モーラを考慮した Fine-tuning による口語短歌生成

浦川通 [1]　新妻巧朗 [1]　田口雄哉 [1]　田森秀明 [1]　岡崎直観 [2]　乾健太郎 [3,4]
[1] 株式会社朝日新聞社　[2] 東京工業大学　[3] 東北大学　[4] 理化学研究所
{urakawa-t,niitsuma-t,taguchi-y2,tamori-h}@asahi.com,
okazaki@c.titech.ac.jp, inui@tohoku.ac.jp

概要

短歌は日本語における詩の形態の一つで，短い文字列をモーラの定型に従わせることで，日本語話者の間に広く伝えられる性質を持つ．短歌の自動生成，特に現代語で表現された口語短歌に焦点を当てた場合，データセットの数が少なく，従来手法を適用することが難しい．そこで本研究では教師データの少ない口語短歌の自動生成を行うため，モーラ情報を考慮した事前学習済み言語モデルの Fine-tuning 手法を提案する．具体的には，GPT-2 にモーラ情報を埋め込み表現として与え，残りモーラ数を考慮しながら生成することで，指定モーラ数を満たす系列生成が可能なことを実験により確認した．

1　はじめに

短歌は日本語における定型詩の一つで，5・7・5・7・7 の計 31 モーラをもつ 5 句から構成される短詩である．短い文字列の中で自然から社会生活，また個人的な日常まで幅広く表現することができ，かつ定型によって読み方が共有されていることから，他者と創造的にコミュニケーションする手段となりうる．本研究では，広告や見出しなどの他分野に短歌の特徴を応用をすることを目的として，自動生成のタスクに取り組む．利用者の性質から，読み手に文語への理解を要求しない口語短歌に対象を絞った上で，日本語として破綻のない 31 モーラの定型に従う系列生成問題として扱う．

モーラは，それぞれの言語における時間的な長さをもった音の分節単位で，日本語では音節と区別される（表 1）．日本語話者が音を数える際にはこのモーラを単位として数えることが多く，和歌 [1] や俳句 [2]，また歌唱旋律 [3] や歌詞 [4] の自動生成において，このモーラをモデル訓練時の素性に取り入れ生成を行う手法の提案が行われてきた．一方で，現代語により表現される口語短歌は和歌と同じ 31 モーラの定型をもちながらも整備されたデータセットが存在せず，これまでに提案されている手法をそのまま適用するのは難しいと考える．

表 1　日本語における音節とモーラ

語	読み	音節	モーラ
新聞	シンブン	シン / ブン	シ / ン / ブ / ン
切手	キッテ	キッ / テ	キ / ッ / テ

近年では事前学習済み言語モデルに対して，タスクに合わせた少量の教師データによる Fine-tuning を行う方法が多く見られる [5]．そこで本研究では，学習データとして現代語の文書から抽出された少量の疑似短歌データを，実際の口語短歌データの代替として用い，GPT-2 [6] に対するモーラ数制約を考慮した口語短歌生成タスクの Fine-tuning 手法を提案する．

実験の結果，提案手法により短歌のもつモーラ数制約を満たす系列が生成できることを，ビーム探索とサンプリング生成におけるモーラの正答率により確認した．また，短歌のモーラ数制約を満たす生成系列の流暢度を人手評価により評価した．

2　提案手法

2.1　口語短歌生成タスク

本研究で取り組む口語短歌生成タスクを以下で定義する．「口語」とは話し言葉・書き言葉に依らず，現代で一般的に扱われる言葉を指す．「口語短歌」とは，この口語によって書かれた全 31 モーラをもった系列とする．このモーラ数を超える，また未満となる，短歌において「破調」と呼ばれる系列については，今回のモーラ数制御の評価対象には含めない．口語短歌には一般的に知られたデータセットが存在しないため，本研究では既存の文書から疑似短歌を抽出するという形で教師データを作成し，また少量の教師データでも訓練が可能なモーラ数制

