再帰的にエンコードを行う階層型 Transformer による マルチターン雑談対話の応答生成

岩間 寛悟 [1]　狩野 芳伸 [2]

静岡大学大学院 総合科学技術研究科

[1]kiwama@kanolab.net, [2]kano@inf.shizuoka.ac.jp

1　はじめに

人々が日常生活で行う雑談対話は、頻繁に行われる話者交替によって、複数のターンから成る。このマルチターンの雑談対話では、返答のきっかけとなる直前の相手の発話より過去の発話の対話履歴が文脈として返答に利用される。本研究では、マルチターンの雑談対話において、自然で首尾一貫性のある返答を行う対話生成を目指す。マルチターン対話の応答文を生成するニューラル対話モデルである Hierarchical Recurrent Encoder Decoder (HRED) [1]の RNN 部分を Transformer[2]に置き換えた階層型 Transformer をベースにし、各発話のエンコード時において過去の発話情報を用いた手法を提案する。BLEU-N および DIST-N[3]を評価基準とした自動評価実験と、文の自然性と首尾一貫性を評価基準とした人手評価実験により、その効果を確認する。

2　関連研究

マルチターンのニューラル対話モデルとして、RNN を用いた Hierarchical Recurrent Encoder Decoder (HRED)[1] が提案された。HRED は Seq2Seq を拡張した、階層的な構造をもつモデルである。各ターンの発話は分けて Encoder に入力される。Encoder RNN と Decoder RNN に加え、新たに追加した Context RNN で文レベルでの文脈も学習する。

他方、マルチターン対話に関連する研究において Attention[4, 5]や Transformer を用いたモデルも提案されている。Hierarchical Recurrent Attention Network[6]では HRED に Attention を導入した。Li ら(2019)は特定の分野の文書を入力の一つとし、その分野に関する対話を行う Document Grounded Task のモデルのベースとして Transformer を用いた[7]。また Su ら(2019)はマルチターン対話において、単語の省略を補完するように応答文を書き換えるモデルで Transformer を用いた[8]。いずれのモデルも RNN を用いたモデルと比較し、自動評価や人手評価の結果より性能の向上が示された。なお雑談対話以外のタスク指向システムにおいても、Vlasov ら(2019)はスロットとシステムの行動を入力とする対話モデルに Transformer を用いるなど[9]、Transformer を用いたニューラル対話モデルが多く提案されている。

3　階層型 Transformer

本研究では、HRED の RNN 部分を Transformer に置き変えたモデルを階層型 Transformer と呼び、階層型 Transformer をベースにしたモデルによるマルチターン対話の応答生成を行う。階層型 Transformer は 2 つの Encoder と 1 つの Decoder から成る。トークン列を処理する Encoder を Utterance Encoder、文の文脈を処理する Encoder を Context Encoder と定義する。図 1 にモデルの概要を示す。

n-1 回のターンから構成される発話列 $U = \{u_1, u_2, \dots, u_{n-1}\}$の各発話をトークン列として入力し、1 ターン分の発話u_nのトークン列を出力するニューラル対話モデルである。まず Utterance Encoder に複数の発話を入力し、各発話の文ベクトルを得る。次に Context Encoder に文ベクトルを入力し、文の文脈を表すコンテキストベクトルを得る。最後に、Decoder は前の単語とコンテキストベクトルを用いて、単語確率分布を計算する。

3.1　Utterance Encoder

Utterance Encoder は、Transformer の Encoder と同一であり、1 文単位でトークン列をエンコードするモジュールである。発話u_mの時刻 i におけるトークンの単語ベクトルをx_{mi}とすると、入力ベクトルx^u_{mi}は以下のように計算される。

$$x^u_{mj} = x_{mi} + PositionalEncoding(i)$$

PositionalEncoding(t)は Transformer の機構で用いられる positional encoding と同一である。Utterance Encoder の入力は単語ベクトルと positional encoding を足し合わせたベクトルである。Self-Attention a^u_mおよび Utterance Encoder の出力ベクトルs^u_mは、トークンの系列長がL^mのと

