

RNN(Recurrent Neural Network)を用いた 2 段階発話意図分析方法の提案と評価

2014SE055 久保井 恵里香 2014SE076 野田 菜月 2014SE096 鈴木 陽子

指導教員：青山 幹雄

1 研究の背景と課題

1.1 研究の背景

近年、システム開発の要求獲得においてステークホルダ分析の重要性が高まっている。大量のデータからステークホルダの意図を抽出するために、教師あり機械学習を用いて発話意図を特定する研究がある[3]。しかし、この研究では、文末の 4 語から発話意図を分類しているの
で、発話者の意図が適切に抽出されない。

1.2 研究の課題

研究背景から、以下の 2 点を研究課題とする。

- (1) 会話文が記述された発話データに対して、会話の前後関係を考慮した発話意図分析を、機械学習を用いて行うことを可能にする。
- (2) 提案した発話意図分析方法を例題に適用し、有用性を確認する。

2 関連研究

(1) 発話意図特定

発話意図とは、発話内容から取得できる発話者の意図である。発話意図を文末の 4 単語と品詞から抽出できる「報告」、「返答」、「受入」、「問い」、「要望」、「示唆」の 6 種類と定義する[2][4]。

(2) 形態素解析

対象言語の文法や単語の品詞等の情報に基づき、単語に分割する技術である。

(3) 機械学習

機械学習とはコンピュータが文字列や数値などのデータから特徴を抽出し、分析や予測をする技術である。

(4) クラス分類

あらかじめ定められた選択肢の中からクラスラベルを予測することである。クラス分類の 1 つであるランダムフォレストは、少しずつ異なる決定木を集めて平均を取る方法である。決定木は、クラス分類に広く用いられるモデルで、木構造を学習して分類する[4]。

(5) RNN (Recurrent Neural Network)

RNN は系列データに対するニューラルネットワークである。RNN では系列が長くなり、深いネットワークになると誤差逆伝播のアルゴリズムでは勾配が消失したり発散したりする問題が生じる。この点を改良したのが LSTM (Long Short Term Memory) である。

3 アプローチ

研究課題を解決するために、文末と文脈の 2 段階の以下のアプローチを示す。

(1) 文末による分類

クラス分類を用いて、教師データから抽出した特徴により、発話意図を発話文の文末 4 単語から、6

種類に分類する。

(2) 文脈による推定

(1) で付与された発話意図ラベルを入力し、文脈教師データを用いた機械学習により、発話意図ラベル間の依存関係から発話意図を再分析する。

4 2 段階発話意図分析方法

4.1 2 段階分析プロセス

図 1 に、発話データに対して 2 段階で特定をするプロセスを示す。

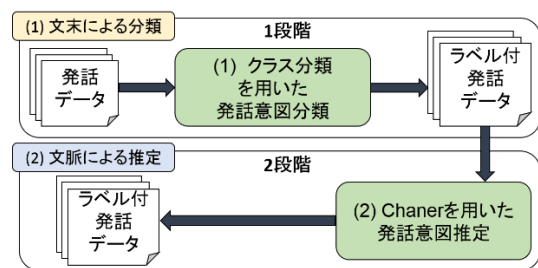


図 1 2 段階分析プロセス

4.2 クラス分類を用いた発話意図分類

発話データの文末の 4 単語から発話意図を分類し、ラベル付けされたデータを生成する。

分類する発話データについて説明する。日本語で発話文が記述された発話データを形態素解析器で単語に分割し、文末の 4 単語を取り出す。この発話データを、人手で 6 項目（「報告」、「問い」、「返答」、「要望」、「受入」、「示唆」）に振り分けて文末教師データを作成する。この文末教師データを用いて、テストデータに対してクラス分類を行う。クラス分類の方法はランダムフォレストを採用する。ランダムフォレストで 6 項目にクラス分類されたテストデータを文末によるラベル付き発話データとする。

2 段階目に入る前に、1 段階目の出力で得られたラベル付き発話データを整形して、2 段階目の文脈による発話意図分析で使用するテストデータを作成する。

4.3 RNN を用いた発話意図推定

(1) 2 段階目の文脈によるラベル付けの方法

4.2 のクラス分類を用いた発話意図分類の結果である、文末を分析して得られた文末によるラベル付き発話データに機械学習を用いて、さらに発話意図を分析するプロセスを説明する。入力を 4.2 のプロセスから得られた発話意図ラベル①と発話者のデータから作成した文脈クラスタ、出力を、文脈を分析して新たに得られた文脈クラスタとする。教師データを、発話意図ラベル①と発話者のデータから作成した文脈クラスタ、人手で分類した正解発話意図ラベルから作成した文脈教師データとする。1 段階目の文末によるラベル付けのから 2 段階目の文脈による

