人工知能学会第二種研究会資料
AIチャレンジ研究会
SIG-Challenge-066-05

動的な語彙拡張を用いたEnd-to-end音声認識の文脈適応

Contextualized End-to-end Automatic Speech Recognition using Dynamic Vocabulary Expansion

周藤 唯[1*]　Muhammad Shakeel[1]　Yifan Peng[2]　渡部 晋治[2]
Yui Sudo[1]　Muhammad Shakeel[1]　Yifan Peng[2]　Shinji Watanabe[2]

[1] (株) ホンダ・リサーチ・インスティチュート・ジャパン
[1] Honda Research Institute Japan Co.，Ltd.
[2] カーネギーメロン大学
[2] Carnegie Mellon University

Abstract: 本稿では，動的な語彙拡張を用いたEnd-to-end音声認識の文脈適応手法を提案する．ユーザが編集可能なフレーズリスト (バイアスリスト) を利用するDeep Biasing手法は，人名や専門用語などといった稀なフレーズの認識精度を向上させる方法として有望なアプローチである．しかし，従来の手法では，事前に定義された語彙のサブワードトークン列としてこれらのフレーズを扱うため，学習データ内で頻度の低いトークンパターンを持つフレーズの認識確率が低下するという課題があった．本研究では，動的な語彙拡張を用いた新たなDeep Biasing手法を提案する．提案手法では，バイアスリストに登録された各フレーズに対し，そのフレーズ全体を表す新しいトークンを動的に導入することで語彙を拡張する．この方法により，従来手法のようにサブワード列に依存する必要がなくなり，頻度の低さに起因する認識精度の低下を回避できる．提案手法は，一般的なEnd-to-end音声認識モデルが共通して持つ埋め込み層と出力層を拡張することで実現可能であり，さまざまなモデルに適用することができる．英語および日本語のデータセットを用いた評価実験において，提案手法は従来のDeep Biasing手法と比較して登録フレーズの単語誤り率（B-WER）を3.1〜4.9ポイント改善した．

1　はじめに

ロボット聴覚は，ロボットが音声や環境音をもとに周囲の環境を認識することを目指す研究分野であるが，その中で，音声認識は人の音声を理解するために必要不可欠な技術である．近年，End-to-end音声認識 [1, 2] の登場により，音声認識の性能が飛躍的に向上している．End-to-end音声認識 [1, 2] は，音響モデル，発音辞書，言語モデルといった複数のコンポーネントで構成される従来の音声認識システムとは異なり，単一のニューラルネットワークを用いて入力音声信号を直接文字列に変換する．これまでに，Conectionist-Temporal-Classification（CTC）[3, 4]，Reccurent Neural Network Transducer（RNN-T）[5, 6]，注意機構（Attention）[7, 8, 9]，およびこれらのハイブリッドモデル [10, 11, 12] など，さまざまなEnd-to-end音声認識モデルが提案されている．End-to-end音声認識モデルの性能は，学習データに強く依存しているため，学習データに含まれていないフレーズを正しく認識することが難しい．例えば，業界特有の専門用語，人名，地名などの固有名詞は，学習データ内での出現頻度が少ないため，正しく認識されないことが多い．これらの単語は，文脈上重要なキーワードとなるため，ロボット聴覚の実用化に向けて大きな課題となる．

このような課題に対し，Deep Biasing [13, 14, 15, 16, 17, 18, 19, 20] と呼ばれる文脈適応手法が提案されている．Deep Biasingでは，編集可能なフレーズリスト（バイアスリスト）を介して，バイアスリストに登録されたフレーズ (バイアスフレーズ) の認識性能を向上させる．多くのDeep Biasing手法は，バイアスフレーズを検出するためにCross-attention層を導入し，補助的な損失関数を用いたマルチタスク学習によって性能を向上させている [15, 16, 17, 18, 19, 20]．しかし，Cross-attention層の導入はモデル構造を複雑にし，マルチタスク学習は学習重みを調整作業を増加させるという課題がある．

また，従来のDeep Biasing手法では，事前に定義された語彙 (静的語彙) のサブワードトークン列としてバ

*連絡先：(株) ホンダ・リサーチ・インスティチュート・ジャパン
〒 351-0188 埼玉県和光市本町 8-1
E-mail: yui.sudo@jp.honda-ri.com

イアスフレーズを扱っていることが性能劣化の一因となっている．例えば，「*Nelly*」という人名は「*N*，*el*，*ly*」のようなサブワードトークン列として処理される．しかし，学習データ内でこのようなサブワードトークンパターンが稀である場合，その認識確率は大幅に低下する．この問題に対処するため，[21, 22] では追加のテキストデータを利用することで，稀なサブワードトークンパターンの問題を緩和している．しかし，このアプローチでは，追加データの収集や外部言語モデルの学習が必要となり，作業負荷が大幅に増加してしまう．その他にも，音素情報 [23, 24, 25] や固有名詞タグ [26]，音声合成 [27, 28] などの追加情報を利用する手法が提案されているが，これらも同様に作業負荷を増加させる．

