

非タスク指向型対話システムの改善

博多 哲也^{1,*} 鍛田 雅輝² 柴里 弘毅¹

Improvement of the Non-task-oriented Dialogue System

Tetsuya Hakata^{1,*}, Masaki Kuwata², Koki Shibasato¹

With the development of information and communication technology, various technologies and services have been born. Among them, the dialogue system is one of the technologies that have received a lot of attention. In this study, proposed non-task-oriented dialogue system is using Recurrent Neural Network (RNN) and word embedding. In dialogue systems, there are two types. One is retrieval model and the other is generative model. The dialogue system using retrieval model has few mistakes of grammar. However, it is like a parrot and takes a few seconds to generate responses. Therefore, we propose the dialogue system using generative model to improve these problems. In the proposed system, method of generating responses is seq2seq. The seq2seq is a RNN model that is trained by training data that pairs of inputs and outputs. Furthermore, the proposed system was improved to use enough vocabulary using Word2Vec. The proposed system can generate responses more natural than the retrieval model. Also, response speed was improved. Although some improvements were found, the proposed system was shown to be suitable as a dialogue system.

キーワード：非タスク指向型対話システム、自然言語処理、再帰型ニューラルネットワーク、Word2Vec、ディープラーニング

Keywords : Non-task-oriented Dialogue System, Natural Language Processing, Recurrent Neural Network, Word2Vec, Deep Learning

1. 緒言

情報通信技術の発展に伴い、様々な技術やサービスが生まれている。その中でも、人の話相手となる対話システムは非常に注目されている技術の一つである⁽¹⁾。対話システムには、タスク指向型対話システムと非タスク指向型対話システムがある。前者はある特定の目的を達成するための対話システムであり、後者は雑談などの対話そのものを継続する対話システムである⁽²⁾。また、インターネットに接続できる環境があれば、既存のサーバ・クライアント型の対話システムを利用することができ、独居老人の話相手や健康管理に応用できると期待されている。しかし、後期高齢者のインターネット利用率は3割程度に留まる⁽³⁾ことから、対話システムを必要としている年齢層に対しては普及が進んでいない。

また、対話システムはその応答生成の仕組みから Retrieval model と Generative model に分類される。前者は、文章を構文解析などの各工程で処理を行い、言語データベースなどから応答を生成する手法である。後者は、解析処理や言語データベースを用いずに応答を生成する手法である。

そこで、後期高齢者の現在の住環境でも使用できるように、インターネット環境に依存しない非タスク指向型対話システムを Retrieval model で開発した⁽⁴⁾。ユーザの入力文章を構文解析し、文章に含まれる話題語を単語分散表現により類似単語に置換することで応答文を生成する。このシステムの特徴は表 1 のように表すことができる。文法ミスが少なく話題展開ができる一方、応答時間や会話の自然さの面で課題が残されていた。

表 1 Retrieval model を用いたシステムの特徴

有意な点	改善点
インターネット環境に依存しない	応答時間が遅い (1 応答に数秒を要する)
類似単語を用いた話題展開ができる	定型文データベース応答補助が必要
文法ミスが少ない	自然さに欠ける

¹ 電子情報システム工学系
〒861-1102 熊本県合志市須屋 2659-2
Department of Control and Information Systems Engineering,
2659-2 Suya, Koshi-shi, Kumamoto, Japan 861-1102

² 電子情報システム工学専攻
〒861-1102 熊本県合志市須屋 2659-2
Electronics and Information Systems Engineering Advanced Course,
22659-2 Suya, Koshi-shi, Kumamoto, Japan 861-1102

* Corresponding author:
E-mail address: hakata@kumamoto-nct.ac.jp (T. Hakata).