ラベル付けのプロセスを図 2 に示す。

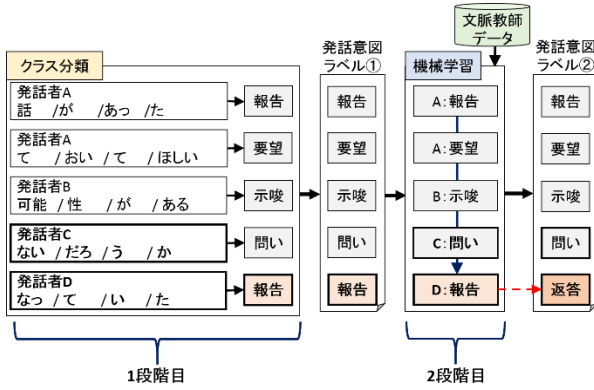


図 2 発話意図再分析プロセスの例

(2) ラベル付けの例

文脈を分析して発話意図を推定するには、発話の順序関係を学習できる機械学習器を構築する必要がある。ここでも、「返答」の推定を例に挙げる。「返答」の定義は、「問いに対して答える」であるので、文の構造からでは分類できない。「問い」に続いて出現する発話意図と発話者を分析することで、「返答」を推定することができる。1段階目の文末の構造を分析した発話意図分類で「報告」と分類された発話データを、文脈教師データの、発話意図ラベル①と発話者のデータの依存関係を学習した学習器によって分析する。それにより、「問い」の後に出現する発話意図ラベル①と発話者の依存関係が「返答」と推定される発話は、2段階目の文脈によるラベル付けの出力である発話意図ラベル②で「返答」に分類することができる。以下に「返答」の推定例を示す。

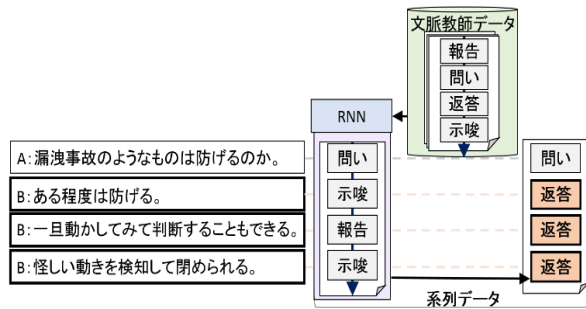


図 3 「返答」の推定例

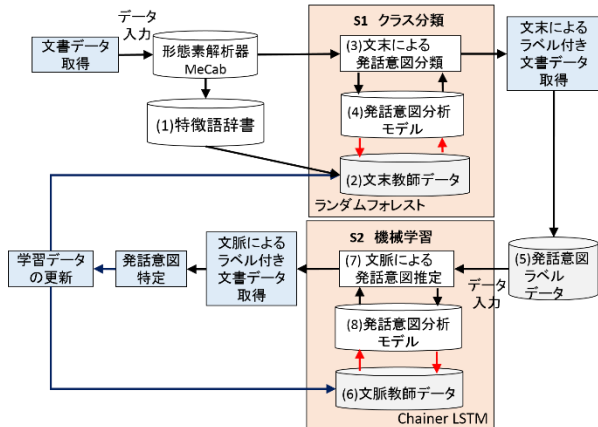


図 4 システム構成図

「返答」の推定パターンは、「問い」の後の 1 文でだけではない。例えば、「問い」と分類される発話の後で同じ人が何文も「問い」に対して答えていけば、その文すべてが「返答」と推定されるべきである。「問い」のすぐ後の発話文が必ずしも「返答」ではない場合もある。ある人が「問い」の発話をした後に、同じ人が続けて発話をしている場合、その発話は「返答」ではない可能性がある。