本研究では，これらの課題を解決するため，動的な語彙拡張を用いた新たな Deep Biasing 手法を提案する．提案手法では，バイアスリストに登録された各フレーズに対し，そのフレーズ全体を表す新しいトークンを動的に導入することで語彙を拡張する．この方法により，従来手法 [21, 22] のようにサブワード列に依存する必要がなくなり，頻度の低さに起因する認識精度の低下を回避することができる．また，[15, 16, 17, 18, 19] とは異なり，Cross-attention 層や補助損失を導入する必要がなく，CTC，RNN-T，Attention などのさまざまな End-to-end 音声認識モデルに適用することができる．

なお，本稿は，[29] をもとに，詳細な性能分析を追加している．具体的には，従来のサブワードベースの手法との比較および日本語システムにおける性能分析を追加している．

2 End-to-end 音声認識モデル

本節では，音響エンコーダとデコーダから構成される一般的な End-to-end 音声認識モデルについて述べる．

2.1 音響エンコーダ

音響エンコーダは，入力音響特徴列 $\boldsymbol{X}$ を長さ T の隠れ状態ベクトル列 $\boldsymbol{H} = [\boldsymbol{h}_1, \cdots, \boldsymbol{h}_T] \in \mathbb{R}^{d \times T}$ に変換する．ここで，d は次元を表す．

$$\boldsymbol{H} = \mathrm{AudioEnc}(\boldsymbol{X}). \tag{1}$$

本稿では，音響エンコーダには Conformer[6] を使用する．Conformer エンコーダは，2 つの畳み込み層，線形射影層，および M_{a} 個の Conformer ブロックで構成される．

2.2 デコーダ

デコーダは，式 (1) で生成された隠れ状態ベクトル列 $\boldsymbol{H}$ と，過去のトークン列 $y_{0:i-1} = [y_0, \cdots, y_{i-1}]$ を用いて，i 番目のトークン y_i を再帰的に推定する．

$$P(y_i | y_{0:i-1}, \boldsymbol{X}) = \mathrm{Decoder}(y_{0:i-1}, \boldsymbol{H}). \tag{2}$$

y_i は事前に定義されたサイズ K の語彙 $\mathcal{V}^{\mathrm{n}}$ に属するサブワードトークンである ($y_i \in \mathcal{V}^{\mathrm{n}}$)．

デコーダは，埋め込み層，メインデコーダブロック (Transformer や RNN-T における Prediction network や Joint network など)，および出力層で構成される．まず，埋め込み層によって，入力された過去のトークン列 $y_{0:i-1}$ が埋め込みベクトル列 $\boldsymbol{E}_{0:i-1} = [\boldsymbol{e}_0, \cdots, \boldsymbol{e}_{i-1}] \in \mathbb{R}^{d \times i}$ に変換される．

$$\boldsymbol{E}_{0:i-1} = \mathrm{Embedding}(y_{0:i-1}). \tag{3}$$

$\boldsymbol{E}_{0:i-1}$ は，式 (1) で生成された隠れ状態ベクトル列 $\boldsymbol{H}$ とともにメインデコーダブロックに入力され，隠れ状態ベクトル $\boldsymbol{u}_i \in \mathbb{R}^d$ が生成される．

$$\boldsymbol{u}_i = \mathrm{MainBlock}(\boldsymbol{H}, \boldsymbol{E}_{0:i-1}). \tag{4}$$

その後，出力層によって，トークンごとのスコア $\boldsymbol{\alpha}^{\mathrm{n}} = [\alpha_1^{\mathrm{n}}, \cdots, \alpha_K^{\mathrm{n}}]^T$ およびその確率が以下のように計算される．

$$\boldsymbol{\alpha}^{\mathrm{n}} = \mathrm{Linear}(\boldsymbol{u}_i), \tag{5}$$

$$P\left(y_i \mid y_{0:i-1}, \boldsymbol{X}\right) = \mathrm{Softmax}(\boldsymbol{\alpha}^{\mathrm{n}}). \tag{6}$$

なお，語彙サイズ K は事前に定義された静的語彙によって固定されている．これらのプロセスを再帰的に繰り返すことで，S 長のトークン列 $Y = [y_0, \cdots, y_S]$ の事後確率は以下のように定式化される．

$$P(Y \mid \boldsymbol{X}) = \prod_{i=1}^{S} P\left(y_i \mid y_{0:i-1}, \boldsymbol{X}\right). \tag{7}$$

モデルパラメータは，以下の負の対数尤度を最小化することにより最適化される．

$$L = -\log P(Y \mid \boldsymbol{X}). \tag{8}$$

式 (3) および式 (6) における埋め込み層および出力層は，提案する Deep Biasing 手法に対応するために第 3.2 節で拡張される．

3 提案手法

提案手法の概要を図 1 に示す．提案手法は 2.1 節で述べた音響エンコーダに加え，バイアスエンコーダおよび埋め込み層，出力層を拡張したデコーダで構成される．以下の節でバイアスエンコーダと拡張デコーダについて説明する．