本研究では従来のシステムの課題であった応答時間や会話の不自然さを改善するために新たに対話システムを Generative model で開発する。構文解析などの中間処理が省かれるため、応答時間の短縮が期待される。また、Recurrent Neural Network (RNN)言語モデルを活用した Seq2seq⁵⁾を採用した。Seq2Seq は Generative model の中でも最も一般的であり機械翻訳の分野などで特に成果を挙げていることから、流暢な応答を生成することが期待される。ただし、Seq2Seq の学習には大規模な言語コーパスが必要である。しかし、収集できるコーパスには限界があることが課題となっている。本研究では、単語分散表現により学習に用いるコーパスを増幅することでこの問題の解決を試みる。また、応答の候補を形態素解析することで、自然な応答を選択する仕組みを提案する。

2. 関連技術

2.1 Seq2Seq

Seq2Seq は Encoder-Decoder モデルと呼ばれるニューラル

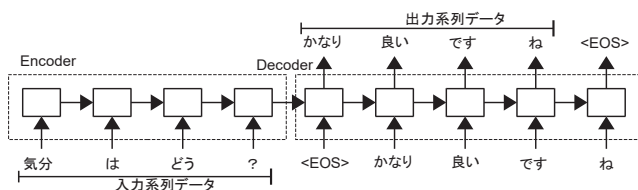


図1 Seq2Seq による生成モデル

ネットワークの一種で、系列データを与えると関連する系列データに変換する (図 1)。Encoder-Decoder モデルは Encoder と Decoder 2 つの RNN を繋ぎ、Encoder の最後の隠れ層が Decoder の隠れ層の初期値となることで構成される。Encoder は入力データの特徴を表すベクトルに変換する機構であり、Decoder は Encoder で生成された特徴ベクトルから新しいデータを生成する機構である。Seq2Seq は RNN に Long Short-Term Memory を用いる。

2.2 単語分散表現

単語分散表現は、単語を文字列から語彙情報を含むベクトルに変換する技術である。図 2 に Mikolov らの手法⁶⁾を示す。Skip-gram モデルの学習により単語ベクトルが作成される。Skip-gram モデルとは、入力単語からその周辺単語を予測するニューラルネットワークモデルのことである。この Skip-gram モデルにおける入力層、中間層間の重み行列が単語ベクトルである。学習に用いられる One-hot ベクトルの

