

# 自己相互情報量を用いた特徴語彙予測に基づく雑談応答生成

高山 隼矢, 荒瀬 由紀

大阪大学大学院情報科学研究科

{takayama.junya, arase}@ist.osaka-u.ac.jp

## 1 はじめに

深層学習技術の発展と Twitter<sup>1</sup> などのソーシャルネットワークサービス (SNS) の普及により, 雑談対話システムを SNS 上に存在する大量の会話データによって構築する取り組みが盛んになされている。深層学習による応答生成では, 特に機械翻訳分野において広く用いられる Sequence to Sequence (seq2seq) [1] モデルを用いる手法 [2] が提案されており, 流暢な応答文を生成することが可能になっている。しかし, seq2seq では訓練時に単語単位で損失計算を行うため, 出力した単語が妥当かどうかにかかわらず参照文と同じ単語でなければ損失は下がらない。そのため既存の seq2seq ベースの応答生成器では「そうですね」や「わかる」などといった, 高頻度に出現する単語で構成される汎用的な応答ばかりを生成するように学習してしまうという問題を抱えている。

本研究では, 自己相互情報量 (Point-wise Mutual Information; PMI) を用いて, ある入力発話文に対する応答に現れやすい特徴的な語彙を事前に予測する。また, 予測語彙に含まれる単語をモデルが出力した際に報酬を与えるように損失関数を改良することで, 入力発話文に対する応答としての自然さを保ちつつ, 多様な語彙を出力するような応答生成器を構築する。モデルが出力した単語を用いた損失関数の計算のためには, 出力層の確率分布から単語をサンプリングする必要がある。しかし, 通常用いられる argmax によるサンプリングでは誤差逆伝播ができない。そこで本研究では確率分布から微分可能な形で擬似的にサンプリングを行う手法である Gumbel-Softmax [3] を用いることでこの問題の解決を図る。

ツイッターからクロールした対話コーパスでの実験において, 自動評価でベースラインを上回る結果を得られた。提案手法の方が発話と強い共起を持つ語を有意に多く生成することも確認している。

<sup>1</sup><https://twitter.com>

## 2 前提知識

### 2.1 ニューラル対話モデル

ニューラル対話モデル (Neural Conversation Model; NCM) [2] とは, 機械翻訳において用いられる Sequence to Sequence (seq2seq) [1] を対話システムにおける応答生成に適用した手法である。このモデルはエンコーダとデコーダと呼ばれる二つの再帰ニューラルネットワーク (Recurrent Neural Networks; RNN) によって構成される。エンコーダは発話の単語列を入力として受け取り, 発話の内容を表すような実数値ベクトルを出力する。デコーダはエンコーダの出力を初期状態として受け取り, 発話内容に沿った応答文を一単語ずつ順に生成する。NCM ではデコーダの各出力単語と参照文の各単語との間の単語単位の交差エントロピー損失によって学習を行う。このような学習方法では, 出力した単語が発話に対する応答として妥当であるかどうかにかかわらず, 参照文と同じ単語を出力しなければ損失が下がらないため, 頻度の高い (正解しやすい) 汎用的な語を多く出力してしまう。

この問題の解決を図った手法の一つに, MMI-bidi [4] がある。この手法では発話に対する相互情報量が高い応答を返すために, 発話文  $Q$  と生成文  $R$  の間の PMI を以下で近似している:

$$\text{PMI}(Q, R) = (1 - \lambda) \log P(R|Q) + \lambda \log P(Q|R) \quad (1)$$

ここで,  $P(R|Q)$ ,  $P(Q|R)$  は共に seq2seq を用いて構築し, 訓練は別々に行う。テスト時に  $P(R|Q)$  モデルによって  $N$ -best の候補文を生成したのち, 式 (1) を用いてリランキングを行うことで, より発話との共起が強い応答の生成を実現している。しかし, この手法は訓練時に相互情報量を最大化しないため,  $P(R|Q)$  による  $N$ -best の生成において妥当性の高い語が出力されなかった場合には効果が期待できない。そのため, 発話文との共起が強い語が出やすくなるような目的関

数を  $P(R|Q)$  の訓練時に加えることで, MMI-bidi を適用した際の効果も高まると期待できる.

