Vol.2021-NL-251 No.19
Vol.2021-SLP-139 No.19
2021/12/2

End-to-End 音声認識のための 粒度の異なるサブワード単位に基づく階層的な条件づけ

樋口 陽祐[1,a)]　軽部 敬太[1]　小川 哲司[1]　小林 哲則[1]

概要：End-to-End 音声認識では，単語を推定するのに適した特徴表現が暗黙的に獲得されることを期待している．しかし，入力の音声信号と出力の単語列では，情報の抽象度が大きく異なるため，目的の特徴表現を End-to-End に学習するのは困難である．本研究では，End-to-End 音声認識において，単語単位の特徴表現を効果的に学習するために，Connectionist Temporal Classification (CTC) に基づいた階層的条件付きモデルを提案する．提案モデルでは，最終層に加えて，複数の中間層に対して CTC 損失を適用し，各 CTC における出力単位の粒度を最終層に向けて段階的に高くする．このとき，粒度の低い単位による予測によって粒度の高い単位による予測を条件付けることで，単語単位の系列に対する生成過程を明示的に学習する．言語情報の抽象度が段階的に組み上がるモデルを構築することで，単語単位の特徴表現が効果的に学習されることを期待する．LibriSpeech-{100h, 960h} と TEDLIUM2 を用いた実験において提案モデルを評価したところ，既存モデルよりも高い認識性能を与えることが明らかとなった．また，詳細な分析の結果，提案モデルによって単語単位の認識に適した特徴表現が学習できることを確認した．

キーワード：End-to-End 音声認識，connectionist temporal classification，acoustic-to-word モデル

End-to-End Speech Recognition with Multi-Granular Subword Units and Hierarchical Conditioning Mechanism

Yosuke Higuchi[1,a)]　Keita Karube[1]　Tetsuji Ogawa[1]　Tetsunori Kobayashi[1]

1. はじめに

End-to-End 音声認識とは，単一の深層ニューラルネットワーク（Deep Neural Network; DNN）を用いて，音声の特徴量列から目的の記号列への変換を直接モデル化する枠組みである [1–3]．従来の音響モデル，発音辞書，言語モデルを組み合わせたシステム [4] に比べ，End-to-End なシステムは構築が容易であり，学習・推論のプロセスを大幅に簡略化することができる．系列変換のためのモデリング技術の確立 [5–8] とネットワーク構造の洗練化 [9–11] により，End-to-End 音声認識モデルは様々な音声認識ベンチマークにおいて，有望な性能を示している [12–14]．

End-to-End モデリングでは一般的に，特定のタスクを解くのに適した特徴表現が，データから暗黙的に学習されることを期待している．例えば，画像分類モデルでは形状に関する特徴 [15] が，言語モデルでは構文構造に関する特徴 [16] が抽出されるように，各タスクに対する適切な DNN を設計することで学習が行われる．一方，音声認識では，単語を推定するのに適した特徴が音声から抽出されることが望ましいが，この特徴表現を End-to-End に学習するのは困難である．セグメンテーションやアラインメントの情報が与えられない End-to-End 音声認識では，フレーム単位の音声信号から単語単位の言語記号を直接推定する必要がある．この入出力列間における抽象度の差が，End-to-End 音声認識モデルの最適化を難しくしており，単語単位の認識に適した特徴表現を獲得することが困難となっている．そのため，大量のデータ [17, 18] や強力な言

[1] 早稲田大学
Waseda University
[a)] higuchi@pcl.cs.waseda.ac.jp

語モデル [19] を用いた学習や推論を行わなければ，単語単位の高精度な認識を実現するのは難しい．

音声認識において，音声から単語を推定するための特徴抽出プロセスを効率的に得るためには，言語情報の抽象度が徐々に上がるように認識システムを構築する必要がある．「音声 → 音素 → 単語 → テキスト」に基づいた変換過程 [20] が合理的とされてきたように，低レベルの（抽象度の低い）情報を組み上げることで高レベルの（抽象度の高い）情報が構成される．End-to-End 音声認識でも，低レベルの言語情報から，高レベルの言語情報が明示的に構成されるようなモデルを構築することで，抽象度の高い単語情報に対する特徴表現が効果的に学習されることが期待できる．

本研究では，上記の End-to-End 音声認識における段階的な特徴表現の学習を実現するために，階層的条件付きモデルを提案する（図 1）．提案モデルでは，Connectionist Temporal Classification（CTC）[5] に基づいた損失を，最終層だけでなく中間層に対しても繰り返し適用する [21–27]．このとき，各 CTC では異なる粒度の言語情報に基づいた損失計算を行い，入力に近い浅い層からは低レベルの言語情報が，出力に近い深い層からは高レベルの言語情報が推定されるようにモデルの学習を行う．具体的には，サブワード（部分文字列）単位を用いることで，単語単位の最終出力に近くなるにつれて，サブワードの語彙サイズを徐々に増やしていく（例：256 → 2k → 16k）．また，この階層的な学習法において，低レベルな単位による予測で条件付けながら高レベルな単位を推定することで，単語情報が細かい単位から明示的に構成されることを期待する．

本研究の主たる貢献を以下にまとめる．

(1) 提案手法により，予測系列の抽象度を徐々に上げるようにモデルを学習することで，認識精度が向上することを示す．スパースな情報を効果的に扱うことができ，単語単位の認識に適した特徴表現が学習できる．
(2) LibriSpeech と TEDLIUM2 に基づいた実験により，データ量や発話スタイルによらず，提案モデルが有効であることを示す．なお，実験に利用したソースコードはすべて公開済みである[*1]．
(3) 既存の類似モデルとの比較を行うことで，提案モデルの優位性を示す．また，実験結果を深く分析することで，提案モデルの利点に関する知見を与える．

