Vol.2022-NL-254 No.8
Vol.2022-SLP-144 No.8
2022/11/29

内容語保存機構を備えた変分自己符号化器に基づくテキスト発話スタイル変換

吉岡 大貴[1,a)]　安田 裕介[1]　松永 悟行[2]　大谷 大和[2]　戸田 智基[1]

概要：テキスト音声合成（TTS）技術は発展し，通常の読み上げでは人間に近い自然性を達成している．次の課題として，感情やキャラクタ性などの「発話スタイル」を付与した音声の合成が盛んに研究されている．本稿では，「音声のスタイルだけでなく読み上げテキストのスタイルも制御可能な TTS」の実現に向けたテキストの発話スタイル変換を行う．ノンパラレルコーパスを用いた手法によるテキストスタイル変換においてコンテンツ保存性が不十分であることに着目し，条件付き変分自己符号化器に対して内容語保存機構を組み合わせることを提案する．また，提案手法に位置埋め込みやサイクル学習を導入することで，さらなる性能改善を試みる．日本語テキストのスタイル変換タスクを対象とした実験的評価の結果から，本手法の有効性を示す．

キーワード：テキストスタイル変換，自然言語処理，テキスト音声合成，発話スタイル

1. はじめに

人が社会生活を送る上で，音声を用いたコミュニケーションは必要不可欠である．近年では，音声アシスタントやスマートスピーカなどの普及により，「人同士」だけでなく「人と計算機」，「計算機を介した人同士」が音声コミュニケーションを行う機会が増加している．その中で，テキストから適切な音声を生成する技術であるテキスト音声合成（Text-to-Speech: TTS）の研究が進んでいる．現在，音声案内や文章読み上げアプリケーションなどで用いられるような通常の読み上げに関しては，人間の音声と比較しても遜色ない自然性を持つ音声合成が可能になっている [1]．次の課題として，感情やキャラクタ性などの「発話スタイル」を付与した音声の合成が盛んに研究されている [2]．一方で，特にキャラクタ性のようなスタイルは，言語情報，つまり発話するテキストそのものにも大きく影響すると考えられる．実際，複数の研究 [3], [4] で，テキスト情報のみから，その発話者（または筆者）のキャラクタ性や個人的特徴を特定できることが示されている．よって，より適切なスタイル制御のために，音声に付与したいスタイルに合わせてテキストのスタイルも制御可能な技術が求められる．

[1] 名古屋大学
NU, Furo, Chikusa, Aichi 464–8601, Japan
[2] 株式会社エーアイ
AI Inc., KDX Kasuga Building 10F, 1-15-15 Nishikata, Bunkyo Ward, Tokyo, 113-0024, Japan
[a)] yoshioka.daiki@g.sp.m.is.nagoya-u.ac.jp

テキストのスタイル制御について，意味を保持したままスタイルだけを別のスタイルへと変換する，「テキストスタイル変換」というタスクが注目されている．テキストスタイル変換において，ルールベースの手法 [5] やパラレルコーパスを用いた深層学習手法 [6], [7] を用いれば，テキストに所望のスタイルを付与することが可能である．しかし，大量の変換ルールを作成したり，大規模なパラレルコーパスを構築したりするには，相応の人手作業が必要である．これは時間的にも労力的にも高コストであり，現実的ではない．対して，「対訳テキストは存在しないが特定のスタイルを持つテキスト」は，様々な媒体から比較的容易に入手することが可能である．これらのデータを利用して構築することができるノンパラレルコーパスは，人手作業を必要とするパラレルコーパスよりも低コストで作成可能である．必要な時間や労力を抑えつつテキストスタイル変換を実現するために，ノンパラレルコーパスを用いて学習可能な手法が研究されている．

ノンパラレルコーパスを用いるテキストスタイル変換手法としては，自己符号化器（Auto-Encoder：AE）や変分自己符号化器（Variational AE：以下，VAE）[8] を利用したモデルが多く研究されている [9], [10]．VAE を用いたモデルでは，テキストの意味（コンテンツ）とスタイルを分けて潜在変数に埋め込み，再構成を学習する．そして，コンテンツを埋め込んだ潜在変数（以下，コンテンツ特徴量）を変えることなく，スタイルを埋め込んだ潜在変数（以下，

スタイル特徴量）を操作することで，パラレルコーパス無しでもスタイル変換を実現可能である．また，Generative Adversarial Network（GAN）[11] で提案された敵対的学習を用いることで性能を向上させている研究もある [12].

