Vol.2019-SLP-129 No.2
2019/10/11

音声波形を入力とする単語単位End-to-End音声認識

上乃 聖[1,a)]　三村 正人[1]　坂井 信輔[1]　河原 達也[1]

概要：単語単位 End-to-End 音声認識は簡潔な構造で非常に高速な認識ができ，高い性能を達成している．現在，単語単位 End-to-End 音声認識を実現する上において入力の音響特徴量として用いられるのは対数メルフィルタバンク特徴量であるが，音声波形から対数メルフィルタバンクへの変換の際に，認識に有用な情報を損失している可能性がある．本研究では，より単語単位 End-to-End 音声認識に合致した特徴量抽出のために，CNN を用い，音声波形を入力とし，特徴量抽出から単語単位認識までを 1 つのネットワークで行う手法を提案する．単語基準の損失関数を特徴量抽出まで誤差逆伝播を行うことでより適合した特徴量抽出を行うことを期待する．また，本モデルではモデル内の設定により窓幅やシフト幅を任意に変更することが可能である．そのため，より多様な特徴量抽出を行うために，提案モデルを拡張し，複数の窓の設定での特徴量抽出を用意するモデルも提案する．実験により，提案手法は従来の対数メルフィルタバンクと同等の結果を示し，対数メルフィルタバンクとは異なる特徴量抽出を行なっていることを示した．

1. はじめに

End-to-End 音声認識は音響特徴量を直接記号系列に変換するシステムであり，非常に簡潔な構造で構築が容易である．End-to-End 音声認識の実現方法として，Connectionist Temporal Classification (CTC) を用いた手法 [1] や，RNN トランデューサ [2] や注意機構モデルを用いた sequence-to-sequence (seq2seq) モデル [3] などが挙げられる．これらの手法は HMM などの潜在状態遷移モデルを必要とせずに音響特徴量を記号系列に変換することができる．End-to-End 音声認識の出力単位に関しては，音響特徴量から単語系列を直接出力する単語単位音声認識モデル [4] が外部デコーダなどを必要としないため，特に高速な認識を実現できる．

現状，単位音声認識モデルを実現するために用いられる入力は対数メルフィルタバンク特徴量であり，特徴量抽出を行う機構では単語基準の損失を逆伝搬を行うことができないため，音声波形から特徴量抽出の際に認識に有用な情報を損失している可能性がある．そのため，音声波形を入力とし，CNN を用いて特徴量抽出を行い，音声認識を実現する研究はいくつか行われている．Tjandra らは CNN を用いて特徴量抽出を行うモデルを提案し，初期学習として対数メルフィルタバンクとの二乗距離を取ることで文字単位の音声認識を実現している [5]．Ravanelli らは SincNet と呼ばれるバンドパスフィルタをベースに CNN のフィルタを学習するモデルを音声認識に用いている [6]．また，Sainath らは時間ドメイン，周波数ドメインに CNN のフィルタを分けて学習するモデルを提案している [7]．Zeghidour らは音声波形を入力とし，CNN の学習を行うフロントエンドを用いて音声波形から文字単位音声認識の学習を一括で行うモデルを提案している [8]．これらのモデルは音素や文字単位にとどまっている．

そこで本研究では，一つのモデルで音声波形から単語系列まで出力するモデル化を行い，適切な特徴量抽出を単語基準で学習するモデルを提案する．本モデルで学習を行うことで，単語基準の損失を特徴量抽出部まで伝搬することができる．また，提案モデルでは任意の窓幅・シフト幅を設定することができるため，提案モデルを拡張し，1 つのモデルの中に複数の窓の設定をもモデルを設計することで，より異なる特徴量抽出を行えることを期待する．

2. End-to-End 音声認識

本研究で提案する音声波形を入力とする音声認識モデルでは一括で音声波形から単語系列を学習するが，特徴量抽出を行う機構と音声認識を行う機構に分割する。本研究では，音声認識を行う機構は注意機構モデルを用いる．

2.1 注意機構モデル

注意機構を用いたモデルはエンコーダとデコーダの 2 つのネットワークから構成される．エンコーダでは LSTM を用いて音響特徴量系列を分散表現にする．デコーダでは

[1] 京都大学情報学研究科
Graduate School of Informatics, Kyoto University, Sakyo-ku, Kyoto 606–8501, Japan
[a)] ueno@sap.ist.i.kyoto-u.ac.jp

