単語の難易度埋め込みを用いた日本語のテキスト平易化

柳本 大輝 [1]　梶原 智之 [2]　二宮 崇 [2]
[1] 愛媛大学工学部　[2] 愛媛大学大学院理工学研究科
{yanamoto@ai., kajiwara@, ninomiya@}cs.ehime-u.ac.jp

概要

本研究では、系列変換モデルの埋め込み層を拡張し、単語埋め込みに加えて単語の難易度埋め込みを用いることで、日本語のテキスト平易化の性能を改善する。日本語のテキスト平易化のために利用可能な文の難易度付きパラレルコーパスが存在しないため、英語の先行研究のように語や文の難易度を考慮したテキスト平易化を日本語で実現することは難しい。提案手法では、2,000 単語からなる基礎語彙に注目し、入力単語が平易か否かを考慮する。やさしい日本語コーパスにおける実験の結果、提案手法は積極的に平易な表現を出力する傾向が見られた。

1　はじめに

日本における在留外国人数は年々増加傾向にあり、その数は今や 290 万人[1)] を超えている。さらに、在留外国人の国籍の多様化も進んでおり、在留外国人の国籍および出身地域の数は約 200 種類[2)] である。そのため、在留外国人をはじめとする日本語非母語話者への情報伝達の際に、各人の母語へ翻訳することは困難である。また、2010 年の調査 [1] では、日常生活に困らない言語として日本語を挙げた在留外国人は約 62%であり、英語や中国語を大きく上回ることが報告されている。そこで、緊急時の情報発信や平時の生活情報案内など、様々な場面においてやさしい日本語[3)] での情報提供が推奨されている。

やさしい日本語などの制限言語の生成は、テキスト平易化 [2] の技術による自動化が期待されている。テキスト平易化では、文の意味を保持したまま、難解な文を平易に変換する。近年は、テキスト平易化を同一言語内の機械翻訳の問題として捉え、難解文と平易文の対からなるパラレルコーパス [3] を用いて系列変換モデルを訓練するアプローチ [4–6] が主流である。この技術を用いて、所与の日本語文をやさしい日本語へ自動的に変換する日本語のテキスト平易化の研究 [7–11] が行われている。

しかし、テキスト平易化の多くの先行研究 [3–5, 10–14] は、機械翻訳などで用いられる標準的な系列変換モデルを適用し、テキスト平易化タスクの特徴である難易度を充分に考慮していない。多段階の文の難易度が付与されたパラレルコーパス [3, 5, 15] を利用可能な状況では、難易度を考慮した手法 [16–18] も提案されているが、日本語を含む英語以外の言語には、そのようなコーパスは存在しない。

日本語には文の難易度が付与されたコーパスは存在しないものの、単語の難易度が登録された辞書 [19] は複数存在する。そこで本研究では、単語の難易度を考慮した日本語のテキスト平易化手法を提案する。提案手法では、やさしい日本語コーパス [7–9] に付属の基礎語彙を使用し、Transformer [20] に基づく系列変換モデルにおいて単語埋め込みに単語難易度の情報を組み込む。やさしい日本語コーパスにおける評価実験の結果、単語難易度の考慮によってテキスト平易化の性能を改善できた。

2　関連研究

テキスト平易化によって、言語障害を持つ人々や子どものテキスト読解支援 [21–23] および言語学習効率の向上 [24] などの効果が期待できる。ただし、子どもや言語学習者の言語能力は年齢や学習期間などの影響を受け、複数の段階を持つ。そのため、対象読者を想定してテキストの難易度を制御する手法が近年盛んに研究されている。

文レベルの難易度を考慮する先行研究では、目標難易度を特殊トークンとして入力文の文頭に付加する手法 [16] や、出力文の推定難易度と目標難易度の誤差最小化の訓練を行う手法 [18] が提案されている。単語レベルの難易度を考慮する先行研究では、目標難易度の文中で出現しやすい単語の出力を促す手法 [17] や、難解な文中で出現しやすい単語を出力

1)　https://www.moj.go.jp/isa/content/001381744.pdf
2)　https://www.moj.go.jp/isa/publications/press/13_00028.html
3)　語彙や文法を制限した初学者にも理解しやすい日本語