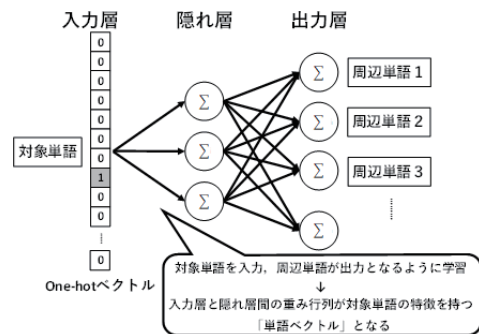


図2 Skip-gram モデル

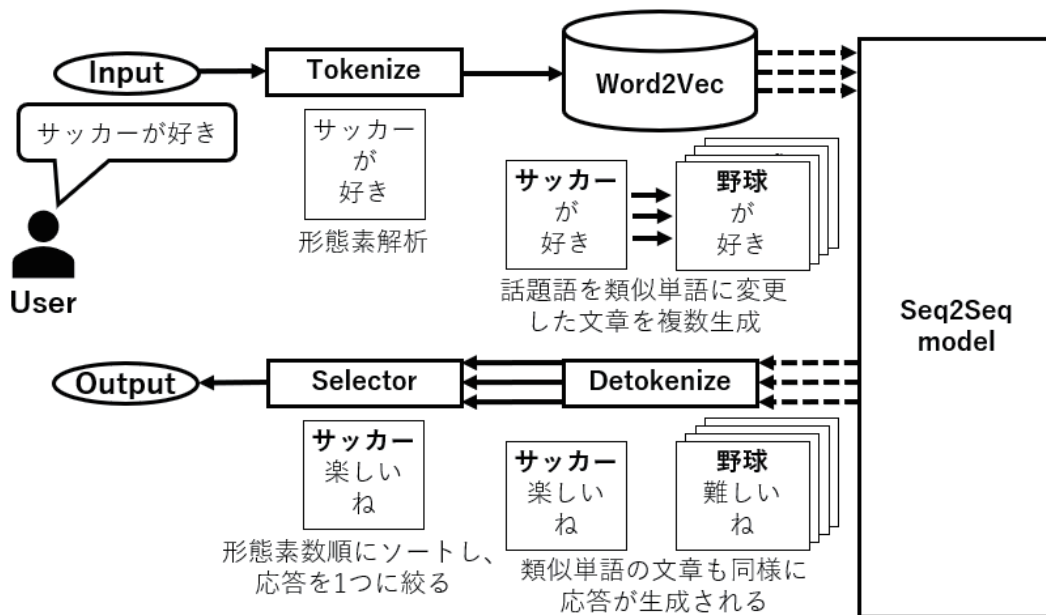


図3 提案システム

次元は単語データの総数と等しいため、言語生成用の単語情報が豊富になるほど次元が大きくなるという問題がある。しかし、単語分散表現を用いることで、その次元を数100次元まで圧縮することができる。このベクトルを用いて単語を分類し、単語同士の類似度を定量的に計算できる。

3. 提案システム

3.1 単語分散表現による応答生成

本研究では、提案システムの応答生成モデルに Sutskever らの Seq2Seq を利用する。図 3 に提案システムにおける Seq2Seq の語彙数や応答パターンを増加させるための流れを示す。一般的な Seq2Seq の応答生成システムでは、Tokenize、Embedding、Sampling、Detokenize の 4 つの処理が行われる。しかし、入力された文章にデータセットにない語彙が用いられている場合、適切な応答を生成することができない。そこで、Word2Vec、Selector という 2 つの処理を追加することで、適切な応答が出力されるようにした。

3.2 処理の流れ

提案システムの内部処理について、例を挙げて説明する。

[step.1]

ユーザから「サッカーが好き」という入力を与えられると、Tokenize で形態素解析が行われ「サッカー/が/好き」と分割される。

[step.2]

話題語「サッカー」を抽出し、その話題語の類似単語を複数個生成した後、元の入力文章の話題語部分と置換して複数の文章を生成する。これら一連の処理は Word2Vec で行われる。話題語「サッカー」の場合は、類似単語として「野球」や「テニス」などが出力され、文章として「野球が好き」や「テニスが好き」などが作成される。

表 2 Seq2Seq データセット

データセット	内容
名大会話コーパス ^[7]	129 会話, 合計約 100 時間の日本語母語話者同士の雑談を文字化したコーパス
NTT 雑談対話コーパス ^[8]	雑談対話システムとユーザが 21 発話からなる対話を行ったデータで, 116 名による 1,146 対話

表 3 Word2Vec データセット

データセット	内容
Wikipedia コーパス ^[9]	Wikipedia に掲載されている全文章データで約 10 億文字, 約 100 万語彙

[step.3]

Seq2Seq に作成した複数の文章を入力する。それぞれの応答である「サッカー楽しいね」や「野球難しいね」などが出力される。

[step.4]

複数の応答文が生成されるため、Selector によって 1 つに絞る。複数の応答文を形態素数順に並べ、形態素が最も多い応答文である「サッカー楽しいね」が選択される。

ここで、Selector が形態素数を用いるのは、形態素数の多さが相手への興味の強さを示す指標⁽⁶⁾とされているためである。

4. 実験

4.1 実験環境

実験に使用した PC の仕様は次の通りである。

CPU : AMD Ryzen7 1800X 8 コア プロセッサ 3.60GHz

Memory : 32GB

GPU : GTX1060 (2 枚)

OS : Ubuntu-16.04 LTS (64 ビット)

4.2 データセット

表 2 に Seq2Seq、表 3 に Word2Vec で用いたデータセットの情報を示す。

提案システムでは TensorFlow で公開されている Seq2Seq⁽¹⁰⁾を応用し実装した。学習の結果はレイヤー層の数とユニット数で大きく異なってくる。そこで、本研究ではいくつかのパラメータを用いて学習を実行し、その結果がどのように変化するかを考察した。具体的には、(a)NTT 雑