現状のテキストスタイル変換研究は，多くがスタイルを効果的に変換する方法を模索している．対して，コンテンツ保存については，最新の大規模事前学習済みモデルを用いた手法 [13] でさえ十分な性能を発揮しているとは言えない．そのため，テキストスタイル変換において，コンテンツを十分に保存するためのアプローチについて研究の余地が大きいと考えられる．本稿では，Conditional VAE（CVAE）[14] に対して内容語保存機構やサイクル学習を組み合わせることで，テキストスタイル変換の性能改善を試みる．日本語テキストのスタイル変換タスクを対象とした実験的評価の結果から，本手法の有効性を示す.

2. CVAE によるテキストスタイル変換

テキストスタイル変換のシンプルな実装として，確率モデルの一種である CVAE を用いることができる．CVAE は VAE を基とした手法で，周辺分布 $p(\hat{\boldsymbol{x}}|y)$ を以下の変分下限 L の最大化によって最適化する.

$$L = \mathbb{E}[\log p(\hat{\boldsymbol{x}}|\boldsymbol{z}, y)] - \mathrm{KL}[q(\boldsymbol{z}|\boldsymbol{x})|p(\boldsymbol{z})] \tag{1}$$

ここで，近似事後分布 $q(\boldsymbol{z}|\boldsymbol{x})$ は入力テキスト $\boldsymbol{x}$ をコンテンツ特徴量 $\boldsymbol{z}$ へ符号化するエンコーダとして，また出力確率 $p(\hat{\boldsymbol{x}}|\boldsymbol{z}, y)$ は出力テキスト $\hat{\boldsymbol{x}}$ をコンテンツ特徴量 $\boldsymbol{z}$ とスタイルラベル y から復号化するデコーダとして，それぞれ実装可能である．CVAE を用いたテキストスタイル変換では，入力テキストから得たコンテンツ特徴量と，クラスラベルから得たスタイル特徴量を結合してデコーダを条件付けする．学習時には再構成のみを学習するため，スタイルについてパラレルなテキスト，つまり対訳データを必要としないが，一方でラベルは既知である必要がある.

具体的な実装について，エンコーダには双方向 LSTM を用いる．エンコーダに入力単語列 $\boldsymbol{x} = (\boldsymbol{x}_1, ..., \boldsymbol{x}_L)$ の埋め込み表現 $\boldsymbol{e}^{(x)}$ を入力し，最終ステップの隠れ層から平均と分散を表すベクトル $\boldsymbol{\mu}$，$\boldsymbol{\sigma}$ を出力する．これを用いてサンプリングを行い，コンテンツ特徴量 $\boldsymbol{z}$ を得る.

$$\boldsymbol{e}_l^{(x)} = \mathrm{Embed}(\boldsymbol{x}_l) \tag{2}$$

$$\overrightarrow{\boldsymbol{h}_l^{(E)}} = \mathrm{LSTM}(\overrightarrow{\boldsymbol{h}_{l-1}^{(E)}}, \boldsymbol{e}_l^{(x)}) \tag{3}$$

$$\overleftarrow{\boldsymbol{h}_l^{(E)}} = \mathrm{LSTM}(\overleftarrow{\boldsymbol{h}_{l+1}^{(E)}}, \boldsymbol{e}_l^{(x)}) \tag{4}$$

$$\boldsymbol{\mu} = \mathrm{Linear}([\overrightarrow{\boldsymbol{h}_L^{(E)}}, \overleftarrow{\boldsymbol{h}_1^{(E)}}]) \tag{5}$$

$$\boldsymbol{\sigma} = \mathrm{Linear}([\overrightarrow{\boldsymbol{h}_L^{(E)}}, \overleftarrow{\boldsymbol{h}_1^{(E)}}]) \tag{6}$$

$$\boldsymbol{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \tag{7}$$

ここで，Embed 関数は埋め込みベクトルへの変換，Linear 関数はバイアス項を含む線形変換を表す．また，$(\overrightarrow{\boldsymbol{h}_1^{(E)}}, ..., \overrightarrow{\boldsymbol{h}_L^{(E)}})$ は順方向 LSTM の隠れ層，$(\overleftarrow{\boldsymbol{h}_1^{(E)}}, ..., \overleftarrow{\boldsymbol{h}_L^{(E)}})$ は逆方向 LSTM の隠れ層を示している.

