seq2seq モデルを用いた日本語テキストからの読み仮名・韻律記号列推定の検討

Phonetic and Prosodic Information Estimation From Japanese Text Using Seq2Seq

懸川直人，原直，阿部匡伸，井島勇祐

Naoto Kakegawa[1], Sunao Hara[1], Masanobu Abe[1], Yûsuke Ijima[2]

[1] 岡山大学 阿部研究室，[2]NTT

[1]Abe Laboratory, Okayama University, [2]NTT Coporation

概要 英語において，End-to-End テキスト音声合成（TTS: text-to-speech）は高品質な音声が生成できることが確認されている．これに対して，日本語 End-to-End TTS の実現には読み仮名，韻律情報の推定が必要となる．本稿では，ニューラル機械翻訳を用いて日本語テキストから読み仮名と韻律情報を高精度に推定できることを確認する．

1 はじめに

近年，TTS でも，End-to-End アプローチが盛んに研究されており [1, 2, 3, 4]，英語において高品質な音声が生成できることが確認されている．これに対して，日本語 End-to-End TTS を行う際に問題となるのが日本語テキストの扱いである．英語の場合，単語はスペースで区切られているが，日本語の場合，文章は分かち書きされていないため単語の同定から行う必要がある．また，英語は大文字・小文字合わせても 52 種類だが，日本語は常用漢字だけで 2,136 種類と，文字の種類が多い．さらに，「辛い（からい，つらい）」のように複数の読み方を持つ漢字や，「あめ（飴，雨）」のように同じ表記でアクセントが異なる単語が数多く存在するなど，読みばかりでなく，韻律情報もテキストから推定しなければならない．End-to-End TTS では，テキスト入力から音響特徴量を出力するモデルを学習するため，テキストから読み・韻律情報を推定することは大きな課題である．これらの理由から，日本語 End-to-End TTS の研究では，日本語テキストに対応する読みや韻律をモデルに入力することが多い [5, 6]．

本稿では，ニューラル機械翻訳 (NMT: Neural Machine Translation) を用いた日本語テキストからの読み仮名・韻律記号列推定方式を提案する．NMT のモデルとして，Attention 付きの RNN を用いた Sequence-to-Sequence モデル [9] を用いる．

この推定方式により得られた読み仮名・韻律記号列を基に音声を生成することで，日本語 End-to-End TTS の実現を目指す．

本稿は以下の通りの構成である．第 2 章では提案方式，学習データの形式について述べる．第 3 章では評価実験について述べる．最後に，第 4 章では結論と今後の課題を述べる．

2 提案方式

2.1 Sequence-to-Sequence モデル

図 1 に提案モデルの概略図を示す．Sequence-to-Sequence(seq2seq) モデルは系列を入力として系列を出力するモデルである．seq2seq モデルの構成は Encoder と Decoder の二要素に大別される．

提案方式の Encoder は Embedding layer とそれに続く Bidirectional Long Short-Term Memory (BiLSTM) から構成される．Encoder の目的は，入力系列の情報を LSTM の最終状態として縮約することである．Encoder での処理の流れを以下に示す．

1. Embedding layer が，入力系列の最小単位（トークン）を受け取り，実数値のベクトルによる分散表現を出力する
2. LSTM が，Embedding layer が出力したベクトルを受け取り，隠れ状態とセルの状態を次のステップの LSTM に渡す
3. 1. と 2. を入力系列の最後まで繰り返す

Decoder の目的は，入力系列の情報が縮約された Encoder の LSTM の最終状態から，出力系列を推定することである．Decoder の構造・処理は Encoder とほぼ同じだが，以下の点が異なる．

- LSTM のセルの状態 c と隠れ状態 h は, Encoder の LSTM の最終ステップでの状態 c，h で初期化される
- 各ステップの出力トークンは，各ステップにおける LSTM の隠れ状態 h を Softmax 関数に通すことで得られる

Sequence-to-Sequence モデルの目的は，入力系列を $(x_1, \cdots, x_T)$，出力系列を $(y_1, \cdots, y_{T'})$ とした時，条件つき確率 $p(y_1, \cdots, y_{T'} \mid x_1, \cdots, x_T)$ を求めることである．

