

LSTM エンコーダを用いた統計的発話選択

Statistical Utterance Selection using LSTM Encoder

稲葉 通将*¹ 高橋 健一*¹
 Michimasa INABA Kenichi TAKAHASHI

*¹ 広島市立大学大学院情報科学研究科

Graduate School of Information Sciences, Hiroshima City University

Web data is a useful resource for response generation in dialogue systems. This paper presents a dialogue model that selects an appropriate utterances from candidates extracted from Web data for non-task-oriented dialogue systems based on the recurrent neural network (RNN). Proposed model ranks candidate utterances by encoding utterances in given context and candidate utterances and calculating scores using the encoded vectors. Experimental results show that our proposed model can appropriately ranks candidate utterances in order of suitability to given contexts.

1. まえがき

非タスク指向型対話システムを実現するための手法として、応答ルールを記述し、そのルールを元に対話を進めるルールベース応答手法が未だに多く用いられている。しかし、ルールは基本的に人手で記述しなければならないため、作成および保守に関するコストが非常に大きい点が欠点である。

そこで近年、大規模な Web データを用いた統計処理により応答を決定する統計ベース応答手法の研究が活発になっている。例えば、Shibata らは、Web ページから抽出した文を用いて、映画に関する対話が可能な手法を提案した [Shibata 09]。また、Sugiyama らはユーザ発話に含まれる係り受け関係を持つ文節ペアを用いて、Twitter から応答発話を生成する手法を提案した [Sugiyama 13]。これらの統計ベース応答手法は、人手による応答の作成が不要であり、必要なコストが小さいという利点である一方で、ノイズが多く含まれたデータを使用する場合、不適切な文がそのまま出力されうという問題が存在する。その問題に対処するため、Web データから応答として使用可能な適切な文のみを抽出する研究が行われている [稲葉 14, Higashinaka 14]。しかし、これらの研究では、文の抽出を目的としており、ユーザとの対話において、どう応答するかに関しては扱っていない。

そこで本研究では、複数の発話候補から文脈に応じて適切な発話を選択することで対話を進める対話システムのための新たな応答手法を提案する。本手法は、文脈に応じて発話を順位付けし、1位の発話を応答として選択する手法である。なお、我々はこれまでに発話を順位付けする手法 [Koshinda 15] を提案しているが、後述する実験でその手法との比較を行う。

提案手法は、対話における2つの系列、すなわち発話中の単語の系列と、文脈中の発話の系列を RNN (Recurrent Neural Network) によって処理する点が大きな特徴である。具体的には、文脈中、および応答候補の発話をそれぞれ RNN により固定長のベクトルにエンコードする。次に、そのベクトルを時系列順に別の RNN に入力し、その結果を用いて発話の順位付けを行う。発話のエンコードにより、適切な応答選択のために必要な情報を抽出することが可能となる。また、文脈の時系列を

捉えることで、より適切な発話を選択可能となる。

2. LSTM エンコーダを用いた発話選択

本章では、LSTM ユニットを中間層に用いた RNN (LSTM-RNN) を用いた発話順位付け手法について述べる。

まず、発話の開始からある時点までの発話の系列を文脈 $c = (u_1, u_2, \dots, u_l)$ とする。ここで u_i は1発話を表し、 l は文脈中の発話数である。また、対話システムとユーザは交互に発話するものとし、文脈最後の発話 u_l は必ず対話システムの発話とする。そして、文脈 c に応じて生成された発話候補リスト $a_c = (a_c^1, a_c^2, \dots, a_c^m)$ と、そのスコア $t_c = (t_c^1, t_c^2, \dots, t_c^m)$ が与えられているとする。本研究における発話の順位付けとは、文脈 c の次のシステムの応答として適切な順に発話候補リスト a_c を並び替えることである。正解の順序は t_c により与えられ、並び替えは発話候補 a_c に対応するモデルの出力 $y_{a_c} = (y_1, y_2, \dots, y_m)$ に基づいて行う。