次にスタイルラベルを線形変換することでスタイル特徴量 $\boldsymbol{a}$ を得る．学習時にはソーススタイル s のラベル $\boldsymbol{y}^{(s)}$ を用いて再構成を学習し，推論時にはターゲットスタイル t のラベル $\boldsymbol{y}^{(t)}$ を用いてスタイル変換を行う.

$$\boldsymbol{a}^{(x)} = \mathrm{Linear}(\boldsymbol{y}^{(s)}) \quad \text{(in training)} \tag{8}$$

$$\boldsymbol{a}^{(\hat{x})} = \mathrm{Linear}(\boldsymbol{y}^{(t)}) \quad \text{(in inference)} \tag{9}$$

デコーダには単方向 LSTM を用いる．隠れ層の初期値を $\boldsymbol{h}_0^{(D)} = \mathrm{Linear}([\boldsymbol{z}, \boldsymbol{a}^{(\hat{x})}])$ とすると，m ステップ目のデコーダの隠れ層 $\boldsymbol{h}_m^{(D)}$ は以下のように求めることができる.

$$\boldsymbol{d}_m = \mathrm{Embed}(\hat{\boldsymbol{x}}_{m-1}) \tag{10}$$

$$\boldsymbol{h}_m^{(D)} = \mathrm{LSTM}(\boldsymbol{h}_{m-1}^{(D)}, [\boldsymbol{d}_m]) \tag{11}$$

ここで $\hat{\boldsymbol{x}}_{m-1}$ はデコーダが前ステップで出力した単語であり，$\boldsymbol{d}_m$ は m ステップ目にデコーダに入力する単語の埋め込み表現である． 最後にデコーダの隠れ層の値からデコーダ出力 $\boldsymbol{o}_m$ を計算し，単語出力確率を $\boldsymbol{p}_m$ 決定する.

$$\boldsymbol{o}_m = \mathrm{Linear}(\boldsymbol{h}_m^{(D)}) \tag{12}$$

$$\boldsymbol{p}_m = \mathrm{Softmax}(\boldsymbol{o}_m) \tag{13}$$

この $\boldsymbol{p}_m$ から，確率が最大となる単語を m ステップ目の出力単語 $\hat{\boldsymbol{x}}_m$ として得る.

3. 提案モデル：CVAE + CWS

本節では，CVAE に内容語保存機構を組み合わせた提案手法について述べる．提案手法の概略図を**図 1** に示す.

3.1 内容語保存機構（Content Word Storage: CWS）

VAE を用いたテキストスタイル変換では，変換前後で保

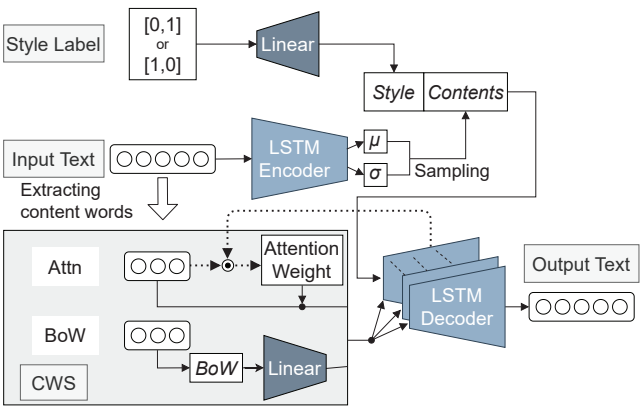

図 1 提案手法：CVAE + CWS の概略図
Fig. 1 Schematic diagram of CVAE + CWS.

持すべき情報も含めてすべて固定次元の潜在表現に埋め込む．しかし，限られたキャパシティを有効活用するためには，保持したい情報は潜在表現から除くことが望ましい．そこで，スタイル変換時に保持すべき情報を明示的に「内容語」として定め，コンテンツ特徴量と切り離し，デコーダに直接伝達する手法である CWS を提案する．

内容語とは「実質的な意味を持ち，自立した要素になり得る語」と説明される [15]．本研究では，「名詞・動詞・形容詞」を内容語と定義する．

3.1.1 Bag-of-Word を用いた CWS

Bag-of-Words（以下，BoW）は古典的なテキスト特徴量の表現方法の 1 つである．これは，テキスト中での単語の出現回数を表す one-hot ベクトルの和で表現されるもので，データ全体の語彙数に等しい次元を持つ．

BoW を用いた CWS（以下，CWS-BoW）では，まず学習データ全体から内容語を抽出し，内容語の語彙辞書を生成する．その後，各入力テキストに対し内容語を one-hot ベクトルの形式で取り出す．これを線形変換して LSTM デコーダの全ステップの入力に加えることで，生成テキストに内容語が出現するような制約を強めることが可能になる．