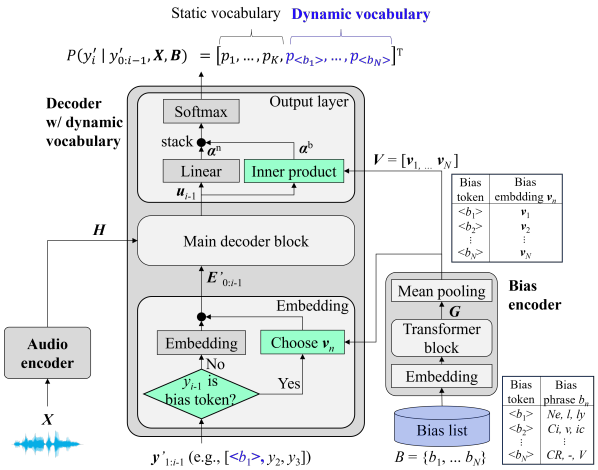

図 1: 提案手法の概要

3.1 バイアスエンコーダ

バイアスエンコーダは，埋め込み層，M_{b} 個の Transformer ブロック，平均プーリング層で構成され，バイアスリスト $\boldsymbol{B} = \{b_1, \cdots, b_N\}$ から N 個のバイアスベクトル $\boldsymbol{V} = [\boldsymbol{v}_1, \cdots, \boldsymbol{v}_N]$ を抽出する．$b_n \in \mathcal{V}^{\mathrm{n}}$ は n 番目のバイアスフレーズを表す I_n 長のサブワードトークン列（例：[*N*, *el*, *ly*]）である．まず，バイアスリスト $\boldsymbol{B}$ は，埋め込み層と Transformer ブロックによって，サブワードレベルの特徴量 $\boldsymbol{G} \in \mathbb{R}^{N \times L_{\max} \times d}$ に変換される．ここで，$L_{\max}$ はバイアスリストのに含まれる最大フレーズ長を表す．

$$\boldsymbol{G} = \mathrm{TransformerEnc}(\mathrm{Embedding}(\boldsymbol{B})). \tag{9}$$

その後，平均プーリング層を用いて，各フレーズごとの特徴ベクトル $\boldsymbol{V} = [\boldsymbol{v}_1, \cdots, \boldsymbol{v}_N] \in \mathbb{R}^{d \times N}$ を抽出する．

$$\boldsymbol{V} = \mathrm{MeanPool}(\boldsymbol{G}). \tag{10}$$

3.2 拡張デコーダ

拡張デコーダでは，事前に定義された静的語彙に加えて，動的に拡張可能な語彙 $\mathcal{V}^{\mathrm{b}} = \{<b_1>, \cdots, <b_N>\}$ を導入する．動的語彙に属する各トークン（動的トークン）は，バイアスフレーズ全体を 1 つのトークンで表現する．式 (2) に示す従来のデコーダとは異なり，式 (1) および (10) に示す $\boldsymbol{H}$，$\boldsymbol{V}$ および過去のトークン列 $y'_{0:i-1}$ に基づいて，拡張デコーダは次のトークン y'_i を拡張語彙 $\mathcal{V}^{\mathrm{n}} \cup \mathcal{V}^{\mathrm{b}}$ から推定する（$y'_i \in \mathcal{V}^{\mathrm{n}} \cup \mathcal{V}^{\mathrm{b}}$）．

$$P(y'_i | y'_{0:i-1}, \boldsymbol{X}, \boldsymbol{B}) = \mathrm{ExDecoder}(y'_{0:i-1}, \boldsymbol{H}, \boldsymbol{V}). \tag{11}$$

ここで，$y'_{0:i-1} = [y'_0, \cdots, y'_{i-1}] \in \mathcal{V}^{\mathrm{n}} \cup \mathcal{V}^{\mathrm{b}}$ は拡張語彙のトークン列を表す．例えば，「*Nelly*」という人名がバイアスフレーズである場合，拡張デコーダは静的トークン列 [*N*, *el*, *el*] ではなく，動的トークン [<*Nelly*>] を出力する．

拡張デコーダは拡張埋め込み層，メインデコーダブロック，および拡張出力層で構成される．まず，入力される過去のトークン列 $y'_{0:i-1}$ は拡張埋め込み層によって，埋め込みベクトル列 $\boldsymbol{E}'_{0:i-1} = [\boldsymbol{e}'_0, \cdots, \boldsymbol{e}'_{i-1}] \in \mathbb{R}^{d \times i}$ に変換される．ただし，式 (3) とは異なり，入力トークン y'_{i-1} が動的トークンである場合，対応するバイアスベクトル $\boldsymbol{v}_n$ が $\boldsymbol{V}$ から抽出される．それ以外の場合は通常の埋め込み層が使用される．

$$\boldsymbol{e}'_{i-1} = \begin{cases} \mathrm{Linear}(\mathrm{Embedding}(y'_{i-1})) & (y'_{i-1} \in \mathcal{V}^{\mathrm{n}}) \\ \mathrm{Linear}(\mathrm{Extract}(\boldsymbol{V}, y'_{i-1})) & (y'_{i-1} \in \mathcal{V}^{\mathrm{b}}). \end{cases} \tag{12}$$