5 2 段階発話意図分析システムのプロトタイプ

発話データをシステムに適用する。本研究のシステム構成図を図 4 に示す。

6 プロトタイプへの適用

6.1 目的

本稿での提案プロセスを実データに適用、評価し、提案プロセスの妥当性を示す。

6.2 発話データの適用プロセス

6.2.1 発話データの整形

本研究で取り扱う議事録には、表形式のデータ、タイトル、日付など、そのままの形式では形態素解析に適さないデータが含まれているので、データを適した形式に整形する。

6.2.2 形態素解析

MeCab を用いて、整形済みデータを形態素解析することで、発話の文末 4 単語を抽出する。

- (1) 整形済みの発話文データに形態素解析器 MeCab を用いて、形態素に分割する。
- (2) 形態素に分割された発話文データから、発話文の文末から 4 単語を取り出す。
- (3) (発話者)のように発話者が括弧で囲まれて書かれているので、括弧内から発話者データを抽出する。

6.2.3 文末教師データ作成

1 段階目の発話意図分類として提案した、文末によるラベル付けて用いる文末教師データを作成する。下記にプロセスを示す。

- (1) 会話文それぞれに対して人手で発話意図を特定し、正解発話意図ラベルを付与する。
- (2) 正解発話意図ラベルと文末 4 単語のデータを結び付け、発話意図ラベルごとに文末 4 単語のデータを分類する。

本研究では 4 回分の議事録 597 文を文末教師データとして使用した。

6.2.4 クラス分類によるラベル付け

文末教師データから発話文の文末の構造の特徴を抽出して、テストデータの発話文の文末 4 単語に対して発話意図ラベル①を付与する。下記にプロセスを示す。

- (1) 作成した文末教師データから特徴語の辞書を作成する。
- (2) 作成した文末教師データの特徴を学習して、ランダムフォレストの決定木を作成する。
- (3) テストデータの発話文から文末 4 単語を抽出して、作成した辞書を用いて特徴ベクトルに変換する。
- (4) テストデータの特徴ベクトルを決定木によって分析し、テストデータの発話文に文末の構造から分類した発話意図ラベル①を付与する。

本研究では、会話文 111 文をテストデータとした。

6.2.5 文脈教師データ作成

1 段階目の文末による発話意図分類で得られた発話意図ラベル①と手で分類した正解発話意図ラベル、発話者データを用いて、文脈教師データを作成する。以下にプロセスを示す。

- (1) 文末教師データ用の発話文に対して、1 段階目の文末によるラベル付けを行い、文末の文構造から分類される発話意図ラベル①を得る。
- (2) 発話者のデータと、発話意図ラベル①、正解発話意図ラベルを用いて、4.3 で提案した文脈教師データを作成する。

6.2.6 文脈によるラベル付け

文脈教師データから、会話文の発話者と文構造から分類された発話意図ラベル①の依存関係を RNN によって学習し、テストデータの会話文に対して発話意図ラベル②を付与する。以下にプロセスを示す。

- (1) LSTM ニューラルネットワークを Chainer で構築し、作成した文脈教師データを系列データとして入力して学習する。
- (2) 1 段階目の文末によるラベル付けで得られた、発話意図ラベル①が付与されたテストデータと、発話者のデータを用いて、4.3 で提案した文脈教師データの作成と同様に文脈クラスタを作成する。
- (3) テストデータの文脈クラスタを LSTM によって系列データとして分析し、テストデータの会話文に続けて出現する発話間の依存関係から、分類した発話意図ラベル②を付与する。

文脈によるラベル付けで用いるテストデータは、文末によるラベル付けで用いたテストデータと同じ発話文 111 文である。

6.3 適用対象

プロトタイプで本研究の提案を適用したデータについて説明する。適用対象としたデータの概要とデータの用途を表 1 に示す。

表 1 適用データ

議事録回数	発話文数 (文)	用途	報告 (文)	示唆 (文)	要望 (文)	問い (文)	受入 (文)	返答 (文)
第3回	146	語尾教師データ (597文)	210	160	39	88	6	94
第4回	140							
第5回	104							
第6回	207							
第7回	201	文脈教師データ (545文)	175	120	20	103	8	119
第8回	91							
第9回	193							
第10回	60	テストデータ (111文)	28	6	5	33	4	35
	111							