3.1.2 注意機構を用いた CWS

注意機構（Attention Mechanism：Attn）[16] は，あるステップでのデコーダの出力を決定する際に，入力のどこに着目するかを決める仕組みであり，ニューラル機械翻訳を中心とした seq2seq [17] モデルの中で提案された．これにより，長期系列においても情報を欠損なく保存し，出力に有用な情報を重点的に伝達することが可能となる．

注意機構を用いた CWS（以下，CWS-Attn）では，入力文に含まれる内容語の埋め込み表現と，LSTM デコーダの各ステップの隠れ層に対して内積を取り，注意重みを計算する．そして，注意重みと内容語の埋め込み表現の積をコンテキストベクトルとしてデコーダ出力と組み合わせて単語予測に用いる．コンテクストベクトルは，データの語彙数に比例しない小さな次元数で，かつ文脈の考慮が可能という点で BoW よりも優れている．これによって，生成時のコンテンツ保存性がさらに改善することが期待される．

3.1.3 CVAE + CWS による学習

コンテンツ特徴量 $\boldsymbol{z}$ とスタイル特徴量 $\boldsymbol{a}$ は 2 節と同じ方法で計算する．CWS-BoW では，入力から内容語を取り出し，内容語系列 $\boldsymbol{x}_{\mathrm{CW}}$ を得る．これを BoW 関数に入力し，ワンホットベクトルの和で表される BoW 特徴量 $\boldsymbol{b}^{(x)}$ に変換，さらに線形変換により CWS 特徴量 $\boldsymbol{z}_{\mathrm{CWS}}$ を得る．

$$\boldsymbol{b}^{(x)} = \mathrm{BoW}(\boldsymbol{x}_{\mathrm{CW}}) \tag{14}$$

$$\boldsymbol{z}_{\mathrm{CWS}} = \mathrm{Linear}(\boldsymbol{b}^{(x)}) \tag{15}$$

デコーダの隠れ層の初期値を $\boldsymbol{h}_0^{(D)} = \mathrm{Linear}([\boldsymbol{z}, \boldsymbol{a}^{(\hat{x})}])$ とすると，CWS-BoW を用いた際の m ステップ目のデコーダの隠れ層，デコーダ出力，単語出力確率は以下のように求めることができる．

$$\boldsymbol{d}_m = \mathrm{Embed}(\hat{\boldsymbol{x}}_{m-1}) \tag{16}$$

$$\boldsymbol{h}_m^{(D)} = \mathrm{LSTM}(\boldsymbol{h}_{m-1}^{(D)}, [\boldsymbol{d}_m, \boldsymbol{z}_{\mathrm{CWS}}]) \tag{17}$$

$$\boldsymbol{o}_m = \mathrm{Linear}(\boldsymbol{h}_m^{(D)}) \tag{18}$$

$$\boldsymbol{p}_m = \mathrm{Softmax}(\boldsymbol{o}_m) \tag{19}$$

次に，CWS-Attn では，まず入力単語列のうち内容語以外の単語をマスクした内容語系列 $\boldsymbol{x}_{\mathrm{CW}}$ を得る．これを埋め込みベクトル $\boldsymbol{e}_{\mathrm{CW}}$ に変換し，注意の対象とする．

$$\boldsymbol{e}_{\mathrm{CW}} = \mathrm{Embed}(\boldsymbol{x}_{\mathrm{CW}}) \tag{20}$$

この内容語埋め込みベクトルとデコーダの隠れ層の注意重み $\boldsymbol{w}_m^{\mathrm{Attn}}$ を計算し，m ステップ目におけるコンテキストベクトル $\boldsymbol{c}_m$ を求める．

$$\boldsymbol{w}_m^{\mathrm{Attn}} = \mathrm{Softmax}(\boldsymbol{e}_{\mathrm{CW}} \odot \boldsymbol{h}_m^{(D)}) \tag{21}$$

$$\boldsymbol{c}_m = \boldsymbol{e}_{\mathrm{CW}} \odot \boldsymbol{w}_m^{\mathrm{Attn}} \tag{22}$$

デコーダの隠れ層の初期値を $\boldsymbol{h}_0^{(D)} = \mathrm{Linear}([\boldsymbol{z}, \boldsymbol{a}^{(\hat{x})}])$ とする．CWS-Attn を用いた際のデコーダの隠れ層と m ステップ目のデコーダ出力は，式 (16) とコンテキストベクト $\boldsymbol{c}_m$ を用いて以下のように求めることができる．

$$\boldsymbol{h}_m^{(D)} = \mathrm{LSTM}(\boldsymbol{h}_{m-1}^{(D)}, \boldsymbol{d}_m) \tag{23}$$

$$\boldsymbol{o}_m = \mathrm{Linear}([\boldsymbol{h}_m^{(D)}, \boldsymbol{c}_m]) \tag{24}$$