本稿の構成は以下の通りである．2 では要素技術および提案手法について説明する．3 では先行研究との関連性について述べる．4 では音声認識実験による提案モデルの評価を行う．最後に 5 で本稿のまとめと今後の課題について述べる．

[*1] https://github.com/YosukeHiguchi/espnet/tree/hierctc

2. 提案手法

2.1 ベースラインの End-to-End 音声認識モデル

End-to-End 音声認識は，長さ T の音響特徴量系列 $X = (\mathbf{x}_t \in \mathbb{R}^D | t = 1, ..., T)$ から，長さ L の記号系列 $Y = (y_l \in \mathcal{V} | l = 1, ..., L)$ へのマッピングを推定する問題である．ここで，$\mathbf{x}_t$ は t 番目のフレームにおける D 次元の音響特徴量，y_l は l 番目の出力記号，$\mathcal{V}$ は出力記号の語彙である．本研究では，ベースラインの End-to-End 音声認識モデルとして，Transformer [28] に基づいたネットワーク構造を採用し，CTC [5] を用いた中間的な損失計算 [27] により学習することを考える．

2.1.1 Transformer Encoder

入力系列 X を潜在特徴量に変換するために，E 個の Self-attention 層から成る Transformer Encoder [28] を構築する．第 i 層では，d_{model} 次元の潜在特徴量の系列 $X^{(i)} = (\mathbf{x}_t^{(i)} \in \mathbb{R}^{d_{\mathsf{model}}} | t = 1, ..., T)$ が以下のように出力される．

$$\tilde{X}^{(i)} = X^{(i-1)} + \mathrm{SelfAttention}(X^{(i-1)}), \quad (1)$$

$$X^{(i)} = \tilde{X}^{(i)} + \mathrm{FeedForward}(\tilde{X}^{(i)}). \quad (2)$$

ここで，$i \in \{1, ..., E\}$ であり，$X^{(0)}$ は X に位置コーディング [28] を適用することで得られる．式 (1) と (2) において，SelfAttention(·) は Self-attention 機構，FeedForward(·) は全結合層であり，それぞれの入力に対して Layer Normalization が適用される．本研究では，畳み込み層を導入することで Transformer Encoder の改良を行なった Conformer [10] によるモデルも検討する．Conformer では，式 (1) と (2) の間に畳み込みモジュールによる処理が追加される．

2.1.2 Connectionist Temporal Classification

CTC [5] では，Transformer Encoder の出力系列 $X^{(E)}$ と目的の記号系列 Y における単調アラインメントを推定するように学習が行われる．フレーム単位でアラインメントを行うために，出力系列 Y にブランク記号 ϵ を導入し，同一記号の連続を許すことで，潜在的な記号列 $Z = (a_t \in \mathcal{V} \cup \{\epsilon\} | t = 1, ..., T)$ を得る．CTC では，条件付き確率 $P_{\mathsf{ctc}}(Y|X^{(E)})$ を全ての潜在記号列 Z に対して周辺化することで，End-to-End 音声認識をモデル化する．

$$P_{\mathsf{ctc}}(Y|X^{(E)}) = \sum_{Z \in \mathcal{B}^{-1}(Y)} \prod_{t=1}^{T} P(z_t | z_{<t}, X^{(E)}), \quad (3)$$

$$\approx \sum_{Z \in \mathcal{B}^{-1}(Y)} \prod_{t=1}^{T} P(z_t | X^{(E)}). \quad (4)$$

ここで，$\mathcal{B}^{-1}(Y)$ は Y に対応した全ての潜在記号列の集合を表す．式 (4) では，各記号の出力確率が独立であることを仮定している．CTC による損失は，式 (4) に対する負の対数尤度を用いて以下より定義される．

$$\mathcal{L}_{\text{ctc}}(Y|X^{(E)}) = -\log P_{\text{ctc}}(Y|X^{(E)}). \quad (5)$$

2.1.3 中間層の出力に対する CTC 損失の適用

モデルの最終層の出力から計算される標準的な CTC 損失（式 (5)）に加えて，中間層の出力に対して補助的な CTC 損失を適用する [26,27]．このような中間的な損失は，モデルの学習における正則化として効果的に働き，認識精度の向上につながることが知られている．出力層と中間層の出力に対して，合計 K 個の CTC 損失を適用すると，モデルの損失は以下のように算出される．

$$\mathcal{L}_{\text{selfctc}} = \frac{1}{K}\left\{\mathcal{L}_{\text{ctc}}(Y|X^{(E)}) + \sum_{k=1}^{K-1}\mathcal{L}_{\text{ctc}}(Y|X^{(\lfloor \frac{kE}{K} \rfloor)})\right\}. \quad (6)$$

ここで，$1 < K \leq E$ であり，各損失に対して均等に重みを分配する [29]．式 (6) の各損失の計算において，下層の CTC で予測された系列を条件付ける．これは Self-conditioned CTC（SelfCTC） [30] に基づいた手法であり，中間的な損失によるモデルの学習を促進することが知られている．CTC の中間予測を用いた条件づけは，式 (2) を変更して以下のように実現される．

$$\tilde{\tilde{X}}^{(i)} = \tilde{X}^{(i)} + \text{FeedForward}(\tilde{X}^{(i)}), \quad (7)$$

$$X^{(i)} = \tilde{\tilde{X}}^{(i)} + \text{Linear}(A^{(i)}). \quad (8)$$

ここで，$i \in \{\lfloor kE/K \rfloor\}_{k=1}^{K-1}$ であり，$A^{(i)} = \text{softmax}(\tilde{\tilde{X}}^{(i)})$ は CTC の各出力における語彙の事後確率分布である．

