社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

# 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

今泉　遼†　増村　亮††　塩田　さやか†　貴家　仁志†

† 首都大学東京　〒191–0065 東京都日野市旭が丘 6–6
†† 日本電信電話株式会社
E-mail: †imaizumi-ryo@ed.tmu.ac.jp

**あらまし**　ある地域特有の言語を方言といい, 入力音声がどの方言かを識別するタスクを方言識別という. 音声認識モデルの多くは標準語で作られており, 標準語のモデルを用いて方言を含む音声を認識した場合, 認識性能が大幅に低下するという問題が知られている. この問題を解決する方法の 1 つとして方言ごとの情報を学習に用い, 標準語および方言それぞれの認識器を用意することが挙げられる. 複数の認識器から入力音声に対してどの認識器を用いるかを判別するために方言識別が重要となる. また方言識別の精度が非常に高ければ, 方言固有の情報から言語モデルを最適化することにより音声認識システムの改善も期待できるため方言識別モデルの精度を向上させることは重要なタスクである. そこで本研究では英語の方言識別に有用であると報告されている系列分類型のニューラルネットワークを日本語方言識別に使用した. 実験では様々なパラメータを用いた系列分類型ニューラルネットワークを使用して, 青森, 広島, 熊本, 名古屋, 札幌, 仙台の 6 つの地域の方言の識別モデルの性能を調査, 分析した.
**キーワード**　日本語方言音声識別, 系列分類型ニューラルネットワーク, LSTM, BLSTM

# Japanese dialect speech classification using sequence-to-one neural networks

Ryo IMAIZUMI†, Ryo MASUMURA††, Sayaka SHIOTA†, and Hitoshi KIYA†

† Tokyo Metropolitan University　6–6 Asahigaoka, Hino-shi, Tokyo, 191–0065 Japan
†† Nippon Telegraph and Telephone Corporation
E-mail: †imaizumi-ryo@ed.tmu.ac.jp

**Abstract**　Dialect is a variety of language and a characteristics spoken by a particular group. Dialect identification is a task to identify dialects against input speeches. Many systems of automatic speech recognition (ASR) are constructed with a standard language. It is well-known that recognition performance of ASR is seriously reduced when input speeches are included some dialects. One solution is to use each dialect information for estimating each recognizer. In this situation prepared some recognizers, dialect identification is regarded as an important module to select a recognizer for each input speech. Additionally, in the case that dialect identification performs very well, it is expected that some information of dialect identification contribute to improve performances of ASR. Therefore, improving accuracies of dialect identification is an important task. Recently, it has been reported that English dialect identification obtained high accuracy by using sequence-to-one neural networks. Following this research, in this study, we propose Japanese dialect identification with sequence-to-one neural networks. The database of Japanese dialect contains some dialects spoken in Aomori, Hiroshima, Kumamoto, Nagoya, Sapporo and, Sendai. To construct some dialect identification systems, several parameters were used. From the results, we showed some tendencies for each dialect affected the performance.
**Key words**　Japanese dialect identification, sequence-to-one neural network, LSTM, BLSTM

## 1. はじめに

ある地域特有の言語を方言といい, 入力音声がどの方言かを識別するタスクを方言識別という. 音声認識モデルの多くは標準語で作られており, 標準語モデルを用いて方言を含む音声を認識した場合, 認識性能が大幅に低下するという問題が知られている [1]. この問題を解決する方法の 1 つとして方言の情報を学習に用い, 標準語および方言それぞれを識別できるモデルを