単語出力確率 $\boldsymbol{p}_m$ は式 (19) と同様である．

3.2 位置埋め込み（Positional Embedding: PE）

注意機構を単体で用いる場合，文脈に沿った単語の出力確率を向上させることは可能だが，各単語の出現順序の情報は含まれず，同じ単語が複数入力に含まれる場合はそれらの単語ベクトルが同一になってしまう．そこで入力テキストのうち，各内容語の単語ベクトルに対して，その単語の位置を明示的に表すような埋め込みベクトルを加算することを提案する．これにより，各内容語の出現数と出現位置も考慮可能になることが期待される．

本稿では，Transformer [18] で採用されている三角関数を用いた絶対位置埋め込み（Absolute Positional Embedding: APE）を用いる．具体的には，入力テキストのうち内容語以外をマスクした単語ベクトル系列に対して，以下の式 (25) で表される位置埋め込みベクトル $p_t^{(i)}$ を加算する．

$$p_t^{(i)} = \begin{cases} \sin\left(t/10000^{i/d}\right) & \text{if } i = 2k \\ \cos\left(t/10000^{i/d}\right) & \text{if } i = 2k+1 \end{cases} \tag{25}$$

3.3 サイクル学習

CVAE をベースとしたモデルでは，パラレルデータを用いない場合，再構成ベースの学習のみを行う．しかし，再構成の学習だけでは，推論時にターゲットクラスラベルを用いてスタイルの条件付けを行っても，スタイルが変換されたテキストを十分に合成するのは難しいと予想される．そこで，前述までの CVAE ＋ CWS ＋ PE を用いて合成したスタイル変換テキストを疑似パラレルデータとし，再構成と同時に，疑似パラレルデータを元のスタイルに戻す再変換を行うサイクル学習を導入する．これにより，パラレルデータを使用しない条件を維持しつつ，テキストスタイル変換の性能向上が期待される．以下，サイクル学習を導入した提案モデルを CycleCVAE ＋ CWS と呼称する．

また，サイクル学習を行う際，再変換時には基本的に入力テキストと出力テキストの系列長が異なるため，絶対位置埋め込みを用いた内容語の保存は正常に機能しないことが予想できる．そこで，疑似的にテキスト全体における内容語の相対的な位置を埋め込む，内容語絶対位置埋め込み（Content Word Absolute Positional Embedding: CWAPE）を提案する．これは入力テキストから取り出した内容語系列だけを見て，その絶対位置を埋め込みベクトルとする方法である．本稿ではスタイル変換前後で内容語は変化しないと仮定しているため，これを用いることで再変換時にも内容語を効果的に保存することが可能となる．

4. 実験的評価

4.1 実験条件

「非流暢性の有無」と「標準語・関西弁」の 2 種類のスタイルを対象として両方向にスタイル変換する実験を行った．実験で比較するシステムとして，ベースラインの CVAE と提案手法の CVAE ＋ CWS-BoW，CWS-Attn，CWS-Attn ＋ APE と CycleCVAE ＋ CWS-Attn を用いた．

実験用データとして，非流暢性変換実験では，日本語話し言葉コーパス（CSJ）[19] から非流暢性のあるテキストを抽出した．付与されたメタ記号を基に非流暢性を取り除いたテキストを作成し，それぞれに非流暢性「あり」「なし」とラベル付けした．最大文長 27，平均文長 15.2，語彙数 40,738 で，train 426,400 文，dev 22,926 文，test 9,170 文とした．ただし，今回は教師なしで学習するために対応付けを外してノンパラレルにした．また，方言変換用のデータとして，CSJ の非流暢性「なし」データに「標準語」，関西弁コーパス（KVJ）[20] のデータに「関西弁」ラベルを付けて使用した．最大文長 26，平均文長 13.7，語彙数 43,921 で，train 337,258 文，dev 18,133 文，test 7,253 文とした．

客観評価実験では，4.2 節で説明する各評価指標について，テストデータ全体の平均を計算した．主観評価実験では，非流暢性変換と方言変換について，日本語が堪能な 20 代〜30 代の男女 10 名と 16 名に対して，変換前後テキストのペアを順に提示し，4.2 節で示す 3 つの項目について評価を依頼した．テストデータからランダムに選択した 100 テキストに対して各テキストが異なる被験者から最低 3 回評価されるように実験を行った．ここで，どのモデルからの生成テキストであるかは隠した状態で実施した．