2.2 学習データ

学習データとして，入力系列と出力系列の対を大量に用意する必要がある．対の例を図 2 に示す．

提案モデルの入力は漢字仮名混じりの日本語テキストである．入力系列の最小単位（トークン）は文字である．

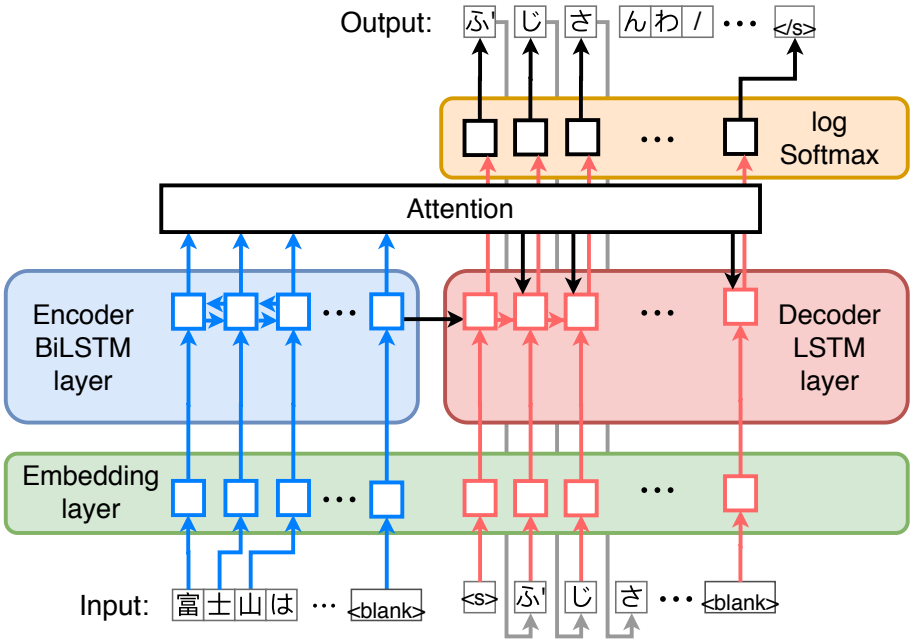

図 1: The proposed model

入力データ | そ | の | 日 | に | 出 | 場 | を | 決 | 断 | し | た |

出力データ | ソ | ノ | ヒ' | ニ | @ | シュ^ | ツ | ジョ | ー | オ | / | ケ | ツ | ダ | ン | シ^ | タ | . |

図 2: Example pair of input and output sequence

提案モデルの出力は読み仮名と韻律記号で構成される系列である．韻律記号は，電子情報技術産業協会による日本語テキスト音声合成用記号 (JEITA IT-4006) [10] を参考に，表 1 のような韻律記号を用いる．出力系列では，読み仮名はモーラ単位で扱い，アクセント核となるモーラにはアクセント核記号を付与し，読み仮名とアクセント核記号を合わせて 1 つのトークンとする．それ以外の韻律記号は，記号単体で 1 つのトークンとして扱う．

「母音の無声化」が発生している読み仮名は，無声化記号 ^ を付与し，読み仮名と無声化記号を合わせて 1 つのトークンとして扱う．ここで，「母音の無声化」とは，言葉を話す時に，普通有声である母音が他の音との並びなどによって聞こえにくくなる現象である．

特殊なトークンとして，<unk>，<blank>，<s> そして</s>を用いる．<unk>は，モデルの語彙に含まれないトークンを表現するためのトークンである．<blank>は，入出力系列の長さを固定する目的で，系列の長さが足りない時に系列を充填するためのトークンである．<s>は系列の開始位置を示すためのトークン，</s>は系列の終了位置を示すためのトークンである．

これらのうち，<s>と</s>は，出力系列だけで利用される．

表 1: Prosodic symbols

Accent boundary		Accent nuclear	
With long pause	.	Accented	’
With short pause	@		
Without pause	/		

3 客観評価実験

3.1 データセット

データセットとして，漢字仮名混じりの日本語テキストと，読み仮名と韻律記号からなる系列のペアデータを用いる．データセットを作成するために，インターネット上のニュース記事から大量の日本語テキストをクロールした．クロールしたデータを句点「。」で区切り，結果として 5,146,894 文の日本語テキストを得た．この日本語テキストに対し，形態素解析 [11]，アクセント推定 [12] を行い，読み仮名と韻律記号の系列データを機械的に作成した．ここで用いた形態素解析システムとアクセント推定システムの精度は 91% であるが，評価時には，機械的に推定した読み仮名と韻律記号の系列データを正解として扱い，モデルの性能評価を行う．データセットの詳細を表 2 に示す．