提案モデルの概要を図1に示した。本モデルは、文脈 $c = (u_1, u_2, \dots, u_l)$ と発話候補 a_c^i を入力として受け取り、 t_c^i を出力する。以降では、モデルの詳細について説明する。

2.1 発話のエンコード

提案モデルでは、対話中のユーザ、およびシステムの各発話から、次のシステムの応答として適切な発話を選択するために必要な情報を抽出するため、LSTM ユニットを中間層に用いた RNN (LSTM-RNN) 発話エンコーダを用いる。

RNN エンコーダを用いた研究として、機械翻訳 [Cho 14] や、対話システムのための応答文生成 [Shang 15] に関する研究が知られている。これらの研究では、入力文に含まれる単語を順に RNN エンコーダに入力し、固定長のベクトルを得る。次に、そのベクトルを別の RNN デコーダに入力することで、目的的文を生成する。そのため、RNN エンコーダは、目的文の生成に必要な情報を埋め込んだベクトルを出力するように学習が行われる。一方、本研究では、RNN デコーダを用いた文生成は行わず、エンコード結果は発話候補を選択するための素性として用いる。したがって、本手法で用いる RNN エンコーダは、発話選択のために重要な特徴を抽出するように学習されるため、RNN エンコーダ自体は類似した構造を使用するものの、文生成のためのエンコーダとは出力されるベクトルの性質が異なると考えられる。

連絡先: 広島市立大学大学院情報科学研究科

〒731-3194 広島市安佐南区大塚東 3-4-1

E-mail: inaba@hiroshima-cu.ac.jp

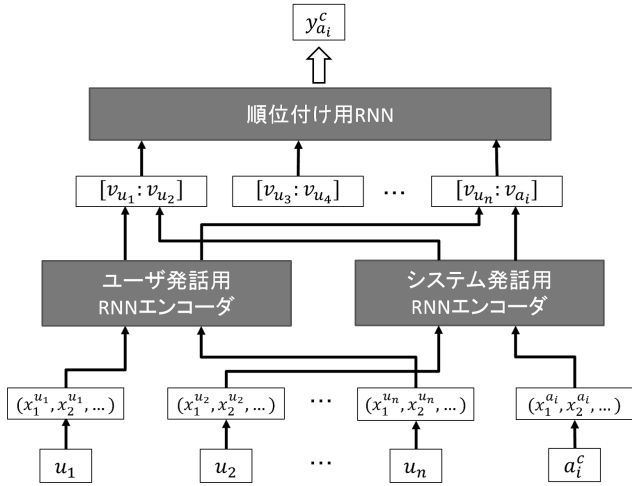


図 1: 提案モデル

本研究ではまず、形態素解析により発話を単語に分割し、単語の系列 $w = (w_1, w_2, \dots, w_n)$ を得る。次に、単語の系列 w を単語の分散表現の系列 $x = (x_1, x_2, \dots, x_n)$ に変換し、この系列を RNN への入力とする。分散表現への変換は Mikolov らの手法 [Mikolov 13] を実装した word2vec を用いる。

LSTM-RNN は順方向と逆方向の 2 種類を用意する。まず、順方向 LSTM-RNN は x を文頭から順に入力として受け取り、各入力に対応する $\vec{h} = (\vec{h}_1, \vec{h}_2, \dots, \vec{h}_n)$ を出力する。逆方向 LSTM-RNN には x を最後から逆順に入力し、 $\overleftarrow{h} = (\overleftarrow{h}_1, \overleftarrow{h}_2, \dots, \overleftarrow{h}_n)$ を出力する。最終的に獲得する発話ベクトル v は順方向と逆方向の LSTM-RNN のそれぞれの最後の出力を連結した $v = [\vec{h}_n : \overleftarrow{h}_1]$ となる (: はベクトルの連結を意味する)。なお、実験では、LSTM の中間層を 2 層重ねた RNN を使用した。

