単語の分散表現を用いた意味予測に基づく雑談応答生成*

☆古舞千暁, 滝口哲也, 有木康雄 (神戸大)

1 はじめに

近年, IoT 化に伴う会話型インターフェースや, 独居老人の増加, 若者の対話的コミュニケーション不足などの社会問題を受けて, 人間と会話できるシステムの研究が盛んに行われている. 既に実用化されているものとして, Apple 社による対話型秘書機能システム「Siri」や, Microsoft 社による会話ボット「りんな」などが挙げられる. 対話システムの応答生成には, あらかじめ人手によって作成した規則によって応答を生成するルールベース手法が存在するが, 多種多様な応答のためにはコストがかかるという問題がある. 本研究で扱う雑談システムは, 特定の話題やタスクを想定したものではなく, 人間との対話そのものに焦点を当てた非タスク指向型と呼ばれるもので, 道案内やチケット予約など特定の目的を持ったタスク指向型システムとは違い, 広い話題への対応が求められるので, ルールベース手法ではなく自動で応答文を生成する手法を用いる必要がある.

現在, 対話システムにおける単語表現は one-hot 表現によるものが主流であるが, 雑談においては扱う単語数が非常に多くなることが予想され, 多種多様な応答に対応できるようにしようとすると, one-hot 表現を用いた場合は単語ベクトルの次元数の増加が避けられず, モデルが複雑化する. また, コーパス中に出現した単語以外で応答文を生成することができず, コーパスへの依存度が高い. そこで, one-hot 表現を用いず, 事前にテキストデータで学習した固定次元の意味表現ベクトル空間を用意し, 入出力時の単語表現を全て統一することで, コーパス中に存在しなかった単語も扱え, モデルの複雑化も防ぐことが期待できる.

本研究では, 事前に学習させた word2vec による単語の分散表現を用いて, Recurrent Neural Network による単語予測を行い, 応答文を生成する手法を提案する.

2 RNN Encoder-Decoder による対話システム

対話システムにおける応答の自動生成手法として多く用いられているものは Vinyals らの Neural Conversational model [1] や Shang らの Neural Responding Machine for Short-Text Conversation [2] で見られるように RNN である. Fig.1 に示すように, 入力単語ベクトルの系列 $X = (x_1, ..., x_{T_x})$ を受け取り, 出力単語ベクトルの系列 $Y = (y_1, ..., y_{T_y})$ を出力する.

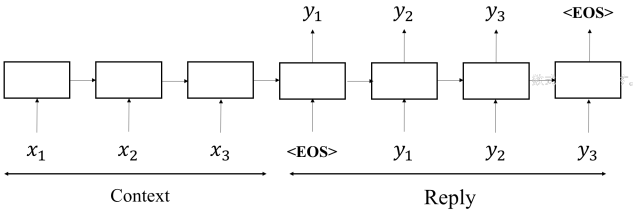

Fig. 1 Recurrent Neural Network による応答生成

ここで, RNN の隠れ層 $h_{(t)}$ は

$$h_{(t)} = f(h_{(t-1)}, x_t)$$

で表すことができる. 入力単語系列 X を処理する RNN を Encoder, 出力単語系列 Y を生成する RNN を Decoder として分け, 隠れ層 $h_{(T_x)}$ を Decoder における $h_{(0)}$ に用いるこのモデルは RNN Encoder-Decoder と呼ばれる. 本研究では RNN Encoder-Decoder モデルを用いる.

3 単語の分散表現

単語の分散表現は分布仮説に基づいたもので, 単語を低次元の実数値ベクトルで表す表現であり, Mikolov ら [3], [4], [5] によって提案された word2vec が主流である. one-hot 表現で単語を扱った場合は単語間の関係を考慮できないのに対し, 分散表現を用いると例えば (King - Man + Woman = Queen) などといった単語の意味を考慮したような演算が可能になることが知られている.

word2vec の学習手法として CBOW(Continuous Bag-of-Words), Skip-gram の 2 つが挙げられるが, 文献 [4] において Skip-gram モデルによる学習の方が良い結果を示しているので, 本研究では Skip-gram モデルを用いて word2vec を学習する. Skip-gram モデルでは, Fig. 2 のようなニューラルネットワークを用い, 中間層を学習させることで単語に対する意味ベクトルを得る. 本研究ではこのようにして得られた各単語のベクトル表現を入力時の変換, 出力時の単語検索に用いる.