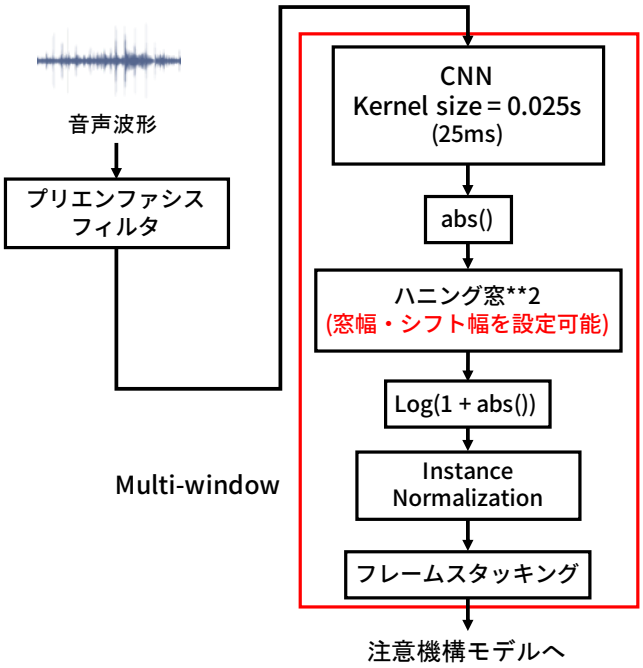

図 1　提案手法の特徴量抽出部の概念図．複数窓を持つ場合は赤枠部分を複数用意し，統合を行うことで統合する．

エンコードされた系列表現と出力記号表現との関連性を考慮して出力記号系列を生成する。本研究ではエンコーダに複数層の双方向 LSTM を用い，デコーダには 1 層の単方向 LSTM，注意機構の計算は [9] をもとに行う．デコーダに LSTM を用いることで前の記号列をもとに次の記号列を予測する．これは言語モデルの構造が注意機構モデルは含まれているとみなすことができる．損失関数は予測記号系列と正解記号系列とのクロスエントロピを用いる．学習時には，デコーダの LSTM の入力に用いる前の記号系列は正解データを用いるが，認識時には予測した記号系列を用い，その系列をもとにビームサーチを行う．

2.2　単語単位 End-to-End 音声認識

End-to-End 音声認識は正解記号系列を選択することができ，もっとも高速に認識を実現できるモデルが，単語系列を出力する単語単位モデルである．音声認識の最終的な目標である単語がそのまま出力できるため，外部機構の処理を一切用いずに非常に簡潔で高速な認識が可能となる．また，損失関数としては正解単語系列とのクロスエントロピとなるため，その誤差を伝搬することで，単語基準の学習が可能になる．

3.　提案手法

3.1　CNN を用いた特徴量抽出

特徴量抽出部は Zeghidour らのモデル [8] をベースに対数メルフィルタバンク特徴量の処理を模したモデルを用いる．音声波形を入力とし，プリエンファシスフィルタで初期化した CNN を構成し，その後に 25ms のフィルタ幅・1

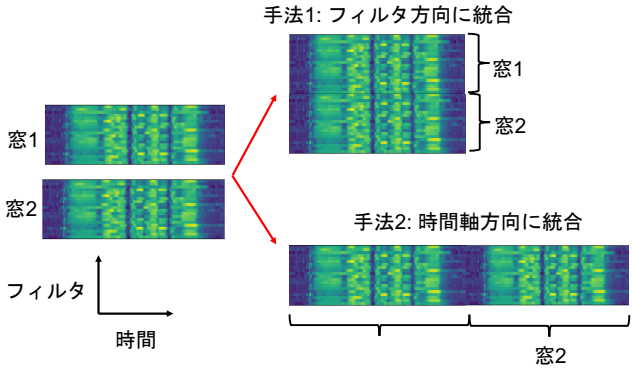

図 2　統合手法の概念図

サンプル分のストライドの複数枚のフィルタをもつ CNN を適用する．この CNN の処理は時間軸上で用いられるが，対数メルフィルタバンク特徴量におけるメルフィルタバンクに対応する．その CNN から出力された波形に対して，正の値になるようにした後に，ハニング窓の二乗を窓関数として適用する。対数化の後，各フィルタの出力ごとに正規化を行う instance normalization [10] を行うことで，特徴量抽出を行う．最終的に出力される特徴量は対数メルフィルタバンクとほぼ同じ次元をもつ特徴量となる．この音響特徴量にフレームスタッキング [11] を適用し，オーバーラップのない 3 フレーム分の音響特徴量を単語単位音声認識モデルの入力とする．図 1 に使用する特徴量抽出部の図を示す．ここで学習するのはプリエンファシスフィルタで初期化した CNN，その後段の CNN である．