4.2 評価指標

客観評価指標には，以下の 3 種類を用いた．

・Accuracy

Accuracy（以下，AC）は正解率を示し，スタイル変換の成功度合いとして評価指標に用いられる．計算方法は，まず学習データを用いて CNN 分類器を事前に訓練した．これを用いて，各モデルが生成したテキストのスタイルを予測し，ターゲットスタイルであると分類できた割合を計算した．方言変換ではスタイル間で出現する語彙の違いから，内容語もスタイルを表す情報として分類器が判断するため，内容語抜きの AC（AC w/o CW）を計算した．

・BLEU

BLEU[21] は機械翻訳の分野で提案されたテキスト全体のコンテンツ保存性を示す指標である．パラレルコーパスを利用できる場合，生成テキストと参照テキストを比較する reference-BLEU（r-BLEU）を計算することができる．しかし，ノンパラレルコーパスを用いる場合は参照テキストが用意できないため，生成テキストとスタイル変換前の入力テキストを比較する self-BLEU（s-BLEU）を用いる場合が多い．本稿では，非流暢性変換では r-BLEU を，方言変換では s-BLEU をそれぞれ計算した．

・Content Word Error Rate（CWER）

CWER は，音声認識の分野で用いられる単語誤り率（WER）を内容語系列に対して適用した指標である．テキストスタイル変換では，入力テキストと生成テキストの内容語は変化しないことが望ましい．そこで，CWER は内容語の保存性を計るため，「生成テキストの内容語系列」と「入力テキストの内容語系列」に対して WER を計算した．

また，主観評価では，「スタイル変換度合い (ST)」「コンテンツ保存性 (CP)」「自然性 (Nat)」の 3 つの基準を用いた．一般的なアンケートなどで用いられるリッカート尺度に則り，ST については 1〜4 までの 4 段階，CP と Nat については 1〜5 までの 5 段階で評価を行った．

4.3 実験結果

各実験の客観評価結果を**表 1** に示す． ベースラインと比較して提案手法の方が各指標で上回り，コンテンツ保存性では CWS-Attn に APE を組み合わせたモデルが最も優れていることが分かった．特に CWER の改善は目覚ましく，これは注意機構の注意重みをヒートマップ表示した**図 2** からも確認することができた．また，サイクル学習の導入により，AC が大幅に改善することが確認できた．し

表 1 客観評価結果．サイクル学習ありとなしでそれぞれ最も良い結果を太字で示している．
Table 1 Subjective evaluation results. The best results are shown in bold.

Method	Disfluency			Dialect		
	AC ↑	r-BLEU ↑	CWER ↓	AC w/o CW ↑	s-BLEU ↑	CWER ↓
Reference	98.35	100.00	0.00	-	-	-
Baseline (CVAE)	51.44	35.72	54.16	51.86	30.85	60.95
CVAE+CWS-BoW	49.55	52.05	30.94	52.27	39.47	47.80
CVAE+CWS-Attn	49.80	57.50	19.73	56.47	51.85	25.38
CVAE+CWS-Attn+APE	**58.02**	**61.57**	**5.62**	**58.48**	**60.14**	**6.82**
CycleCVAE+CWS+APE	**87.99**	24.36	15.11	–	–	–
CycleCVAE+CWS+CWAPE	78.20	**58.79**	**6.30**	81.78	41.38	18.29

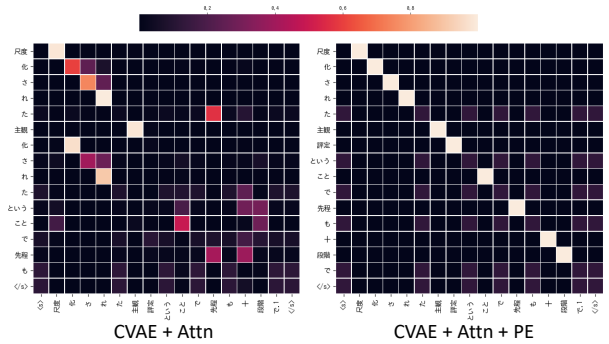

図 2 注意重み行列の比較．縦軸が生成テキスト，横軸が入力テキストを示す．
Fig. 2 Comparison of attention weight heatmaps. The vertical axis is generated text and the horizontal axis is the input text.