2.2 サブワード分割

サブワード分割 [31] は，テキスト中の単語を部分文字列単位に分割する手法であり，自然言語処理における未知語問題の解決に広く用いられている．サブワードの語彙を構築するための一般的なアルゴリズムでは，連結した際にテキストデータの中で最も頻度が高くなる 2 つの文字またはサブワード単位を選び，それらを結合して新たな語彙を生成する．この手続きは決められた語彙サイズに達するまで繰り返される．

本研究では，音声認識のテキストデータ（書き起こし文）に対してサブワード分割を適用する．文字列に比べ，サブワード列は長さが短く，出力間の依存関係を捉えやすくなることが期待できる．これは，記号間の条件付き独立性を仮定した CTC による学習では，特に重要であることが予想される（式 (4) 参照）．ただし，サブワードは語彙サイズを大きくすることで，単語に近いスパースな単位となり，End-to-End 音声認識の学習が困難となることに注意する必要がある [17]．

2.3 階層的条件付き End-to-End 音声認識モデル

図 1 に，提案する階層的条件付き End-to-End 音声認識モデルの構造を示す．提案モデルは，2.1.3 の中間的な

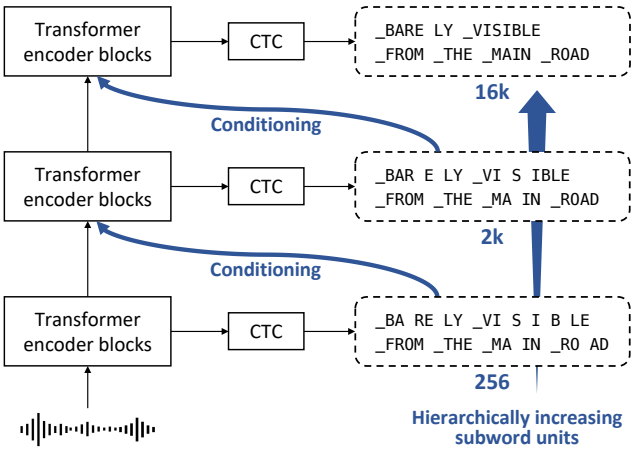


図 1 提案の階層的条件つき End-to-End 音声認識モデル

CTC 損失を用いたモデルに似ているが，各 CTC における出力単位の粒度が徐々に高くなっており，最終層では単語に近い単位の系列を予測する．長さ $L^{(k)}$ のサブワード列 $Y^{(k)} = (y_l^{(k)} \in \mathcal{V}^{(k)} | l = 1, \ldots, L^{(k)})$ を k 番目の CTC 損失に対する目的の系列とする．ここで，$\mathcal{V}^{(k)}$ は k 番目の CTC が出力するサブワードの語彙である．CTC 損失の位置がモデルの出力層に近づくにつれて，サブワードの語彙サイズを階層的に増やす（$|\mathcal{V}^{(<K)}| < |\mathcal{V}^{(K)}|$）．異なる語彙サイズに基づいたサブワード列を用いて，提案モデルの損失は次のように定義される．

$$\mathcal{L}_{\text{hc-ctc}} = \frac{1}{K}\left\{\mathcal{L}_{\text{ctc}}(Y^{(K)}|X^{(E)}) + \sum_{k=1}^{K-1}\mathcal{L}_{\text{ctc}}(Y^{(k)}|X^{(\lfloor \frac{kE}{K} \rfloor)})\right\}. \quad (9)$$

各目的サブワード列において語彙サイズが同じ場合，式 (9) と式 (6) は等しくなる．式 (7) に基づいた条件づけにより，式 (9) における各 CTC 損失の計算は，下層で予測されたより細かい単位のサブワード列によって次のように条件付けられる．

$$\mathcal{L}_{\text{ctc}}(Y^{(k)}|X^{(k)}) = -\log P_{\text{ctc}}(Y^{(k)}|\hat{Y}^{(1)}, ..., \hat{Y}^{(k-1)}, X^{(k)}), \quad (10)$$

ここで，$\hat{Y}^{(k)}$ は k 番目の CTC によって予測された系列である．

提案モデルにより，単語単位の系列が粒度の低い単位から階層的に構成されることが期待できる．浅い層からは細かい単位の高頻度なサブワードを，深い層からは粗い単位の低頻度なサブワードを予測するようにモデルの学習を行う（式 (9)）．このとき，各単位の予測において，より細かい単位の予測情報が明示的に考慮されるように，式 (7) と (10) に基づいた条件づけを行う．このように，提案モデルでは言語表現の階層性を考慮することで，スパースな単語に対する特徴表現が効果的に学習され，単語単位の認識性能が向上することを期待する．

2.4 CTC 損失を並列に適用したモデル

提案モデルにおける階層構造の有効性を検証するために，最終層の出力に対して複数の CTC 損失を並列に適用したモデルも検討する [24, 32–34]．CTC 損失を並列に適用したモデルの損失関数は，式 (9) を変更することで次のように定義される．

$$\mathcal{L}_{\mathsf{paractc}} = \frac{1}{K}\left\{\mathcal{L}_{\mathsf{ctc}}(Y^{(K)}|X^{(E)}) + \sum_{k=1}^{K-1}\mathcal{L}_{\mathsf{ctc}}(Y^{(k)}|X^{(E)})\right\}. \tag{11}$$

ここで，モデルの出力特徴量を各サブワード単位による認識に適応させるために，$X^{(E)}$ に対して線形層を 1 層適用する．このモデルでは，異なる粒度での系列予測を並列に学習することで，細かい単位の予測による粗い単位の予測に対する帰納的バイアス（inductive bias）が働き，認識精度が向上することが知られている [33]．