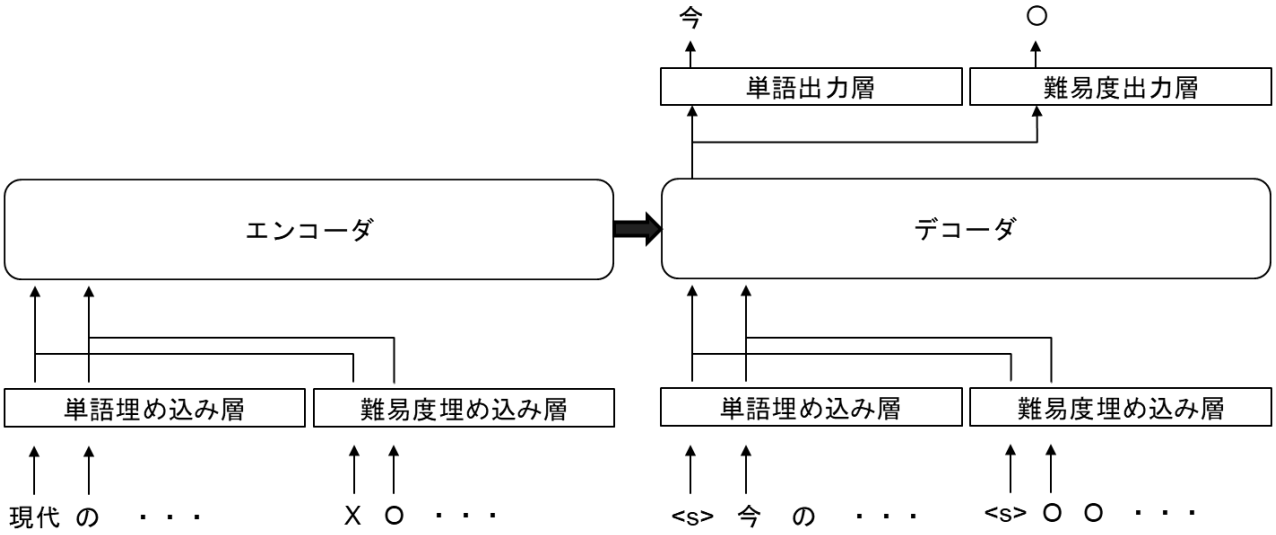

図 1 提案手法の概要

しないように強制する手法 [25] が提案されている。

しかし、これらの先行研究では、難易度を考慮するモデルを訓練するために、多段階の文の難易度が付与されたパラレルコーパス [3, 5, 15] を必要とする。このようなパラレルコーパスは英語以外の言語では使用できないため、日本語においては難易度を考慮するテキスト平易化モデルを構築できない。

3 前提知識

3.1 やさしい日本語コーパスと基礎語彙

日本語のテキスト平易化のための言語資源として、やさしい日本語コーパス [7–9] がある。これは、8.5 万文の日本語文を人手で平易化した日本語のテキスト平易化のためのパラレルコーパスである。原文は、日英対訳コーパスである田中コーパス[4]から選択されている。このうち、5 万文対[5]は日本語母語話者の大学生によって平易化され、残りの 3.5 万文対[6]はクラウドソーシングによって雇用された日本語母語話者によって平易化されたものである。日本語のテキスト平易化の先行研究 [10, 11] では、やさしい日本語コーパスを用いて Transformer [20] に基づくテキスト平易化モデルが構築されている。

やさしい日本語コーパスの作者らは、やさしい日本語として、2,000 単語からなる基礎語彙[7]を選定している。平易化の際には、記号や固有名詞は例外としつつも、基本的にはこの基礎語彙のみを用いて原文を言い換えている。なお、この基礎語彙は、UniDic の単語分割基準に従っている。

3.2 系列変換モデルと追加特徴量

ニューラル機械翻訳などの深層学習に基づく系列変換モデルの性能を改善するために、埋め込み層に追加の特徴量を組み込む手法 [26] が提案されている。この手法は、品詞タグや係り受けラベルなど、単語埋め込みにおいて明示的に考慮されていない言語的特徴を考慮するために提案されたものである。具体的には、各特徴量に対してそれぞれ特徴ベクトルを作成し、それらを単語埋め込みと連結する。