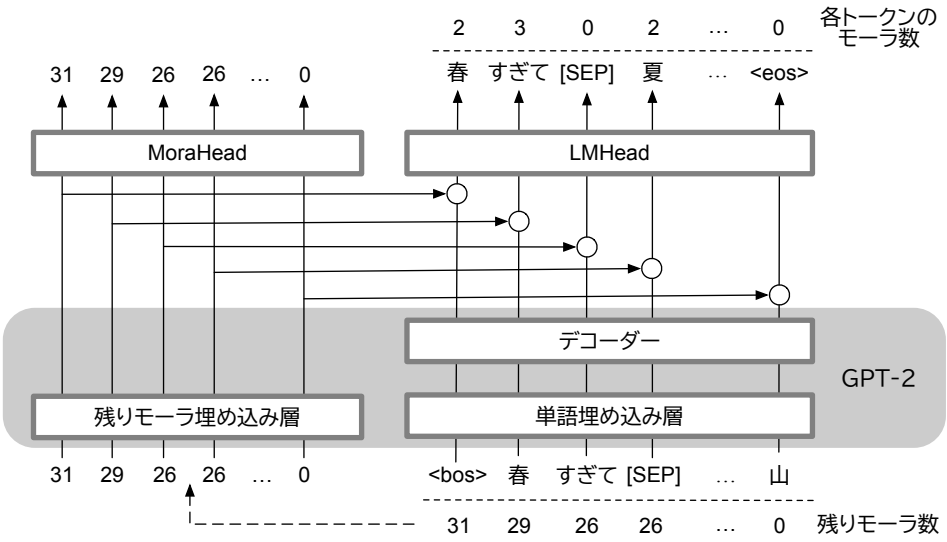

図 1　提案手法の概要

約を考慮した Fine-tuning 手法を検討する．

2.2　モデル

モデルの概要を図 1 に示す．GPT-2 にモーラ情報を取り入れるモーラ埋め込み層を用意し，入力系列の各トークンに対応する残りモーラ数を埋め込みとして計算する．この埋め込みは，GPT-2 を生成タスクで Fine-tuning するための層（本稿では LMHead と呼ぶ）と，新たに追加した MoraHead に直接与えられ，LMHead ではデコーダーの出力と結合した形でこれを受け，次トークンの尤度を計算する．MoraHead ではモーラ埋め込みから，入力で与えられているモーラ残数を推測する．これにより，モデルが残りモーラ数を考慮しながら生成を行うことが期待される．

訓練時の損失関数 $\mathscr{L}$ は以下のように定式化する．なお，入力系列 S に属するトークン集合を $W = \{w_1, w_2, \cdots . w_n | w \in S\}$ とし，入力系列中の各トークン位置に対する残りモーラ数集合を $M = \{m_1, m_2, \cdots . m_n\}$ とする．

$$\mathscr{L} = (1 - \lambda)\, \mathrm{CE}_{\mathrm{token}}(W) + \lambda\, \mathrm{CE}_{\mathrm{mora}}(M) \qquad (1)$$

ここで，$\mathrm{CE}_{\mathrm{token}}$, $\mathrm{CE}_{\mathrm{mora}}$ はそれぞれ LMHead と MoraHead のクロスエントロピー誤差を表し，これらを λ によって重み付けした上で足し合わせたものをモデル全体の損失とする．モデルに与えられる各トークンのモーラは，入力系列をモデルのトークナイザーによって分解した上で，各トークンの読みがなを MeCab[1] IPA 辞書により取得して計算する．なお，アルファベットおよび特殊トークンといった読み方の一意に定まらないトークンに関しては，すべてモーラ数を 0 とする．

表 2　疑似短歌データ

粘液の [SEP] 入った管が [SEP] あったりと [SEP] その状態は [SEP] さまざまである
偏光を [SEP] かけて重ねて [SEP] 投影し [SEP] これを偏光 [SEP] フィルタの付いた

2.3　疑似短歌データによる学習

口語短歌に関して，広く一般に知られた公開データは存在しない．そこで本研究では，短歌の定型を満たす文字列を既存の文書から抽出し，これを疑似短歌データとしてモデルの訓練に利用する．疑似短歌データは，口語短歌と同様に現代語によって書かれた，全 31 モーラからなる文字列集合とする．また実世界における応用を見据え，オープンデータからの作成を行う．今回は文字列から偶然に短歌の定型を満たす系列を抽出する偶然短歌プロジェクト[2] の公開するスクリプト[3] を用いて，Wikipedia の日本語ダンプデータから 11,344 件の教師データを作成した．なお，教師データでは各句の間に特殊トークンを挿入することで，句切れを明示した（表 2）．