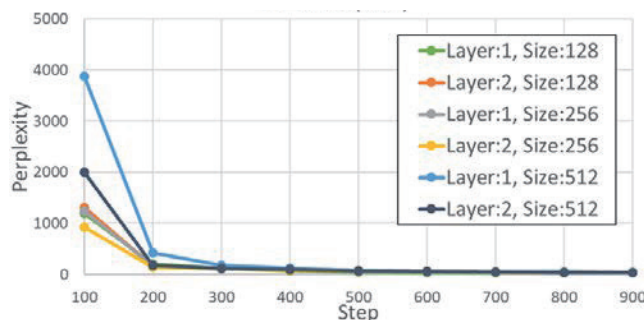


図 4 学習結果((a)NTT)

表 4 コーパス内の総行数((a)NTT)

項目	総行数
テストデータ (input)	1460
テストデータ (output)	1460
学習データ (input)	10000
学習データ (output)	10000
合計	22920

談対話コーパスのみ (以下、NTT)、(b)NTT 雑談対話コーパスと名大会話コーパスの組み合わせ (以下、NTT+名大)、(c)NTT 雑談対話コーパスを単語分散表現によって増幅した新たなコーパス (以下、提案手法) の3つについて述べる。

4.3 学習結果

コーパス毎の学習結果を図4~6に示す。横軸は学習のステップ数であり、縦軸は Perplexity である。Perplexity はモデルが応答における次の候補を絞り込んでいるかを表す指

標である。学習の進行状況を示す値であり、この値が 1 に近づく程十分な学習が行えたことを意味する。また、それぞれのデータの総行数を表4~6に示す。図4より、Perplexity の初期値は、レイヤー数やユニット数の多いモデルの方が高いことがわかる。また、レイヤー数やユニット数によらず 200 ステップまでは急速に減少し、それ以降は緩やかなカーブを描いている。この傾向はコーパスを変更しても同

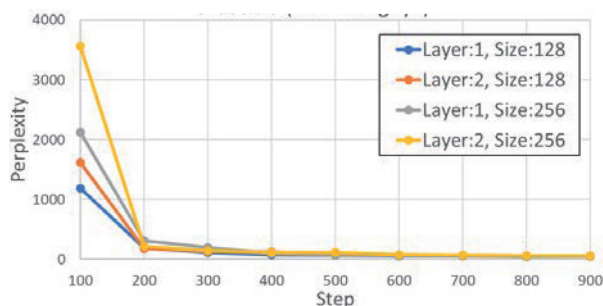


図5 学習結果((b)NTT+名大)

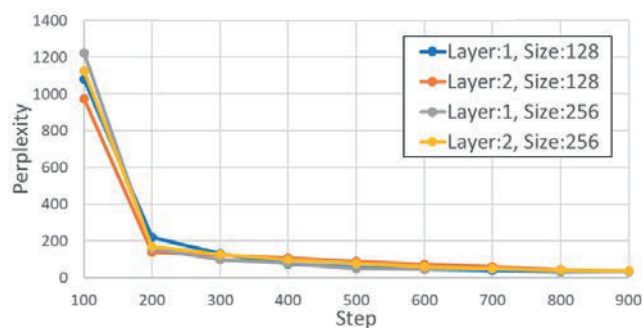


図6 学習結果((c)提案手法)

表5 コーパス内の総行数((b)NTT+名大)

項目	総行数
テストデータ (input)	4821
テストデータ (output)	4821
学習データ (input)	40000
学習データ (output)	40000
合計	89642

表6 コーパス内の総行数((c)提案手法)

項目	総行数
テストデータ (input)	6210
テストデータ (output)	6210
学習データ (input)	30000
学習データ (output)	30000
合計	72420