4 提案手法

本研究では、単語の難易度を考慮して日本語のテキスト平易化に取り組む。3.2 節で説明した単語埋め込みにおける追加特徴量を用いて、ある単語が 3.1 節の基礎語彙に含まれるか否かの情報を考慮する。本手法では、基礎語彙に含まれる単語の生成および入力文中の基礎語彙に含まれる単語の出力への保持を促進し、入力文中の基礎語彙に含まれない単語の出力を抑制することが期待できるため、テキスト平易化の性能改善が見込める。

提案手法の概要を図 1 に示す。提案手法では、言語学的特徴を考慮する機械翻訳 [26] から着想を得て、系列変換モデルのエンコーダおよびデコーダにおいて、通常の単語埋め込み層とは別に、新たに難易度埋め込み層を用意する。難易度埋め込み層

4) https://github.com/odashi/small_parallel_enja
5) https://www.jnlp.org/GengoHouse/snow/t15
6) https://www.jnlp.org/GengoHouse/snow/t23
7) https://www.jnlp.org/GengoHouse/list/語彙

表 1 実験結果

モデル	次元	BLEU	SARI	add	keep	del	基礎語彙の割合
ベースライン	512	75.55	63.88	16.78	**88.23**	86.63	77.79
提案手法	500 + 12	**75.59**	**64.00**	**17.10**	88.19	**86.70**	**78.14**

は、ある単語が基礎語彙に含まれるか否かで 2 種類の埋め込みを生成する役割を担う。入力された単語およびその難易度について、それぞれ異なる埋め込みを作成し、それらを連結して系列変換モデルに与える。つまり、エンコーダは単語の系列 $X = (x_1, ..., x_m)$ と難易度の系列 $G = (g_1, ..., g_m)$ を読み込み、エンコーダへの入力埋め込みは

$$E_{\text{enc}} = Wx_i \| Fg_i$$

となり、デコーダは推定単語の系列 $\hat{y} = (\hat{y}_1, ..., \hat{y}_n)$ と推定難易度の系列 $\hat{g} = (\hat{g}_1, ..., \hat{g}_n)$ を読み込み、デコーダへの入力埋め込みは

$$E_{\text{dec}} = W\hat{y}_i \| F\hat{g}_i$$

となる。

ここで、$\|$ はベクトルの連結、E は d_{model} 次元のベクトル、$W \in \mathbb{R}^{V_{\text{word}} \times d_{\text{word}}}$ および $F \in \mathbb{R}^{V_{\text{grade}} \times d_{\text{grade}}}$ は単語埋め込み行列および難易度埋め込み行列であり、V は語彙サイズ、d は次元数である。

また、デコーダは、Softmax 層の直前に単語出力用および難易度出力用の 2 つの線形変換層を持つ。これらの単語出力層および難易度出力層の次元はそれぞれの埋め込み次元数と一致している。デコーダの出力は、単語埋め込み次元とそれ以降の次元の 2 つに分かれ、それぞれの出力層に入力される。

単語難易度は、やさしい日本語コーパスにおける 2,000 単語の基礎語彙に基づくが、頻出する記号（句読点、感嘆符、疑問符、括弧）は充分に平易だと考え、基礎語彙に含めた。図 1 の例では、単語 x_i が基礎語彙に含まれる場合は g_i として「O」を、単語 x_i が基礎語彙に含まれない場合は g_i として「X」を、それぞれ単語難易度として入力している。

5 評価実験

提案手法の有効性を検証するために、やさしい日本語コーパスおよび基礎語彙を用いて、日本語のテキスト平易化の評価実験を行う。

5.1 実験設定

データ 本実験では、日本語のテキスト平易化のためのパラレルコーパスであるやさしい日本語コーパス [7–9] を使用した。難解文に対して 7 種類の平易文が付与されたマルチリファレンスの 100 文対を評価データ、その他のシングルリファレンス部分から無作為抽出された 2,000 文対を検証データ、残りの 82,300 文対を訓練データとして使用した。