3　実験

1）https://taku910.github.io/mecab/

2）http://inaniwa3.github.io/guuzen-tanka/

3）https://github.com/inaniwa3/guuzen-tanka

表 3　モーラ正答率（%）

デコード手法	beam				top-p (p = 0.6)			
	Acc@1	Acc@3	Acc@5	Acc@10	Acc@1	Acc@3	Acc@5	Acc@10
Fine-tuning	6.6	16.0	22.1	33.6	8.5	23.9	36.6	58.6
Fine-tuning with Mora (λ=0.2)	43.7	70.3	82.1	92.1	48.9	82.8	93.7	98.9
Fine-tuning with Mora (λ=0.4)	**69.0**	**88.6**	**95.3**	**98.9**	**54.5**	**88.5**	**96.2**	98.9
Fine-tuning with Mora (λ=0.6)	66.5	88.3	93.8	98.3	52.1	86.7	95.3	99.1
Fine-tuning with Mora (λ=0.8)	57.7	86.0	91.2	97.3	44.8	81.9	94.4	**99.2**

表 4　生成結果に対する人手評価（%）

	流暢だと判定された割合
Fine-tuning	76
Fine-tuning with Mora (λ=0.4)	72

表 5　より小さい学習データでのモーラ正答率（%）

	beam			top-p (p = 0.6)
データ数	Acc@3	Acc@5	Acc@10	Acc@10
1,000	66.9	77.3	88.0	94.5
3,000	75.8	83.9	91.6	95.3
5,000	83.6	92.4	97.3	98.6

3.1　実験設定

データセット　上記の Wikipedia から作成した疑似短歌コーパスを，10,000 件の学習データ，672 件の開発データ，672 件の評価データとして分割しデータセットを作成した．

モデル設定　GPT-2 の事前学習済みモデルには，HuggingFace[4] 上で公開されているモデルを使用した[5]．トークナイザーは日本語 Wikipedia を学習した Sentencepiece[7] で，語彙数は 32,000 である．ベースラインとして，GPT-2 を提案手法と同じ学習データでそのまま Fine-tuning したものを用意した．

評価　本実験では，評価データから短歌の第一句に相当する 5 モーラ分のトークンを入力し，得られる生成結果に対して評価を行う．モーラ数制約に対する性能評価として，評価データ全体のうちで 31 モーラの系列が出力候補中に含まれる割合（モーラ正答率）を計算する．ビーム幅を 10 としたビーム探索の出力から top-1,3,5,10 の正答率を得るほか，サンプリングによる生成でも出力数 n=1,3,5,10 で評価する．サンプリング手法としては，top-p サンプリング [8](p = 0.6) を採用し，3 回生成した上での平均値を取る．加えて，モーラ数制約によって結果の流暢さが損なわれないかを確認するため，生成結果が意味の通る内容となっているか否かを 3 人のアノテーターで人手評価する．アノテーション対象となる系列は，ビーム幅を 10 としたビーム探索の出力からモーラ数制約を満たすものを 50 文サンプリングし，各件で 2 人以上が選択した結果を採用として，その割合を計算する．

4）https://huggingface.co/models
5）https://huggingface.co/rinna/japanese-gpt2-medium

3.2　実験結果

表 3 にモーラ制御の正答率を示す．ベースラインではビーム探索にて最大でも 30%程度，サンプリングでは 60%程度にとどまる一方で，提案手法ではすべてのモデルで最大 90%を超える正答率となり，提案するモーラ情報の埋め込みが実際にモーラ数制約へ寄与していることが確かめられた，

