイニングのサブタスクである関係抽出を行うモデルの構築を行う。また、複数人議論の特有の特徴である話者情報や時間情報などを捉えた素性の作成を行う。

3. データセット

複数人議論に議論マイニングのサブタスクである関係抽出を行うためには、素性に必要なアノテーションや発話間の関係がアノテーションされた複数人議論コーパスが必要となる。そこ

表 1 談話行為タグの詳細

タグ	詳細
Backchannel	相槌などの発話
Stall	フィラーなどの話し始めの言いよどみ
Fragment	言いかけてやめた発話
Inform	話者が聞き手に情報を与える発話
Elicit-inform	話者が聞き手に情報を求める発話
Suggest	話題の提案など 聞き手に働きかける発話
Offer	話者自身の行動を申し出る発話
Elicit-Offer-or-suggestion	話題の提案や 行動の申し出を求める発話
Assess	前の発話に対して 評価を与える発話
Comment-about-Understanding	話者自身の理解、 不理解を示す発話
Elicit-Assessment	自分の発話を相手が 理解できたか確認する発話
Elicit-Comment-about-Understanding	話者自身の発話を聞き手が 理解できたか確認する発話
Be-Positive	聞き手との信頼関係に 良い影響を与える発話
Be-Negative	聞き手の気分を害する 冗談や攻撃的な発話
Other	上記に該当しない 発話意図をもつ発話

表 2 Unit label の詳細

label	詳細
Statement	肯定的な主張
Weak Statement	否定的な主張
Open Issue	自由な受け答えできる質問
A/B Issue	選択肢を与える質問
Yes/No Issue	Yes か No で答える質問

で、本研究では複数人議論コーパスである AMI corpus [10] を用いる。AMI corpus はあらかじめ議題が与えられている状態で行われているシナリオ会議を収録したコーパスで、書き起こしデータが公開されている。議論の設定として、架空の家電企業に務める異なる役職の 4 人の従業員が、市場に出回っているユーザーに不親切で魅力の無い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。AMI corpus の各発話には話者 ID や時間情報に加えて、談話行為タグが付与されている。談話行為タグとは、各発話がどのような機能を持つかを示すタグである。談話行為タグの詳細を表 1 に示す。何らかの情報を提供している発話に付与される「Inform」や相槌などの発話に付与される「Backchannel」など計 15 種類の談話行為タグが付与されている。これらのタグを用いれば、複数人議論の特徴を捉える素性を作成することができる。

