方言ラベルを補助特徴量としたEnd-to-End日本語方言音声認識*

☆今泉遼 (都立大), 増村亮 (NTT), 塩田さやか, △貴家仁志 (都立大)

1 はじめに

音声認識とは話し言葉を文字列に変換する技術である. 近年, 深層学習の発展により Deep Neural Network (DNN) を用いた手法の 1 つである End-to-End 音声認識システムが提案され, 高い認識性能が得られることから活発に研究されている [1]. 高性能な End-to-End 音声認識の構築には大量のデータが必要であることが知られているが, 用いられるデータは標準語で話されているものが基本となっている. そのため, 地域特有の方言音声を End-to-End 音声認識に入力すると, 標準語に比べて認識性能が大幅に低下することが知られている. しかしながら, 方言音声データとして公開されているデータベースが少ないため各方言に合わせた高性能な音声認識器を作ることは難しい. この問題の解決法の 1 つとして方言と標準語を合わせて学習することが挙げられる. 標準語のデータベースを用いることで方言データの不足を解決することができ, かつ全体のデータ量が増えることから方言だけでなく標準語の認識精度が上がるとも期待できるためである. しかし, 方言と標準語を合わせると各方言より標準語のデータ量の方が圧倒的に多いため各方言および標準語それぞれの方言というドメイン依存性が薄まってしまい, 特に標準語の認識性能が下がってしまうという懸念もある. そこで本研究では, 各方言ラベルを補助特徴量として用いて学習することで上記の問題を解決することを目指す. 具体的にはモデル学習時に音声データと正解テキストだけでなく方言ラベルも用いることによって入力音声のドメイン依存性を高め, 方言だけでなく標準語の認識性能も改善することを目指す. 本研究では多言語音声認識の分野 [2] において高い性能を得られることが知られている Transformer に基づく End-to-End 音声認識システム [3,4] に対して方言ラベルを補助特徴量として用いて学習し, 標準語および方言音声の認識システムを構築することを提案する. 実験では学習データおよびテストデータに方言のみ, 標準語のみ, 両方を混ぜたものを用いて実験を行い, 補助特徴量の有効性を調査した. 実験結果から方言ラベルを補助特徴量として用いる場合と方言ラベルを用いない場合を比較したところ提案法の方の CER が 19.2%程度改善したことを報告する.

2 Transformer に基づく End-to-End 音声認識

本節では Transformer に基づく自己回帰生成モデルを用いた End-to-End 音声認識 [5] について説明する. このモデルは入力される音響特徴量系列 $\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_M\}$ から得られるテキスト $\boldsymbol{W} = \{w_1, ..., w_N\}$ の生成確率を予測する. ここで w_n はテキストの n 番目のトークン, $\boldsymbol{x}_m$ は発話の m 番目の音響特徴量を示す. N はテキスト内のトークンの数, M は音声に含まれる音響特徴量の数である. 自己回帰生成モデルでは, $\boldsymbol{W}$ の生成確率を次のように定義する.

$$P(\boldsymbol{W}|\boldsymbol{X};\boldsymbol{\Theta}) = \prod_{n=1}^{N} P(w_n|\boldsymbol{W}_{1:n-1}, \boldsymbol{X};\boldsymbol{\Theta}) \quad (1)$$

ここで, $\boldsymbol{\Theta}$ はモデルパラメータセットを表し, $\boldsymbol{W}_{1:n-1} = \{w_1, ..., w_{n-1}\}$ となる.

2.1 ネットワーク構造

Transformer に基づく End-to-End 音声認識モデルでは確率 $P(w_n|\boldsymbol{W}_{1:n-1}\boldsymbol{X};\boldsymbol{\Theta})$ は音声エンコーダとテキストデコーダを使用して計算できる. これらはそれぞれ Transformer ブロックを積み重ねることで構成される. モデルパラメータは音声エンコーダのパラメータセット θ_{enc} とテキストデコーダのパラメータセット θ_{dec} に分けられる.

