E-011

拍数を考慮したサブワード単位の俳句自動生成
Haiku Generation Based on Subwords Considering the Number of Moras

仲村勇馬 † Yuma Nakamura
山本幹雄 † Mikio Yamamoto

1 はじめに

俳句は世界最短の定型詩の一つとされており、定型と呼ばれる 5·7·5 の韻律を持つ、季節を想起させる語である季語を 1 つ含むといった特徴を持っている。また、伊藤園お〜いお茶新俳句大賞*1に毎年 150 万句を超える数の俳句が投稿されているように、俳句は現代日本において多くの人に親しまれている身近な詩であるといえる。

そのため、日本語における詩生成として俳句を対象とした研究が現在盛んに行われており、近年ではニューラルネットワークを使用したものが多くを占める。先行研究では単語の拍数を素性として利用することで、生成される俳句の拍数が 5 · 7 · 5· の 17 拍により近くなることがわかっている [1]。また、文をトークン分割する際の単位に単語ではなく SentencePiece[2] のようなサブワードを使用する事で語彙サイズを削減し、更に生成器が学習データと同一の俳句を生成する割合を低下させることができる [3]。しかし、サブワードを使用する場合、サブワード単体では自身の拍数を決定することが困難となる場合がある。

本研究ではサブワードに単語の拍数の情報を効果的に付与することで、俳句自動生成の性能を向上させることを目的とする。

2 関連研究

俳句自動生成には土佐ら [4] のようなルールベースの手法と、太田ら [1] や横山ら [3] のようなニューラルネットワークを使用した手法があり、本稿では後者をベースに改良する。太田ら [1] は俳句の生成時に単語の拍数・季語の季節といった追加の素性を使用し、それによって生成される俳句の拍数・季語が改善されることを示した。横山ら [3] は文字・単語・サブワードの 3 つのトークン単位で俳句の生成を行い、トークン単位の違いが俳句自動生成に与える影響について調査した。その結果、サブワードの採用によって語彙数が削減されるだけでなく、単語単位で俳句を生成する際に 10% 強の割合で発生していた「盗作」と呼ばれる問題の発生率を低下させることを示した。盗作は生成器が学習データと全く同一の俳句を生成してしまう問題で、俳句生成のような芸術分野の生成に特有の問題である。

3 提案手法:拍数を考慮したサブワードの利用

拍数素性とサブワードの使用という前述した 2 つの手法を併用する場合、サブワードに対して一意な拍数を与える必要があるが、それが困難となる場合がある。例としては熟字訓が挙げられ、熟字全体に対して 1 つの読みが割り当てられるため、1 つの熟字訓が複数のサブワードに分割された場合それらの拍数を決定することができない。その他に、文全体が与えられた際は各サブワードの拍数が決定できるが、生成途中ではサブワードの拍数が不定であるという場合も存在する。例としては単語「暑い」がサブワード「暑」「い」と分割された場合が挙げられる。この場合ではサブワード「暑」だけが与えられた場合、その続きとしては「暑」の拍数が 2(あつ) となる「い」以外に、拍数が 1(しょ) となる「中」などが考えられる。そのため、サブワード「暑」のみではその拍数を決定することができない。

本研究で提案する手法では図 1 に示すように、モデルの入出力におけるトークン単位としてはサブワードを使用しつつ、拍数の付与を単語単位で行うことでこの問題を解決する。この手法ではサブワードと単語は多対 1 で対応している。サブワードと単語の相互変換には階層的 NMT デコーダ [5] で使用されている、文字分散表現と単語分散表現を相互変換する機構を導入する。本稿ではこの機構を変換機構、変換機構を導入した言語モデルを階層的言語モデルと呼称する。階層的言語モデルを使用することによって、単語による正しい拍数の割り当てとサブワードによる語彙サイズの削減・盗作の発生率低下を両立することができると考えた。

言語モデルと変換機構のアーキテクチャには transformer[6] を使用した。拍数の情報の付加については、transformer の位置エンコーディングと同様に単語の拍数に対応したベクトルを単語分散表現に加算する形で行った。サブワードから単語への変換機構は、1 つの単語を構成するサブワード列を入力として得られるベクトル列の内、単語開始トークンに対応する位置の出力ベクトルを単語全体の分散表現とした。また、単語からサブワードへの変換については、単語分散表現を transformer に対する長さ 1 の入力ベクトル列とし、その単語を構成するサブワード列をデコードする形で行った。

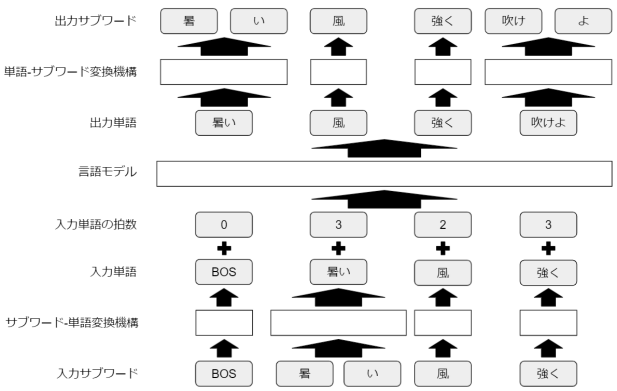

図 1 提案手法による俳句の生成イメージ

† 筑波大学 University of Tsukuba
*1 https://itoen-shinhaiku.jp/

表 1　自動評価：言語モデルのパープレキシティと自動生成俳句が 575 の条件を満たす割合

トークン化単位・手法	拍数素性	パープレキシティ	文パープレキシティ	575 率 [%]	575′ 率 [%]
サブワード	無	86.26	1.55e+21	49.3	82.7
サブワード	有	86.35	1.62e+21	48.0	81.6
単語	無	169.59	9.17e+20	47.3	81.3
単語	有	163.75	6.74e+20	55.6	86.4
階層的言語モデル	無	10.42	5.83e+20	59.5	89.3
階層的言語モデル	有	10.36	$\mathbf{5.13e+20}$	**62.1**	**90.8**