その後，式 (4) と同様に，メインデコーダブロックは，$\boldsymbol{E}'_{0:s-1}$ を隠れ状態ベクトル $\boldsymbol{u}'_s$ に変換する．

$$\boldsymbol{u}'_i = \mathrm{MainBlock}(\boldsymbol{H}, \boldsymbol{E}'_{0:i-1}). \tag{13}$$

続いて，静的語彙のサブワードトークンスコア $\boldsymbol{\alpha}^{\mathrm{n}} = [\alpha^{\mathrm{n}}_1, \cdots, \alpha^{\mathrm{n}}_K]^T$ に加えて，内積計算を用いて動的トークンスコア $\boldsymbol{\alpha}^{\mathrm{b}} = [\alpha^{\mathrm{b}}_1, \cdots, \alpha^{\mathrm{b}}_N]^T$ を算出する．

$$\boldsymbol{\alpha}^{\mathrm{n}} = \mathrm{Linear}(\boldsymbol{u}'_i), \tag{14}$$

$$\boldsymbol{\alpha}^{\mathrm{b}} = \frac{\mathrm{Linear}(\boldsymbol{u}'_i)\mathrm{Linear}(\boldsymbol{V}^T)}{\sqrt{d}}. \tag{15}$$

$\boldsymbol{\alpha}^{\mathrm{n}}$，$\boldsymbol{\alpha}^{\mathrm{b}}$ は結合された後 ($\boldsymbol{\alpha} = [\alpha^{\mathrm{n}}_1, \cdots, \alpha^{\mathrm{n}}_K, \alpha^{\mathrm{b}}_1, \cdots, \alpha^{\mathrm{b}}_N]^T$) Softmax 関数を用いて確率に変換される．

$$P\left(y'_i \mid y'_{0:i-1}, \boldsymbol{X}, \boldsymbol{B}\right) = \mathrm{Softmax}(\mathrm{Concat}(\boldsymbol{\alpha}^{\mathrm{n}}, \boldsymbol{\alpha}^{\mathrm{b}})). \tag{16}$$

式 (7) および (8) と同様に，事後確率および損失関数は以下のように定式化される．

$$P(Y' \mid \boldsymbol{X}, \boldsymbol{B}) = \prod_{i=1}^{S'} P\left(y'_i \mid y'_{0:i-1}, \boldsymbol{X}, \boldsymbol{B}\right), \tag{17}$$

$$L' = -\log P(Y' \mid \boldsymbol{X}, \boldsymbol{B}). \tag{18}$$

ここで，$Y' = [y'_0, \cdots, y'_{S'}]$ は，動的に拡張された語彙に基づいた S' 長のトークン列を表す．式 (15) および式 (16) は，バイアスリストのサイズ N に依存する学習可能なパラメータを保持していないため，推論中にバイアスリストを動的に置き換えることができる．また，提案手法は補助的な損失関数を導入することなく，式 (16) のみで最適化される．

さらに，提案手法は，ストリーミングシステムや多言語システムを含む様々な End-to-end 音声認識モデル（例：CTC，RNN-T，Attention），およびそれらのハイブリッドシステムに適用することができる [9, 10, 11, 12, 30, 31, 32, 33] また，既存の推論アルゴリズムを変更せずに実現可能であるため，様々なジョイントデコード手法にも適用することができる [10, 11, 12, 34, 35, 36].

3.3 学習方法

訓練時には，バッチごとに正解テキストからランダムにバイアスリスト $\boldsymbol{B}$ を作成する．具体的には，各発話からトークン長が I である N_{utt} 個のバイアスフレーズが抽出され，合計で $N = N_{\mathrm{utt}} \times \mathrm{batch}$ のバイアスフレーズがバイアスリストに登録される．次に，バイアスリスト $\boldsymbol{B}$ をもとに拡張された動的語彙に基づいて，正解テキストを変更する．例えば，$y_{\mathrm{gt}} = [Hi,\ N,\ el,\ ly]$ という正解テキストから $[N,\ el,\ ly]$ $(N_{\mathrm{utt}} = 1, I = 3)$ というフレーズが抽出された場合，正解テキストは $y'_{\mathrm{gt}} = [Hi,\ <Nelly>]$ に変更される．

3.4 推論中のバイアス重み

過剰なバイアスやバイアス不足を避けるため，式 (16) に対して推論時のバイアス重みを導入する．

$$\mathrm{WeightSoftmax}_j(\boldsymbol{\alpha}, \boldsymbol{w}) = \frac{w_j \exp(\alpha_j)}{\sum_{l=1}^{(K+N)} w_l \exp(\alpha_l)}, \quad (19)$$

ここで，$\boldsymbol{w} = [w_1 . \cdots, w_{(K+N)}]^T$ は重みベクトル，j は，結合スコア $\boldsymbol{\alpha} = [\alpha_1^{\mathrm{n}}, \cdots, \alpha_K^{\mathrm{n}}, \alpha_1^{\mathrm{b}}, \cdots, \alpha_N^{\mathrm{b}}]^T$ におけるインデックスを表す．本稿では，すべての動的トークンに対し，同一のバイアス重み μ を適用する．

$$w_j = \begin{cases} 1.0 & (j \leqq K) \\ \mu & (j > K), \end{cases} \quad (20)$$