また本研究では、発話間の関係の推定を行うため、Twente Argument Schema (TAS) [11] を用いる。TAS はシナリオ

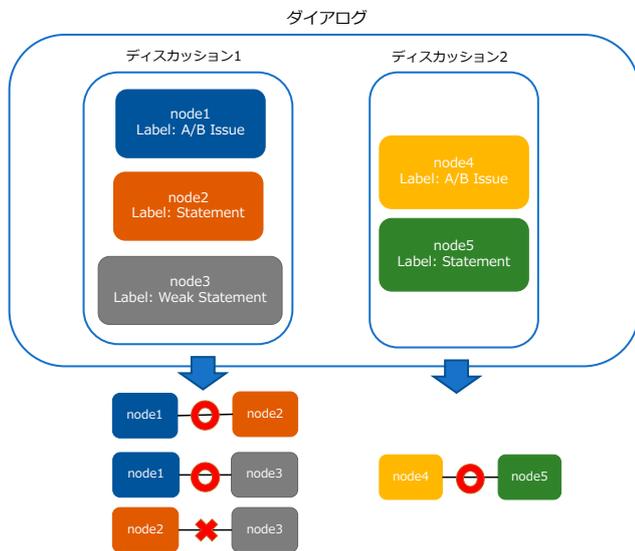


図1 TASを用いた関係抽出のイメージ図

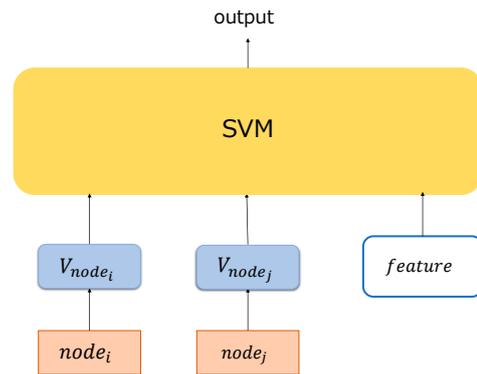


図2 SVMを用いたモデルの概要図

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における議論構造は、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つノードと、そのノード間を結ぶエッジから成り立っている。また、ノードがどのような役割を持つかを表している Unit Label が付与されている。Unit label の詳細を表 2 に示す。肯定的な主張を表す「Statement」や回答に対して選択しを与える質問を表す「A/B Issue」など計 5 種類の Unit Label 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の一つの議題をディスカッションと定義されている。各ディスカッションは、一つのダイアログの中に一つ、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。また、各ノードには一つ Unit label が付与されている。

図 1 に TAS を用いた関係抽出の例を示す。図 1 では、ディスカッション 1 に 3 つのノードが存在し、ディスカッション 2 に 2 つのノードが存在するダイアログの例となっている。このダイアログ中の各ディスカッションの中から 2 つのノードを総当たりで抽出する。図 1 では、ディスカッション 1 から $node_1 \sim node_3$ の 2 つのノードを総当たりで 3 組抽出する。ディスカッション 2 からは $node_4$ と $node_5$ の 1 組を抽出する。次に、抽出したノードペアに対して関係の有無を推定する。図 1 では、 $node_2$ と $node_3$ でペアで関係がないとみなされ、他のペアは関係があるとみなされた例である。本研究では、抽出したノードペアに対して関係の有無の推定を行う。

4. 提案手法

本節では機械学習を用いて発話間の関係の有無を推定するモデルを構築する。4.1 節では、SVM と bi-LSTM の 2 つのモデルを作成する。2 つのモデルではノード間の関係の有無を推定するために単語の分散表現を用いて構築している。しかし、単語の分散表現のみでは複数人議論特有の特徴を捉えることができない。そこで、4.2 節では複数人議論の特徴を捉え作成した素性について説明する。

4.1 モデル

本節では、発話間の関係の有無を推定する 2 種類のモデルを構築する。

4.1.1 SVM

SVM による発話間に関係の有無を推定を行うモデルを構築する。モデルの概要図を図 2 に示す。モデルの入力として、ディスカッションの内の任意のノードペアである $node_i = (w_1, w_2, \dots, w_k)$, $node_j = (w_1, w_2, \dots, w_l)$ 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 Google が公開している Word2Vec^(注1)を用いて得られた単語ベクトル $V_i = (v_1, v_2, \dots, v_l)$, $V_j = (v_1, v_2, \dots, v_l)$ に変換する。ここでの v_n は w_n の単語ベクトル、 V はノードベクトルを表す。そして、ノード内の単語ベクトルの和 V_{node_n} を以下の式から算出する。

$$V_{node_n} = \sum_{x=1}^m v_x$$

ここでの v_x は $node_n$ の単語ベクトルを表し、 m はノードの単語長を表している。最後に二つのノードベクトルを連結させる。この時に、4.2 節で説明する素性を用いる場合、連結したノードベクトルに素性を連結させる。最後に、ノードベクトルまたはノードベクトルと素性を連結したものをを用いてノード間の関係の有無を推定する。

4.1.2 bi-LSTM

bi-LSTM[12] による発話間に関係の有無を推定するモデルを構築する。モデルの概要図を図 3 に示す。モデルの入力として、一つのディスカッション内の任意のノードペアである $node_i = (w_1, w_2, \dots, w_k)$, $node_j = (w_1, w_2, \dots, w_l)$ を入力とする。次にノード内の単語を Google が公開している Word2Vec を用いて得られた単語ベクトル $V_i = (v_1, v_2, \dots, v_k)$, $V_j = (v_1, v_2, \dots, v_l)$ を入力する。ここでの $node_i, node_j, V_i, V_j$ は SVM と同様の定義である。bi-LSTM によってノード内の単語ベクトルからノードベクトル V_{node_i}, V_{node_j} を作成する。そして、二つのノードベクトルを連結させ、次の dense 層へ送る。この時に、4.2 節