以上が発話のエンコード法であるが、発話のエンコードはシステム発話・発話候補とユーザー発話で別の RNN を使用する。このことは、システムの応答傾向を踏まえた学習を行うため、重要であると考えられる。

2.2 発話候補の順位付け

2.2.1 順位の学習

発話の順位付けには、発話のエンコードに用いた RNN とは別の RNN を用いる。この順位付け用の RNN は入力側から順に、LSTM の層を 2 層、ReLU を活性化関数に用いた通常のニューラルネットの中間層を 2 層、最後に出力層という構造を持つ。この RNN はエンコードされた発話のベクトルを入力とし、スコア (実数値) を出力する。

ただし本モデルでは、RNN の学習で多くの場合に用いられる、正解とモデル出力との間の平均二乗誤差を損失関数として使うのではなく、各発話候補のスコアに基づく順位と、正解スコアに基づく順位の距離を損失関数として用いる。本研究では順位の距離の計算のため、Plackett-Luce モデル [Plackett 75, Luce 59] を用いる。

Plackett-Luce モデルは順位付けのためのスコアのリストを確率分布に変換するためのモデルであり、リスト中で順位が高いスコアに対し、高い確率が与えられるような変換が行われる。スコアリスト $t_c = (t_1^c, t_2^c, \dots, t_m^c)$ において、 t_i^c が 1 位となる確率は、Plackett-Luce モデルを用いると次式で表さ

れる。

$$p(t_i^c) = \frac{\exp(t_i^c)}{\sum_{k=1}^m \exp(t_k^c)}$$

また、同式により、順位付け用の RNN が出力したスコア $y_{a_i}^c$ を確率分布に変換することが可能である。本手法では、Plackett-Luce モデルにより得られた正解データとモデルの確率分布のクロスエントロピー誤差を損失関数として用い、RNN の学習を行う。

2.2.2 文脈-発話候補ベクトル系列

与えられた文脈に対して適切な応答を選択するためには、文脈最後のユーザーの発話に対し、発話候補が適切であるかを判断するだけでなく、それよりも前の文脈における、システムとユーザーのやり取りの系列も考慮する必要がある。そこで、順位付け用の RNN には、文脈と発話候補から構成したベクトル系列を時系列順に入力することで、発話候補のスコアを得る。

文脈-発話候補ベクトル系列 $v_{a_i}^c$ は、文脈ベクトル系列 $v_c = (v_{u_1}, v_{u_2}, \dots, v_{u_l})$ と、 i 番目の発話候補のベクトル v_{a_i} を用いて以下のように構成する。

$$v_{a_i}^c = \begin{cases} ([v_{u_1} : v_{u_2}], [v_{u_3} : v_{u_4}], \dots, [v_{u_l} : v_{a_i}^c]), & \text{if } l \text{ is odd} \\ ([\mathbf{0} : v_{u_1}], [v_{u_2} : v_{u_3}], \dots, [v_{u_l} : v_{a_i}^c]), & \text{if } l \text{ is even} \end{cases}$$

式中の $\mathbf{0}$ はゼロベクトルである。このように、RNN にユーザーの発話とそれに対するシステムの発話をペアで入力することで、発話候補とその直前のユーザー発話の関連性を捉えることに加え、それよりも前のユーザーとシステムのやり取りを考慮することが可能となる。

3. 評価実験

提案手法の評価のため、実験を行った。本実験では、ユーザーとシステムが 1 対 1 で対話した文脈と複数の発話候補が与えられ、発話候補を文脈に対して適切な順に順位付け可能であることを確認する。性能比較のため、ベースライン手法との比較も実施した。

3.1 データセット

我々は、非タスク指向型対話システムを Twitter 上で公開している (アカウント名: @KELDIC)*1。この対話システムは、発話候補獲得手法 [稲葉 14] により獲得した複数の発話候補から、ランキング学習を用いて発話を 1 つ選択し、応答を行うシステムである。本実験ではデータセットとして、Twitter 上におけるこの対話システムとユーザーの対話データを使用した。