*Chat response generation based on semantic prediction using distributed representations of words, by Kazuaki Furumai, Tetsuya Takiguchi, Yasuo Ariki (Kobe univ.)

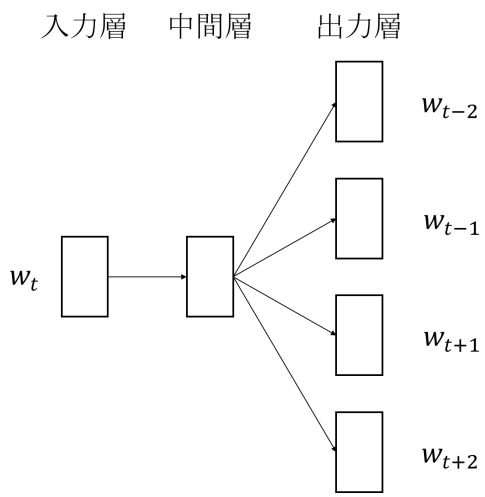

Fig. 2　Skip-gram モデル

4　提案手法

本研究では, RNN Encoder-Decoder の入出力ベクトルに, word2vec による分散表現ベクトルを用いて応答文を生成する．モデルの概略図を Fig.3 に示す．入力単語列を事前に学習された word2vec によって d_{word} 次元ベクトルへと変換し Encoder へ入力する．次に, Encoder で生成された隠れ層 $h_{(T_x)}$ を, Decoder の隠れ層の初期状態 $h_{(0)}$ とする．Decoder の出力ベクトルは意味予測ベクトル y_{mean_t} と扱うことができ, 各要素実数値をとる d_{word} 次元ベクトルである．単語への変換時は, この意味予測ベクトル y_{mean_t} を用いて, word2vec によって作成された単語ベクトル集合 V の中で, 最も cos 類似度が高いものを該当単語として応答文を出力する．ここで, 語彙数を N, word2vec で学習した単語ベクトルを $W_k \in V(k = 1, ..., N)$ とすると,

$$y_t = \arg\max_{W_k} cos(y_{mean_t}, W_k)$$

と表すことができる．　また，正解単語列を $T = (t_1, ..., t_{T_t})$ とすると, 学習時に用いる損失関数 L は

$$L = \sum_i |t_i - y_{mean_i}|$$

である．

5　データセット

word2vec を学習するデータセットと, 応答文生成を学習するデータセットは異なっていても構わないので, 本研究では Twitter で収集した対話コーパスと, 日本語 Wikipedia 記事から作成したデータセットを用意した．それぞれ, 適当な形式に整形した後に, MeCab [6] を用いた形態素解析による分かち書きを行なっている．

5.1　Twitter 対話コーパス

本研究では話者性や対話履歴を考慮しないため, Twitter から表 1 のような日本語の Tweet/Reply のペアを集めた 36 万ペア (72 万発話) で対話コーパスを作成した．ただし, 画像や URL を持つ発話を含むペア, 改行による複数文を用いたツイート, 非公開ツイートは使用しない．

5.2　word2vec 学習に用いるテキストデータ

word2vec の学習には, 収集した対話コーパスに加え, 日本語の Wikipedia 全記事 3G 分を用いた．これらデータセットを用いて word2vec を学習させた後に, Twitter 対話コーパスを対話学習に用いている．

6　実験

6.1　実験条件

word2vec による単語の分散表現次元数 $d_{word} = 128$, 出現回数が 10 回以下の単語は除外し, Skip-gram モデルで単語間の最大スキップ長は 3 単語で学習を行い, 結果として語彙数は 20 万単語となった．RNN Encoder-Decoder については, LSTM セルを用い, ユニット数 256, 隠れ層 3 層のモデルとした．学習時の最適化手法は Adam [7] を用いて，学習係数は $\alpha = 0.0001$, $\beta_1 = 0.9$, $\beta_2 = 0.99$ とした．特殊記号として, 文頭を示す $< GO >$ と文末を示す $< EOS >$ も word2vec で単語として学習し, RNN Decoder による応答文生成時には $< GO >$ を最初の単語として入力し, $< EOS >$ が出力されるまで応答を生成している．また, cos 類似度が 0.5 以下, または 1 つ前の出力時の cos 類似度の 60 %以下の場合は除くといった処理を行った．