情報システム開発についての事務局会議でされた議論が記述された発話データである。

第 3 回から第 6 回までの議事録を、1 段階目の文末によるラベル付けで用いる文末教師データの作成に使用し、第 7 回から第 9 回までの議事録と第 10 回の議事録の 60 文を 2 段階目の文脈によるラベル付けで用いる文脈教師データの作成に使用する。第 10 回の議事録の 111 文を、1 段階目の文末によるラベル付けから 2 段階目の文

脈によるラベル付けを適用して効果を検証するためのテストデータとした。

文脈教師データとテストデータについて、文脈クラスタの発話データを作成するときに必要な話題数と発話者数を表 2 に示す。

表 2 文脈教師データとテストデータ

議事録回	話題数 [個]	発話者数 [人]	発話文 [文]
第7回	7	9	201
第8回	6	13	91
第9回	8	12	193
第10回	3	7	60
	6	8	111

7 評価

本稿は先行研究[4]との比較方法として、人手でラベル付けしたものと機械学習によってラベル付けされたものを比較し正答率を算出する。

「正確率」は、人手で振り分けた正解数と、機械学習の正答数との比率とする。

「正当率」は、機械学習の正解数に対する、誤りを含めた機械学習の全回答数の比率とする。

「全正答数」は、各発話者回答数に各発話正答数を重みづけ、全回答数で割ったものである。

算出方法はそれぞれ式(1)、式(2)、式(3)で定義する。

$$\text{正確率} = \frac{\text{機械学習による正答データ数}}{\text{人手による正答データ数}} \quad (1)$$

$$\text{正当率} = \frac{\text{機械学習による正答データ数}}{\text{機械学習による回答数}} \quad (2)$$

$$\text{全正当率} = \frac{\text{各発話回答数} \times \text{各発話正当率}}{\text{全回答数}} \quad (3)$$

また 1 段階目の出力結果から「問い」、「要望」、「受入」は文脈に依存しない発話意図分類であるので、1 段階目の結果を評価対象とする。

テストデータに 2 段階発話意図分析方法を適用した結果の正答率を以下に示す。

- (1) 1 段階目の正確率と正当率

表 3 1 段階目の正確率と正当率

出力/正解	報告	示唆	要望	問い	受入	返答	合計	正当率
報告	17	0	0	0	1	23	41	0.42
示唆	9	4	0	0	0	10	23	0.17
要望	0	0	5	0	0	0	5	1.00
問い	1	0	0	32	0	0	33	0.97
受入	0	0	0	0	3	0	3	1.00
返答	1	2	0	1	0	2	6	0.33
合計	28	6	5	33	4	35	111	
正確率	0.61	0.67	1.00	0.97	0.75	0.06		

1 段階目の結果について、文末による分類で、「要望」、「問い」、「受入」は、正当率と正確率共に高いため、適切に分類されている。しかし、今回対象とする「返答」は両方共に低いため、適切に分類されていないことが分かる。この結果から、1 段階目の文末から発話意図を分類するだけでは、適切な判断ができないことが明らかになった。

(2) 2段階目の正確率と正当率

表4 2段階目の正確率と正当率

出力/正解	報告	示唆	要望	問い	受入	返答	合計	正当率
報告	15	0	1	0	0	0	16	0.94
示唆	10	6	0	0	0	1	17	0.35
要望	0	0	4	0	0	0	4	1.00
問い	1	0	0	32	0	0	33	0.97
受入	2	0	0	0	2	1	5	0.40
返答	0	0	0	1	2	33	36	0.92
合計	28	6	5	33	4	35	111	
正確率	0.54	1.00	0.80	0.97	0.50	0.94		

2段階目の結果に対し文脈による意図の推定を行った。赤枠は2段階目での向上を、青枠は低下を示す。1段階目で適切に分類できなかった「返答」は、正確率と正当率共に向上した。

(3) 正答率の遷移図

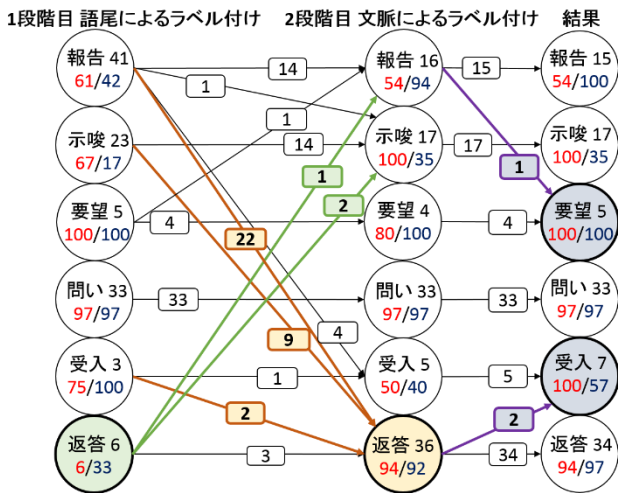


図5 正答率の遷移図

「返答」の1段階目と2段階目の発話意図の変化の遷移を図5に示す。丸の中は1段階目と2段階目、結果のそれぞれ振り分けられた数を示し、左側が正確率、右側が正当率を示す。