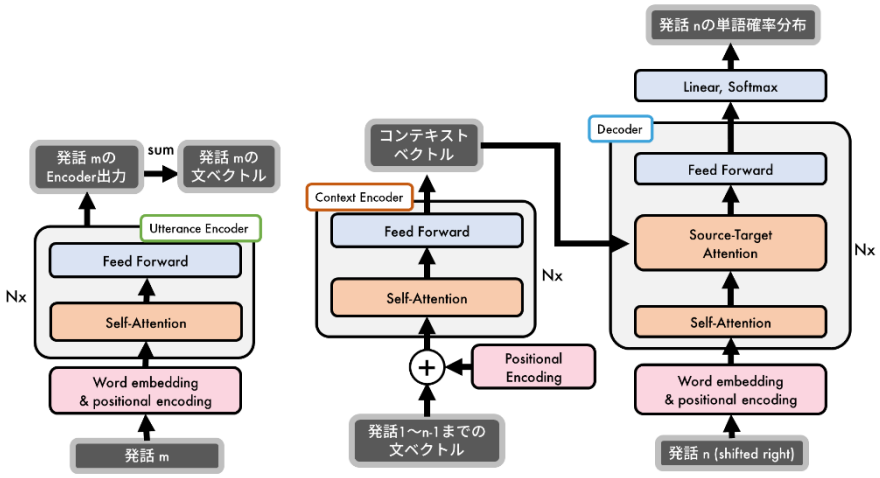

図 1: 階層型 Transformer のモデル概要

き、以下のように計算される。なお文ベクトルの算出は Memory Networks[10]の手法を参考とした。

$$a_m^u = MultiHead(x_m^u, x_m^u, x_m^u)$$

$$s_m^u = \tanh(\sum_j^{L^m} \mathrm{FFN}(a_{mj}^u))$$

MultiHead(Query, Key, Value)はクエリ、キー、バリューの 3 種類の入力によって計算される Multi-Head Attention である。FFN(x)はフィードフォワードニューラルネットワークである。

3.2 Context Encoder

Context Encoder の各モジュールは Utterance Encoder と同一である。したがって発話 m の文ベクトルをs_m^uとすると、入力ベクトルx_m^cは以下のように計算される。

$$x_m^c = s_m^u + PositionalEncoding(m)$$

また Self-Attention a^cおよび Context Encoder の出力s^cは以下のように計算される。

$$a^c = MultiHead(x^c, x^c, x^c)$$
$$s^c = \mathrm{FFN}(a^u)$$

3.3 Decoder

モジュールの構成は Transformer の Decoder に等しい。発話u_nの時刻 i におけるトークンの単語ベクトルをx_{ni}とすると、入力ベクトルx_i^dは以下のように計算される。

$$x_i^d = x_{ni} + PositionalEncoding(i)$$

Self-Attention a^{dself}およびコンテキストベクトルとの Source-Target Attention a^{dst}の出力s^cは以下のように計算される。

$$a^{dself} = MultiHead(x^d, x^d, x^d)$$
$$a^{dst} = MultiHead(a^{dself}, s^c, s^c)$$

最終的な単語ラベルの確率分布P_wは以下のように計算される。

$$s^d = \mathrm{FFN}(a^{dst})$$
$$P_w = \mathrm{softmax}(w_g s^d + b_g)$$

3.4 特徴

HRED や階層型 Transformer では、各ターンの発話のエンコードを独立して行う。HRED や階層型 Transformer では n 回のターンの対話に続く応答を生成する際、1 ターン目から n ターン目までの各発話を独立した文として Utterance Encoder でエンコードする。つまり各発話の文ベクトルの算出時には、各発話同士が干渉していないものとされている。

4 提案手法

普段の会話において、n 回のターンにわたって会話をした際、過去のターンの発話はそれ以前のターンの発話の文脈の影響を受ける。たとえば過去の 2 ターン目や 3 ターン目の発話は 1〜2 ターン目の発話に影響を受けていると考えられる。また n ターン目以前の各ターンの時点においても会話としてみなすことができる。一方、HRED や階層型 Transformer の Utterance Encoder では、過去の n ターン分の発話は独立した文としてまずエンコードされていた。以上を踏まえ、本研究では、各ターンの発話のエンコードを該当ターンより前のすべての発話エンコード結果も用いて再帰的に行う手法を提案する。

Utterance Encoder に Source-Target Attention を導入し、以前のターンまでで計算された文ベクトルとの Attention を計算する。図 2 にモジュールの概要を示す。提案手法の Utterance Encoder において、発話u_mのエン