## 2.2 Gumbel-Softmax

応答生成においては通常デコーダの出力層に対して  $\text{argmax}$  操作を行うことで確率分布からの単語サンプリングを行う. しかし  $\text{argmax}$  操作の出力は離散的であり, 誤差逆伝播ができない. Jang ら [3] は, この問題を解決するために, 誤差逆伝播可能な形で確率分布から擬似的なサンプリングを行う Gumbel-Softmax を提案している. Gumbel-Softmax では, ある  $k$  クラスの確率分布  $\pi$  (応答生成器における出力層に相当) に対して以下の計算を行う.

$$y_i = \frac{\exp((\log(\pi_i) + g_i)/\tau)}{\sum_{j=1}^k \exp((\log(\pi_j) + g_j)/\tau)}$$

ここで,  $\tau$  は温度と呼ばれるハイパーパラメータで, 小さいほどワンホットに近いベクトルが得られるが, 勾配の分散が大きくなる.  $g_i$  は一様分布  $u_i \sim \text{Uniform}(0, 1)$  を用いて以下の計算によって得られる.

$$g_i = -\log(-\log(u_i))$$

## 3 提案手法

提案手法の概要を図 1 に示す. 提案手法ではまず PMI に基づいて入力発話と強く共起する特徴語彙を予測し (第 3.1 節), 次に Gumbel-Softmax を用いてデコーダが出力した語彙をサンプリング (第 3.2 節) する. 最後に予測した特徴語彙のうち, デコーダが生成した語の割合を計算し, その値を損失関数に加える (第 3.3 節).

### 3.1 自己相互情報量に基づく特徴語彙予測

特徴語彙予測器は, 学習コーパス全体から PMI を計算し, ある入力発話文に対する応答文に現れやすい特徴的な語彙を予測する. 単語  $x$  が発話文・応答文中に現れる確率をそれぞれ  $P_Q(x)$ ,  $P_R(x)$  とし, 単語  $x, y$  が発話・応答対の発話・応答文中にそれぞれ存在する確率を  $P(x, y)$  とする. このとき PMI は以下で計算される.

$$\text{PMI}(x, y) = \log_2 \frac{P(x, y)}{P_Q(x) \cdot P_R(y)}$$

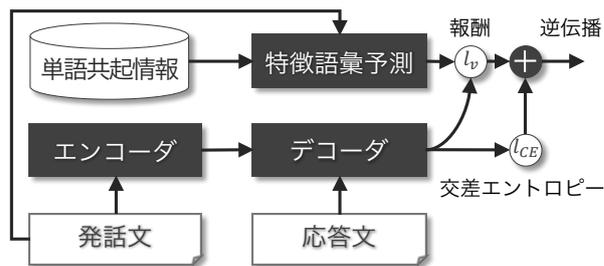


図 1: 提案手法の概要

また, ある発話文の単語集合を  $Q = \{q_1, q_2, \dots, q_L\}$  応答文の語彙の候補を  $V_R = \{v_{R_1}, v_{R_2}, \dots, v_{R_N}\}$  とする. このとき, ある語彙  $v_{R_n} \in V_R$  のスコアを以下で定義する.

$$\sum_{q \in Q} \text{PMI}(q, v_{R_n})$$

$V_R$  の全てに対してスコアを計算し, トップ  $k$  件のみからなる語彙の集合を予測語彙集合  $V_{\text{Pred}}$  とする.