3.2 実験条件

seq2seq モデルの訓練時の条件を表 3 に示す．モデルの構築には OpenNMT-py [13] を使用し，ほとんどのパラメータはデフォルトのものを利用した．

表 2: Details of generated dataset

		Kinds	Total
Input tokens	Kana(Hiragana)	83	83,868,304
	Kana(Katakana)	84	23,349,595
	Kanji	4,922	95,234,262
	Numbers	20	10,960,332
	English alphabets	104	2,861,472
	Symbols	223	18,850,611
Output tokens	Syllables	138	248,225,352
	Accented	151	41,196,975
	Devocalized	43	11,506,943
	Without pause	1	33,410,709
	With short pause	1	13,350,395
	With long pause	1	15,735,781

表 3: Training condition for Transformer

Datasets	
Validation data	5,000 pairs
Test data	5,000 pairs
NMT	
Maximum numbers of tokens in a sentence	300
Encoder layers	2
Decoder layers	2
Dropout	0.3
Optimizer	SGD
Starting learning rate	1.0
Epoch	20
Batch size	512
Batch type	Sentence
Loss function	Cross entropy

3.3 ベースライン

Encoder 側の BiLSTM を LSTM に変更したモデルをベースラインとし，提案モデルと比較する．学習データとテストデータは，提案モデルと同じである．

3.4 評価指標

評価指標として，翻訳タスクの評価尺度として用いられる単語正解精度を拡張して使用する．単語正解精度は以下のように計算される．

$$\text{単語正解精度} = 100 \times (N - S - D - I)/N \quad (1)$$

ここで，N は入力文中の総単語数，S は置換単語数，D は脱落単語数，I は挿入単語数である．本稿では，モデルの推定系列と正解系列のトークン同士の比較を行い，トークン単位での正解精度を算出する．また，正解精度の算出を行う際には，推定系列と正解系列で動的計画法（DP: Dynamic Programming）によるマッチングをおこなった上で各系列のトークンを比較する．

表 4: Correct estimation ratio in token based evaluation

	BiLSTM Enc. (proposed)	LSTM Enc.
P-accuracy	84.3%	80.2%
PP-accuracy	97.1%	96.8%
B-accuracy	97.4%	96.4%
N-accuracy	99.0%	98.6%

3.5 実験結果

3.5.1 トークン単位での評価

表 4 にトークン単位での性能評価の結果を示す．ここで, P-accuracy は読み仮名の正解精度, PP-accuracy は読み仮名と韻律情報の正解精度，B-accuracy は読み仮名が正しく推定できた文におけるアクセント句境界の正解精度，N-accuracy は読み仮名，アクセント句境界が正しく推定できた文におけるアクセント核の正解精度である．結果より，提案手法が全ての評価尺度においてベースラインを上回っており，読み仮名，アクセント句境界，アクセント核それぞれについても高い精度で推定が行えていることが分かる．

3.5.2 文単位での評価

トークン単位で評価する読み仮名・韻律正解精度では，誤りが 1 文中に複数含まれる可能性がある．より厳格な尺度として，文単位で評価する評価尺度を用いる．表 5 に結果を示す．ここで，S-P-accuracy は読み仮名が全て正しく推定できた文の割合, S-PB-accuracy は読み仮名・アクセント句境界が全て正しく推定できた文の割合，そして S-PBN-accuracy は読み仮名・韻律が全て正しく推定できた文の割合である．提案手法とベースラインで共通して，読み仮名だけを評価する場合よりも，アクセント句境界を含めて評価する場合の方が精度が大きく低下しており，アクセント句境界の推定がタスクとして困難であることが分かる．提案手法とベースラインの比較では，全ての評価尺度において提案手法が上回っており，読み仮名・韻律推定タスクにおける提案手法の有効性が確認できた．