かし，方言変換についてみると，サイクル学習を導入した際にコンテンツ保存性をある程度損なってしまうことも分かった．

より詳細に結果を確認するため，変換方向毎の評価結果を**表 2** に示す．非流暢性変換の AC を見ると，「なし→あり」よりも「あり→なし」の方が高いという傾向がみられ，非流暢性を除去するよりも挿入する方が困難であることが分かった．同時に，位置埋め込みやサイクル学習の導入により，「なし→あり」の方向の性能を大きく改善できていることも分かった．一方，方言変換では方向による AC の差は非流暢性変換程大きくないという結果であった．

最後に CVAE と CVAE + CWS-Attn の主観評価結果を**表 3** に示す．対応のある t 検定から，提案手法についてコンテンツ保存性を示す CP が有意に改善していることが分かったが，その他の指標に有意な差は見られなかった．

4.4 考察

非流暢性変換，方言変換の両タスクにおいて，提案手法での CWER が大幅に改善していることから，特に CWS-Attn で位置埋め込みを導入することにより，内容語を正しい回数，位置へ出力可能になると考えられる．位置埋め込み導入による AC の向上については，位置埋め込みによって CWS だけで内容語の情報の大部分を保存できるようになったことで，エンコーダが内容語以外の情報をより多く含むような潜在表現を獲得可能になったと予想できる．

また，サイクル学習を導入したモデルが方言変換でコンテンツ保存性を損なっている点について，方言変換における疑似パラレルデータの質が悪いことが予想できる．これは客観評価でコンテンツ保存性を示す指標について非流暢性変換よりも方言変換の方が劣ること，主観評価における方言変換の結果もコンテンツ保存性と自然性で劣ることが示唆している．また，方言変換で用いている標準語データは「講演」，関西弁データは「日常会話」という方言以外の異なるスタイルを持つデータであることも，今回の方言変換を困難にしている一因であり，この点についても改善の余地がある．

5. まとめと今後の展望

TTS への応用を想定したテキストスタイル変換において，CVAE に内容語保存機構やサイクル学習を組み合わせ，コンテンツ保存性やスタイル変換性能を向上させた．今後の展望として，さらなる性能向上を目指し，現実的に準備可能な少量の正解変換データを利用する条件で実験を行う．例えば，サイクル学習で疑似パラレルデータに対して少量の正解データを混合する，CVAE + CWS で事前学習したモデルを少量の正解データを用いて fine-tuning する，などの方法が考えられる．また，TTS への応用も実施することを視野に入れ，提案モデルと既存の TTS モデルを単純にカスケードしたシステムの性能や，現在の提案モデルが捉えている潜在表現が TTS モデルの学習に対して有用であるか否かを調査する予定である．

謝辞 本研究は名古屋大学及び JST 科学技術イノベーション創出に向けた大学フェローシップ創設事業 JPMJFS2120 による「名古屋大学融合フロンティアフェローシップ」の支援と，NEDO の委託事業である JPNP20006，及び JSPS 科研費 JP21H05054 の支援を受けたものである．

表 2　スタイル変換方向毎の性能比較
Table 2 Performance comparison between different style transfer directions

Method	Direction	AC ↑	r-BLEU ↑	Direction	AC w/o CW ↑	s-BLEU ↑
CVAE+CWS-Attn	なし→あり	15.83	54.39	標準語→関西弁	52.13	53.95
	あり→なし	83.77	52.32	関西弁→標準語	63.91	47.72
CVAE+CWS-Attn+APE	なし→あり	38.34	60.99	標準語→関西弁	55.32	63.00
	あり→なし	77.69	54.88	関西弁→標準語	63.91	54.78
CycleCVAE+CWS+CWAPE	なし→あり	63.66	55.36	標準語→関西弁	75.97	45.17
	あり→なし	92.74	54.37	関西弁→標準語	91.75	34.54

表 3　主観評価結果（95%信頼区間を併記）
Table 3 Objective evaluation results shown with 95% confidence interval.

Method	ST ↑	CP ↑	Nat ↑
	Disfluency		
CVAE	2.40 ± 0.15	2.46 ± 0.18	4.01 ± 0.24
+ CWS-Attn	2.39 ± 0.08	**3.93** ± 0.20	3.96 ± 0.20
	Dialect		
CVAE	2.56 ± 0.33	2.00 ± 0.15	3.61 ± 0.33
+ CWS-Attn	2.38 ± 0.24	**3.11** ± 0.23	3.39 ± 0.27

参考文献

[1] Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Ryan, R., Saurous, R. A., Agiomyrgiannakis, Y. and Wu, Y.: Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions, *Proc. ICASSP*, IEEE, pp. 4779–4783 (2018).

[2] Um, S., Oh, S., Byun, K., Jang, I., Ahn, C. and Kang, H.: Emotional Speech Synthesis with Rich and Granularized Control, *Proc. ICASSP*, pp. 7254–7258 (2020).