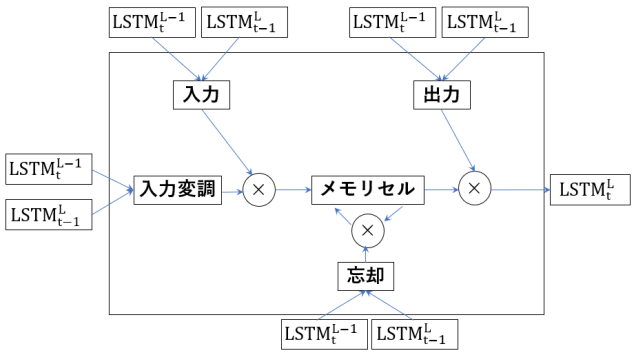


図 1　LSTM の内部構造

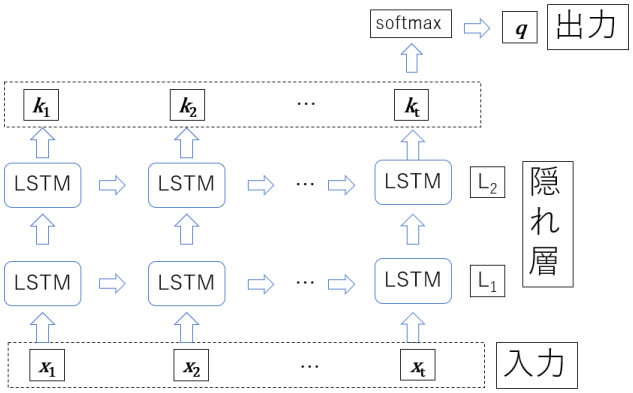


図 2　LSTM (隠れ層が 2 段の例)

用意することが挙げられる．そこで入力音声に対してどの認識器を用いるかを判別する方言識別が重要なモジュールとなる．また方言識別の精度が非常に高ければ, 方言固有の情報から言語モデルを最適化することにより音声認識システムの改善も期待できるため方言識別モデルの精度を向上させることは重要なタスクである．しかしながら方言は基となる言語が同じであるため標準語と音響特徴が似ていることが多く, 識別が難しいタスクとされていた．

近年のニューラルネットワークの発展により, 英語などの方言識別においてニューラルネットによるモデルを適用した際, 高い識別性能が得られることが報告されている [2] [3]．識別のためのニューラルネットワークのアプローチとして, ボトルネック特徴量や埋め込み層などを特徴抽出部として用いる手法 [4] [5] と, 入力から出力までを直接モデル化する手法 [6] [7] があり, どちらも活発に研究がされている．本研究では入力から出力までを直接モデル化する手法に着目し, 英語の方言識別で有効とされている Long short-term memory (LSTM) [8] [9] および注意機構付き Bidirectional LSTM (BLSTM) [10] の 2 種類の系列分類型ニューラルネットワークによる方言識別が, 日本語においても有効かを調査する．本実験ではデータセットに青森, 広島, 熊本, 名古屋, 札幌, 仙台の 6 つの地域の日本語方言を用いた．また LSTM および BLSTM のモデル学習において様々なパラメータを用いることで性能の違いを調査した．実験結果よりパラメータによって正解率が大きく変化することが分かった．また同じモデルでも方言ごとに間違いやすい方言が異なることも確認できた．

## 2.　系列分類型ニューラルネットワーク

近年, 深層ニューラルネットワークの発展にともない, 音声認識にも様々な深層学習に基づくモデル構造が提案されている．入力音響特徴量系列 $\mathbf{X} = \{\mathbf{x_1}, ..., \mathbf{x_T}\}$ から各方言ラベルに対する事後確率を推定することで, 方言識別を実現する．本研究では音声認識の分野において広く用いられている LSTM と, 注意機構付きの BLSTM の 2 つのモデル構造について述べる．