実験用データは以下のように作成した。まず、システムが応答を行った際の文脈と、その際の発話候補をすべて取得する。次に、実際にシステムが Twitter 上で応答に使用した発話 1 個と、応答に使用しなかった発話候補に対し、複数のアノテータが文脈に対する発話の自然さを評価した。

発話の評価基準は対話破綻検出チャレンジ [東中 15] を参考に $\circ \cdot \triangle \cdot \times$ の 3 段階とし、それぞれ「破綻はなく、自然な応答である」、「破綻とは言い切れないが違和感を感じる」、「破綻している」と定義した。

アノテータは我々が構築したツール上でアノテーションを行った。アノテータは最初に文脈と、実際に対話システムが Twitter 上で応答に使用した発話 1 個を含む複数の発話候補が提示され、各発話候補に $\circ \cdot \triangle \cdot \times$ のいずれかを付与する。アノテータには、1 つの文脈に対して提示された発話候補に対

*1 <https://twitter.com/KELDIC>

し、○の評価を最低でも1つ含めるよう指示した。もし最初に提示された発話中に○と評価可能なものが存在しない場合は、アノテータは随時ツール上で評価対象となる発話候補を増やすことができるようにした。また、それでも○と評価可能な発話が存在しない場合は、その文脈に対する評価をパスすることができるようにした。アノテータはクラウドソーシングサイトである CrowdWorks*2 で募集した。

上記の手順で収集したデータのうち、1つの発話候補につき、3名以上の評価が集まった1257個の文脈に関するデータを実験に用いた。1257個のデータのうち、評価を行ったアノテータの半分以上が○と評価した発話を1個以上含む200個のデータを評価用データ、それ以外の1057個のデータを学習データとして用いた。

学習時には、各発話候補について順位を定義するためのスコアが必要であるが、スコア y_i^c は以下の式で計算する。

$$y_i^c = s_o \frac{n_o}{N} + s_t \frac{n_t}{N} + s_x \frac{n_x}{N}$$

$$N = n_o + n_t + n_x$$

式中の n_o, n_t, n_x はそれぞれ○・△・×のアノテーションを行った人数であり、 s_o, s_t, s_x は○・△・×の点数を意味するパラメータである。今回の実験では、 $(s_o, s_t, s_x) = (10.0, -5.0, -10.0)$ を用いた。

応答性能の評価の際には、文脈に対して発話が自然か否かという点が重要であることから、評価用データでは、50%以上のアノテータが○と判定した発話を正解発話、それ以外を不正解発話として扱う。

3.2 実験設定

発話のエンコードに用いる word2vec については、次元数1000、ウィンドウサイズは5、最小出現頻度は10と設定し、約100GBのTwitterデータで学習を行った。

発話のエンコードに使用したRNNの入力層、LSTMの次元数、および出力層の次元はすべて1000次元とし、順位付けのためのRNNでは、入力層、LSTM、および中間層の次元は2000次元、出力層は1次元とした。また、学習時には各層の入力の50%をdropoutさせ、AdaGradで最適化を行う。

また、実験では、以下に挙げる手法との比較も実施した。

3.2.1 比較手法1: BoW + DNN

近年広く使われるようになったDeep Learningによって順位付けを行う手法である。BoW(Bag of Words)素性を用い、全8層のDNN(Deep Neural Network)を平均二乗誤差で学習したモデルとした。入力は「発話候補」、「文脈最後のユーザ発話」、および「最後のユーザ発話を除いた文脈中の発話をすべて連結した文」のそれぞれから作成したBoWベクトルとした。BoWの単語辞書は学習データ中の出現頻度が2回以上であった6203単語とし、上記の3つのベクトルを連結した18609次元を入力ベクトルとした。6つの中間層の次元はすべて5000次元とした。活性化関数にはReLUを使用し、学習時には各層の入力の50%をdropoutさせ、AdaGradで最適化を行う。正解となるスコアは提案手法のものと同じのものを用いる。