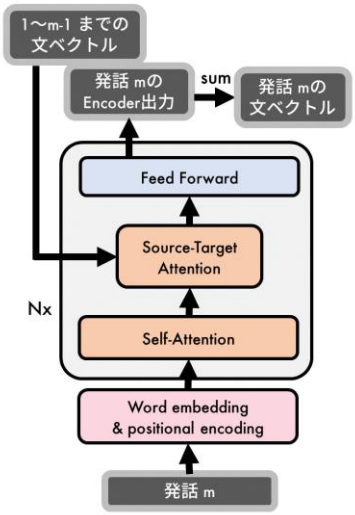

図 2: 提案手法における Utterence Encoder

コード時における Self-Attention a_m^{uself}および 1 ターン目から直前のターンまでで計算された文ベクトルを連結したs_{m-1}^uとの Source-Target Attentiona_m^{ust}、Utterance Encoder の出力s_m^uは以下のように計算される。

$$a_m^{uself} = MultiHead(x_m^u, x_m^u, x_m^u)$$
$$a_m^{ust} = MultiHead\left(a_m^{uself}, s_{m-1}^u, s_{m-1}^u\right)$$
$$s_m^u = \tanh(\sum_j^{L^m} \mathrm{FFN}(a_{mj}^{ust}))$$

なお 1 ターン目のエンコード時に Source-Target Attention の計算に用いるベクトルs_0^uは零ベクトルである。

5 評価実験

本研究では、ベースラインおよび提案手法による自動生成文を用いた自動評価および人手評価実験を実施した。また入力となる発話のターン数による影響を調べるため、入力に用いる対話のターン数が 5 ターンと 10 ターンからなる 2 種類のデータを用いた。

5.1 データセット

本研究では Twitter API を用いて収集した日本語のマルチターン対話データを用いる。自身へのリプライ、bot によるツイート、2 文字以上かつ 50 文字以下でないツイートは除外した。前処理として、ツイート内に含まれるリプライ先のスクリーンネーム、URI、ハッシュタグ、改行を正規表現で除去した。トークンへの分割には訓練データを用いて学習した SentencePiece[1]を用いた。入力が 5 ターンの対話データは 170 万件、10 ターンの対話データは 94 万件用いた。そのうち 1 万件をテストデータ、1,000 件を検証データに用い、その他を訓練データに用いた。

5.2 訓練設定

モデルの入力および出力に用いる語彙数は 30,000 に設定した。単語埋め込みの次元数は 256 次元、フィードフォワードネットワークの次元数は 1,024 次元に設定した。Utterance Encoder、Context Encoder、Decoder のレイヤーは 6 層とした。Multi-Head Attention のヘッド数は 8 に設定した。最適化関数には Adam を用い、学習率は 1e-4 に固定した。訓練時には Label Smoothing を導入し、ε =0.1 に設定した。バッチサイズは 64 に設定した。モデルの実装には PyTorch (ver.1.1.0)[2]を用いた。訓練は 20Epoch 行い、各 Epoch で検証データに対する損失を算出し、評価時に用いるモデルを選択した。推論時の探索手法として、ビーム幅が 4 の Beam Search を用いた。なお評価時に生成結果から文末を表す EOS タグや文頭を表す SOS タグ、未知語タグを除外した。

5.3 自動評価

自動評価の指標として、N-gram の適合率をもとに参照文との一致度を表す BLEU-N (N=1, 2, 3, 4)および、系列長に対する語彙数の割合によって多様性を表す DIST-N (N=1, 2)[3]を用いた。表 1 に自動評価の実験結果を示す。なおベースラインは階層型 Transformer であり、Human は実際に行われた発話を表す。5 ターンの発話からなる入力を用いた評価では、ベースラインと比較して提案手法の BLEU スコアの上昇がみられ、特に BLEU-2,3,4 において 0.2〜0.4 ポイント上昇した。他方、10 ターンの発話からなる入力を用いた場合では、提案手法の BLEU スコアはベースラインとほぼ同じであったが、DIST-N のスコアは上昇した。自動評価では、入力となる発話のターン数によって傾向が異なる結果となった。

5.4 人手評価

Li ら[7]が実施した対話システムに対する人手評価を参考にし、1〜4 の 4 段階のリッカート尺度を用い、評価基準として文の自然性と首尾一貫性の 2 種類を設定した。文の自然性は、生成文の日本語文としての流暢性を表す。これは対話履歴にかかわらず、生成文のみ