3. 関連研究

End-to-End 音声認識モデルの中間層に補助的な CTC 損失を導入することで，認識性能を改善する手法が多く検討されている．例えば，Attention-based Sequence-to-Sequence [35, 36]，Recurrent Neural Network Transducer [37]，CTC [26, 27, 38, 39] に基づいた End-to-End 音声認識モデルにおいて，その有効性が示されている．CTC に基づいたモデルでは，中間層の CTC 損失に対して音素といった低レベルの（抽象度の低い）教師情報を利用することで，最終層の CTC における高レベルな系列の予測を改善できることが知られている [21–25]．提案モデルは，このような出力単位を階層的に変化させたモデルの拡張として捉えられるが，以下の観点で従来研究とは異なる．1) 各 CTC による系列予測は，下層で予測された低レベルの系列情報に条件付いている．高レベルな系列の生成に寄与する情報を明示的に考慮することで，単語単位の認識が効果的に学習されることを期待する．2) 最近の研究 [26, 27] では，最終層における CTC と共通した単位の CTC 損失を中間層に適用することで，性能が改善することを示しているが，階層的なモデリングとの比較は行われていない．本研究では，複数の CTC 損失を用いた類似手法との比較を行うことで，階層的なモデリングの有効性について慎重に評価・分析する．3) 各出力にはサブワード単位を用いる．音素アラインメントといった追加のラベリング作業を必要とせず，出力系列の粒度を容易に調整することが可能である．4) Transformer [28] および Conformer [10] に基づいた state-of-the-art な End-to-End 音声認識モデルの構造を用いて実験評価を行う．

4. 音声認識実験

4.1 比較モデル

- **CTC:** 式 (5) の $\mathcal{L}_{\mathsf{ctc}}$ より学習した標準的なモデル [1].
- **SelfCTC:** 複数の中間層に CTC 損失を適用し [26,27], 式 (6) の $\mathcal{L}_{\mathsf{selfctc}}$ より学習した既存のモデル [30].
- **HC-CTC:** 式 (9) の $\mathcal{L}_{\mathsf{hcctc}}$ より学習した提案の階層的条件付きモデル.
- **ParaCTC:** 複数の CTC 損失を最終層に適用し，式 (11) の $\mathcal{L}_{\mathsf{paractc}}$ より学習した既存のモデル [24,32,33].

4.2 実験データ

英語のデータセットである LibriSpeech (LS) [40] および TEDLIUM2 (TED2) [41] を用いた．LibriSpeech はオーディオブックの読み上げ発話を収録したコーパスである．訓練データとして，100 時間のサブセット (LS-100) または 960 時間のフルセット (LS-960) を用いた．TEDLIUM2 は TED Talks の自然発話音声から構成されるコーパスであり，モデルの学習には 210 時間の訓練データを用いた．モデルのチューニングおよび評価には，各コーパスの開発および評価データを用いた．モデルの入力は，対数メル尺度フィルタバンクにピッチ情報を加えた 83 次元の音響特徴量とし，特徴量抽出には Kaldi [42] を用いた．データ拡張として，訓練データに speed perturbation [43]，SpecAugment [44] を適用した．サブワード語彙は SentencePiece[*2] [45] を用いて各訓練データから構築した．

4.3 学習・推論条件

全ての実験は ESPnet[*3] [46] を用いて行った．End-to-End 音声認識モデルとして，エンコーダブロック 18 層（$E = 18$）から成る Transformer [28] を構築した．各 Self-attention 層において, Multi-head Attention のヘッド数 d_{h}，埋め込み次元 d_{model}，全結合層のユニット数 d_{ff} は，それぞれ 4, 256, 2048 とした．Transformer に加え，Conformer [10] に基づいたモデルも構築した．畳み込みモジュールにおけるカーネルサイズは 15 とし，Self-attention 層における d_{h}, d_{model}, d_{ff} は，それぞれ 4, 256, 1024 とした．モデルの学習におけるエポック数は，LS-100 と TED2 では 100, LS-960 は 70 とした．複数の CTC 損失を適用した各モデル（**SelfCTC**, **HC-CTC**, **ParaCTC**）において，損失の合計数は 3 ($K = 3$) とした．出力におけるサブワード単位の語彙サイズは，LS100 は 16384, LS960 は 32768, TED2 は 16384 とした．これらの語彙サイズは，SentencePiece において各訓練データに対して設定できた最大のサイズに基づいて決定され，単語単位に近い値である．**SelfCTC** における中間損失における語彙サイズは，出力と同じであ

[*2] https://github.com/google/sentencepiece
[*3] https://github.com/espnet/espnet

表 1 LibriSpeech-{100h, 960h}, TEDLIUM2 における各モデルの単語誤り率（Word Error Rate; WER）[%]. 出力のサブワード語彙サイズは，LibriSpeech-100h と TEDLIUM2 は 16k，LibriSpeech-960h は 32k とした．外部言語モデルとビームサーチは用いずに推論を行った．

Model		LibriSpeech-100h				LibriSpeech-960h				TEDLIUM2	
		Dev WER		Test WER		Dev WER		Test WER		Dev WER	Test WER
		clean	other	clean	other	clean	other	clean	other		
Transformer	CTC	11.5	24.8	11.8	25.5	4.2	10.0	4.5	9.9	11.8	10.7
	SelfCTC	8.9	21.0	9.1	21.7	3.2	8.2	3.5	8.2	9.4	**8.6**
	HC-CTC	**8.2**	**19.9**	**8.4**	**20.6**	**3.1**	**8.0**	**3.4**	**8.0**	**9.1**	**8.6**
	ParaCTC	10.4	24.0	10.9	24.3	4.6	10.3	4.8	10.3	10.9	10.2
Conformer	SelfCTC	7.1	17.7	7.7	18.3	**2.8**	**6.7**	**3.0**	6.9	8.5	7.8
	HC-CTC	**6.9**	**17.1**	**7.1**	**17.8**	**2.8**	6.9	**3.0**	**6.8**	**8.0**	**7.6**