3.2　複数窓を持つ特徴量抽出

より多様な特徴量抽出のためには，1 つの窓幅ではなく，複数の窓幅・シフト幅から獲得された特徴量の方が良いと考えられる．3.1 節のハニング窓は窓幅・シフト幅を調整することができるため，異なる窓幅・シフト幅をもつハニング窓を用意することで複数窓をもつ特徴量抽出を実現する (図 1，赤枠)．最終的に統合することで注意機構の入力として用いるが統合には以下の 2 つの手法を考えることができる．

手法 1. フィルタ方向に統合する
手法 2. 時間軸方向に統合する

図 2 に統合手法の概念図を示す．手法 1 では，シフト幅が異なる場合に，統合した特徴量がそれぞれ別の時間フレームを参照してしまうため，シフト幅を複数設計することができない．手法 2 は時間軸方向に対して統合するため，シフト幅が異なる特徴量でも設計可能である．

4.　評価実験

4.1　データセット

本研究では『日本語話し言葉コーパス』(CSJ) を用いる。CSJ は CSJ-APS と CSJ-SPS の 2 つのサブコーパスで構成

表 1 『日本語話し言葉コーパス』(CSJ) 中の学会講演 (APS) テストセットにおける単語誤り率 (%)．ベースラインシステムでは 40 次元の対数メルフィルタバンク特徴量を用いて単語単位モデルを学習．

	WER(%)
ベースラインシステム	12.10
単一窓 [窓幅 25ms, シフト幅 10ms]	11.86
単一窓 [窓幅 50ms, シフト幅 10ms]	11.81
単一窓 [窓幅 50ms, シフト幅 20ms]	12.66
複数窓 (フィルタ方向に結合) [窓幅 25ms，シフト幅 10ms + 窓幅 50ms，シフト幅 10ms]	**11.66**
複数窓 (時間軸方向に結合) [窓幅 25ms，シフト幅 10ms + 窓幅 50ms，シフト幅 10ms]	12.78
複数窓 (時間軸方向に結合) [窓幅 25ms，シフト幅 10ms + 窓幅 25ms，シフト幅 20ms]	12.10
複数窓 (フィルタ方向に結合) [窓幅 12.5ms，シフト幅 10ms + 窓幅 25ms，シフト幅 10ms + 窓幅 50ms，シフト幅 10ms + 窓幅 125ms，シフト幅 10ms]	11.87

されている。CSJ-APS は学会講演を収録したコーパスで，訓練データは 247.9 時間のデータで構成される。CSJ-SPS は 3 つのテーマでスピーチを行った模擬講演コーパスで，訓練データは 281 時間のデータで構成される．それぞれのサブコーパスでテストセットが提供されており，本研究ではテストセット 1 (CSJ-APS) とテストセット 3 (CSJ-SPS) を使用する．語彙には 2 回以上出現した単語と ⟨sos⟩, ⟨eos⟩, ⟨UNK⟩ といった特殊なラベルを使用する．語彙サイズは APS では 19,146，SPS では 24,286 である．

4.2 システム構成

4.2.1 特徴量抽出部

提案するモデルの特徴量抽出部の入力として入力は 16kHz のサンプリングレートの音声波形を用いる．入力する前に，1 発話に対して平均と分散を計算し，平均を 0，分散を 1 になるように計算する．CNN の出力フォルタ枚数は単一窓であっても，複数窓であっても，合計が 40 になるように設定する．例えば，フィルタ方向に統合する複数窓を用いるモデルの場合，1 つ目の設定の窓からは 20 枚，2 つ目の設定の窓からは 20 枚を出力するようにし，統合することで合計の出力フィルタ枚数を 40 枚にする．

4.2.2 単語単位音声認識モデル

特徴量抽出部の処理を終えたのちに，注意機構モデルをベースとし，単語単位音声認識を実現する．エンコーダは 5 層の 320 次元の隠れ層を持つ双方向 LSTM で構成する．また，ドロップアウトを 0.2 に設定し，各双方向 LSTM に適用する．注意機構を用いたデコーダは，1 層の 320 次元の隠れ層を持つ単方向 LSTM で構成し，その後出力単語数分の単語数のノードを持つ softmax の出力層となる．最適化アルゴリズムは Adam [12] を用い，Gradient Clipping の閾値を 5.0 とした．正則化のためにラベルスムージング [13] を用いる．また，認識時のビーム幅は 4 とした．これらは PyTorch を用いて実装されている [14].