音声エンコーダ: 音声エンコーダは, I 個の Transformer エンコーダブロックを使用して, 入力音響特徴量系列を隠れ表現 $\boldsymbol{H}^{(I)}$ に変換する. i 番目の Transformer エンコーダーブロックは, 以下の式のように, 下位層の入力 $\boldsymbol{H}^{(i-1)}$ から i 番目隠れた表現 $\boldsymbol{H}^{(i)}$ 構成する.

$$\boldsymbol{H}^{(i)} = \text{TransformerEncoderBlock}(\boldsymbol{H}^{(i-1)};\theta_{\text{enc}}) \quad (2)$$

ここで TransformerEncoderBlock 関数はスケーリングされた内積マルチヘッド自己注意層と位置ごとのフィードフォワードネットワークで構成される Transformer エンコーダブロックである. 隠れ表現 $\boldsymbol{H}^{(i)} = \{\boldsymbol{h}_1^{(0)}, ..., \boldsymbol{h}_{M'}^{(0)}\}$ は, 位置情報を埋め込んだ連続ベクトルを追加する AddPostionalEncoding 関数を用いて以下のように定義される.

$$\boldsymbol{h}_{m'}^{(0)} = \text{AddPostionalEncoding}(\boldsymbol{h}_{m'}) \quad (3)$$

$\boldsymbol{H} = \{\boldsymbol{h}_1, ..., \boldsymbol{h}_{M'}\}$ は, 畳み込み層とプーリング層で構成される ConvolutionPooling 関数で定義される.

$$\boldsymbol{H} = \text{ConvolutionPooling}(\boldsymbol{x}_1, ..., \boldsymbol{x}_{M'};\theta_{\text{enc}}) \quad (4)$$

ここで M' はサブサンプリングされたシーケンスの長さを表す.

テキストデコーダ: テキストデコーダは, 先行するトー

---

*Using Dialect Label as Auxiliary Feature for End-to-End Japanese Dialect Speech Recognition, by Ryo Imaizumi (TMU), Ryo Masumura (NTT), Sayaka Shiota, Hitoshi Kiya (TMU)

クンと音声の隠れ表現からトークンの生成確率を計算する．n 番目のトークン w_n の予測確率は，次のように計算される．

$$P(w_n|\boldsymbol{W}_{1:n-1}, \boldsymbol{X}; \boldsymbol{\Theta}) = \mathrm{Softmax}(\boldsymbol{u}_{n-1}^{(J)}; \theta_{\mathrm{dec}}) \quad (5)$$

ここで，Softmax 関数は線形変換を使用したソフトマックス層を表し $\boldsymbol{u}_{n-1}^{(J)}$ は，J 個の Transformer デコーダーブロックから計算される．j 番目の Transformer デコーダーブロックは，以下の式によって，下位の入力 $\boldsymbol{U}_{1:n-1}^{(j-1)} = \{\boldsymbol{u}_1^{(j-1)}, ..., \boldsymbol{u}_{n-1}^{(j-1)}\}$ から j 番目の隠れ表現 $\boldsymbol{u}_{n-1}^{(j)}$ を構成する．

$$\boldsymbol{u}_{n-1}^{(j)} = \mathrm{TransformerDecoderBlock}(\boldsymbol{U}_{1:n-1}^{(j-1)}, \boldsymbol{H}^{(I)}; \theta_{\mathrm{dec}}) \quad (6)$$

TransformerDecoderBlock 関数は，スケーリングされた内積マルチヘッド自己注意層とスケール内積マルチヘッドソースターゲットアテンション層，および位置ごとのフィードフォワードネットワークで構成される Transformer デコーダブロックである．隠れ表現 $\boldsymbol{U}_{1:n-1}^{(0)} = \{\boldsymbol{u}_1^{(0)}, ..., \boldsymbol{u}_{n-1}^{(0)}\}$ は，

$$\boldsymbol{u}_{n-1}^{(0)} = \mathrm{AddPostionalEncoding}(\boldsymbol{w}_{n-1}) \quad (7)$$

$$\boldsymbol{w}_{n-1} = \mathrm{Embedding}(w_{n-1}; \boldsymbol{\theta}_{\mathrm{dec}}) \quad (8)$$

で表される．Embedding 関数は，入力トークンを連続ベクトルに埋め込む線形層である．

2.2 教師あり学習

End-to-End 音声認識では，モデルパラメータセットを発話単位のラベル付きデータから最適化する．次式に音声とテキストのペアデータの場合を示す．

$$\mathcal{D} = \{(\boldsymbol{X}^1, \boldsymbol{W}^1), ..., (\boldsymbol{X}^T, \boldsymbol{W}^T)\} \quad (9)$$

T は，学習データセット内の発話の数を表す．最尤推定に基づく目的関数は次のように定義される．

$$\mathcal{L}_{\mathrm{mle}}(\theta_{\mathrm{enc}}, \theta_{\mathrm{dec}}) = -\sum_{t=1}^{T}\sum_{n=1}^{N^t} \log P(w_n^t|\boldsymbol{W}_{1:n-1}^t, \boldsymbol{X}^t; \theta_{\mathrm{enc}}, \theta_{\mathrm{dec}}) \quad (10)$$