$\mu < 1.0$ の場合，動的トークンは通常トークンに比べて重みが小さく，反対に，$\mu > 1.0$ の場合，動的トークンは通常トークンに比べて重みが大きくなる．

4 評価実験

提案手法の効果を検証するため，LibriSpeech-960 (英語) および日本語の社内データセットを用いて評価実験を行った．

4.1 実験条件

提案した統合モデルの入力には，サンプリング周波数 16kHz，窓長 512 サンプル，ホップ長 128 サンプルの 80 次元メルフィルタバンク特徴量を使用し，SpecAugment [37] を適用した．音響エンコーダは，ストライド 2 の 2 層の畳み込み層，256 次元の線形射影層，および 12 層の Conformer ブロックで構成される．バイアスエンコーダと拡張デコーダは，それぞれ 6 層の Transformer ブロックを持つ．これらのエンコーダとデコーダは，Attention 層の次元は 256 の 4 つの Multi-head attention を持つ．モデル全体のパラメータ数は，バイアスエンコーダを含めて 40.58M であった．

学習時には，各バッチごとに $N_{\mathrm{utt}} = [2\text{ - }10]$ および $I = [2\text{ - }10]$ でランダムに抽出したフレーズをもとにバイアスリスト $\boldsymbol{B}$ を作成した (Section 3.3 参照)．モデルは，学習率 0.0025 で 150 エポック学習を行った．また，式 (20) に示すバイアス重み μ は，0.8 に設定して推論を行った．

提案手法は，ESPnet ツールキット [38] を用いて，Librispeech-960 [39] および日本語の社内データセットを用いて評価した．社内データセットは，日本語話し言葉コーパス (CSJ) [40]，国際電気通信基礎技術研究所が作成した日本語音声データベース (ATR-APP) [41] に加え，会議や朝礼など様々な場面で収集された 93 時間の日本語音声データで構成されている．評価指標としては，[21] と同様に，単語/文字誤り率（WER/CER），バイアスフレーズ誤り率（B-WER/B-CER），および非バイアスフレーズ誤り率（U-WER/U-CER）を用いた．提案手法の目標は，U-WER/U-CER の悪化を最小限にしながら，B-WER/B-CER を改善することである．

4.2 LibriSpeech-960 における実験結果

本節では，LibriSpeech-960 を用いた英語音声認識システムを対象とした実験結果について議論する．具体的には，4.2.1 節で WER (B-WER/U-WER) による性能評価を行った後，4.2.2 節，4.2.3 節でバイアスフレーズ長の影響および推論時における累積確率の推移を分析する．

4.2.1 WER (B-WER/U-WER) 評価

表 1 に，Librispeech-960 における，異なるバイアスリストサイズ N に対する実験結果を示している．バイアスリストサイズが $N > 0$ の場合，提案手法は U-WER をわずかに増加させるものの，B-WER を大幅に改善し，結果として WER を顕著に改善した．N が大きくなるにつれて B-WER および U-WER は悪化する傾向がみられたが，提案手法はすべてのバイアスリストサイズにおいて従来の Deep Biasing 手法 [15, 19] を上回った．

さらに，提案手法は学習データに含まれていない未知のフレーズに対しても顕著な改善を示した．具体的には，$N = 1000$ の場合の test-other において，未知のフレーズに対するベースラインの B-WER が 73.5%であったのに対し，提案手法は 19.0%に低減した．すなわち，提案手法は学習データに含まれていない未知のフレーズに対しても効果的であることがわかった．

表 1: Librispeech-960 における実験結果 ($N = 0 - 1000$)

Model	$N = 0$ (no-bias)		$N = 100$		$N = 500$		$N = 1000$	
	test-clean	test-other	test-clean	test-other	test-clean	test-other	test-clean	test-other
Baseline	**2.57**	**5.98**	2.57	5.98	2.57	5.98	2.57	5.98
(CTC/attention)	(**1.5**/**10.9**)	(**4.0**/**23.1**)	(**1.5**/10.9)	(**4.0**/23.1)	(**1.5**/10.9)	(**4.0**/23.1)	(**1.5**/10.9)	(**4.0**/23.1)
CPPNet [15]	4.29	9.16	3.40	7.77	3.68	8.31	3.81	8.75
	(2.6/18.3)	(5.9/37.5)	(2.6/10.4)	(6.0/23.0)	(2.8/10.9)	(6.5/24.3)	(2.9/11.4)	(6.9/25.3)
Attention-based Biasing	5.05	8.81	2.75	5.60	3.21	6.28	3.47	7.34
+ BPB beam search [19]	(3.9/14.1)	(6.6/27.9)	(2.3/6.0)	(4.9/12.0)	(2.7/7.0)	(5.5/13.5)	(3.0/7.7)	(6.4/15.8)
Proposed	3.16	6.95	**1.80**	**4.63**	**1.92**	**4.81**	**2.01**	**4.97**
	(1.9/13.8)	(4.6/27.5)	(1.7/**2.8**)	(4.3/**7.1**)	(1.8/**3.1**)	(4.5/**7.9**)	(1.9/**3.3**)	(4.6/**8.5**)