る．**HC-CTC** と **ParaCTC** の ($|\mathcal{V}^{(1)}|$, $|\mathcal{V}^{(2)}|$, $|\mathcal{V}^{(3)}|$) は，LS-100 と TED2 は (256, 2048, 16384)，LS-960 は (512, 4096, 32768) とした．モデルの学習後，開発セットに対する精度が最も高い上位 10-20 個のチェックポイントを用いてモデルパラメータを平均し，最終的なモデルを得た．推論時，言語モデルは用いず，CTC による Best Path Decoding [5] を行った．

4.4 実験結果

LS-100, LS-960, TED2 に対する，各モデルの単語誤り率（Word Error Rate; WER）を表 1 に示す．Transformer を用いた結果を比較すると，複数の損失を用いてモデルを学習すること（**SelfCTC**, **HC-CTC**, **ParaCTC**）で，**CTC** を上回る性能を与えた．特に，**SelfCTC** と **HC-CTC** では，全てのタスクにおいて **CTC** よりも大幅に WER を削減した．LS-100 において，**HC-CTC** は **SelfCTC** よりも良好な性能を与えたことから，出力単位の粒度を階層的に増加させることの有効性を示唆している．一方，データ量の多い LS-960 と TED2 では，**HC-CTC** と **SelfCTC** の性能差は縮まり，**HC-CTC** がわずかに低い WER を与えた．したがって，**HC-CTC** は，単語情報がよりスパースになる小規模なデータに対して特に有効であり，単語単位の特徴表現を効果的に学習できたことがわかる．**SelfCTC** は，十分なデータ量があれば，単語単位の認識を精度良く行うことができた．しかし，数万の語彙サイズに基づいた Softmax 層（Eq. (8) 参照）は膨大な計算コストを必要とし，これを繰り返す **SelfCTC** では学習・推論速度が著しく低下した．一方，**HC-CTC** では，中間層の損失に対してより細かい単位を利用するため，学習・推論速度が大きく低下することはなかった．また，語彙サイズに関する同様の理由から，**HC-CTC** のモデルサイズは **SelfCTC** よりも遥かに小さかった（例：LS-960 で 36.4M vs. 67.6M）．**HC-CTC** と **ParaCTC** を比較すると，**HC-CTC** が全てのタスクにおいて高い認識精度を達成していることから，CTC 損失を中間層に適用し，出力単位を段階的に増加させることの有効性が確認できた．

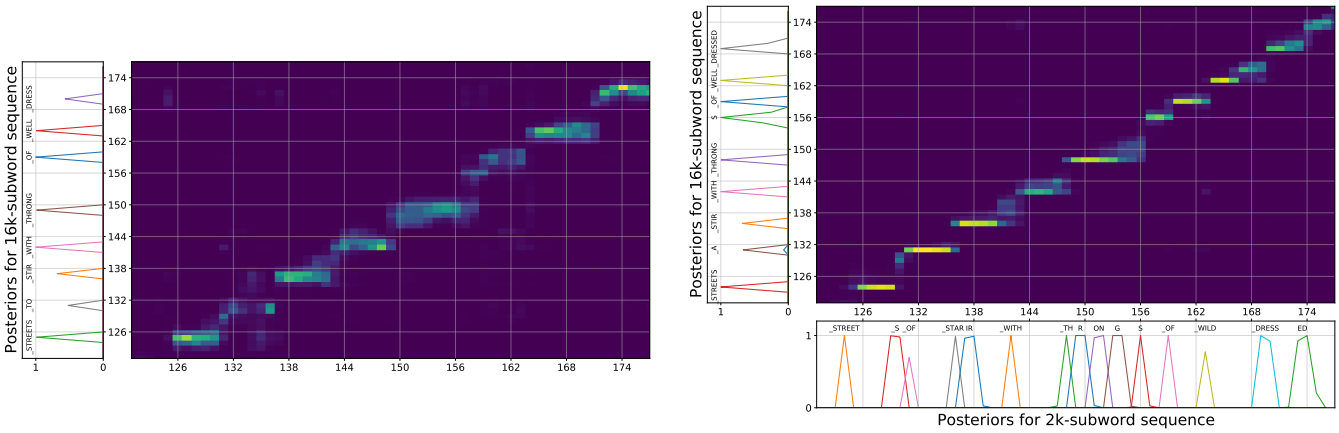
表 2 LS-100 の開発データ（dev-clean, dev-other）における単語誤り率 [%]．Transformer を用いた **SelfCTC** と **HC-CTC** において，異なるサブワード語彙サイズの組み合わせを評価した．

Model	$\|\mathcal{V}^{(1)}\|$-$\|\mathcal{V}^{(2)}\|$-$\|\mathcal{V}^{(3)}\|$	dev-clean	dev-other
SelfCTC	256 - 256 - 256	8.4	22.8
SelfCTC	2k - 2k - 2k	8.5	22.0
SelfCTC	16k - 16k - 16k	8.9	21.0
HC-CTC	256 - 256 - 16k	**8.2**	20.2
HC-CTC	2k - 2k - 16k	8.4	20.2
HC-CTC	256 - 2k - 16k	**8.2**	**19.9**