6.2　主観評価

機械翻訳のタスクなどでは, 部分的な単語列の一致度でスコアを計算する BLEU [8] が評価に用いられる．しかし, BLEU による評価と人手による評価に差異がある場合が指摘されている [9]. 例えば, 英語とフランス語など文法的構造が似ている言語間の翻訳タスクなどでは人手評価と近いが, 英語と日本語など文法的構造が似ていない言語間の翻訳タスクでは差異が発生する．本研究で扱っている雑談システムでは, 入力文と出力文は様々な組み合わせが考えられ, より複雑なタスクとなっているため, 同様に BLEU 評価と人手評価に差異が生まれることが考えられる．そこで, 本研究では, 以下に示す 2 つの評価指標を作成した．

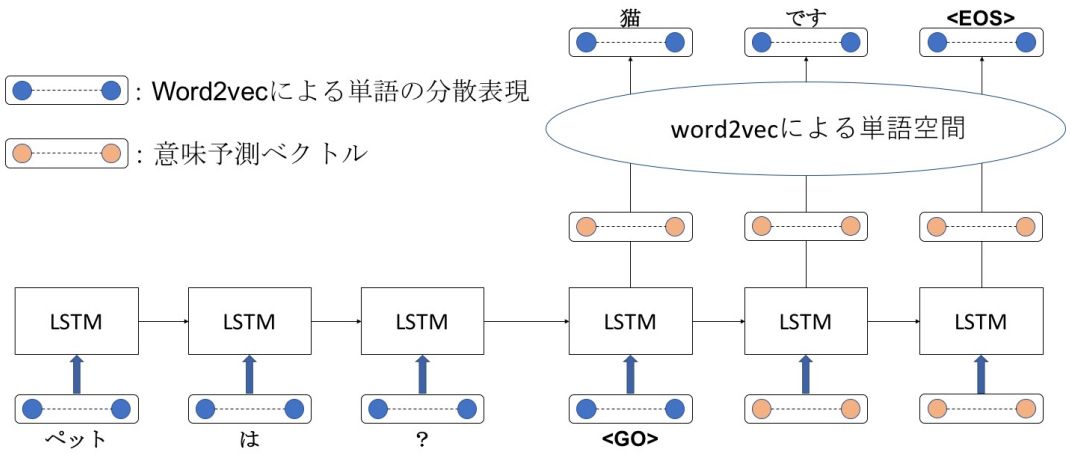

Fig. 3　提案手法モデル

Table 1　Tweet/Reply ペアの例

Tweet	Reply
やっと ちゃんと 寝 れ た おはよう	しっかり 睡眠 は 取っ て ください ね (‘ _ ‘)
がんばれ ！ うち も 靴擦れ の なか 頑張る 笑	頑張っ た よ (;_;)
はいはい 。 いけ めん です ねー ()	棒読み 感 はんぱ ない 、

- 適切性：入力文に反応, また理解していると感じるかどうか
- 多様性：多様な返答が行えているかどうか

多様性については, 当たり障りのない相槌などではなく, その会話特有の返答を行えているかを評価基準としている. 適切性に関しては主観評価（5: とても良い, 4: 良い, 3: 普通, 2: 悪い, 1: とても悪い）, 多様性に関しては（5: 面白い, 4: やや面白い 又は 気が利いている, 3: 普通 (一般的: 当たり障りがない), 2: やや面白くない 又は 気が利いていない, 1: 面白くない）を用いた. Twitter から収集し, 学習に用いなかった 46 文で応答文生成を行い, それぞれの生成文に関して各評価について複数の評価者による 5 段階評価を行った.

6.3　実験結果・考察

Fig.4 に実験結果の比較を示す. 各指標に関して, それぞれの評価値を平均した結果を示している. one-hot 表現を用いた従来手法と比べて, 提案手法は, 多様性が向上していることが確認できる. one-hot 表現では考慮していなかった類義語を処理できる点から, 適切性の向上が期待されたが, 実際はほとんど差がみられなかった. しかし, Twitter から選んだ入力文（ユーザ発話）の意味が分からないものだと, 適切性の判断が難しかったという意見もあり, アンケートの改善が