表 4 に生成結果に対する人手評価の結果を示す．提案手法とベースラインとで大きな開きはなく，モーラ数制約を与える Fine-tuning が生成系列の流暢度を著しく下げるような影響は与えないことが確かめられた．一方で，教師データの中にはすべて名詞だけで構成される例なども存在し（「芝離宮 [SEP] 恩賜庭園 [SEP] 浜離宮 [SEP] 恩賜庭園 [SEP] 世界貿易」など），これらをノイズとして取り除くことでより良い結果が得られることが期待できる．

また，より少量データでの提案手法の効果をみるために，学習データを 1,000~5,000 件と絞った上でモデルを訓練し (λ = 0.6)，モーラの正答率を計算した（表 5）．結果から，少量データでの訓練においてもモーラ数制約を満たすモデルを学習できることが確認できた．これは，例えばある特定の歌人の歌など，ごく少量の教師データからモデルを訓練できることを示唆する結果であると考える．

最後に，実際の生成例を表 6 に示す．人手により第一句のみ入力し，モーラ数制約を満たすものを正例として記載した．正例では，31 モーラという制約を満たしつつ，句切れも学習データにあるような 5・7・5・7・7 となる系列がある程度出力できてい

表 6 提案手法による生成例. λ=0.6, top_p=0.6

正例	揺れだした [SEP] 雲の間から [SEP] 太陽が昇るのを見た [SEP] 群衆はただ
	さっきまで [SEP] 人間だった [SEP] 人形が [SEP] 喋ってみせる [SEP] 世界であった
	画面では [SEP] 表示されない [SEP] バーコード [SEP] セグメントを非 [SEP] 表示にできる
	まぶしくて [SEP] 見ていてつらい [SEP] 夕焼けの [SEP] 空にまにまに [SEP] オレンジの花
	光ってる [SEP] 森の中を [SEP] 移動して [SEP] 見えるのは今 [SEP] ここだというの
負例	揺れだした [SEP] 人々の間で [SEP] 共有の [SEP] 感情が生まれ [SEP] 伝播していく
	さっきまで [SEP] 平和だったのに [SEP] 今度会ったら [SEP] 違う人になってるかも
	画面では [SEP] 表示されない [SEP] メッセージ [SEP] ウィンドウが出る [SEP] 場合や [SEP] 何らかの
	まぶしくて [SEP] 見ていて気分が [SEP] わるいや [SEP] うれしくないや [SEP] わかってるや [SEP] でも
	光ってる [SEP] 波か? 光ってる [SEP] 波か? 光ってる [SEP] 波か? 光って

ることがわかる．また同様に学習データ中に存在する句またがり（「セグメントを非 [SEP] 表示にできる」というように，句切れと語の区切りが一致しないもの）が確認され，句切れを表すトークンにより教師データ中の短歌の句構造を捉えながら生成していることがうかがえる．これは，モーラ数制約を句ごとに行うことでより改善されると考える．一方負例では，指定モーラ数を超えるものが多く見られた．これは文脈によりもっともらしい系列を生成する中で，指定モーラに収まらないトークンを選択してしまう結果によるものと考えられる．また正例中にある「光ってる...」の例では，モデルの計算上では系列全体のモーラ数が 31 となっているが，実際の読みからモーラを数えると 30 とる．これは，初句に与えた「光ってる（5 モーラ）」が，トークナイザーにより「光 (ひかり・3), って (2), る (1)」と分解される中で計 6 モーラとして数えられてしまっていることが原因である．この，トークンにおけるモーラ数と実際の系列中のモーラ数の齟齬を埋める形での生成が，より精度の高いモーラ数制御には必要であることがわかった．これは例えば，サブワードを用いない事前学習済み言語モデルを利用することなどで解消されるだろうと考える．

4 関連研究

和歌や俳句，また歌詞といった詩の自動生成において，モーラ数制約の考慮や実世界での応用を視野に入れた研究として以下が挙げられる．

土佐ら [9] は，ユーザーの入力に応じたフレーズをデータベースから抽出し，ルールに基づき俳句を生成するモデルを提案している．ニューラルネットワークモデルを用いた生成では，小西 [10] らが GAN を用いて俳人による現代俳句と一般の投稿俳句を分けて学習する，幅広い層へむけた俳句生成手法を提案している．太田ら [2] は LSTM ベースの seq2seq モデルを用いて，拍数また季節の素性をモデルに与えることで，俳句の制約を満たす系列生成と内容の制御を行なっている．