3.2.2 比較手法2: 対話システム(@KELDIC)

2つ目の比較手法として、データセットの作成時に使用したシステムの出力をそのまま使用したものを用いた。システムは、ランキング学習手法のListNet[Cao 07]により発話を順位づけし、応答を行う。順位付けのための素性ベクトルは、発話候補と文脈から作成しており、発話候補と文脈中の発話にそれ

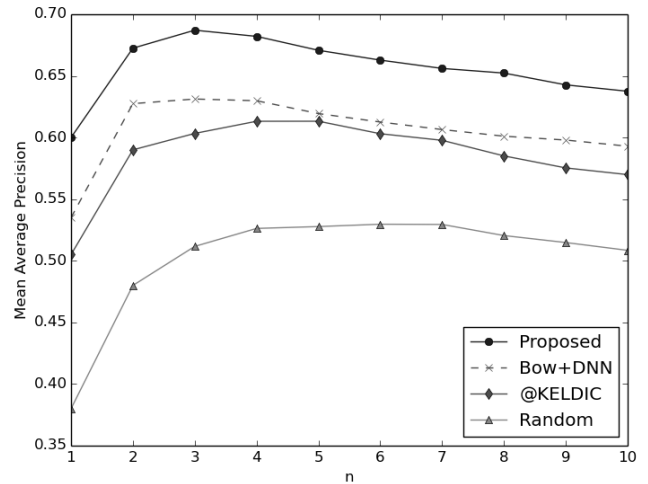


図 2: 実験結果

ぞれ含まれる特定の単語のペアなどを素性としている。アルゴリズムの詳細は[Koshinda 15]を参照のこと。

3.2.3 比較手法3: ランダム

発話候補をランダムで並び替える手法とも比較を行う。本手法の結果が、順序付け性能の下限值である。

3.3 実験結果

順位付けの評価は Mean Average Precision (MAP) によって行った。MAPの計算に用いる順位 n を上位1位~10位まで変更させた場合の結果を図2に示した。図より、提案手法(Proposed)が他の手法と比べ、すべての n について最も性能が良いことがわかる。BoW+DNNについては、BoW素性を用いたことから、文脈中の発話の順序関係や発話の意味を正しく扱うことができず、十分な性能が得られなかったと考えられる。@KELDICもランダムよりは高い性能であったが、やはり文脈の処理に課題があり、BoW+DNNよりも低い結果となった。

また、 $n=1$ のMAPの値は、1位に正解発話を正しく順位付けできた割合と等しいことから、提案手法は60%の文脈に対し、正しく応答可能であったことがわかる。ただし、本実験で使用した評価用データは、最低1つは正解発話が含まれるように用意したものであることから、実際にユーザと対話を行った際は、自然な応答ができる割合は、これよりも小さくすると予想される。

表1に評価データにおける文脈を、表2に表1の文脈に対する応答候補と、それぞれの手法における発話候補の順位、および発話候補ごとの「○」とアノテーションしたアノテータの割合を示した。表より、提案手法は○アノテーションの割合の大きい発話候補を上位に、そうでない発話を下位に順位付け可能であることがわかる。

4. まとめ

本論文では、非タスク指向型対話システムのためのRNNを用いた発話選択法を提案した。提案モデルは、文脈中、および応答候補の発話をそれぞれRNNにより固定長のベクトルにエンコードし、そのベクトルを時系列順に別のRNNに入力し、その結果を用いて発話の順位付けを行う。実験の結果、Deep Learningを用いたベースライン、および既存の手法と