### 2.1　LSTM

ニューラルネットワークの 1 種に Recurrent Neural Network (RNN) というものがある．RNN とは文章や音声など連続なデータの情報を使えるようになったニューラルネットワークであり, 音声認識などに広く用いられている [11]．一方, LSTM は RNN の拡張とされるモデルの 1 種であり, RNN で認識が難しい長い時系列データに主に使われるモデルである．音声認識や時系列情報を用いるタスクにおいてはある時点までの音声特徴量を入力として, その時点での発話されている音素を予測する役割を果たす．LSTM の利点は RNN では出来なかった長期依存を学習できることである．長期依存なデータとはある時点の単語を予測するにあたって 1, 2 個前の単語情報では予測できず, 数十個前の単語情報を必要とすることである．LSTM は従来の RNN では不可能な数十個前の必要な単語情報を保存することを可能とするために提案された．長期記憶が可能になったことから LSTM は音声認識などのタスクで広く用いられるようになった．LSTM の基本的な構造を図 1 に示す．図 1 より LSTM の内部は入力, 出力, 入力変調, メモリセル, 忘却部の 5 つの要素で構成されている．この中のメモリセル以外はゲートとなり入力を受け入れるかの取捨選択を管理している．次に LSTM を用いた簡単なネットワーク構造を図 2 に示す [12]．ネットワーク構造の隠れ層の数は指定することが可能で使うデータに合わせて変化させることが重要となる．LSTM を用いたモデル学習では特徴量を前のフレームの特徴量に依存した隠れ表現に変換できる．$t$ 番目の入力フレーム $\boldsymbol{x_t}$ に対応する隠れ表現は式 (1) に従い計算する．

$$\boldsymbol{k_t} = \mathrm{LSTM}(\boldsymbol{x_1}, ..., \boldsymbol{x_t}; \boldsymbol{\theta_k}) \tag{1}$$

ここで LSTM() は, LSTM の関数であり, $\boldsymbol{\theta_k}$ がモデルパラメータを表す．この場合 $\boldsymbol{k_T}$ は発話全体が隠れ表現に変換されたことに相当する．出力層では, ラベルの予測確率分布 $\boldsymbol{q}$ を式 (2) に従い生成する．

$$\boldsymbol{q} = \mathrm{SOFTMAX}(\boldsymbol{k_t}; \boldsymbol{\theta_q}) \tag{2}$$

ここで SOFTMAX() はソフトマックス関数を表し, $\boldsymbol{\theta_q}$ はモデルパラメータを表す．モデル全体をまとめると,$\boldsymbol{\Theta}$ は $\{\boldsymbol{\theta_k}, \boldsymbol{\theta_q}\}$ に対応する．

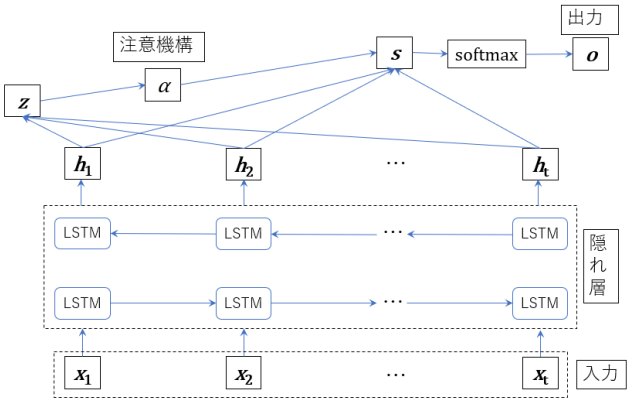


図 3　注意機構つき BLSTM (隠れ層が 1 段の例)

### 2.2　注意機構付き BLSTM

LSTM の応用の 1 つとして BLSTM が提案された. BLSTM は未来の入力から過去の出力を予測するという逆方向のレイヤを導入して, 過去と未来の両方の情報を活用できるようになったものである. また注意機構とは各隠れ表現の重要性を考慮して発話全体を連続表現にまとめるものである. BLSTM の簡潔な構造を図 3 に示す [12]. 図より LSTM の層を左から順に読み込み, 過去の情報から予測するように学習する層と右から順に読み込んで未来の情報を用いて予測する層があることでフレームの前後の情報を用いるモデルとなっている. BLSTM を用いたモデル学習では特徴量を前後のフレームに依存した隠れ表現に変換する. $t$ 番目のフレームに対応する隠れ表現は式 (3) に従い計算する.

$$\boldsymbol{h_t} = \mathrm{BLSTM}(\boldsymbol{x_1}, ..., \boldsymbol{x_T}, t; \boldsymbol{\theta_h}) \tag{3}$$