前処理として、難解文および平易文には Mecab[8] [27] による単語分割を行った。基礎語彙と単語分割基準を合わせるために、Mecab の辞書には UniDic を使用した。

モデル テキスト平易化モデルには Transformer [20] を使用し、Sockeye[9] [28] を用いて実装した。モデルの構成は、層数を 6 層、自己注意機構のヘッド数を 8、埋め込み次元数を 512、全結合層の次元数を 2,048 とした。提案手法では、500 次元を単語埋め込み、12 次元を難易度埋め込みに割り当て、それらを結合して 512 次元の埋め込みをモデルに入力した。また、エンコーダおよびデコーダの単語埋め込み層と Softmax 層直前の線形変換の 3 つのパラメタを共有した。

訓練時には、バッチサイズを 4,096 トークンとし、最適化手法として Adam [29]（$\beta_1 = 0.9$、$\beta_2 = 0.98$）を使用した。初期学習率は 0.001 に設定し、学習率スケジューリングとして plateau-reduce を使用した。また、200 ステップごとに検証用データにおいて SARI [30] を評価し、10 回連続で改善が見られない場合に訓練を終了する early-stopping を採用した。

評価 評価には、テキスト平易化タスクで一般的に用いられている BLEU [31] および SARI [30] の自動評価指標を使用した。BLEU は、出力文と参照文を比較し、単語 n-gram の一致度を評価する。SARI は、入力文・出力文・参照文を比較し、単語 n-gram の追加・保持・削除の 3 つの編集操作の適切さを評価する。BLEU および SARI の計算には、自動評価パッケージである EASSE[10] [32] を使用した。なお、詳細な分析のために、SARI における各編集操作の F 値も確認する。

8） https://taku910.github.io/mecab/
9） https://github.com/awslabs/sockeye
10） https://github.com/feralvam/easse

表 2　出力例（太字は基礎語彙に含まれる単語）

入力文	**私たちのところに**、不意**の**来客**があった**。
ベースライン	**私たちのところに**、思わ**ない客があった**。
提案手法	**私たちのところに**、**偶然に客があった**。
入力文	**彼は来る日も来る日も**熱心**に勉強した**。
ベースライン	**彼は来る日も来る日も**頑張っ**て勉強した**。
提案手法	**彼は来る日も来る日も集中して勉強した**。

また、平易性に関する自動評価として、出力文中の基礎語彙の割合を算出する。この割合が高くなると、テキスト平易化モデルは基礎語彙、つまり、やさしい日本語をより多く出力できたと考えられる。参照文における基礎語彙の割合は 73.13% であった。

5.2　実験結果

表 1 に実験結果を示す。シード値を変更しつつ 5 回の実験を行い、その平均値によってテキスト平易化モデルの性能を評価した。提案手法は、ベースラインと比較して BLEU が 0.04 ポイント、SARI が 0.12 ポイント、それぞれ向上した。語句の追加・保持・削除の編集操作に関しては、特に追加の操作において、ベースラインと比較して 0.32 ポイントと大きな改善が見られた。基礎語彙の割合についても、提案手法はベースラインを上回った。追加操作（add）が向上していることと併せて、提案手法は基礎語彙に含まれる単語を積極的に用いて書き換えを行っていると考えられる。

5.3　定性評価

各モデルによる生成文の例を表 2 に示す。上段の例では、入力文中の「不意」と「来客」の 2 単語が基礎語彙に含まれていない難解語であり、ベースラインおよび提案手法のどちらもこの 2 単語を書き換えた。「来客」については、両モデルとも基礎語彙に含まれている「客」に書き換えることができた。一方で、「不意」については、ベースラインは「思わ」「ない」の 2 単語、提案手法は「偶然」に書き換えた。ベースラインの出力した「思わ」は基礎語彙に含まれていないが、提案手法は基礎語彙に含まれる「偶然」を出力できた。

下段の例では、入力文中の「熱心」が基礎語彙に含まれていない難解語であり、両モデルともこの単語を含む「熱心」「に」の 2 単語に対して書き換えを行った。上段の例と同様に、提案手法の出力である「集中」は基礎語彙に含まれており、ベースラインの出力である「頑張っ」は含まれていない。これらの出力例で確認できるように、提案手法は単語の難易度を考慮しないベースラインよりも、入力文中の難解な表現を基礎語彙を用いる平易な表現に積極的に書き換える傾向がある。