Conformer を用いることで，**SelfCTC** と **HC-CTC** の WER はさらに改善し，学習・推論速度，モデルサイズ，認識精度の観点で **HC-CTC** は **SelfCTC** よりも良好な性能を与えた．これら Conformer による結果は，特にチューニングを行わずに得られたものであり，CTC に基づいた最先端のモデル [11, 47, 48] と比べて遜色のない性能を達成している．

4.5 サブワード語彙サイズに関する分析

End-to-End 音声認識において，スパースな単語単位の認識を学習するのは，一般的に困難であることが知られている [17]．しかし，本実験において，単語レベルのサブワード単位を用いることで，Transformer に基づいた **CTC** の性能が向上することがわかった．出力の語彙サイズを 256 から 16k に増やすことで，開発データに対する WER は，LS-100 では 11.1/28.1% から 11.5/24.8% に，TED2 では 12.3% から 11.8% に変化した．また，LS-960 においても，語彙サイズを 2k から 32k に増やすことで，WER が 4.6/12.1% から 4.4/10.5% に変化した．これら改善の理由として，サブワードの語彙サイズを増加させることで，出力記号間の依存性が軽減され，CTC に基づいた学習が効



(a) CTC **(b) HC-CTC**

図 2 表 1 における LS-100 で学習された（a）CTC と（b）HC-CTC の Attention 重みを可視化した例．dev-other の発話（116-288045-0000）を入力とし，正解列は “*STREETS ASTIR WITH THRONGS OF WELL DRESSED*” であった．

果的に行われたことが考えられる（2.2 参照）．

以上を踏まえ，Transformer を用いた **SelfCTC** と **HC-CTC** に対する語彙サイズの影響について分析する．各モデルにおいて，語彙サイズの組み合わせを変えた時の評価結果を表 2 に示す．dev-other に対する **SelfCTC** の結果を比較すると，語彙サイズを大きくすることで CTC による学習が効果的に行われ，語彙サイズ 16k を用いたモデルが最良の性能を与えた．出力の語彙サイズ $|\mathcal{V}^{(3)}|$ が 16k の **HC-CTC** と **SelfCTC** を比べると，**HC-CTC** によるモデルは **SelfCTC** よりも低い WER を達成したことがわかる．このことから，**HC-CTC** は大きい語彙サイズを用いた CTC による学習の利点に加えて，単語単位の認識に効果的なモデリングが行えたことが確認できる．dev-clean に対する **SelfCTC** の性能は，語彙サイズを増やすことで低下した．一方，**HC-CTC** は語彙サイズ 16k でも低い WER を達成していることから，単語単位の特徴表現学習がより頑健に行われたことを意味する．**HC-CTC** の結果を比較すると，語彙サイズを徐々に増加させる方が，中間損失に対して同じ語彙サイズを使うよりも良好な性能を与えた．単語単位の特徴表現学習において，出力の抽象度を徐々に上げていくことの有効性が示唆された．

4.6 条件づけの重要性

提案モデルの構成要素である，条件づけの有効性について調査した（式 (10) 参照）．Transformer を用いた **HC-CTC** を対象とし，LS-100 における評価を行った．このとき，各 CTC 損失に対する条件づけは行わず，式 (1) と (2) に基づいた計算処理を全ての中間層に適用した．ここで得られるモデルは，従来のモデル [21–25] と類似したものである．CTC 損失に対する条件づけを行わなかった場合，**HC-CTC** の開発，評価データに対する WER は，それぞれ 8.7/20.7% および 9.0/21.3% であった．これらの結果は，表 1 において，**CTC**, **SelfCTC**, **ParaCTC** の性能を上回っているが，条件づけを行った **HC-CTC** と比べると性能が劣化したことがわかる．4.5 の結果も踏まえると，単語単位の特徴表現を効果的に学習するには，1) 出力単位の粒度を徐々に粗くした階層的なモデリングと 2) 低レベルな単位による予測で条件付けながら高レベルな単位による予測を行うことの重要性が確認できた．

4.7 Attention 重みの可視化

図 2 は，入力系列（x 軸）の出力系列（y 軸）に対する Attention 重みを可視化した結果であり，表 1 の LS-100 における (a) **CTC** と (b) **HC-CTC** を比較している．各モデルで，語彙サイズ 16k の系列予測に寄与したと思われる Attention 重みを，18 番目の Self-attention 層から選んだ．HC-CTC については，12 層目の CTC によって予測された，語彙サイズ 2k の系列に対する事後確率を示している．図 2 (a) における **CTC** の Attention 結果を見ると，各出力記号に対する重みが全体的に不明瞭であることがわかる．これに対し，図 2 (b) の **HC-CTC** では，より信頼度の高い Attention 重みが学習されており，各記号の予測において重要な領域が明瞭に検出されている．この検出された領域が，語彙サイズ 16k の記号に対応した，語彙サイズ 2k における記号列に一致していることから，**HC-CTC** では低レベルの予測情報を利用することで，より複雑な単語レベルの記号を構成していることがわかる．例えば，**HC-CTC** は，“THRONGS” や “DRESSED” といった単語に対して，下位の “S” や “ED” の認識結果を用いることで適切な活用形で認識が行えている．一方，**CTC** はこれらの出現頻度が低い単語を適切に扱えず，“THRONG” や “DRESS” のように誤った認識結果となっている．

5. まとめ

本研究では，End-to-End 音声認識において，単語単位の認識を行うための特徴表現を獲得することを検討し，階層的条件付きモデルを提案した．提案モデルは，複数の中間層に対して CTC 損失を適用し，入力から出力にかけて予測系列の単位が段階的に大きくなるように学習を行なった．また，各単位における予測を，より大きい単位の予測に明示的に条件付けることで，言語表現の階層性を考慮した認識が行えることを期待した．実験による詳細な評価・分析の結果，提案モデルにより単語単位の特徴表現が効果的に学習され，音声認識性能が格段に向上することを確認した．