ここで, BLSTM() は BLSTM の関数であり, $\boldsymbol{\theta_h}$ はモデルパラメータを表す. 次に, 注意機構をふくめた発話全体は式 (4) から式 (6) に従う.

$$\boldsymbol{z_t} = \tanh(\boldsymbol{h_t}; \boldsymbol{\theta_z}) \tag{4}$$

$$\alpha_t = \frac{\exp(\boldsymbol{z_t^T}\bar{\boldsymbol{z}})}{\Sigma_{j=1}^{T}\exp(\boldsymbol{z_j^T}\bar{\boldsymbol{z}})} \tag{5}$$

$$\boldsymbol{s} = \Sigma_{i=1}^{n}\alpha_t \boldsymbol{h_i} \tag{6}$$

ここで, tanh() は双曲線正接関数によるアクティベーションを含む非線形関数を表し, $\boldsymbol{\theta_z}$ はそのパラメータを表す, $\bar{\boldsymbol{z}}$ は学習可能なコンテキストベクトルであり, 隠れ表現の重要性を測るために用いられる. そして $\alpha_t$ は隠れ表現の重要性を重みにする. 出力層では, ラベルの予測確率分布を式 (7) に従い生成する.

$$\boldsymbol{o} = \mathrm{SOFTMAX}(\boldsymbol{s}; \boldsymbol{\theta_o}) \tag{7}$$

モデル全体をまとめると, $\boldsymbol{\Theta}$ は $\{\boldsymbol{\theta_h}, \boldsymbol{\theta_z}, \bar{\boldsymbol{z}}, \boldsymbol{\theta_o}\}$ に対応する.

## 3.　日本語音声の方言識別とデータベース

方言識別とは入力音声がどの地方の方言かを識別するタスクである. これまでにも, 英語の方言 (イギリス英語, アメリカ英語, オーストラリア英語など) の識別に関する研究がされており, ニューラルネットによるモデルを適用した際に高い識別性能が得られることが報告されている. 本研究では, これまでに報告があまりされていない日本語音声の方言識別のために 2 章で述べた系列分類型ニューラルネットワークを用いることを検討した.

本実験で用いる日本語方言のデータベースについて述べる. 含まれる方言は青森, 広島, 熊本, 名古屋, 札幌, 仙台の 6 地方となっており, 各方言の発話数は表 1 に示す通りである. 表 1 より, 広島と名古屋の学習及び開発夜の発話数が多く, 青森と熊本は少ないことがわかる. また, 表 2 は各方言の発話者数を示している. 表 2 より男女の偏りがないことがわかる. 各発話は iPhone5 と XperiaZ1 の 2 つのスマートフォンを用いて収録されており, 実際の発話内容は日常会話をメインとした 7 秒程度のものとなっている. 収録発話の例を表 3 に示す. またデータベースの正解テキストおよび正解ラベルは人手で付与されている.

表 1　日本語方言データベースの方言ごとの発話数の内訳

| | 学習発話数 | 開発発話数 | テスト発話数 | 全体 |
|---|---|---|---|---|
| 青森 | 5,188 | 538 | 1,352 | 7,078 |
| 広島 | 8,787 | 934 | 1,133 | 10,854 |
| 熊本 | 4,450 | 467 | 1,438 | 6,355 |
| 名古屋 | 8,801 | 931 | 1,102 | 10,834 |
| 札幌 | 7,528 | 798 | 1,356 | 9,682 |
| 仙台 | 7,907 | 826 | 1,070 | 9,803 |
| 全体 | 42,661 | 4,494 | 7,451 | |

表 2　日本語方言データの方言ごとの話者数

| | 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 |

表 3　日本語方言データの発話内容

| | 発話内容 (例) |
|---|---|
| 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.　実　　験

日本語方言識別を行うために 3 章で述べた日本語方言音声データベースを用いて識別実験を行った.