表 2 『日本語話し言葉コーパス』(CSJ) 中の学会講演 (APS) テストセットと模擬講演 (SPS) における単語誤り率 (%)．ベースラインシステムでは 40 次元の対数メルフィルタバンク特徴量を用いて単語単位モデルを学習．[] 内は [窓幅 (ms), シフト幅 (ms)] を示す．

	APS	SPS
ベースラインシステム	12.10	9.69
単一窓 [50, 10]	11.81	9.08
複数窓 (フィルタ方向に結合) [25，10 + 50，10]	11.66	9.27

4.3 結果

表 1 に CSJ-APS に対しての音声認識結果を示す。提案手法はほとんどがベースラインシステムである 40 次元の対数メルフィルタバンクで学習されたモデルと同等の結果を達成した．単一窓に関しては，窓幅はほとんど性能に影響しないが，シフト幅は 10ms の方が良いという結果になった．2 つの窓を用いた場合は出力フィルタ方向で複数窓幅のフィルタから出力された特徴量を統合する手法が一番良いという結果になった．手法 1 に関して，フレームスタッキングの係数をシフト幅ごとに変えることにより，統合される特徴量の時間フレームを合致させる手法も試したが，非常に悪い結果となった．時間軸に対して特徴量を統合する手法 2 に関してはシフト幅を変更した方が良いという結果になったが，図 3 の上図のようにシフト幅 20ms の特徴量が注意機構の計算で無視されてしまっているため，抽出される特徴量が単一窓と変わらない．図 3 の下図のように同一シフト幅で行なったものに関しては 2 つの窓幅から出力された特徴量に対して，注意機構の計算時点で確率が割り振られていることがわかる．しかし，時間軸方向に

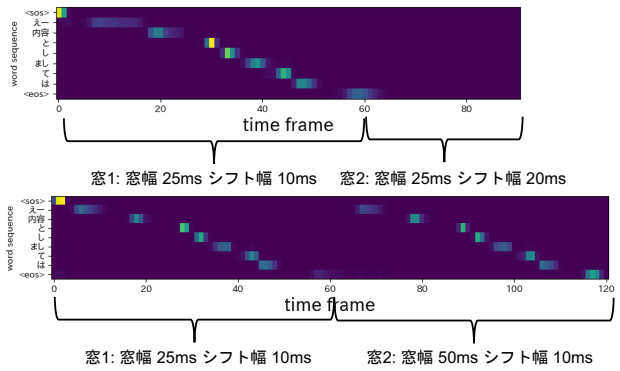

図 3 上図: 複数窓 (時間軸方向に結合) [窓幅 25ms，シフト幅 10ms + 窓幅 50ms，シフト幅 10ms] の注意機構の重み．下図: 複数窓 (時間軸方向に結合) [窓幅 25ms，シフト幅 10ms + 窓幅 50ms，シフト幅 10ms] の注意機構の重み．

統合しているため，BiLSTM の計算の時系列が窓 1 から窓 2 に変わる際に時系列が戻ってしまうという問題点があり，デコードがうまくいかないことがあった．また，本実験では窓の設定を 4 つまで増やしたが効果は見られなかった．表 2 に CSJ-SPS の単語誤り率も含めた結果を示す．単一窓でも，複数窓でも APS の結果同様，ベースラインシステムと同程度の性能を示した．

図 4 に特徴量抽出部で学習されたフィルタの周波数応答を示す．学習されたフィルタはいずれもメルフィルタバンク (上図) と異なり，1 フィルタごとに広範囲の周波数を見るものが多いことがわかる．しかし，メルフィルタバンクと同様に低周波帯域に集中していることがわかる．複数窓では各窓幅のフィルタが低周波から高周波まで見ていることがわかる．

5. おわりに

本研究では音声波形を入力とし，単語を出力するモデルを提案し，また単一の窓の設定ではなく，複数の窓の設定を行うことで，より多くの種類の特徴量抽出を行うモデルも提案した．『日本語話し言葉コーパス』による実験により，従来用いられている対数メルフィルタバンク特徴量を入力とするモデルと同等の結果を得た．今後の予定としては提案モデルを拡張し，複数のエンコーダを用いた手法や，マルチチャネル化などを目指す．