今後は，提案モデルにデコーダネットワーク [49] を導入することで，単語間の文脈を考慮した学習を行うことを検討する予定である．また，音響情報に基づいたサブワード単位 [50,51] を利用することを考えている．

謝辞 本研究の一部は，JST，ACT-X，JPMJAX210J の支援を受けたものである．

参考文献

[1] Graves, A. and Jaitly, N.: Towards end-to-end speech recognition with recurrent neural networks, *Proc. ICML*, pp. 1764–1772 (2014).
[2] Chorowski, J. K., Bahdanau, D., Serdyuk, D., Cho, K. and Bengio, Y.: Attention-based models for speech recognition, *Proc. NeurIPS*, pp. 577–585 (2015).
[3] Chan, W., Jaitly, N., Le, Q. and Vinyals, O.: Listen, attend and spell: A neural network for large vocabulary conversational speech recognition, *Proc. ICASSP*, pp. 4960–4964 (2016).
[4] Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N. et al.: Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, *IEEE Signal processing magazine*, Vol. 29, No. 6, pp. 82–97 (2012).
[5] Graves, A., Fernández, S., Gomez, F. and Schmidhuber, J.: Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks, *Proc. ICML*, pp. 369–376 (2006).
[6] Graves, A.: Sequence transduction with recurrent neural networks, *arXiv preprint arXiv:1211.3711* (2012).
[7] Sutskever, I., Vinyals, O. and Le, Q. V.: Sequence to sequence learning with neural networks, *Proc. NeurIPS*, pp. 3104–3112 (2014).
[8] Bahdanau, D., Cho, K. and Bengio, Y.: Neural machine translation by jointly learning to align and translate, *Proc. ICLR* (2014).
[9] Dong, L., Xu, S. and Xu, B.: Speech-Transformer: A no-recurrence sequence-to-sequence model for speech recognition, *Proc. ICASSP*, pp. 5884–5888 (2018).
[10] Gulati, A., Qin, J., Chiu, C.-C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y. and Pang, R.: Conformer: Convolution-augmented Transformer for Speech Recognition, *Proc. Interspeech*, pp. 5036–5040 (2020).
[11] Majumdar, S., Balam, J., Hrinchuk, O., Lavrukhin, V., Noroozi, V. and Ginsburg, B.: Citrinet: Closing the Gap between Non-Autoregressive and Autoregressive End-to-End Models for Automatic Speech Recognition, *arXiv preprint arXiv:2104.01721* (2021).
[12] Chiu, C.-C., Sainath, T. N., Wu, Y., Prabhavalkar, R., Nguyen, P., Chen, Z., Kannan, A., Weiss, R. J., Rao, K., Gonina, E. et al.: State-of-the-art speech recognition with sequence-to-sequence models, *Proc. ICASSP*, pp. 4774–4778 (2018).
[13] Lüscher, C., Beck, E., Irie, K., Kitza, M., Michel, W., Zeyer, A., Schlüter, R. and Ney, H.: RWTH ASR Systems for LibriSpeech: Hybrid vs Attention, *Proc. Interspeech*, pp. 231–235 (2019).
[14] Karita, S., Chen, N., Hayashi, T., Hori, T., Inaguma, H., Jiang, Z., Someki, M., Soplin, N. E. Y., Yamamoto, R., Wang, X. et al.: A comparative study on Transformer vs RNN in speech applications, *Proc. ASRU*, pp. 449–456 (2019).
[15] Zeiler, M. D. and Fergus, R.: Visualizing and understanding convolutional networks, *Proc. ECCV*, pp. 818–833 (2014).
[16] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. and Zettlemoyer, L.: Deep contextualized word representations, *Proc. NAACL-HLT*, pp. 2227–2237 (2018).
[17] Soltau, H., Liao, H. and Sak, H.: Neural speech recognizer: Acoustic-to-word LSTM model for large vocabulary speech recognition, *arXiv preprint arXiv:1610.09975* (2016).
[18] Zhang, Y., Qin, J., Park, D. S., Han, W., Chiu, C.-C., Pang, R., Le, Q. V. and Wu, Y.: Pushing the limits of semi-supervised learning for automatic speech recognition, *arXiv preprint arXiv:2010.10504* (2020).
[19] Irie, K., Zeyer, A., Schlüter, R. and Ney, H.: Language modeling with deep Transformers, *Proc. Interspeech*, pp. 3905–3909 (2019).
[20] Jelinek, F.: Continuous speech recognition by statistical methods, *Proc. IEEE*, Vol. 64, No. 4, pp. 532–556 (1976).
[21] Fernández, S., Graves, A. and Schmidhuber, J.: Sequence Labelling in Structured Domains with Hierarchical Recurrent Neural Networks, *Proc. IJCAI*, pp. 774–779 (2007).
[22] Rao, K. and Sak, H.: Multi-accent speech recognition with hierarchical grapheme based models, *Proc. ICASSP*, pp. 4815–4819 (2017).
[23] Toshniwal, S., Tang, H., Lu, L. and Livescu, K.: Multitask learning with low-level auxiliary tasks for encoder-decoder based speech recognition, *arXiv preprint arXiv:1704.01631* (2017).
[24] Sanabria, R. and Metze, F.: Hierarchical multitask learning with CTC, *Proc. SLT*, pp. 485–490 (2018).
[25] Krishna, K., Toshniwal, S. and Livescu, K.: Hierarchical multitask learning for CTC-based speech recognition, *arXiv preprint arXiv:1807.06234* (2018).
[26] Tjandra, A., Liu, C., Zhang, F., Zhang, X., Wang, Y., Synnaeve, G., Nakamura, S. and Zweig, G.: Deja-vu: Double feature presentation and iterated loss in deep Transformer networks, *Proc. ICASSP*, pp. 6899–6903 (2020).
[27] Lee, J. and Watanabe, S.: Intermediate loss regularization for CTC-based speech recognition, *Proc. ICASSP*, pp. 6224–6228 (2021).
[28] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J.,

Jones, L., Gomez, A. N., Kaiser, Ł. and Polosukhin, I.: Attention is all you need, *Proc. NeurIPS*, pp. 5998–6008 (2017).