### 3.2 Gumbel-Softmax を用いた生成語彙の擬似サンプリング

PMI に基づいて予測した語彙をデコーダが生成したかどうかを判断するためには, デコーダが生成した語彙をサンプリングする必要がある. 提案手法では, 各タイムステップ  $i$  におけるデコーダの出力ベクトル  $\pi$  に対して Gumbel-Softmax を式 (2) のように適用することで, 微分可能な形で擬似的な Bag-of-Words ベクトル  $\mathbf{B}$  が得られる.

$$\mathbf{B} = \sum_{i=1}^T \text{GumbelSoftmax}(\pi_i) \quad (2)$$

### 3.3 損失関数の設計

入力発話と強く共起する語をデコーダがより多く出力できるように,  $V_{\text{Pred}}$  に含まれる語彙を生成した分だけ値が下がるような損失関数  $l_v$  を設計する. 具体的には,  $t(b_n)$  を, ベクトル  $\mathbf{B}$  中の  $n$  番目のインデックスに対応する単語とした時,  $l_v$  を以下で定義する.

$$l_v = -\sum_{n=0}^N f(b_n, V_{\text{Pred}}),$$

$$f(b_n, V_{\text{Pred}}) = \begin{cases} \min(b_n, 1) & (t(b_n) \in V_{\text{Pred}}), \\ 0 & (\text{otherwise}), \end{cases}$$

表 1: 実験結果

	BLEU-1	BLEU-2	NIST	dist-1	dist-2
Proposal + MMI	8.970	2.881	0.618	0.025	0.063
Proposal	<b>9.022</b>	<b>2.906</b>	<b>0.640</b>	0.023	0.055
MMI	7.930	2.568	0.479	<b>0.038</b>	<b>0.093</b>
Baseline	7.833	2.527	0.484	0.035	0.082

$f(b_n, V_{pred})$  において  $\min(b_n, 1)$  としているのは、単語  $t(b_n)$  が複数回生成された場合には報酬を加えない（その単語を使ったかどうかのバイナリ判定に基づいて報酬を加える）ためであり、同じ単語が何度も出力されるのを抑制する狙いがある。

最終的に訓練において最小化する損失関数  $\mathcal{L}$  は、交差エントロピー誤差の総和  $l_{CE}$  に  $l_v$  を加え、以下のように定義される。

$$\mathcal{L} = (1 - \alpha) \cdot l_{CE} + \alpha \cdot l_v$$

ただし、 $\alpha$  は語彙による報酬の度合いを調整するハイパーパラメータである。

## 4 実験

提案手法によって高頻度な応答を抑制し、発話に対する妥当な応答が生成されることを検証するため、大規模雑談データを用いて応答生成の実験を行う。

### 4.1 実験設定

Twitter からクロールした約 130 万件の発話応答データのうち、1 万件を検証データ、1 万件をテストデータ、残りを訓練データとして実験を行った。実験にあたり、提案手法を適用してビームサーチを行ったモデル (Proposal) と、そのモデルのデコード時に MMI-bidi を適用したモデル (Proposal+MMI) を用意した。また、比較対象としては交差エントロピー損失によってのみ訓練したモデル (seq2seq) と、そのモデルのデコード時に MMI-bidi によってリランキングを行ったモデル (MMI) を用意した。ビーム幅は全てのモデルにおいて 10 とした。また、学習時にはオプティマイザとして Adam [5] を用いており、全てのモデルについて学習率は 0.0002 とし、その他のパラメータは推奨値を用いた。語彙予測における  $k$  は 100 とし、Gumbel-Softmax の温度パラメータ  $\tau$  は 0.1 とした。

表 2: 予測語彙使用率の比較

	使用率
Proposal + MMI	5.7%
MMI	5.0%

### 4.2 実験結果

全体の評価は正解の応答文に対する BLEU, distinct [4], NIST [6] で行った。ここで distinct は、システムが生成した全ての生成文に含まれる n-gram の種類数を n-gram の総数で割ったものであり、システムが生成した n-gram の多様性を評価する指標である。また、NIST は BLEU をベースとした生成文と参照文との一致度を測る指標であるが、BLEU とは異なり n-gram に対して情報量の大きさを用いて重み付けを行うため、機能語よりも内容語の一致が重視される。実験結果を表 1 に示す。