(注1) : <http://code.google.com/archive/p/word2vec/>

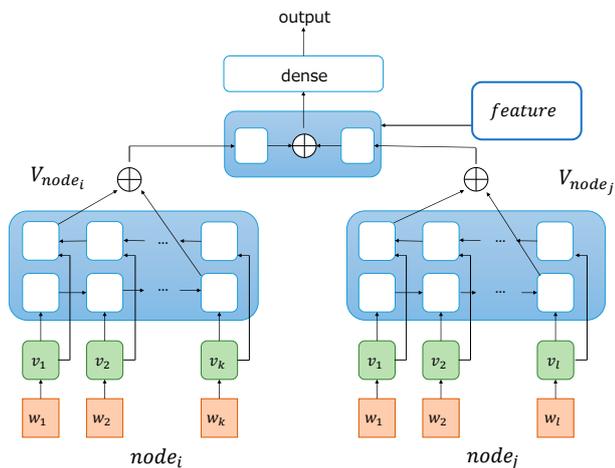


図 3 bi-LSTM を用いたモデルの概要図

で説明する素性を用いる場合、連結したノードベクトルに素性を連結させ、次の dense 層へ送る。最後に、ソフトマックス関数でノード間の関係の有無の推定を行う。

4.2 素性

本節では 4.1 節で説明した 2 つのモデルに使用する複数人議論の特徴を捉えた素性について説明する。本節で作成される素性は図 2, 3 で共通して *feature* の箇所である。

- ノードペアの単語数

主張を表す発話は発話者の考えている情報を多く伝達したいため、発話長が長くなる。一方で、主張に対する発話でも支持したり反論したりするため、発話長が長くなる。また、主張に対して相槌や相手の発話に対する同意のように一言で相手の発話に対して反応を示す場合に発話長が短くなる。このことから関係を持つようなペアには発話長の特徴があることが考えられる。そこで、各ノードを単語単位で分割し、各ノードの単語数を素性として利用する。

- ノードペアの共通単語数

主張を表す発話に対して関係を持つ発話は同一の議題について発話していることが想定される。主張に対して反応を表す発話が相槌のような一言で返答する発話でなければ、主張も返答も同じ議題に触れるような発話であることが考えられる。同じ議題に触れる場合、どちらも同じ固有名詞や議題に関連する動詞が発話内に出現することが考えられる。そこで、各ノードを単語単位で分割し、各ノードで現れる共通単語の数を算出し素性として利用する。

- 話者情報

ある話者の主張に対して、肯定意見を述べたり問題点を指摘したりする話者は、主張を述べた話者と同一の話者でないことが多い。また、同一の話者が主張に対して肯定意見を述べたり問題点を挙げたりすることは非常に稀な場合である。このことからノードペアにおいて関係の有無を推定することは、話者情報が重要になることが考えられる。そこで、各ノードの話者情報を素性として利用する。

- 時間情報

複数人議論では議論が活発になれば主張とその主張を支持し

たり反論したりする発話間の時間が短くなることが考えられる。また、議論の序盤に述べた発話と終盤に述べた発話の関係を持つことは稀な場合といえる。このことから時間情報はノード間の関係の有無の推定において重要な特徴である。そこで、対象となるノードペアの一つのノードの終了時から、もう一つのノードの開始時までの経過時間を算出し、素性として利用する。

- ノード間の距離

時間情報の素性の場合と異なり、議論が停滞している場合を考えると発話間の時間は長くなるが、発話間の距離は短くなることが考えられる。一方で、発話間の距離が離れている場合、議論が活発でなければ同じ議題の中で論点が変わることが考えられるため、発話間に関係があることがほとんどない。このことから、ノード間の関係の有無の推定において重要な特徴である。そこで、ディスカッション内のノードを発話時間順に並び、任意のノードペアの間にいくつノードが存在するのかを算出し、素性として利用する。