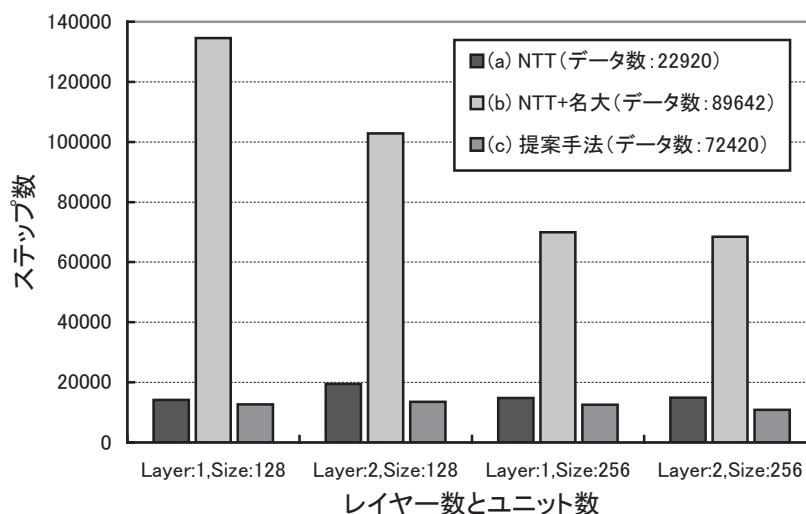


図7 学習結果の比較

様であった（図 5、6）。

次に、各学習における収束までのステップ数を図 7 に示す。横軸はレイヤー数とユニット数であり、縦軸は Perplexity が 4 以下となるステップ数を示している。この値が大きいくほど学習の収束に多くのステップを要することを意味する。図 7 より、(b)NTT+名大を学習させたモデルのステップ数が最も多く、学習に時間を要することが分かる。また、(c)提案手法を用いたモデルが最もステップ数が少なくなり、学習時間が短くなった。(a)NTT、(b) NTT+名大の 2 つのコーパスに含まれる文章データは対話データであり、様々な文法の文章が含まれている。それに対して、(c)提案手法は、データ量としては増加しているものの、話題語を類似単語に置換した文章であることから文法的变化がないため、総文法数は(a)NTT と同様と言える。そのため、データセットの量が多いにも関わらず、学習が速く収束したものと考えられる。

4.4 対話結果

表 7 に対話結果を示す。第 1 列はユーザの入力、第 2 列は従来システム⁽⁴⁾の応答、第 3 列は提案システムの応答である。例えば、「最近すごく暑いよね。」という入力に対して、従来システムは「最近すごく寒いよね。」と応答している。しかし、提案システムは「熱中症に気を付けようね。」と応答している。この例では、提案システムの方が従来システムよりも自然な対話をしている。従来システムの応答は、「暑い」と「寒い」という対義語が話題語となって対話を展開しているが、会話が噛み合っておらず、破綻している。それに対し、提案システムの応答は話題としても適切であり、次の会話へと繋がる。また、従来システムでは応答を出力するまでに数秒必要としていたが、提案システムでは 1

秒以内で応答が可能となった。応答速度の面でも提案システムの有効性を確認することができた。

次に、レイヤー数やユニット数を変更した提案システムの応答を表 8 に示す。「えっ。」や「何が。」のように応答が極端に単調になってしまう場合や、「スイカは面白いですね。」のように意味が通らない応答が確認できる。これらの応答は Perplexity が 2 以下となったモデルを用いており、十分な学習を行ったモデルである。それにも関わらず不自然な応答となっていることから、レイヤー数やユニット数などのパラメータの選択が極めて重要で、応答の精度に大きな影響を与えていることが分かる。

5. 結言

本研究では、非タスク指向型対話システムの改善について述べた。提案システムではインターネット環境に依存することなく動作できるよう、Generative model によって実装されている。一般的な Seq2Seq 対話システムにおいて、データセットに無い語彙が入力された場合、適切な応答を生成することができない。そこで、単語分散表現によって類似単語に置換された文章を Seq2Seq に入力し、複数の出力文章から形態素数の最も多いものを応答として選択する仕組みを提案した。その結果、破綻がない自然な応答ができることを確認した。また、応答時間が 1 秒以内に短縮され、従来システムと比較して大幅に改善された。