$\boldsymbol{w}_n^t$ は t 番目の発話の n 番目のトークンを表し，$\boldsymbol{W}_{1:n-1}^t = \{w_1^t, ..., w_{n-1}^t\}$ であり，N^t は，t 番目の発話のトークンの数を表す．

2.3 方言音声認識における課題

2.2 節までで述べたのは一般的な End-to-End 音声認識の枠組みである．しかし標準語に含まれない単語やイントネーションの多い方言音声を入力する場合，従来の End-to-End 音声認識では認識率が大幅に低下してしまう．これまでにもドメイン適応など未知のドメインに対応する手法は多く提案されてきたが，方言音声に対する有効性は報告されていない．特に方言は標準語に同じような発音があっても意味や使い方が全く異なる場合もあるため標準語か，もしくは，どこの方言かなどの情報がないと音声認識時に認識率が大幅に低下してしまう．そこで方言音声認識という課題をドメイン適応の課題の一つと捉えて研究を行った．

3 提案法

本研究の提案法である方言ラベルを補助特徴量として用いた学習法について説明する．2.3 節でも述べた通り，方言音声認識はドメイン適応の一つとして捉えることができる．そこでドメイン，つまり方言を陽に指定して用いる方言音声認識について考える．そのために，提案法では入力音声 $\boldsymbol{X}$ の方言ラベル D が観測できる状況で，出力単語系列 $\boldsymbol{W}$ を予測する End-to-End 音声認識をモデル化する．提案法では，$\boldsymbol{W}$ の生成確率を次のように定義する．

$$P(\boldsymbol{W}|\boldsymbol{X}, D; \boldsymbol{\Theta}) = \prod_{n=1}^{N} P(w_n|\boldsymbol{W}_{1:n-1}, D, \boldsymbol{X}; \boldsymbol{\Theta}) \quad (11)$$

このように 1 発話につき方言ラベル 1 つの生成確率を計算して，$\boldsymbol{W}$ の生成確率を計算した．式 (11) を Transformer でモデル化する方法を図 1 に示す．図 1 の左側が音声エンコーダ部分，右側がテキストデコーダ部分を表す．音声エンコーダでは，入力音響特徴量を隠れ表現に変換するために，式 (2)〜 (4) と同様のモデル化を採用する．一方テキストデコーダでは，方言ラベルと隠れ表現を用いてトークンの生成確率を計算するために，デコーダの最初の入力でトークン記号を連続値ベクトルに変換して入力し，コンテキストの条件付けを行う．具体的には，従来手法で音響特徴量に依存しない始端記号を入力する部分で，方言を表すラベルを入力することで方言に依存したデコーダのモデル化を行う．また 2.2 節で示したモデルパラメータセットの最適化に用いるセットを音声とテキストのペアから次式のような音声，方言ラベル，テキストのセットに代えて最適化する．

$$\mathcal{D} = \{(\boldsymbol{X}^1, D^1, \boldsymbol{W}^1), ..., (\boldsymbol{X}^T, D^T, \boldsymbol{W}^T)\} \quad (12)$$

式 (10) と式 (12) から提案法で用いる目的関数は次のように定義される．

$$\mathcal{L}_{\mathrm{mle}}(\theta_{\mathrm{enc}}, \theta_{\mathrm{dec}}) = -\sum_{t=1}^{T}\sum_{n=1}^{N^t} \log P(w_n^t|\boldsymbol{W}_{1:n-1}^t, D^t, \boldsymbol{X}^t; \theta_{\mathrm{enc}}, \theta_{\mathrm{dec}}) \quad (13)$$

音声，テキストのペアに加えて方言ラベルも用いて最適化を行い，生成確率を求めることで各方言の情報が強まり，方言依存性が高まることから方言を含む音声だけでなく標準語においても音声認識性能が上がることが期待できる．