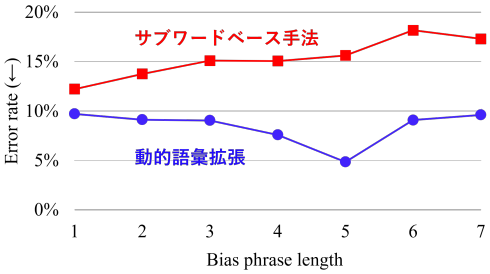

図 2: バイアスフレーズ長の影響

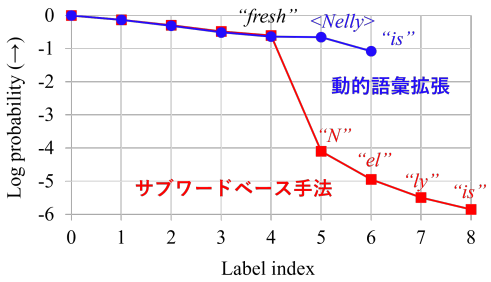

図 3: ビームサーチ中の累積対数確率

4.2.2 バイアスフレーズ長の影響

図 2 は，test-other ($N = 1000$) における，バイアスフレーズ長ごとのエラー率を示している．赤線と青線はそれぞれ，サブワードベースの従来手法 [19] と提案手法の結果を示している．サブワードベースの従来手法では，バイアスフレーズが長くなるにつれてエラー率が増加しているのに対し，提案手法はバイアスフレーズ長の増加にロバストであることがわかる．これは，従来手法は各サブワードに対して再帰的な推論を繰り返すこと累積確率が低下してしまうのに対し，提案手法は 1 つの動的トークンでバイアスフレーズ全体を扱うことでその問題を回避するためであると考えられる．これについては，次節でさらに分析を行う．

4.2.3 動的語彙の累積確率

図 3 は，式 (17) で示される累積対数確率の例を示している．動的語彙を使用しない場合 (赤線)，サブワードトークン列の確率が大きく低下しているのに対し，提案手法では (青線)，動的トークン（<*Nelly*>）に高いスコアが割り当てられている．また，動的トークンは推論中に動的に拡張されているにもかかわらず，動的トークンの前後に位置するトークン (*fresh* および *is*) の対数確率が安定している．この結果は，提案手法が動的トークンの文脈情報を適切に維持しつつ，バイアスフレーズ内のサブワードトークン間の依存関係の問題を回避していることを示している．

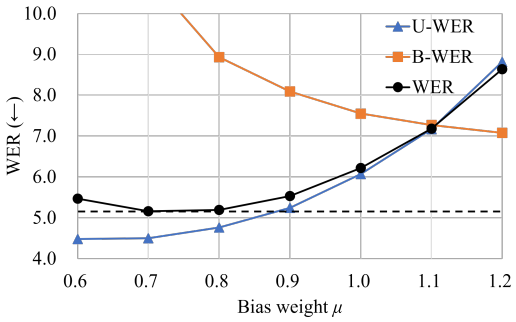

図 4: Bias weight μ の効果

4.2.4 Bias weight の効果

図 4 は，$N = 2000$ の場合の test-other における，3.4 節で述べたバイアス重み μ の WER，U-WER，および B-WER に対する効果を示している．バイアス重み μ を増加させると B-WER が改善する一方で，過剰にバイアスがかかる傾向があるため U-WER が悪化した．本実験条件では，$\mu = 0.8$ に設定することで最適な性能を示した．

適切なバイアスの程度はユーザのドメイン (バイアスフレーズの総数やバイアスフレーズが発話される頻度) によって異なるため，提案手法のように推論時に簡単にバイアス重みを調整できるこのメカニズムは有効であると考えられる．

表 2: 日本語データにおける実験結果

Model	CER	U-CER	B-CER
Baseline (CTC/attention)	9.85	8.17	21.76
NEA-ASR [26]	9.75	**8.11**	21.90
BPB beam search [19]	9.67	9.20	13.16
Intermediate Deep Biasing [20]	9.28	8.23	16.93
提案手法	**9.03**	8.93	**9.73**

表 3: 文字種ごとの性能比較

分類	認識精度 (↑) [%] 従来手法 [19]	提案手法	具体例
カタカナ	85.9	**92.7**	ジャイロ
アルファベット	81.3	**85.4**	CFD，PDCA
漢字	64.6	**83.1**	接種，風洞
英語	25.7	25.7	Sketch，Teams

4.3 日本語データセットにおける検証

続いて，社内データセットを用いた日本語音声認識システムにおける実験結果について議論する．特に，日本語は英語とは異なり，カタカナや漢字，アルファベットなど複数の文字種類を持つため，それぞれの文字種類に対する性能を分析する．

4.3.1 CER (B-CER/U-CER) 評価

表 2 は，日本語の社内データセットを用いた評価実験の結果を示している．評価時には，我々のユーザから提供されたバイアスリストを用いた．このバイアスリストには，CFD や風洞といった専門用語や接種や車高といった同音異義語 (摂取，社交) が存在する単語など，合計 N=203 単語が含まれている．