表 4　LSTM を用いてモデル化した際の取得したパラメータ数

| | | 隠れ層の数 | |
|---|---|---|---|
| LSTM の次元数 | 入力特徴量の次元数 | 2 | 4 |
| 256 | 80 | 873,990 | 1,926,662 |
| | 240 | 1,037,830 | 2,090,502 |
| | 400 | 1,201,670 | 2,254,342 |
| 512 | 80 | 3,320,838 | 7,523,334 |
| | 240 | 3,648,518 | 7,851,014 |
| | 400 | 3,976,198 | 8,178,694 |

表 5　BLSTM を用いてモデル化した際の取得したパラメータ数

| | | 隠れ層の数 | |
|---|---|---|---|
| LSTM の次元数 | 入力特徴量の次元数 | 2 | 4 |
| 256 | 80 | 677,894 | 1,468,422 |
| | 240 | 841,734 | 1,632,262 |
| | 400 | 1,005,574 | 1,796,102 |
| 512 | 80 | 2,535,430 | 5,689,350 |
| | 240 | 2,863,110 | 6,017,030 |
| | 400 | 3,190,790 | 6,344,710 |

表 6　LSTM を用いてモデル化した際の ACC (%)

| | | 隠れ層の数 | |
|---|---|---|---|
| LSTM の次元数 | 入力特徴量の次元数 | 2 | 4 |
| 256 | 80 | 68.6 | **74.6** |
| | 240 | 62.5 | 73.0 |
| | 400 | **71.5** | 65.1 |
| 512 | 80 | **75.2** | **72.4** |
| | 240 | 65.1 | 69.6 |
| | 400 | 71.9 | 67.5 |

表 7　BLSTM を用いてモデル化した際の ACC (%)

| | | 隠れ層の数 | |
|---|---|---|---|
| LSTM の次元数 | 入力特徴量の次元数 | 2 | 4 |
| 256 | 80 | **71.4** | 68.5 |
| | 240 | 69.6 | 68.0 |
| | 400 | 69.9 | **72.6** |
| 512 | 80 | 66.0 | 67.2 |
| | 240 | 73.9 | 65.7 |
| | 400 | **74.0** | **72.0** |

### 4.1 実験条件

データベースには 3 章で述べたものを使用し, 音響特徴量には 80 次元のメルフィルタバンクを用いている. モデル化には 2 章で述べた LSTM と注意機構付き BLSTM の 2 種類のモデル構造を用い, システムを構築する際にいくつかのパラメータを変更して実験を行った. その際, ミニバッチサイズは 16, ドロップアウトの割合は 0.2 と固定した. 次に実験において変更したパラメータについて説明する. 入力の次元数では特徴量を 80 次元メルフィルタバンクであるため 1 フレーム 80 次元であるが, 前後のフレームをつなぐことで入力の次元数を 80, 240, 400 と変化させた. 前後のフレームをつないだ時はサブサンプリングするフレーム数を変化させてオーバーラップが起きないようにした. 次に隠れ層の次元数 ($D$) は 256, 512 のどちらかに指定した. 2 章で述べたように隠れ層の数も指定可能なため, 本実験では層数を 2 または 4 とした. また隠れ層では学習のときにパラメータを計算して取得する. LSTM 関数が取得するパラメータ $g_1$ は入力に対しての重み + 隠れ状態に対しての重み + バイアスとなっており, 以下の式 (8) で表される.

$$g_1 = \text{入力次元数} * D * 4 + D * D * 4 + D * 2 * 4 \quad (8)$$

このとき 4 は LSTM の 2 つの入力ゲート, 忘却ゲート, 出力ゲートを表している. これは 1 段目の計算式であり 2 段目以降は式 (9) で表される.

$$g_2 = D * D * 4 * 2 + D * 2 * 4 \quad (9)$$

同様に BLSTM が取得するパラメータ $f_1$ は次の式 (10) で表される.

$$f_1 = \text{入力次元数} * \frac{D}{2} * 4 + \frac{D}{2} * \frac{D}{2} * 4 + \frac{D}{2} * 2 * 4 \quad (10)$$