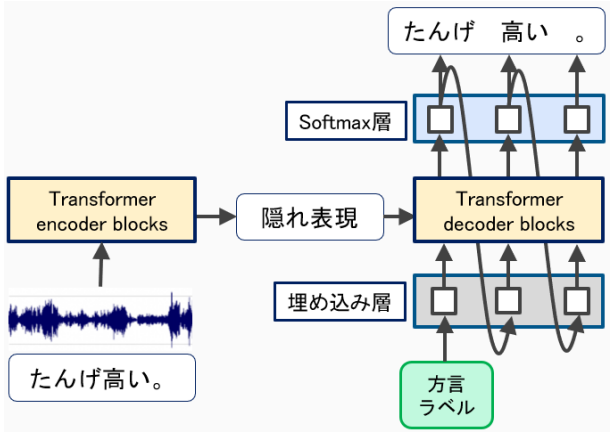

Fig. 1　方言ラベルを補助特徴量とした
Transformer encoder-decoder モデル
(入力：音響特徴量, 出力：テキスト, 例：青森方言)

Table 1　日本語方言データベースの方言ごとの発話数の内訳

	地方	学習	開発	テスト	全体
方言	青森	10,741	676	676	12,093
	広島	18,670	566	566	19,803
	熊本	9,328	719	719	10,766
	名古屋	18,611	551	550	19,712
	札幌	15,955	678	678	17,311
	仙台	16,512	535	535	17,582
標準語		162,243	1,292	2,573	166,108

4　実験

提案法の有効性を示すために日本語方言音声認識実験を行った.

4.1　データベース

本実験で使用するデータベースとして日本語方言音声データベース [6] と標準語音声データベースの 2 つを用いた. 方言データベースは, 青森, 広島, 熊本, 名古屋, 札幌, 仙台の 6 地方の方言から構成されており, 標準語データベースには, 学会講演と自由対話で構成されている日本語話し言葉コーパス (CSJ) [7] を用いた. 各方言と CSJ の発話数を表 1 に示す. CSJ には eval1, eval2, eval3 と 3 組のテストデータが用意されており, eval1, eval2 は学会講演, eval3 は模擬講演となっている. 本実験では eval2 を開発データ, eval1, eval3 をテストデータとして用いた. 方言データにおける方言ごとの男女比は表 2 に示すとおり偏りがない. 各方言発話は iPhone5 または XperiaZ1 を用いて収録されており, 日常会話をメインとした 7 秒程度のものとなっている. 方言の発話内容の例を表 3 に示す. 方言音声データベースのテキストおよび方言ラベルは人手で付与されている. 全データベースのサンプリング周波数は 16kHz, 量子化ビットは 16bit となっている.

Table 2　日本語方言データの方言ごとの話者数

地方	女	男
青森	36	34
広島	44	41
熊本	31	41
名古屋	43	38
札幌	44	42
仙台	47	47

Table 3　日本語方言データの発話内容例

地方	発話内容
青森	なんもわやってね
広島	いいや私はやっとらんよ
熊本	いえ私はやっとりません
名古屋	いや私はやっとらんよ
札幌	いや自分やってないわ
仙台	いやおらはやってねえ

4.2　実験条件

End-to-End 音声認識のモデルに Transformer に基づくエンコーダーデコーダーを用いた. エンコーダーブロックには $I = 8$, デコーダーブロックには $J = 6$ を設定する. 次に Transformer ブロックの条件を示す. 出力連続表現は 256 の次元, 位置ごとのフィードフォワードネットワークの内部出力は 2,048 次元, マルチヘッドアテンションのヘッド数は 4 に設定した. スピーチエンコーダーでは, 40 次元のログメルスケールフィルターバンクにデルタおよび加速係数を追加した. フレーム長は 25 ms, フレームシフトは 10 ms である. 音響特徴は, ストライドが 2 の 2 つの畳み込み層と最大プーリング層を通過したため, 時間軸に沿って 1/4 にダウンサンプリングする. テキストデコーダでは, 256 次元の単語埋め込みを使用した. 最適化には $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-9}$ の radam オプティマイザーを用いた. ミニバッチサイズは 16, Transformer ブロックのドロップアウト率は 0.1 に設定した. SpecAugment は, 2 つのマスクを使用する周波数マスキング及び時間マスキングを適用した. 周波数マスキング幅は 0〜20 の周波数ビンからランダムに選択され, 時間マスキング幅は 0〜100 フレームからランダムに選択される. ASR デコードでは, ビームサイズが 20 に設定されたビーム検索アルゴリズムを使用した. 提案法においてはさらに 6 方言及び標準語のラベルを補助特徴量として入力している. 本実験では学習及びテストどちらにおいてもラベル情報は既知のものとした. 評価指標は次式による文字誤り率 (CER) を用いた.

$$\mathrm{CER} = \left(1 - \frac{\text{文字正解率} - \text{挿入語数}}{\text{全文字数}}\right) \times 100(\%) \tag{14}$$