3.5.3 推定文の分析

複数の読み方を持つ漢字に対して読み仮名を高精度に推定できることを確認する．複数の読み方を持つ漢字の推定結果を表 6 に示す．ここで，“OK” と “NG” はそれぞれ正しく読み仮名を推定できた例，失敗した例を示している．「行」の場合，「い」，「おこな」，「ぎょう」などの読み方があるが，提案モデルは正しい読み仮

表 5: Correct estimation ratio in token based evaluation

	BiLSTM Enc. (proposed)	LSTM Enc.
S-P-accuracy	87.7%	84.1%
S-PB-accuracy	70.6%	61.7%
S-PBN-accuracy	67.3%	58.0%

表 6: Estimation results for Kanji characters with multiple pronunciations

		Estimation
OK	学校へ行った	ガッコーエイッタ
OK	運動会を行った	ウンドーカイオオコナッタ
OK	長い行列だった	ナガイギョーレツダッタ
NG	十分かかる	ジューブンカカル
OK	もう十分だ	モージューブンダ

名を推定できている．一方，「十分」の場合「じゅっぷん」と読むべき文において「じゅうぶん」と誤って推定していた．この原因は，「十分」を含む文がデータセットに足りなかったためと考えられる．

表 7: Estimation results for unknown words

	Unknown word	Reference	Estimation
OK	爬行	ハコー	ハコー
OK	傲岸	ゴーガン	ゴーガン
OK	活眼	カツガン	カツガン
NG	嫌厭	ケンエン	ケンショー

自然言語処理において，テキストを単語単位や形態素単位で扱う場合未知語への対処が問題となることが多い．一方，提案モデルでは入力系列を文字単位で扱っており，データセットに出現する文字は全て語彙として持っている．そのため，データセットに存在しない単語に対しても，単語を構成する漢字を一文字ずつ扱うことで読み仮名を有効に推定できると考えられる．表 7 にデータセットに存在しない単語の読み仮名の結果を示す．ただし，ここでの “OK” と “NG” は表 6 と同じ意味で用いている．推定結果から，多くの未知語の例において正しく推定できており，テキストを一文字単位で扱う提案モデルが未知の単語に対して有効であることが確認できた．

4 まとめ

本稿では，Attention 付きの RNN を用いた seq2seq モデルを用いた日本語テキストからの読み仮名・韻律記号列推定方式を提案した．提案方式による推定精度を評価するために客観評価を行い，読み仮名，アクセント句境界，アクセント核のそれぞれについて高い精度で推定できることを確認した．

今後，この推定方式により得られた読み仮名・韻律記号列を基に音声を生成することで，日本語 End-to-End TTS の実現を目指していく．また，このモデルで推定された読み仮名・韻律記号をもとに合成した音声の品質を確認することが今後の課題として挙げられる．

参考文献

[1] Y. Wang *et al.*, in *Proceedings of Interspeech*, 2017, pp. 4006–4010.

[2] J. Shen *et al.*, in *Proceedings of ICASSP*, 2018, pp. 4779–4783.

[3] W. Ping *et al.*, in *Proceedings of ICLR*, 2018.

[4] J. Sotelo *et al.*, in *Proceedings of ICLR Workshop*, 2017.

[5] 栗原清 *et al.*, **研究報告音声言語情報処理** *(SLP)*, vol. 2018, no. 19, pp. 1–6, 2018.

[6] Y. Yasuda *et al.*, in *Proceedings of ICASSP*. IEEE, 2019, pp. 6905–6909.

[7] “Open JTalk,” http://open-jtalk.sourceforge.net/.

[8] 工藤拓 *et al.*, **情報処理学会研究報告自然言語処理** *(NL)*, vol. 2004, no. 47 (2004-NL-161), pp. 89–96, 2004.

[9] T. Luong, H. Pham, C. D. Manning, in *Proceedings of EMNLP*, Lisbon, Portugal, Sept. 2015, pp. 1412–1421, Association for Computational Linguistics.

[10] 音声入出力方式標準化専門委員会, “JEITA IT-4006 日本語テキスト音声合成用記号,” 電子情報技術産業協会, 2010.

[11] T. Fuchi, S. Takagi, in *Proceedings of ICCL*, 1998, pp. 409–413.

[12] K. Matsuoka *et al.*, in *Proceedings of IVTTA*, Sep. 1996, pp. 33–36.

[13] G. Klein *et al.*, in *Proceedings of ACL*, July 2017, pp. 67–72.