[29] Lee, J., Kang, J. and Watanabe, S.: Layer Pruning on Demand with Intermediate CTC, *Proc. Interspeech*, pp. 3745–3749 (2021).

[30] Nozaki, J. and Komatsu, T.: Relaxing the Conditional Independence Assumption of CTC-Based ASR by Conditioning on Intermediate Predictions, *Proc. Interspeech*, pp. 3735–3739 (2021).

[31] Sennrich, R., Haddow, B. and Birch, A.: Neural Machine Translation of Rare Words with Subword Units, *Proc. ACL*, pp. 1715–1725 (2016).

[32] Li, J., Ye, G., Zhao, R., Droppo, J. and Gong, Y.: Acoustic-to-word model without OOV, *Proc. ASRU*, pp. 111–117 (2017).

[33] Kremer, J., Borgholt, L. and Maaløe, L.: On the inductive bias of word-character-level multi-task learning for speech recognition, *arXiv preprint arXiv:1812.02308* (2018).

[34] Heba, A., Pellegrini, T., Lorré, J.-P. and Andre-Obrecht, R.: Char+CV-CTC: Combining Graphemes and Consonant/Vowel Units for CTC-Based ASR Using Multitask Learning, *Proc. Interspeech*, pp. 1611–1615 (2019).

[35] Kim, S., Hori, T. and Watanabe, S.: Joint CTC-attention based end-to-end speech recognition using multi-task learning, *Proc. ICASSP*, pp. 4835–4839 (2017).

[36] Moriya, T., Ueno, S., Shinohara, Y., Delcroix, M., Yamaguchi, Y. and Aono, Y.: Multi-task Learning with Augmentation Strategy for Acoustic-to-word Attention-based Encoder-decoder Speech Recognition., *Proc. Interspeech*, pp. 2399–2403 (2018).

[37] Jeon, J.-J. and Kim, E.: Multitask Learning and Joint Optimization for Transformer-RNN-Transducer Speech Recognition, *Proc. ICASSP*, pp. 6793–6797 (2021).

[38] Zweig, G., Yu, C., Droppo, J. and Stolcke, A.: Advances in all-neural speech recognition, *Proc. ICASSP*, pp. 4805–4809 (2017).

[39] Chi, E. A., Salazar, J. and Kirchhoff, K.: Align-Refine: Non-Autoregressive Speech Recognition via Iterative Realignment, *Proc. NAACL-HLT*, pp. 1920–1927 (2021).

[40] Panayotov, V., Chen, G., Povey, D. and Khudanpur, S.: Librispeech: An ASR corpus based on public domain audio books, *Proc. ICASSP*, pp. 5206–5210 (2015).

[41] Rousseau, A., Deléglise, P., Esteve, Y. et al.: Enhancing the TED-LIUM Corpus with Selected Data for Language Modeling and More TED Talks, *Porc. LREC*, pp. 3935–3939 (2014).

[42] Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P. et al.: The Kaldi speech recognition toolkit, *Proc. ASRU* (2011).

[43] Ko, T., Peddinti, V., Povey, D. and Khudanpur, S.: Audio Augmentation for Speech Recognition, *Proc. Interspeech*, pp. 3586–3589 (2015).

[44] Park, D. S., Chan, W., Zhang, Y., Chiu, C.-C., Zoph, B., Cubuk, E. D. and Le, Q. V.: SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition, *Proc. Interspeech*, pp. 2613–2617 (2019).

[45] Kudo, T.: Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates, *Proc. ACL*, pp. 66–75 (2018).

[46] Watanabe, S., Hori, T., Karita, S., Hayashi, T., Nishitoba, J., Unno, Y., Enrique Yalta Soplin, N., Heymann, J., Wiesner, M., Chen, N., Renduchintala, A. and Ochiai, T.: ESPnet: End-to-End Speech Processing Toolkit, *Proc. Interspeech*, pp. 2207–2211 (2018).

[47] Ng, E. G., Chiu, C.-C., Zhang, Y. and Chan, W.: Pushing the Limits of Non-Autoregressive Speech Recognition, *Proc. Interspeech*, pp. 3725–3729 (2021).

[48] Higuchi, Y., Chen, N., Fujita, Y., Inaguma, H., Komatsu, T., Lee, J., Nozaki, J., Wang, T. and Watanabe, S.: A Comparative Study on Non-Autoregressive Modelings for Speech-to-Text Generation, *arXiv preprint arXiv:2110.05249* (2021).

[49] Higuchi, Y., Watanabe, S., Chen, N., Ogawa, T. and Kobayashi, T.: Mask CTC: Non-Autoregressive End-to-End ASR with CTC and Mask Predict, *Proc. Interspeech*, pp. 3655–3659 (2020).

[50] Xu, H., Ding, S. and Watanabe, S.: Improving end-to-end speech recognition with pronunciation-assisted subword modeling, *Proc. ICASSP*, pp. 7110–7114 (2019).

[51] Zhou, W., Zeineldeen, M., Zheng, Z., Schlüter, R. and Ney, H.: Acoustic Data-Driven Subword Modeling for End-to-End Speech Recognition, *Proc. Interspeech*, pp. 2886–2890 (2021).