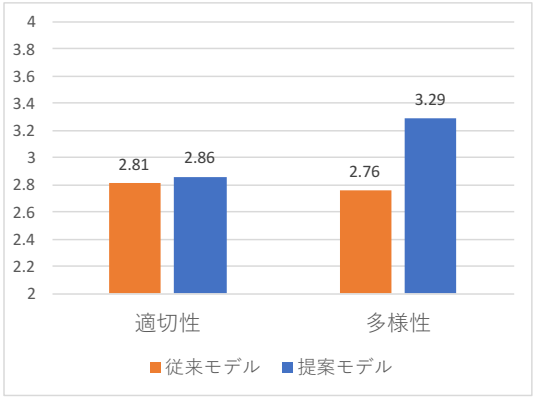

Fig. 4　主観評価実験

必要と考えられる. 多様性に関しては, 従来手法で用いられている, 単語列における各単語の生成確率の積が高くなるよう生成を行うビームサーチ [10] の影響も考えられるので, 精査が必要である. 生成された応答文の例を表 2 に示す. ここで, 「ガイル」, 「ダース・ベイダー」という単語は今回の学習データ中には含まれておらず, 従来手法ではどちらも $< unk >$ として処理されており, 応答が変化していないが, 提案手法ではそれぞれ別のベクトルに変換されており, 応答文も変化していることが確認できる.

Table 2　応答文生成例

入力文	生成文
初めまして〜お話しましょう	(提案手法) ええよ！
	(従来手法) ありがとうございます（ ´∀｀ ）
仲良くしてください！	(提案手法) ほんなら、何て呼びましょ？
	(従来手法) なんて呼んでください（ ´∀｀ ）
ガイル強いよね	(提案手法) 本当に...?
	(従来手法) そうだった（ ¯ － ¯ ）
ダース・ベイダー強いよね	(提案手法) 本当に !!!
	(従来手法) そうだった（ ¯ － ¯ ）

7　おわりに

本稿では, 単語の分散表現を入出力に用いて応答文を生成する手法について検討を行った. 従来の one-hot 表現による応答文生成よりも, 多様性のある返答が行えることを示した. しかし, 現状のモデルでは cos 類似度の最も高いものを出力としており, one-hot 表現モデルで用いられているようなビームサーチにあたる処理が行われておらず, 文法的誤りの多い応答文を生成することも多かった. また, 出力生成時に, 対話コーパスに現れなかった単語が出現したとしても文全体で見ると意味が不明瞭なものとなることが多かった. 今後は seqGAN [11] やその他の言語モデルの使用を検討し, 精度向上を目指したい. また, データセットに用いた Twitter コーパスはノイズの多いものであるので, 正解データと類似していると感じるような生成文でも, 悪い評価となることがあった. 今後は, よりノイズが少なく, 対話履歴も考慮できるような複数ターン会話のデータセットも考える必要がある.

謝辞

本研究の一部は，JSPS 科研費 JP17K00236 の支援を受けたものである

参考文献

[1] O. Vinyals and Q. Le, “A neural conversational model,” ICML Deep Learning Workshop, 2015.

[2] L. Shang *et al.*, “Neural responding machine for short-text conversation,” Proc. of ACL 2015, pp. 1577–1586, 2015.

[3] T. Mikolov *et al.*, “Linguistic regularities in continuous space word representations,” Proc. of NAACL-HLT 2013, pp. 746–751, 2013.

[4] T. Mikolov *et al.*, “Efficient estimation of word representations in vector space,” arXiv:1301.3781, 2013.

[5] T. Mikolov *et al.*, “Distributed representations of words and phrases and their compositionality,” Proc. of NIPS, pp. 3111–3119, 2013.

[6] T. KUDO, “Mecab : Yet another part-of-speech and morphological analyzer,” http://mecab.sourceforge.net/, 2005.

[7] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv:1412.6980, 2014.

[8] G. Doddington, “Automatic evaluation of machine translation quality using n-gram co-occurrence statistics,” Proceedings of the Second International Conference on Human Language Technology Research 2002 (HLT ’02), pp. 138–145, 2002.

[9] 東江恵介 *et al.*, “日英方向におけるハイブ リッド翻訳とルールベース翻訳の人手評価,” 言語処理学会第 17 回年次大会, D5-5, pp. 1127–1130, 2011.

[10] S. Wiseman and A. M. Rush, “Sequence-to-sequence learning as beam-search optimization,” Proc. of EMNLP, pp. 1296–1306, 2016.

[11] L. Yu *et al.*, “SeqGAN: Sequence generative adversarial nets with policy gradient,” arXiv:1609.05473, 2016.