4　評価実験

本研究ではトークン単位とその処理法の違い、拍数素性の有無が俳句生成に与える影響について自動評価と人手評価の 2 つの方法で検証する。

4.1　実験条件

データは俳句例句データベース[*2]に収録されている俳句のうち、拍数が 16 以上 18 以下の約 40 万句を使用した。データの内訳は学習データ 386,586 句、開発データ 10,000 句、テストデータ 10,000 句である。俳句の単語へのトークン化及び拍数の取得には MeCab[7]、サブワードへのトークン化には SentencePiece[2] を使用した。

モデルのパラメータは分散表現次元数を 256、隠れ層次元数を 1024、言語モデル層数を 4、言語モデルヘッド数を 8、変換機構層数を 2、変換機構ヘッド数を 4 とした。トークンの語彙数はサブワード語彙数が 8,000、単語語彙数が 75,263 である。学習の際の損失関数は交差エントロピー、最適化手法は Adam[8]、バッチサイズは 128、学習率は 0.0001 とし、EarlyStopping を使用した。

4.2　自動評価

自動評価では言語モデルのパープレキシティと、生成した俳句が 575 を満たす割合の 2 つを測定する。評価するモデルはトークン単位をサブワードとしたもの、単語としたもの、変換機構によって両者を併用したものの 3 種類と、それぞれに拍数素性を適用した 3 種類の計 6 種類である。

パープレキシティの測定については、異なるトークン単位のパープレキシティは比較できないため、文のトークン数での正規化を行わない文パープレキシティも併せて測定する。

生成した俳句が 575 を満たす割合の測定は、俳句が 575 の定型を完全に満たす割合 (575 率) と 1 字の字余り・字足らずの範囲内に俳句が収まる割合 (575′ 率) の 2 つを測定する。俳句の生成は言語モデルの出力する確率密度に従って行い、各手法でそれぞれ 1 万句の俳句を生成する。

4.3　人手評価

人手評価では提示された俳句の作者が人間であるか機械であるかを回答する判別テストを行い、作者が人間であると回答された数とその割合を測定した。判別テストを 3 名の学生に対して行い、その際に提示した俳句は人間による俳句、トークン化単位を単語として拍数素性を使用した俳句、階層的言語モデルを使用し拍数素性を使用した俳句の 3 手法についてそれぞれ 30 句、計 90 句である。

4.4　結果

自動評価の結果を表 1、人手評価の結果を表 2 に示す。人手評価では提案手法の使用による有意な改善は見られなかったが、自動評価においてはすべての指標で拍数素性を使用した提案手法が最も良い結果となっている。提案手法では非階層的な言語モデルでトークン単位をサブワードとした場合と異なり、拍数素性の利用により生成モデルの性能の改善が見られる。

表 2　人手評価：判別テストで俳句が人間によるものと回答された数と割合

手法	人間	単語	階層的言語モデル
回答数	68/(30×3)	31/(30×3)	32/(30×3)
割合	0.756	0.344	0.356

5　おわりに

本研究ではサブワードに単語の拍数の情報を効果的に付与することで、俳句自動生成の性能を向上させることを目的とし、そのための手法の提案とその評価を行った。その結果、階層的言語モデルの使用によってより 575 の定型に沿った俳句が生成されることが確かめられた。

参考文献

[1] 太田 瑶子, 進藤 裕之, 松本 祐治, ”深層学習を用いた俳句の自動生成”, 情報処理学会 研究報告自然言語処理（NL）, 2018-NL-235, 1, pp. 1-8(2018).

[2] Taku Kudo and John Richardson, ”SentencePiece: A simpleand language independent subword tokenizer and detokenizer for Neural Text Processing”, EMNLP 2018, pp. 66-71(2018).

[3] 横山 想一郎, 高橋 遼, 山下 倫央, 川村 秀憲, ”深層学習を用いた言語モデルによる俳句生成におけるトークン単位選択”, 情報処理学会 研究報告知能システム（ICS）, 2020-ICS-198, 5, pp. 1-7(2020).

[4] Naoko Tosa, Hideto Obara, and Michihiko Minoh, ”Hitch haiku: An interactive supporting system for composing haiku poem”, in Stevens S. M. and Sa;damacro S. J. (eds.), Entertainment Computing - ICEC 2008, pp. 209-216(2008).

[5] Duygu Ataman, Orhan Firat, Mattia A. Di Gangi, Marcello Federico, and Alexandra Birch, ”On the importance of word boundaries in character-level neural machine translation”, Proceedings of the 3rd Workshop on Neural Generation and Translation, pp. 118-127(2019).

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, ”Attention is all you need”, NIPS 2017, pp. 5998-6008(2017).

[7] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto, ”Applying conditional random fields to japanese morphological analysis”, EMNLP 2014, pp. 230-237(2004).

[8] Diederik P. Kingma and Jimmy Ba, ”Adam: A method for stochastic optimization”, in Yoshua Bengio and Yann LeCun(eds.), ICLR(2015).

[*2] http://taka.no.coocan.jp/a5/cgi-bin/HAIKUreikuDB/ZOU.htm