*2 <https://crowdworks.jp>

表 1: 評価データにおける文脈の例

話者	発話
ユーザ	大学行くモチベーションが見た目に如実に表れてる
対話システム	わかります。一夜が明け、仕事へのモチベーションはますますであるもののテンションは非常に低いよね
ユーザ	わかる
対話システム	だよ！ 普段のテンションが高い分、落ち込んだらとことん落ち込むよ
ユーザ	そうなんだ。落ち込むな元気出せ！

表 2: 発話候補と順位付けの例

Proposed	BoW + DNN	@KELDIC	○アノテーションの割合	発話候補
1	6	3	1.00	相当なモチベーションが必要だよ
2	8	6	0.33	独学はモチベーションを保つのが大変だよ
3	10	2	0.00	どんなモチベーションでチャリこげばいいよね
4	1	8	0.33	勉強へのモチベーションがすごい
5	9	4	0.33	モチベーションには繋がるよ
6	4	9	0.00	ポケモンのモチベーションが皆無だよ
7	7	10	0.00	実習のモチベーション保つのは大変だね
8	3	5	0.00	東方のモチベーションがすごくなってよ
9	5	1	0.00	PC に対するモチベーション低いしやる気でない
10	2	7	0.00	モチベーション低い幹事は良くない

比べ、適切に発話が順位付け可能であることが確認された。

しかし、今回の実験では、文脈中の話題の認識や、その話題に基づく発話候補の生成は対象とせず、正解の含まれる発話候補の順位付けのみを対象とした。今後は、話題認識・発話生成までを含めた対話システムとして実装を行い、総合的な評価を実施する予定である。

参考文献

- [Cao 07] Cao, Z., Qin, T., Liu, T., Tsai, M., and Li, H.: Learning to rank: from pairwise approach to listwise approach, in *Proceedings of the 24th international conference on Machine learning*, pp. 129–136 (2007)
- [Cho 14] Cho, K., Merriënboer, van B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y.: Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734 (2014)
- [Higashinaka 14] Higashinaka, R., Kobayashi, N., Hirano, T., Miyazaki, C., Meguro, T., Makino, T., and Matsuo, Y.: Syntactic filtering and content-based retrieval of Twitter sentences for the generation of system utterances in dialogue systems, *Proc. IWSDS*, pp. 113–123 (2014)
- [Koshinda 15] Koshinda, M., Inaba, M., and Takahashi, K.: Machine-Learned Ranking Based Non-Task-Oriented Dialogue Agent Using Twitter Data, in *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, Vol. 3, pp. 5–8 (2015)
- [Luce 59] Luce, R.: *Individual choice behavior: A theoretical analysis*, New York: Wiley (1959)
- [Mikolov 13] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J.: Distributed representations of words and phrases and their compositionality, in *Advances in neural information processing systems*, pp. 3111–3119 (2013)
- [Plackett 75] Plackett, R.: The analysis of permutations, *Applied Statistics*, pp. 193–202 (1975)
- [Shang 15] Shang, L., Lu, Z., and Li, H.: Neural Responding Machine for Short Text Conversation, *Proceedings of the 53th Annual Meeting of Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp. 1577–1586 (2015)
- [Shibata 09] Shibata, M., Nishiguchi, T., and Tomiura, Y.: Dialog system for open-ended conversation using web documents, *Informatica*, Vol. 33, No. 3 (2009)
- [Sugiyama 13] Sugiyama, H., Meguro, T., Higashinaka, R., and Minami, Y.: Open-domain Utterance Generation for Conversational Dialogue Systems using Web-scale Dependency Structures, in *Proc. SIGDIAL*, pp. 334–338 (2013)
- [稲葉 14] 稲葉通将, 神園彩香, 高橋健一: Twitter を用いた非タスク指向型対話システムのための発話候補文獲得, *人工知能学会論文誌*, Vol. 29, No. 1, pp. 21–31 (2014)
- [東中 15] 東中 竜一郎, 船越 孝太郎, 小林 優佳, 稲葉 通将: 対話破綻検出チャレンジ, 第 6 回対話システムシンポジウム (2015)