[1]https://github.com/google/sentencepiece
[2]https://pytorch.org

表 1: 各モデルの入力ターン数ごとの BLEU-N スコアおよび DIST-N スコア

	5 ターン						10 ターン					
	BLEU-N				DIST-N		BLEU-N				DIST-N	
	1	2	3	4	1	2	1	2	3	4	1	2
Baseline	13.00	7.54	4.66	2.73	3.87	13.84	13.48	8.08	5.18	3.18	3.98	14.35
提案手法	13.03	7.75	4.97	3.13	3.83	13.05	13.26	8.03	5.16	3.13	4.11	15.11
Human	-	-	-	-	21.12	83.79	-	-	-	-	21.67	83.99

表 2: 各モデルの入力ターン数ごとの人手評価結果 (括弧内は標準偏差)

	5 ターン		10 ターン	
	文の自然性	首尾一貫性	文の自然性	首尾一貫性
Baseline	3.29 (±0.56)	2.51 (±0.79)	3.11 (±0.57)	2.44 (±0.64)
提案手法	3.34 (±0.90)	2.46 (±0.80)	3.41 (±0.27)	2.33 (±0.66)
Human	3.35 (±0.56)	2.93 (±0.72)	3.57 (±0.29)	3.01 (±0.47)

を見て評価する。首尾一貫性は、生成文の内容が文脈を考慮しているかどうかを表す。なおスコアは、各評価基準に合致しているほど高い。

人手評価には、テストデータから無作為に抽出した、入力となる発話が 5 ターンと 10 ターンの各 10 件、計 20 件のデータを用いた。評価者には各 20 件について 5 ターンおよび 10 ターンの対話を提示し、それに続く 2 種類の自動生成文および実際に Twitter 上でユーザがおこなった発話の計 3 種類の応答文を提示した。表 2 に人手評価の平均値を示す。入力の発話数が 5 ターンと 10 ターンの双方において同様の傾向を示し、特にベースラインと比較して提案手法の文の自然性スコアは上昇した。入力が 5 ターンの場合では実際の発話と同程度のスコアであり、10 ターンの場合では 0.3 ポイント上昇した。一方、首尾一貫性のスコアは上昇しなかった。またベースラインと比較して単語の繰り返しが抑制される傾向がみられた。

6 おわりに

本研究では、マルチターン雑談対話の応答文の生成において、HRED の RNN 部分を Transformer に置き換えた階層型 Transformer を、各発話のエンコード時に過去の発話情報を用いて再帰的なエンコードを行う手法を提案した。提案手法はベースラインと比較し、自動評価では、ターンの短い対話において BLEU スコアが上昇し、長い対話では多様性が向上した。また人手評価では、文の自然性の向上がみられた。今後は、テキストだけでなく知識ベースなどを入力に用いる Document Grounded Task や雑談対話ではないタスク指向のシステムに対して提案手法を適用した場合の効果を検証していきたい。また首尾一貫性の向上に効果的な文脈情報の生成手法についてさらに検討していきたい。

参考文献

[1] Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, Joelle Pineau. Building end-To-end dialogue systems using generative hierarchical neural network models. In AAAI 2016, pp. 3776–3783, 2016.

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin. Attention Is All You Need. In Advances in Neural Information Processing Systems, pp. 6000–6010, 2017.

[3] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, Bill Dolan. A Diversity-Promoting Objective Function for Neural Conversation Models. In NAACL HLT 2016, pp. 110–119, 2016.

[4] Minh Thang Luong, Hieu Pham, Christopher D. Manning. Effective approaches to attention-based neural machine translation. In EMNLP 2015, pp. 1412–1421, 2015.

[5] Dzmitry Bahdanau, Kyung Hyun Cho, Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In ICLR 2015, 2015.

[6] Chen Xing, Yu Wu, Wei Wu, Yalou Huang, Ming Zhou. Hierarchical recurrent attention network for response generation. In AAAI 2018, pp. 5610–5617, 2018.

[7] Zekang Li, Cheng Niu, Fandong Meng, Yang Feng, Qian Li, Jie Zhou. Incremental Transformer with Deliberation Decoder for Document Grounded Conversations. In ACL 2019, pp. 12–21, 2019.

[8] Hui Su, Xiaoyu Shen, Rongzhi Zhang, Fei Sun, Pengwei Hu, Cheng Niu, Jie Zhou. Improving Multi-turn Dialogue Modelling with Utterance ReWriter. In ACL 2019, pp. 22–31, 2019.

[9] Vladimir Vlasov, Johannes E. M. Mosig, Alan Nichol. Dialogue Transformers. CoRR, abs/1910.00486, 2019.

[10] Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, Rob Fergus. End-to-end memory networks. In Advances in Neural Information Processing Systems, pp. 2440–2448, 2015.