- 談話行為タグ

各ノード内の発話の役割を判断できれば、ノードがどんな機能を表しているか理解できる。ノード内の発話に「Inform」のような何らかの情報を与える談話行為タグが付与されていることを考える。このノードと関係を持つノードは「Backchannel」のような相槌を表す談話行為タグが付与されていたり、「Assess」のような直前の発話に対して何らかの評価を与える談話行為タグが付与されていたりする発話を持つノードであることが考えられる。一方で、「Inform」の談話行為タグが付与されている発話を持つノードに対して、「Elicit-Inform」のような他の話者に情報を求める談話行為タグが付与された発話を持つノードと関係を持つことは考えにくい。そこで、各ノード内に出現する計 15 種類の談話行為タグの頻度を算出し、素性として利用する。

- Unit Label

3. 節で述べた TAS では、各ノードに対して議論要素における役割を表す Unit Label が付与されている。Unit Label には、表 2 より自由な受け答えができる質問を表す「Open Issue」や、限られた選択肢の中から回答の選択に迫る質問を表す「A/B Issue」、はいかいいえで答えられるような質問を表す「Yes/No Issue」の 3 種類の質問を表すタグがある。これらの役割を持つノードに対して関係を持つノードは質問への回答を表す肯定意見や否定意見のノードであることが考えられる。一方で、質問を表すタグが付与されたノードに対して、質問を表すタグが付与されたノードが関係を持つことは議論としては破綻しており、関係を持つと考えにくい。そこで、各ノードの Unit Label を素性として利用する。

- ノードペアの極性値

ある話者の主張に対して肯定意見であったり否定意見であったり、主張を表す際に話者の感情が入る。また、ある話者の質問に対しての回答で賛成や反対するときも話者の感情が入ることが考えられる。このことから、発話間の関係を推定するときに発話の極性を考慮することは重要である。そこで、Stanford CoreNLP[13] を用いて各ノードの極性値を算出し、素性として利用する。

表 3 各データのノードペアの分布

	ダイアログ	ディスカッション	関係あり	関係なし
訓練データ	84	201	3850	38530
開発データ	4	13	235	1822
評価データ	4	12	238	1875

5. 実験と分析

本節では、4.1 節で述べたモデルのパラメータについて説明する。次に、3. 節で述べた複数人議論コーパスを用いた実験データについて説明する。最後に、各モデルで発話間の関係の有無を推定した結果、推定した結果に対するエラー分析および各素性の有効性について考察する。

5.1 モデルのパラメータ

各モデルで設定したパラメータについて説明する。

SVM のパラメータは、カーネルは rbf、コストパラメータは 100 と設定した。

bi-LSTM のパラメータは次のように設定した。入力層の次元数 300、隠れ層の次元数を 200、ドロップアウトは 0.2 に設定し、最適化手法には Adam[14] を利用し、学習率は 0.001 とした。また、訓練データを bi-LSTM モデルで学習する際、1 エポックごとに開発データで評価を行い、25 エポック内で最も開発データの損失関数の値が最も小さいものを最終的なモデルとした。

5.2 実験データ

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。本実験では、各ディスカッションでノードはすべて抽出されており、ノードに付与される Unit Label は分類されていると仮定した。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 対話に分割した。また、それぞれのディスカッションの数は 201 個、13 個、12 個となっている。ディスカッション内の 2 つのノードを総当たりでノードペアを作成した。訓練データ、開発データ、評価データのダイアログの数、ディスカッションの数、ノードペアの数の分布を表 3 に示す。表 3 より、訓練データの総数 42380 ペアに対して、関係ありが 3850 ペア、関係なしが 38530 ペアとデータに偏りがある。そこで、SVM と bi-LSTM モデルに訓練する際はデータを関係があるペアの数に合わせて関係なしの訓練データから無作為に 3850 ペア抽出し、関係ありの 3580 ペアと均衡なデータに成形して学習させた。

5.3 実験結果

素性を加えたモデルと素性を加えていないモデルで、ノード間の関係の有無を推定した結果を表 4~7 に示す。

表 4 は 4.1.1 節の手法で推定実験を行った結果である。また、表 5 は 4.1.1 節の手法に 4.2 節で説明した素性を連結して推定実験を行った結果である。表 4 と 5 の F-score より SVM では連結したノードベクトルに素性を連結したモデルの方が推定結果がよいことを確認できた。