LSTM との違いは隠れ層の次元数が 1/2 されていることであるが, これは LSTM が 1 層なのに対して, BLSTM はある時点までの過去の情報とある時点からの未来の情報を用いる 2 層で 1 層とするための措置である. 注意機構付き BLSTM によるモデル化はこれに加えて注意機構層が含まれる. 各隠れ表現の重要性を考慮して重みを BLSTM の出力に加える. 注意機構層においてパラメータ $a$ を取得する計算式は式 (11) に示す.

$$a = D * D + D * 2 \quad (11)$$

blstm における 2 段目以降は式 (12) で表される.

$$f_2 = \frac{D}{2} * \frac{D}{2} * 4 + \frac{D}{2} * \frac{D}{2} * 4 + a + \frac{D}{2} * 2 * 4 \quad (12)$$

最後に出力層で SOFTMAX 関数を用いてラベル予測確率分布を生成する. SOFTMAX 層の取得パラメータ $m$ は式 (13) で表す.

$$m = \text{ラベル数} * D + \text{ラベル数} \quad (13)$$

各条件におけるパラメータの合計数を表 4, 5 に示す. 最適化手法には adam を用いた. 学習において開発データに対するロスが 5 回連続で改善しないときアーリーストッピングした. 評価の際には学習したモデルにテスト発話を入力してラベルを予測し, その予測したラベルが正解ラベルといくつ一致しているかを正解率 (ACC (%) ) として算出した. ACC は式 (14) で定義される.

$$\text{ACC}(\%) = \frac{\text{正解発話}}{\text{全テスト発話数}} * 100 \quad (14)$$

### 4.2 実験結果

実験条件で示したパラメータを用いて, 方言識別のための LSTM および BLSTM を用いたシステムを構築した結果の

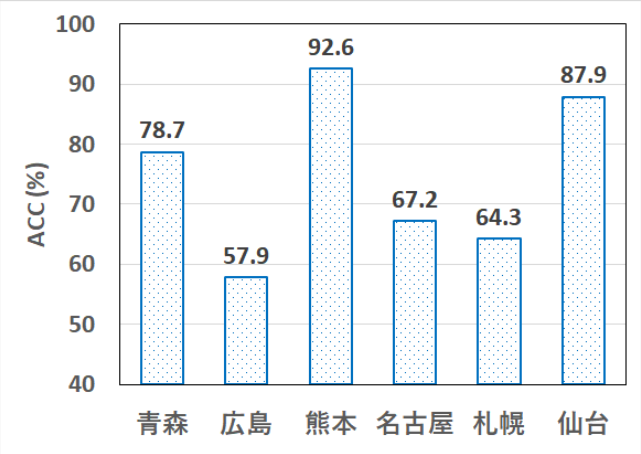


図 4　LSTM の中で最も ACC が高いモデル化における方言ごとの ACC (LSTM の次元数 512, 入力特徴量の次元数 80, 隠れ層の数 2)

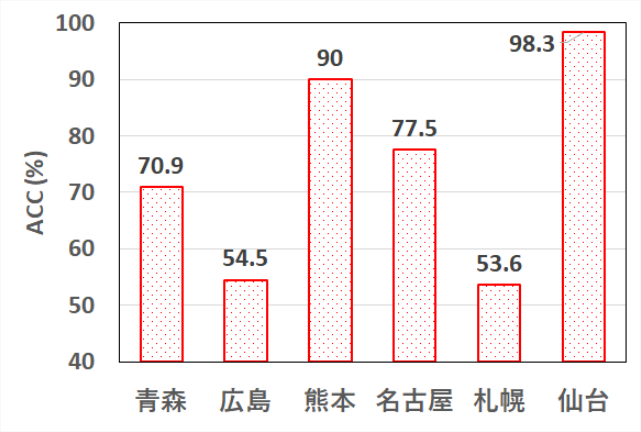


図 5　BLSTM の中で最も ACC が高いモデル化における方言ごとの ACC (LSTM の次元数 512, 入力特徴量の次元数 400, 隠れ層の数 2)