6　おわりに

本研究では、やさしい日本語コーパスにおける基礎語彙を用いて、単語の難易度を考慮したテキスト平易化手法を提案した。提案手法では、系列変換モデルのエンコーダおよびデコーダの埋め込み層を拡張し、単語埋め込みおよび難易度埋め込みの組み合わせとして表現した。やさしい日本語コーパスにおける評価実験の結果、提案手法は積極的に平易な表現を出力し、全ての自動評価指標において比較手法を上回る性能を達成した。

今後の課題として、他の単語難易度辞書を用いて、提案手法によるテキスト平易化モデルの性能向上に取り組みたい。例えば、梶原ら [19] の単語難易度辞書[11]では、各単語に初級・中級・上級の 3 段階の難易度を付与しているため、より詳細な難易度の情報を得られる可能性がある。また、英語の単語難易度辞書[12] [33] を用いるなど、提案手法の他の言語への適用にも取り組みたい。

謝辞

本研究は国立研究開発法人情報通信研究機構の委託研究（課題番号：225）の助成を受けたものです。

参考文献

[1] 岩田一成. 言語サービスにおける英語志向 :「生活のための日本語: 全国調査」結果と広島の事例から. 社会言語科学, Vol. 13, No. 1, pp. 81–94, 2010.

[2] Fernando Alva-Manchego, Carolina Scarton, and Lucia Specia. Data-Driven Sentence Simplification: Survey and Benchmark. **Computational Linguistics**, Vol. 46, No. 1, pp. 135–187, 2020.

11）https://github.com/Nishihara-Daiki/lsj
12）https://github.com/mounicam/lexical_simplification

[3] Chao Jiang, Mounica Maddela, Wuwei Lan, Yang Zhong, and Wei Xu. Neural CRF Model for Sentence Alignment in Text Simplification. In **Proc. of ACL**, pp. 7943–7960, 2020.

[4] Sergiu Nisioi, Sanja Štajner, Simone Paolo Ponzetto, and Liviu P. Dinu. Exploring Neural Text Simplification Models. In **Proc. of ACL**, pp. 85–91, 2017.

[5] Xingxing Zhang and Mirella Lapata. Sentence Simplification with Deep Reinforcement Learning. In **Proc. of EMNLP**, pp. 584–594, 2017.

[6] Sanqiang Zhao, Rui Meng, Daqing He, Andi Saptono, and Bambang Parmanto. Integrating Transformer and Paraphrase Rules for Sentence Simplification. In **Proc. of EMNLP**, pp. 3164–3173, 2018.

[7] 山本和英, 丸山拓海, 角張竜晴, 稲岡夢人, 小川耀一朗, 勝田哲弘, 高橋寛治. やさしい日本語対訳コーパスの構築. 言語処理学会第 23 回年次大会, pp. 763–766, 2017.

[8] Takumi Maruyama and Kazuhide Yamamoto. Simplified Corpus with Core Vocabulary. In **Proc. of LREC**, pp. 1153–1160, 2018.

[9] Akihiro Katsuta and Kazuhide Yamamoto. Crowdsourced Corpus of Sentence Simplification with Core Vocabulary. In **Proc. of LREC**, pp. 461–466.

[10] 中町礼文, 梶原智之. 事前訓練済み系列変換モデルに基づくやさしい日本語への平易化. 情報処理学会第 83 回全国大会, pp. 607–608, 2021.

[11] 畠垣光希, 梶原智之, 二宮崇. やさしい日本語へのテキスト平易化のための訓練データの精選. 第 21 回情報科学技術フォーラム, pp. 293–300, 2022.

[12] Lucia Specia. Translating from Complex to Simplified Sentences. In **Proc. of PROPOR**, pp. 30–39, 2010.

[13] Sanja Štajner, Hannah Béchara, and Horacio Saggion. A Deeper Exploration of the Standard PB-SMT Approach to Text Simplification and its Evaluation. In **Proc. of ACL**, pp. 823–828, 2015.