しかしながら、同じシステム構成であってもレイヤー数やユニット数によっては不自然な会話となる事例も確認された。パラメータの最適なチューニングについて明らかにし、さらに自然な会話を実現することが今後の課題である。

表 7 対話の例

入力	従来システム ⁽⁴⁾	提案システム (レイヤー数 1, ユニット数 256)
U: サッカーは好き?	S: ラグビーは好き。	S: サッカーは面白いですね。
U: スイカおいしいよね。	S: ダイコンおいしいよね。	S: スイカは実が大きいですね。
U: 最近すごく暑いよね。	S: 最近すごく寒いよね。	S: 熱中症に気を付けようねー。
U: 今日はテニスしてきた。	S: 今日全米オープンしてきた。	S: テニスにハマってますか?

表 8 不自然な応答の例

入力	提案システム 1((b)NTT+名大) レイヤー数 2, ユニット数 256	提案システム 2((a)NTT) レイヤー数 1, ユニット数 128
U: サッカーは好き?	S: よかった。	S: いえいえ。
U: スイカおいしいよね。	S: スイカを知りますか?	S: スイカは面白いですね。
U: 最近すごく暑いよね。	S: よかった。	S: えっ。
U: 今日はテニスしてきた。	S: えーってそんな感じじゃない?	S: 何が。

謝辞

本研究の一部は JSPS 科研費 JP20K12738 の助成を受けたものです。

(令和2年9月25日受付)

(令和2年12月7日受理)

参考文献

- (1) 河原達也 : 「音声対話システムの進化と淘汰 : 歴史と最近の技術動向」, 人工知能学会誌, 2013, 28, No.1, pp.45-51, (2018)
- (2) 畑健治, 小倉卓也, 萩原将文 : 「言語資源を用いた非タスク指向型対話システム」, 日本感性工学会論文誌, 2011, Vol.10, No.4, pp.515-522, (2019)
- (3) 内閣府.平成 28 年版高齢社会白書.
http://www8.cao.go.jp/kourei/whitepaper/w-2017/html/zenbun/s1_2_6.html, (2018.2.8 閲覧)
- (4) Masaki KUWATA, Koki SHIBASATO : "Nontask-oriented Dialogue System specialized in distinguishing multiple meaningful words", 12th Asian Control Conference IEEE, 2019, Paper MoA5.1 (2019)
- (5) Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le : "Sequence to sequence learning with neural networks", 2014, Advances in neural information processing systems, (2019)
- (6) 岡田将吾, 松儀良広 : 「マルチモーダル情報に基づくグループ会話におけるコミュニケーション能力の推定」, 人工知能学会, 2016, Vol.31, No.6, AI30-E, (2019)
- (7) Fujimura, Itsuko, Shoju Chiba, Mieko Ohso : "Lexical and Grammatical Features of Spoken and Written Japanese in Contrast: Exploring a lexical profiling approach to comparing spoken and Written corpora", 2012, Proceedings of the VIIth GSCP International Conference, Speech and Corpora, 393-398, (2019)
- (8) 東中竜一郎, 船越孝太郎 : 「Project Next NLP 対話タスクにおける雑談対話データの収集と対話破綻アノテーション」, 人工知能学会 言語・音声理解と対話処理研究会 第 72 回, 2014, pp.45-50, (2019)
- (9) Wikimedia Downloads: Index of /jawiki/latest/,
<https://dumps.wikimedia.org/jawiki/laTest/jawiki-latest-pages-articles.xml.bz2>, (2018.2.8 閲覧)
- (10) Tutorials : Sequence-to-Sequence model,
<https://www.tensorflow.org/tutorials/seq2seq>, (2019.6.9 閲覧)
- (11) 赤間怜奈, 稲田和明 : 「転移学習を用いた対話応答のスタイル制御」, 2017, 言語処理学会第 23 回年次大会発表論文集, (2019)