ACC を表 6, 7 に示す．表の太字はパラメータを変化させた各区分で最も高い ACC を表している．LSTM, BLSTM の両方において LSTM の次元数 256, 隠れ層の数 2 のときだけ傾向は違うが, LSTM において他の区分は入力特徴量の次元数が小さいほうが ACC が高い．反対に BLSTM では入力特徴量の次元数が高いほうが ACC が高いことがわかる．先に述べたように BLSTM は LSTM の応用であり, 前後のフレームから情報を取得して現在の情報を予測できるという利点がある．そのため入力の次元数が 400 と大きくても情報を有効活用して適切に学習することが可能であったと考えられる．一方 LSTM では, BLSTM よりもモデル構造が単純であるため入力次元数が低いほうが方言の特徴を頑健にとらえるという傾向があると考えられる．次に LSTM, BLSTM のそれぞれで最も高い ACC が示された条件において方言ごとの ACC を図 4, 5 に示す．図 4, 5 どちらにおいても熊本と仙台は ACC が高く, 広島と札幌は ACC

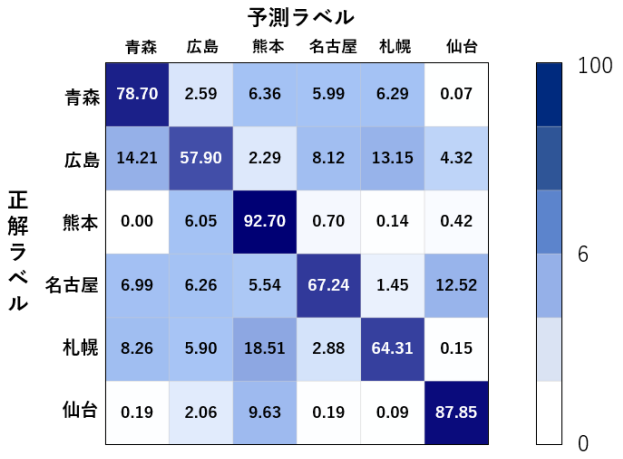


図 6　LSTM で最も ACC が高い条件の際に認識した発話が選択した方言の割合

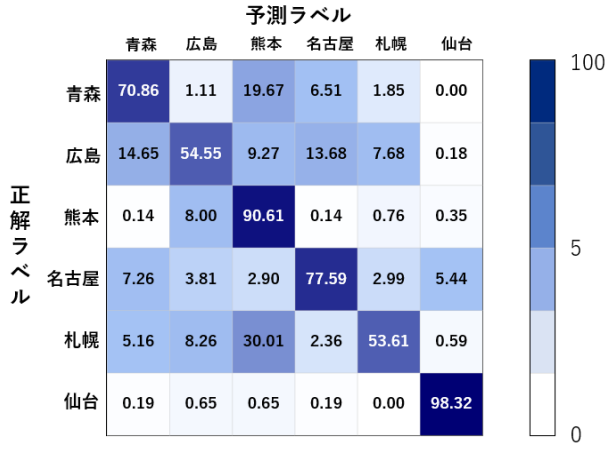


図 7　BLSTM で最も ACC が高い条件の際に認識した発話が選択した方言の割合

が低いことがわかる．このことから方言によって識別のしやすさが異なることが確認できた．そこでさらに識別の傾向を調査するために, LSTM および BLSTM それぞれの ACC が最も高かった条件において予測ラベルと正解ラベルのコンフュージョンマトリックスを図 6, 7 に示す．図 6 より LSTM では誤認識した発話の中で正解ラベルが札幌の発話を熊本だと間違えたものが 18.51% と高い値になっている．同様に図 7 の BLSTM においても正解ラベルが札幌の発話を熊本だと誤認識した割合が 30.01% と高くなっており, どちらのモデルにおいても札幌は熊本に間違いやすい方言であると考えられる．さらに BLSTM のモデルでは正解ラベルが青森の発話を熊本と誤認識した割合が 19.67% となっており, LSTM のときの誤認識した割合と比較するとおよそ 3 倍になっている．このことから BLSTM は LSTM に比べ青森の方言を熊本の方言だと認識しやすいことがわかる．また図 6, 7 両方のモデルにおいて予測ラベルが熊本の列の色が全体的に濃くなっていることから LSTM および