Table 4　学習テストによる CER (%)

		テストデータ		
	学習データ	方言のみ	標準語のみ	方言+標準語
従来法	方言のみ	52.9	-	86.2
	標準語のみ	52.5	14.3	35.4
	方言+標準語	9.3	16.6	12.5
提案法	方言+標準語	**7.1**	**13.8**	**10.1**

Table 5　学習データが方言+標準語の際の方言ごとの CER (%) と改善率 (%)

地方	従来法	提案法	改善率
青森	5.9	2.7	54.2
広島	7.5	5.8	22.7
熊本	4.0	2.2	45.0
名古屋	10.8	8.7	19.4
札幌	17.3	16.0	7.5
仙台	10.2	7.3	28.4

4.3　結果

表 4 に学習およびテストデータに用いるデータベースの条件をかえた際の CER を示す. 方言ラベルを補助特徴量として用いていないものを従来法, 用いたものを提案法としている. 表 4 の従来法の行からテストデータが方言のみの場合は方言データと標準語を合わせて学習に入れた方が方言データもしくは標準語のみで学習した時よりもはるかに CER は良くなる. ここから, 方言データのみでは学習データが足りなく音声認識システムの構築が難しいが, データ量の十分な標準語を学習に用いても方言依存性が高いことから性能が全く出ていないことがわかる. 次にテストデータが標準語のみのときをみると, 方言のみで学習したモデルを用いた場合, 大量の挿入誤りが発生することから CER が 100%以上となってしまったが, 学習データに標準語のみと方言データと標準語をあわせた場合で比較すると、標準語のみの方が CER が低いことがわかる. この結果から標準語のみのテストにおいては標準語のみで学習した方が方言による情報の分散が起きない分性能が安定していることがわかる. 次に, 提案法である方言ラベルを補助特徴量に用いた音声認識の結果 (表 4 提案法) に着目する. データの条件は従来法の方言＋標準語と同様である. 表 4 から提案法の CER は従来法のすべての条件で改善していることがわかる. 特に, 全テストデータを用いた場合誤り改善率は 19.2%にも及ぶ. これにより方言, 標準語のどちらに対しても方言ラベル情報を付与することが有用であると確認できた.

次に学習データに方言データと標準語を用いた場合の従来法と提案法それぞれの方言ごとの CER を表 5 に示す. 表 5 から青森と熊本は CER の改善率が 50%前後と CER が大幅に改善していることがわかる. 表 1 から青森と熊本はデータ数が少ないことがわかる. しかし, データ量の多い地方の改善幅が比例して小さくなるわけではない. これらの結果よりデータ量が認識性能に影響を与える可能性が考えられるため今後の調査が必要であるといえる.

5　まとめ

本論文では, 方言ラベルを補助特徴量として用いた Transformer に基づく End–to-End 音声認識を提案した. 実験結果から提案法を用いたとき方言および標準語の両方において CER が良くなることがわかった. また各方言データの CER の改善率からデータ量が CER に影響する可能性があることがわかった. 今後の課題として, データ量が認識性能に影響を与える可能性があると考えられるため調査する. またテスト音声が事前にどこの言語かわからない場合, 正しく予測することでその予測に基づいて認識できるシステムをつくることがあげられる.

参考文献

[1] S. Watanebe, et al., “Hybrid CTC/attention architecture for end-to-end speech recognition,” IEEE Journal of Selected Topics in Signal Processing, Vol. 11, No. 8, pp. 1240-1253, 2017.

[2] Y. Zhao, et al., “End-to-end-based tibetan multitask speech recognition,” IEEE Access, Vol 7, pp. 162519-162529, 2019.

[3] L. Dong, et al., “Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition,” In Proc. ICASSP, pp. 5884-5888, 2018.

[4] T. Nakatani, “Improving transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration,” In Proc. Interspeech, sep, pp. 1408-1412, 2019.

[5] R. Masumura, et al., “Sequence-Level Consistency Training for Semi-Supervised End-to-End Automatic Speech Recognition,” In Proc. ICASSP, pp. 7054–7058, 2020.

[6] 今泉遼ら, “系列分類型ニューラルネットワークを用いた日本語方言識別の検討,” 信学技報 SP2019-57, 第 119 巻, pp. 41-46, 2020.

[7] K. Maekawa, “Corpus of Spontaneous Japanese: Its design and evaluation,” In Proc. ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition, 2013.