和歌の生成では，RNN を用いた自動生成として Masada ら [11] がある．Masada らは，LDA を用いて同一トピックを扱う語の多い歌にスコアを付与し，歌全体での内容の一貫性を保ちながらの生成を試みている．また Yang ら [12] は，入力されるテキストから感情情報を抽出し，遺伝的アルゴリズムを用いた和歌生成を提案している．近年の例としては，Takeishi ら [1] が挙げられる．Takeishi らは Transformer+VAE を用い，尤度に対するマスクやアテンション機構を用いて，モーラ数制約や和歌の構造を考慮したモデルを構築している．

歌詞の生成では，Watanabe ら [13] は旋律によって条件付けられた歌詞の生成を，音節をモデル内部に取り入れることで実現する手法を提案している．

5 おわりに

本研究では，口語短歌の自動生成を行うために事前学習済み言語モデルに対するモーラ情報を考慮した Fine-tuning 手法を提案した．GPT-2 にモーラ情報を埋め込み，残りモーラ数を考慮しながら次にくるトークンを予測させることで，指定モーラ数を満たす系列生成が可能であることを確認した．

今回，疑似短歌を教師データとして生成を試みたが，今後より実際の短歌に近い生成モデルを学習するためのデータセットや評価手法の検討を行うとともに，指定語を含む短歌生成など，より創造性の高いアプリケーションとして扱うためのモデルの拡張に取り組みたい．

参考文献

[1] Yuka Takeishi, Mingxuan Niu, Jing Luo, Zhong Jin, and Xinyu Yang. Wakavt: A sequential variational transformer for waka generation. **Neural Processing Letters**, 2021.

[2] 太田瑶子, 進藤裕之, 松本裕治ほか. 深層学習を用いた俳句の自動生成. 研究報告自然言語処理 (NL), Vol. 2018, No. 1, pp. 1–8, 2018.

[3] 深山覚, 中妻啓, 酒向慎司, 西本卓也, 小野順貴, 嵯峨山茂樹ほか. 音楽要素の分解再構成に基づく日本語歌詞からの旋律自動作曲. 情報処理学会論文誌, Vol. 54, No. 5, pp. 1709–1720, 2013.

[4] 渡邉研斗, 松林優一郎, 乾健太郎, 後藤真孝. 大局的な構造を考慮した歌詞自動生成システムの提案. 言語処理学会第 20 回年次大会発表論文集, pp. 694–697, 2014.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[6] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. **Technical Report**, 2019.

[7] Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In **Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations**, pp. 66–71, Brussels, Belgium, November 2018. Association for Computational Linguistics.

[8] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In **8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020**. OpenReview.net, 2020.

[9] 土佐尚子, 尾原秀登, 美濃導彦, 松岡正剛. Hitch haiku˜ コンピュータによる俳句創作支援システム˜. 映像情報メディア学会誌, Vol. 62, No. 2, pp. 247–255, 2008.

[10] 小西文昂, 廣田敦士, 松尾星吾, 家原瞭, 小原宗一郎, 加賀ゆうた, 鶴田穣士, 脇上幸洋, 金尻良介, 深田智ほか. Seqgan を用いた一般人に好まれやすい俳句の生成. 2017 年度 情報処理学会関西支部 支部大会 講演論文集, Vol. 2017, , 2017.

[11] Tomonari Masada and Atsuhiro Takasu. Lda-based scoring of sequences generated by rnn for automatic tanka composition. In **International Conference on Computational Science**, pp. 395–402. Springer, 2018.

[12] Ming Yang and Masafumi Hagiwara. A text-based automatic waka generation system using kansei. **International Journal of Affective Engineering**, Vol. 15, No. 2, pp. 125–134, 2016.

[13] Kento Watanabe, Yuichiroh Matsubayashi, Satoru Fukayama, Masataka Goto, Kentaro Inui, and Tomoyasu Nakano. A melody-conditioned lyrics language model. In **Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)**, pp. 163–172, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.