表 6 は 4.1.2 節の手法で推定実験を行った結果である。また、

表 4 SVM モデルの関係推定の評価値

	Precision	Recall	F-score
関係あり	0.12	0.55	0.20
関係なし	0.90	0.49	0.64

表 5 素性を連結した SVM モデルの関係推定の評価値

	Precision	Recall	F-score
関係あり	0.41	0.84	0.56
関係なし	0.98	0.85	0.91

表 6 bi-LSTM モデルの関係推定の評価値

	Precision	Recall	F-score
関係あり	0.15	0.55	0.24
関係なし	0.91	0.61	0.73

表 7 素性を連結した bi-LSTM モデルの関係推定の評価値

	Precision	Recall	F-score
関係あり	0.42	0.46	0.44
関係なし	0.93	0.92	0.92

表 7 は 4.1.2 節の手法に 4.2 節で説明した素性を連結して推定実験を行った結果である。表 6 と 7 の F-score より bi-LSTM も同様に連結したノードベクトルに素性を連結したモデルの方が推定結果が良いことが確認できた。

以上の結果から両手法において、本研究で作成した素性が複数人議論の特徴を捉えた有効な素性であることがわかる。また、素性を連結した各モデルに着目すると、関係がないノードペアの Precision が高いことが確認できた。このことから、4.2 節の素性によって議論構造を構築するときに必要なエッジを引かない傾向があることがわかる。しかし、関係があるノードペアの Precision は低い。これは、議論構造を構築するとき議論構造として正しいノードのペアが獲得できないことを表している。議論構造の構築において Precision が低いことは問題であり、向上させることが課題となる。

5.4 エラー分析

実際に SVM モデルと bi-LSTM モデルが誤って正解ラベルを割り当てたペアを確認して、考察を行った。相槌のような一言で返答しているノードが、主張を表しているノードや質問を表すノードと関係を持つと誤って推定されていることが多く見られた。ノード間の特徴として、相槌のような一言で返答しているノードは、あらゆるノードと関係を持つことが多い。そのため、主張や質問を表すノードに対して関係を持たない相槌のような一言で返答するノードを誤って関係ありと推定したためと考えられる。また、どちらも発話長が長いペアが関係を持つと推定されていることが多く見られた。発話長が長いノードは発話内容の情報を多く持っている。しかし、誤って関係を推定していることから、どのような発話同士が関係するのか捉えられていないことが考えられる。このことから、4.1 節のモデルで用いた 2 つのノードベクトルを連結するだけでは、ノード間の関係の有無の推定に有効に働かないことが考えられる。

表 8 関係推定での素性の有効性に対する評価値

	Precision	Recall	F-score
ALL	0.41	0.84	0.56
– ノードペアの単語数	0.40	0.86	0.55
– ノードペアの共通単語数	0.41	0.84	0.55
– 話者 ID	0.40	0.84	0.54
– 時間情報	0.30	0.84	0.44
– ノード間の距離	0.38	0.85	0.52
– 談話行為タグ	0.41	0.84	0.55
– ノードの種類	0.41	0.84	0.55
– ノードペアの極性値	0.40	0.84	0.55

5.5 素性の分析

各素性の有効性について実験を行った。素性を連結したモデルで関係があるペアの F-score が高かった SVM のモデルに対して各素性を一つずつ除いて実験を行った。各素性を一つだけ除いて推定実験を行った実験結果を表 8 に示す。議論構造を構築することに重要なノード間に関係があるクラスの評価値のみを示した。表 8 の太字は各評価指標で全ての素性を加えた推定実験より最も下がった評価値を表す。

表 8 より、時間情報の素性を学習から抜いたものが F-score, Precision で最も低くなり、有効な素性であることが確認できた。このことから、参加者が意見を述べた直後に、他の参加者が追加意見を述べたり、反応を示したりすることが多いという議論の特徴であるため、関係推定に効いていることがわかる。また、ノード間の距離の素性に注目すると Precision が時間情報の次に低下していることがわかる。複数人議論において議論が活発であれば、ノード間の時間情報が短くなり距離が長くなることや、停滞していればノード間の時間情報が長くなりノード間の距離が短くなる相関がある。このことから、ノード間の距離と時間情報の二つの特徴に相関があり、その二つの特徴がともに大きな影響を及ぼしている。