提案手法は，LibriSpeech-960 における実験結果と同様に，U-CER がわずかに悪化したものの，B-CER を大幅に改善し，その結果全体の CER を大きく改善した．

4.3.2 文字種類に対する性能分析

表 3 は，バイアスフレーズを文字種別 (カタカナ，アルファベット，漢字，英語) に分類した際の認識精度の分析結果を示している．アルファベットは「PR (ピーアール)」のようなアルファベット読みの単語を指し，英語は「Teams (チームズ)」のようにアルファベット読みではない単語を指す．複数の文字種類を含む単語については，最も多く使用されている文字に基づいて分類した (例:「CFD 計算」はアルファベットとして分類)．

表 3 より，カタカナやアルファベット，漢字で表記されたバイアスフレーズに対しては，いずれも提案手法が従来手法 [19] を上回り，80%以上の認識精度を達成した．一方，英語で表記されたバイアスフレーズに対しては，提案手法の認識精度は 25.7%にとどまった．

図 5 は代表的な推論結果例を示している．太字はバイアスフレーズ，赤字は誤認識された文字，青字は正しく認識された文字を表す．提案手法は,「風洞」や「エアロ CFD グローバル」といった，漢字，カタカナ，アルファベットで表記されたバイアスフレーズを正確に

Reference: 一番のネックは**風洞**だって事だ
Baseline: 一番のネッカーフードだって事だ
Proposed : 一番のネックは<風洞>だって事だ

Reference: それとは別の**職域接種**
Baseline: それとは別の職域摂取
Proposed: それとは別の<職域接種>

Reference: **エアロCFDグローバル**というコマンド
Baseline: このARFCDグローバルというコマンド
Proposed: <エアロCFDグローバル>というコマンド

Reference: **Sketch**モードを開いて
Baseline: スケッチモードを開いて
Proposed: スケッチモードを開いて

図 5: 推論結果例

認識したのに対し,「Sketch (スケッチ)」のような英単語は正しく認識できなかった．

これは，学習データからランダムにフレーズを抽出することでバイアスエンコーダを学習していることに起因すると考えられる．日本語データセットには「Sketch」のようなアルファベットで表記された単語が少ないため，表記と入力音声の関係が十分に学習されなかったと考えられる．とはいえ，実運用上はカタカナ表記でバイアスリストに登録しておき，後処理によってアルファベット表記に置換することで対処可能であると考えられる．

5 結論

本稿では，動的な語彙拡張を用いた新たな Deep Biasing 手法を提案した．提案手法は，バイアスリストに登録されたフレーズ全体を表現する新しいトークンを動的に導入することで，従来手法の課題であったサブワード列への依存を解消し，頻度の低さに起因する認識精度の低下を効果的に回避した．提案手法は埋め込み層と出力層の拡張のみで実現可能であり，様々な End-to-end 音声認識モデルに適用可能である．英語および日本語データセットを用いた実験では，従来手法と比較して B-WER を 3.1〜4.9 ポイント改善した．

今後は，バイアスリストのサイズが増加した際の性能改善や計算速度改善に取り組む予定である．

参考文献

[1] R. Prabhavalkar, T. Hori, T. N. Sainath, R. Schluter, and S. Watanabe, "End-to-end speech recognition: A survey," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 325–351, 2023.

[2] J. Li, "Recent advances in end-to-end automatic speech recognition," *APSIPA Transactions on Signal and Information Processing*, vol. 11, no. 1, 2022.

[3] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proc. ICML*, 2006, pp. 369–376.

[4] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proc. ICML*, 2014, pp. 1764–1772.

[5] A. Graves, "Sequence transduction with recurrent neural networks," in *Proc. ICML*, 2012.

[6] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," in *Proc. Interspeech*, 2020, pp. 5036–5040.

[7] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," *Advances in neural information processing systems*, vol. 28, 2015.

[8] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Proc. ICASSP*, 2016, pp. 4960–4964.

[9] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *Proc. ICML*, 2023, pp. 28 492–28 518.

[10] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "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.

[11] K. Hu, T. N. Sainath, R. Pang, and R. Prabhavalkar, "Deliberation model based two-pass end-to-end speech recognition," in *Proc. ICASSP*, 2020, pp. 7799–7803.

[12] Y. Sudo, M. Shakeel, B. Yan, J. Shi, and S. Watanabe, "4d asr: Joint modeling of ctc, attention, transducer, and mask-predict decoders," in *Proc. Interspeech*, 2023, pp. 3312–3316.

[13] G. Pundak, T. N. Sainath, R. Prabhavalkar, A. Kannan, and D. Zhao, "Deep context: End-to-end contextual speech recognition," in *Proc. SLT*, 2018, pp. 418–425.

[14] M. Jain, G. Keren, J. Mahadeokar, and Y. Saraf, "Contextual rnn-t for open domain asr," in *Proc. Interspeech*, 2020, pp. 11–15.

[15] K. Huang, A. Zhang, Z. Yang, P. Guo, B. Mu *et al.*, "Contextualized End-to-End Speech Recognition with Contextual Phrase Prediction Network," in *Proc. Interspeech*, 2023, pp. 4933–4937.