[14] Tomoyuki Kajiwara and Mamoru Komachi. Building a Monolingual Parallel Corpus for Text Simplification Using Sentence Similarity Based on Alignment between Word Embeddings. In **Proc. of COLING**, pp. 1147–1158, 2016.

[15] Wei Xu, Chris Callison-Burch, and Courtney Napoles. Problems in Current Text Simplification Research: New Data Can Help. **TACL**, Vol. 3, pp. 283–297, 2015.

[16] Carolina Scarton and Lucia Specia. Learning Simplifications for Specific Target Audiences. In **Proc. of ACL**, pp. 712–718, 2018.

[17] Daiki Nishihara, Tomoyuki Kajiwara, and Yuki Arase. Controllable Text Simplification with Lexical Constraint Loss. In **Proc. of ACL-SRW**, pp. 260–266, 2019.

[18] Daiki Yanamoto, Tomoki Ikawa, Tomoyuki Kajiwara, Takashi Ninomiya, Satoru Uchida, and Yuki Arase. Controllable Text Simplification with Deep Reinforcement Learning. In **Proc. of AACL**, pp. 398–404, 2022.

[19] 梶原智之, 西原大貴, 小平知範, 小町守. 日本語の語彙平易化のための言語資源の整備. 自然言語処理, Vol. 27, No. 4, pp. 801–824, 2020.

[20] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In **Proc. of NIPS**, pp. 5998–6008, 2017.

[21] John Carroll, Guido Minnen, Yvonne Canning, Siobhan Devlin, and John Tait. Practical Simplification of English Newspaper Text to Assist Aphasic Readers. In **Proceedings of the Workshop on Integrating Artificial Intelligence and Assistive Technology**, pp. 7–10, 1998.

[22] Jan De Belder and Marie-Francine Moens. Text Simplification for Children. In **Proceedings of the SIGIR 2010 Workshop on Accessible Search Systems**, pp. 19–26, 2010.

[23] Tomoyuki Kajiwara, Hiroshi Matsumoto, and Kazuhide Yamamoto. Selecting Proper Lexical Paraphrase for Children. In **Proc. of ROCLING**, pp. 59–73, 2013.

[24] Sarah E Petersen and Mari Ostendorf. Text Simplification for Language Learners: A Corpus Analysis. In **Proceedings of the Workshop on Speech and Language Technology in Education**, pp. 69–72, 2007.

[25] Tomoyuki Kajiwara. Negative Lexically Constrained Decoding for Paraphrase Generation. In **Proc. of ACL**, pp. 6047–6052, 2019.

[26] Rico Sennrich and Barry Haddow. Linguistic Input Features Improve Neural Machine Translation. In **Proc. of WMT**, pp. 83–91, 2016.

[27] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying Conditional Random Fields to Japanese Morphological Analysis. In **Proc. of EMNLP**, pp. 230–237, 2004.

[28] Felix Hieber, Michael Denkowski, Tobias Domhan, Barbara Darques Barros, Celina Dong Ye, Xing Niu, Cuong Hoang, Ke Tran, Benjamin Hsu, Maria Nadejde, Surafel Lakew, Prashant Mathur, Anna Currey, and Marcello Federico. Sockeye 3: Fast Neural Machine Translation with PyTorch. **arXiv:2207.05851**, 2022.

[29] Diederik P. Kingma and Jimmy Lei Ba. Adam: A Method for Stochastic Optimization. In **Proc. of ICLR**, 2015.

[30] Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. Optimizing Statistical Machine Translation for Text Simplification. **Transactions of the Association for Computational Linguistics**, Vol. 4, pp. 401–415, 2016.

[31] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a Method for Automatic Evaluation of Machine Translation. In **Proc. of ACL**, pp. 311–318, 2002.

[32] Fernando Alva-Manchego, Louis Martin, Carolina Scarton, and Lucia Specia. EASSE: Easier Automatic Sentence Simplification Evaluation. In **Proc. of EMNLP**, pp. 49–54, 2019.

[33] Mounica Maddela and Wei Xu. A Word-Complexity Lexicon and A Neural Readability Ranking Model for Lexical Simplification. In **Proc. of EMNLP**, pp. 3749–3760, 2018.