6. おわりに

本研究では、複数人対話コーパスを対象に議論構造解析に向けた発話間の関係の有無の推定を行った。複数人対話コーパスには AMI corpus を用いた。次に、発話間の関係の有無を推定するためのモデルの構築及び、素性を作成し、モデルでの推定実験と素性を利用したモデルでの推定実験を行った。その結果、素性を利用したモデルが利用しなかったモデルよりも推定精度が高くなることが確認でき、素性の有効性を確認した。さらに、各素性の有効性を調査するために各素性を一つずつ除く実験を行い、時間情報とノード間の距離の素性が重要であることを確認できた。

今後の課題としては、モデルを改良することが考えられる。本研究の bi-LSTM モデルでは、ノードベクトルを連結しているだけであるため、現在は 2 つのノードの情報を独立に考えてしまっている。そこで、ノード間の関係を表すベクトル関係に改良することなどが挙げられる。また、主張内容を強調するために副詞や助動詞が用いられ、発話間の関係では主張に対して

補足の説明や主張を助けるために接続詞を用いることが考えられる。そこで、品詞分布に着目した素性の作成にも今後取り組んでいく。

文 献

- [1] C. Stab and I. Gurevych, “Parsing argumentation structures in persuasive essays,” *Computational Linguistics*, vol.43, no.3, pp.619–659, 2017.
- [2] Y. Mehdad, G. Carenini, F. Tompa, and R.T. NG, “Abstractive meeting summarization with entailment and fusion,” *Proceedings of the 14th European Workshop on Natural Language Generation*, pp.136–146, Association for Computational Linguistics, Sofia, Bulgaria, Aug. 2013.
- [3] A. Peldszus, “Towards segment-based recognition of argumentation structure in short texts,” *Proceedings of the First Workshop on Argumentation Mining*, pp.88–97, 2014.
- [4] H. Nguyen and D. Litman, “Context-aware argumentative relation mining,” *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp.1127–1137, 2016.
- [5] E. Barker and R. Gaizauskas, “Summarizing multi-party argumentative conversations in reader comment on news,” *Proceedings of the Third Workshop on Argument Mining (ArgMining2016)*, pp.12–20, 2016.
- [6] D. Ghosh, A. Khanam, Y. Han, and S. Muresan, “Coarse-grained argumentation features for scoring persuasive essays,” *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp.549–554, 2016.
- [7] C. Stab and I. Gurevych, “Training argumentation skills with argumentative writing support,” *Proc. SEMDIAL 2017 (SaarDial) Workshop on the Semantics and Pragmatics of Dialogue*, pp.166–167, 2017.
- [8] C. Stab and I. Gurevych, “Identifying argumentative discourse structures in persuasive essays,” *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp.46–56, 2014.
- [9] P. Potash, A. Romanov, and A. Rumshisky, “Here’s my point: Joint pointer architecture for argument mining,” *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp.1364–1373, Association for Computational Linguistics, Copenhagen, Denmark, Sept. 2017.
- [10] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, et al., “The ami meeting corpus: A pre-announcement,” *International workshop on machine learning for multimodal interaction* Springer, pp.28–39 2005.
- [11] R. Rienks, D. Heylen, and E. van derWeijden, “Argument diagramming of meeting conversations,” *Multimodal Multiparty Meeting Processing, Workshop at the 7th International Conference on Multimodal Interfaces*, pp.85–92, 2005.
- [12] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol.9, no.8, pp.1735–1780, Nov. 1997.
- [13] C.D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S.J. Bethard, and D. McClosky, “The Stanford CoreNLP natural language processing toolkit,” *Association for Computational Linguistics (ACL) System Demonstrations*, pp.55–60, 2014.
- [14] D.P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *The International Conference on Learning Representations (ICLR)*, pp.1–15, 2015.