[3] 望月 朝香，鈴木 泰博：小説における文体印象解析の試み，技術報告 128(2007-BIO-011)，名古屋大学大学院情報科学研究科複雑系科学専攻, 名古屋大学大学院情報科学研究科複雑系科学専攻 (2007).

[4] 岸本 千秋：ウエブ日記に見られる話しことば的文体 (話し言葉の日本語) – (話し言葉の語彙と文法)，日本語学，Vol. 27, No. 5, pp. 168–176 (2008).

[5] 宮崎 千明，平野 徹，東中 竜一郎，牧野 俊朗，松尾 義博，佐藤 理史：文節機能部の確率的書き換えによるキャラクタ性変換，言語処理学会第 21 回年次大会, B1-4，pp. 277–280 (2015).

[6] Jhamtani, H., Gangal, V., Hovy, E. H. and Nyberg, E.: Shakespearizing Modern Language Using Copy-Enriched Sequence-to-Sequence Models, *Proc. the Workshop on Stylistic Variation*, pp. 10–19 (2017).

[7] Carlson, K., Riddell, A. and Rockmore, D.: Evaluating prose style transfer with the Bible, *Royal Society Open Science*, Vol. 5, No. 10, p. 171920 (2018).

[8] Kingma, D. P. and Welling, M.: Auto-Encoding Variational Bayes, *Proc ICLR* (Bengio, Y. and LeCun, Y., eds.) (2014).

[9] Hu, Z., Yang, Z., Liang, X., Salakhutdinov, R. and Xing, E. P.: Toward Controlled Generation of Text, *Proc. ICML* (Precup, D. and Teh, Y. W., eds.), Vol. 70, pp. 1587–1596 (2017).

[10] Lample, G., Subramanian, S., Smith, E. M., Denoyer, L., Ranzato, M. and Boureau, Y.: Multiple-Attribute Text Rewriting, *Proc. ICLR* (2019).

[11] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C. and Bengio, Y.: Generative Adversarial Nets, *Proc. NeurIPS*, pp. 2672–2680 (2014).

[12] Zhao, J. J., Kim, Y., Zhang, K., Rush, A. M. and LeCun, Y.: Adversarially Regularized Autoencoders, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018* (Dy, J. G. and Krause, A., eds.), Proceedings of Machine Learning Research, Vol. 80, PMLR, pp. 5897–5906 (online), available from ⟨http://proceedings.mlr.press/v80/zhao18b.html⟩ (2018).

[13] Laugier, L., Pavlopoulos, J., Sorensen, J. and Dixon, L.: Civil Rephrases Of Toxic Texts With Self-Supervised Transformers, *Proc. EACL* (Merlo, P., Tiedemann, J. and Tsarfaty, R., eds.), pp. 1442–1461 (2021).

[14] Kingma, D. P., Mohamed, S., Rezende, D. J. and Welling, M.: Semi-supervised Learning with Deep Generative Models, *Proc. NeurIPS* (Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D. and Weinberger, K. Q., eds.), pp. 3581–3589 (2014).

[15] 三宅 知宏：現代日本語における文法化: 内容語と機能語の連続性をめぐって (< 特集 > 日本語における文法化・機能語化)，日本語の研究， Vol. 1, No. 3, pp. 61–76 (2005).

[16] Luong, T., Pham, H. and Manning, C. D.: Effective Approaches to Attention-based Neural Machine Translation, *Proc. EMNLP* (Màrquez, L., Callison-Burch, C., Su, J., Pighin, D. and Marton, Y., eds.), pp. 1412–1421 (2015).

[17] Sutskever, I., Vinyals, O. and Le, Q. V.: Sequence to sequence learning with neural networks, *Proc. NeurIPS*, pp. 3104–3112 (2014).

[18] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. and Polosukhin, I.: Attention is All you Need, *Proc. NeurIPS* (Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N. and Garnett, R., eds.), pp. 5998–6008 (2017).

[19] Maekawa, K., Koiso, H., Furui, S. and Isahara, H.: Spontaneous Speech Corpus of Japanese, *Proc. LREC*, pp. 947–952 (2000).

[20] Kevin, H.: An introduction to the Kansai dialect corpus, *Journal of Policy Studies*, No. 41, pp. 157–164 (2012).

[21] Papineni, K., Roukos, S., Ward, T. and Zhu, W.: Bleu: a Method for Automatic Evaluation of Machine Translation, *Proc. ACL*, pp. 311–318 (2002).