(4) 全正当率の改善

全正当率は1段階目の57%から2段階目で86%に改善した。

8 考察

8.1 研究課題の解決

(1) RNNを用いた効果

2段階目で発話意図を推定するために、文脈クラスターごとに異なった柔軟なネットワークを構築して学習する必要がある。また文脈は、前に出現した発話が後続の発話に影響を与える系列データであるので、系列データを学習して分析する必要がある。そのため、系列データを学習するニューラルネットワークであるRNNを用いることが適切である。

(2) 発話意図分析方法の有用性

1段階目は文末から、2段階目は文脈で発話意図を分析する提案をした。プロトタイプでは「返答」の分類精度を上げることが目標に、「返答」の分類基準である「問い」を基準に文脈クラスターを作成した。1段階目では「返答」の正当率は低く、適切に分類されていない。しかし2段階目で文脈から推定することで正当率が向上した。このプロトタイプの結果から2段階発話意図分析の有用であると言える。

脈から推定することで正当率が向上した。このプロトタイプの結果から2段階発話意図分析の有用であると言える。

8.2 先行研究[3]との比較

7の評価から、1段階目の文末によるラベル付けでは、「要望」、「問い」、「受入」の結果を、2段階目の文脈によるラベル付けでは、「報告」、「示唆」、「返答」の結果に基づき先行研究と比較して考察する。「要望」、「問い」、「受入」は1段階目の文末4単語から発話意図分類が適切に行われている。今回対象とした「返答」は、1段階目では適切な発話意図分類はされなかったが、2段階目では正答率の向上が見られた。よって、適切な発話意図を分類するには、文末4単語だけでなく、提案した2段階分析プロセスが有効であることが明らかになった。

9 今後の課題

今後の課題を以下に示す。

- (1) 1段階目で「受入」、2段階目で「返答」と推定された文章があった。文章構造から「受入」とも「返答」とも判断できるので受け取り手の判断になる。そのため定義をより正確にする必要がある。
- (2) 2段階目で文脈から発話意図を推定するため、分析対象を「返答」とした結果、「示唆」、「報告」は正解データに対して分析が不十分であった。今後はこの2つも「返答」と同様に、文脈クラスターの基準となる発話意図ラベルを定義、作成を行い、分析する必要がある。
- (3) 「要望」に対して返事をする発話は、「返答」と考える場合がある。しかし、今回は「問い」に対して返す発話を「返答」と定義したため、受け取り手によっては違う分類になったものがある。そのため「返答」の定義を見直す必要がある。

10 まとめ

本研究では、発話データに対して、文末4単語と文脈を考慮して2段階で発話意図を分析する方法を提案した。発話データに本提案方法を適用し、1段階目に文末4単語から発話意図を分類し、2段階目に文脈から発話意図を推定した。発話者の意図が抽出されたことを検証するために、正確率と正当率を定義して評価した。この評価に基づいて、先行研究からの向上を示した。2段階で発話意図を特定することで、発話意図の正確性の向上が得られ、提案方法の有用性を示した。

11 参考文献

[1] Preferred Networks, chainer, <https://chainer.org/>.
 [2] 藤本 玲子 ほか, セマンティックグラフモデルによるデータ駆動要求獲得方法の提案とステークホルダ分析への適用評価, 情報処理学会 SES2016 論文集, Aug. 2016, pp. 179-186.
 [3] 市川 裕也 ほか, 機械学習を用いたステークホルダ分析方法の提案と評価. 電子情報通信学会 KBSE2016-49, Mar. 2017. pp. 61-66.
 [4] 新納 浩幸, Chainerv2 による実践深層学習, オーム社, 2017.
 [5] MeCab, <http://taku910.github.io/mecab/>.