提案手法とベースラインを比較すると、提案手法では distinct が下がった一方で、BLEU はやや上昇、NIST は大幅に上昇していることがわかる。この結果より、提案手法を適用することでデコーダが生成する語彙の多様性は多少低下するが、妥当な内容語を多く生成できていると考えられる。また、提案手法に MMI を適用したモデルは、提案手法単体に比べて BLEU や NIST が多少低下した一方で、distinct は大きく上昇していることから、2つの手法を組み合わせることにより応答の妥当性を担保しつつ多様性を高めることができると考えられる。

提案手法を適用することにより実際に予測した特徴語彙が多く生成されるようになったかどうかを、Proposal+MMI と MMI の間で比較した。評価の方法としては、テストデータを用いて応答を生成したあと、その応答のなかに PMI による予測語彙上位 100 件のうち何 % が用いられているかを算出し、その平均をとった。結果は表 2 の通りで、提案手法の方が予測語彙を多く生成していることがわかる。なおそれぞ

表 3: 提案手法と比較手法の応答生成例

<b>Input:</b>	いっぱい絡んでいただけると嬉しいです
<b>Proposal+MMI:</b>	もちろんです！絡みましょう！
<b>MMI:</b>	こちらこそ！
<b>Input:</b>	これは絶対みた方がいいってアニメ教えて！
<b>Proposal+MMI:</b>	七つの大罪も見てみます！
<b>MMI:</b>	SAO の曲を聴いてみたいと思います！
<b>Input:</b>	昨日サンタさんに会ってきましたよ
<b>Proposal+MMI:</b>	おー！よかったね！私も会いたい！
<b>MMI:</b>	しゃしん

れの使用率の分布に正規性は確認できなかったため、Wilcoxon の t 検定を実施したところ、有意水準 0.1% で有意差が確認できた。

Proposal+MMI と MMI の応答例をテストデータより抽出したものを表 3 に示す。MMI では全体的に短い応答が生成される傾向にあり、提案手法の方が長文の応答を生成していることがわかる。実際、文長（単語数）の平均は Proposal+MMI が 8.32, MMI が 7.76 で、Wilcoxon の t 検定を実施したところ、有意水準 1% で有意差が確認できた。また二つ目の例を見るとわかるように、MMI では「アニメ」という語に対して「SAO」と、妥当な語を生成できている一方で、「みた」という動詞を処理できていない。一方提案手法では「絡ん」に対して「絡み」、「みた」に対して「見て」、「会っ」に対して「会い」など、名詞のみでなく発話中でキーワードとなるような動詞に対する妥当性が高い語を出力する傾向がみて取れる。一方、生成文中の名詞の多様性については、三つ目の例からもわかるように、MMI の方が全体的にユニークな名詞を生成する傾向があった。提案手法では PMI ベースで特徴語彙予測を行うため、参照文集合での出現頻度が低すぎる名詞（発話文集合に強く共起する単語がない名詞）は出力されにくいと考えられる。

## 5 おわりに

本研究では、雑談対話ドメインにおけるニューラル応答生成器が応答としての自然さを保ちつつ多様な応答を生成できるよう、PMI に基づく特徴語彙予測を用いて生成する語のヒントを与えながら学習を進める手法を提案した。実験では既存手法に比べ高い BLEU, NIST スコアを得られることを確認した。特に、提案手法では語彙予測によって得られた共起が強い語を既存手法よりも有意に多く生成していることが確認でき

た。今後は人手評価や定性的な分析をさらに進める予定である。また、単語単位ではなく文単位での共起の強さを評価し、それが大きくなるように訓練を行う手法を開発する予定である。

## 謝辞

本研究は JSPS 科研費 JP18K11435 の助成を受けたものです。

## 参考文献

- [1] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Proceedings of NIPS 2014*, 2014.
- [2] Oriol Vinyals and Quoc V Le. A neural conversational model. In *Proceedings of ICML'15*, 2015.
- [3] Eric Jang, Google Brain, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *ICLR2017*, 2017.
- [4] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. In *Proceedings of NAACL-HLT 2016*, pages 110–119, 2016.
- [5] Diederik P Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *ICLR2015*, 2015.
- [6] George Doddington. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of HLT 2002*, 2002.