BLSTM のモデルで他の方言に比べ熊本と予測された発話が全体的に多いことがわかる. 以上より, どちらのモデルもテスト発話を熊本と認識しやすい傾向にあり, その結果, 熊本の ACC が高くなったと考えられる. 一方, 両方のモデルで ACC が低い広島の誤り方をみると, 図 6 より LSTM では青森や札幌と認識した割合が 13% 以上と高く, 図 7 の BLSTM は青森や名古屋と認識した割合が 13% 以上と高い. 広島以外の方言は熊本と認識しやすい傾向にあるが, 広島の方言は他の方言と比べ熊本と認識されることが比較的少ないが逆に他の方言に間違える割合が高く, その結果, 広島の全体の ACC が低くなっていると考えられる.

## 5. ま と め

本研究では系列分類型ニューラルネットワークに様々なパラメータを用いて日本語方言識別を行い, 識別正解率や方言ごとの識別精度の違いを調査した. またどの方言がどの方言に間違いやすいかという傾向についても調査した. 実験結果より, LSTM, BLSTM どちらのモデルにおいても方言ごとの ACC にはばらつきがあることが分かり, 特に熊本に判定しやすい傾向があることが確認できた.

今後の課題としては, より大きなデータを用いた評価や, 他の地方の方言を加えて識別ラベルを増やした場合, 本実験で調査したモデルにどのような影響を及ぼすか確認することなどが挙げられる.

### 文 献


[1] Xiaoxiao Miao and Ian McLoughlin. Lstm-tdnn with convolutional front-end for dialect identification in the 2019 multi-genre broadcast challenge. *arXiv preprint arXiv:1912.09003*, 2019.

[2] Thomas Purnell, William Idsardi, and John Baugh. Perceptual and phonetic experiments on american english dialect identification. *Journal of language and social psychology*, Vol. 18, No. 1, pp. 10–30, 1999.

[3] Omar F Zaidan and Chris Callison-Burch. Arabic dialect identification. *Computational Linguistics*, Vol. 40, No. 1, pp. 171–202, 2014.

[4] David Snyder, Daniel Garcia-Romero, Alan McCree, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur. Spoken language recognition using x-vectors. In *Odyssey*, pp. 105–111, 2018.

[5] Maxim Tkachenko, Alexander Yamshinin, Nikolay Lyubimov, Mikhail Kotov, and Marina Nastasenko. Language identification using time delay neural network d-vector on short utterances. In *International Conference on Speech and Computer*, pp. 443–449. Springer, 2016.

[6] Wang Geng, Wenfu Wang, Yuanyuan Zhao, Xinyuan Cai, Bo Xu, Cai Xinyuan, et al. End-to-end language identification using attention-based recurrent neural networks. 2016.

[7] Weicheng Cai, Danwei Cai, Shen Huang, and Ming Li. Utterance-level end-to-end language identification using attention-based cnn-blstm. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5991–5995. IEEE, 2019.

[8] Haşim Sak, Andrew Senior, and Françoise Beaufays. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *arXiv preprint arXiv:1402.1128*, 2014.

[9] Suman Ravuri and Andreas Stolcke. A comparative study of recurrent neural network models for lexical domain classification. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6075–6079. IEEE, 2016.

[10] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pp. 1480–1489, 2016.

[11] Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Eleventh annual conference of the international speech communication association*, 2010.

[12] 増村亮, 井島勇祐, 浅見太一, 政瀧浩和, 東中竜一郎. 音声認識誤りに頑健なニューラル発話意図推定のためのコンフュージョンネットワークの連続表現. 人工知能学会全国大会論文集 第 32 回全国大会 (2018), pp. 3G204–3G204. 一般社団法人 人工知能学会, 2018.