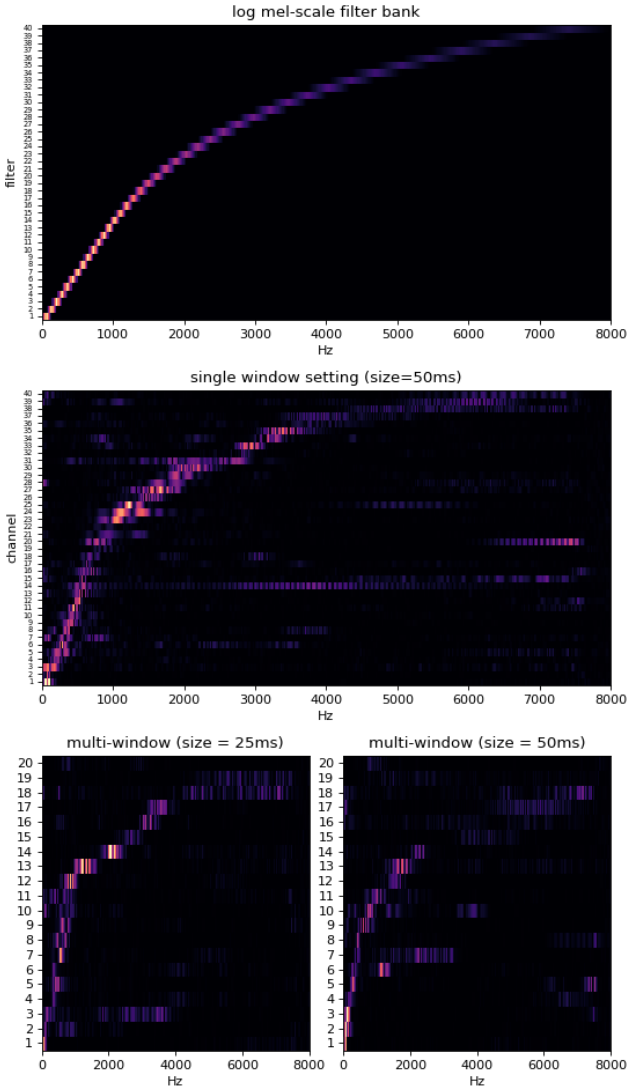

図 4 対数メルフィルタバンクと学習されたフィルタの図示．上図: 対数メルフィルタバンク，中央図: 単一窓により学習されたフィルタ，下図: 複数窓により学習されたフィルタ．

参考文献

[1] Graves, A., Fernandez, S., Gomez, F. and Schmidhuber, J.: Connectionist Temporal Classification : Labelling Unsegmented Sequence Data with Recurrent Neural Networks, *Proc ACM*, pp. 369–376 (2006).
[2] Graves, A.: Sequence transduction with recurrent neural networks (2012).
[3] Battenberg, E., Chen, J., Child, R., Coates, A., Gaur Yi Li, Y., Liu, H., Satheesh, S., Sriram, A. and Zhu, Z.: Exploring neural transducers for end-to-end speech recognition, *Proc. ASRU*, pp. 206–213 (2017).
[4] Sak, H., Senior, A., Rao, K., Irsoy, O., Graves, A., Beaufays, F. and Schalkwyk, J.: Learning acoustic frame labeling for speech recognition with recurrent neural networks, *Proc. ICASSP*, pp. 4280–4284 (2015).
[5] Tjandra, A., Sakti, S. and Nakamura, S.: Attention-based wav2text with feature transfer learning, *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, IEEE, pp. 309–315 (2017).
[6] Ravanelli, M. and Bengio, Y.: Speech and speaker recognition from raw waveform with sincnet, *arXiv preprint arXiv:1812.05920* (2018).
[7] Sainath, T. N., Weiss, R. J., Senior, A., Wilson, K. W. and Vinyals, O.: Learning the speech front-end with raw waveform CLDNNs, *INTERSPEECH* (2015).
[8] Neil, Z., Qiantong, X., Vitaliy, L., Nicolas, U., Gabriel, S. and Ronan, C.: Fully Convolutional Speech Recognition, *arXiv preprint arXiv:1812.06864* (2018).
[9] Chorowski, J., Bahdanau, D., Serdyuk, D., Cho, K. and Bengio, Y.: Attention-Based Models for Speech Recognition, *Proc. NIPS*, pp. 577–585 (2015).
[10] Ulyanov, D., Vedaldi, A. and Lempitsky, V.: Instance

normalization: The missing ingredient for fast stylization, *arXiv preprint arXiv:1607.08022* (2016).

[11] Sak, H., Senior, A., Rao, K. and Beaufays, F.: Fast and Accurate Recurrent Neural Network Acoustic Models for Speech Recognition, *INTERSPEECH*, pp. 1468–1472 (2015).

[12] Kingma, D. P. and Ba, J.: Adam: A Method for Stochastic Optimization, *arXiv preprint, 1412.6980*, pp. 1–15 (2014).

[13] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. and Wojna, Z.: Rethinking the inception architecture for computer vision, *Proc. CVPR*, pp. 2818–2826 (2016).

[14] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. and Lerer, A.: Automatic differentiation in PyTorch, *NIPS-W* (2017).