[16] M. Han, L. Dong, Z. Liang, M. Cai, S. Zhou *et al.*, "Improving end-to-end contextual speech recognition with fine-grained contextual knowledge selection," in *Proc. ICASSP*, 2022, pp. 491–495.

[17] C. Huber, J. Hussain, S. Stüker, and A. Waibel, "Instant one-shot word-learning for context-specific neural sequence-to-sequence speech recognition," in *Proc. ASRU*, 2021, pp. 1–7.

[18] S. Zhou, Z. Li, Y. Hong, M. Zhang, Z. Wang, and B. Huai, "Copyne: Better contextual asr by copying named entities," *arXiv preprint arXiv:2305.12839*, 2023.

[19] Y. Sudo, M. Shakeel, Y. Fukumoto, Y. Peng, and S. Watanabe, "Contextualized automatic speech recognition with attention-based bias phrase boosted beam search," in *Proc. ICASSP*, 2024, pp. 10 896–10 900.

[20] M. Shakeel, Y. Sudo, Y. Peng, and S. Watanabe, "Contextualized end-to-end automatic speech recognition with intermediate biasing loss," in *Proc. Interspeech*, 2024, pp. 3909–3913.

[21] D. Le, M. Jain, G. Keren, S. Kim *et al.*, "Contextualized streaming end-to-end speech recognition with trie-based deep biasing and shallow fusion," in *Proc. Interspeech*, 2021, pp. 1772–1776.

[22] J. Qiu, L. Huang, B. Li, J. Zhang, L. Lu, and Z. Ma, "Improving large-scale deep biasing with phoneme features and text-only data in streaming transducer," in *Proc. ASRU*, 2023, pp. 1–8.

[23] A. Bruguier, R. Prabhavalkar, G. Pundak, and T. N. Sainath, "Phoebe: Pronunciation-aware contextualization for end-to-end speech recognition," in *Proc. ICASSP*, 2019, pp. 6171–6175.

[24] Z. Chen, M. Jain, Y. Wang, M. L. Seltzer, and C. Fuegen, "Joint grapheme and phoneme embeddings for contextual end-to-end asr." in *Proc. Interspeech*, 2019, pp. 3490–3494.

[25] H. Futami, E. Tsunoo, Y. Kashiwagi, H. Ogawa, S. Arora, and S. Watanabe, "Phoneme-aware encoding for prefix-tree-based contextual asr," in *Proc. ICASSP*, 2024.

[26] Y. Sudo, K. Hata, and K. Nakadai, "Retraining-free customized asr for enharmonic words based on a named-entity-aware model and phoneme similarity estimation," in *Proc. Interspeech*, 2023, pp. 3312–3316.

[27] X. Wang, Y. Liu, J. Li, V. Miljanic, S. Zhao, and H. Khalil, "Towards contextual spelling correction for customization of end-to-end speech recognition systems," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 30, pp. 3089–3097, 2022.

[28] X. Wang, Y. Liu, J. Li, and S. Zhao, "Improving contextual spelling correction by external acoustics attention and semantic aware data augmentation," in *Proc. ICASSP*, 2023, pp. 1–5.

[29] Y. Sudo, Y. Fukumoto, M. Shakeel, Y. Peng, and S. Watanabe, "Contextualized automatic speech recognition with dynamic vocabulary," in *Proc. SLT*, 2024.

[30] Y. Wang, Z. Chen, C. Zheng, Y. Zhang, W. Han, and P. Haghani, "Accelerating rnn-t training and inference using ctc guidance," in *Proc. ICASSP*, 2023, pp. 1–5.

[31] Y. Peng, Y. Sudo, M. Shakeel, and S. Watanabe, "Owsm-ctc: An open encoder-only speech foundation model for speech recognition, translation, and language identification," in *Proc. ACL*, 2024, pp. 10 192–10 209.

[32] Y. Peng, J. Tian, B. Yan, D. Berrebbi, X. Chang, X. Li, J. Shi, S. Arora, W. Chen *et al.*, "Reproducing whisper-style training using an open-source toolkit and publicly available data," in *Proc. ASRU*, 2023, pp. 1–8.

[33] Y. Peng, J. Tian, W. Chen, S. Arora, B. Yan, Y. Sudo, M. Shakeel, K. Choi, J. Shi *et al.*, "Owsm v3.1: Better and faster open whisper-style speech models based on e-branchformer," in *Proc. Interspeech*, 2024, pp. 352–356.

[34] Y. Sudo, M. Shakeel, Y. Fukumoto, B. Yan, J. Shi, Y. Peng, and S. Watanabe, "4d asr: Joint beam search integrating ctc, attention, transducer, and mask predict decoders," *arXiv preprint*, 2024.

[35] Y. Sudo, M. Shakeel, Y. Peng, and S. Watanabe, "Time-synchronous one-pass beam search for parallel online and offline transducers with dynamic block training," in *Proc. Interspeech*, 2023, pp. 4479–4483.

[36] E. Tsunoo, H. Futami, Y. Kashiwagi, S. Arora, and S. Watanabe, "Integration of frame- and label-synchronous beam search for streaming encoder-decoder speech recognition," in *Proc. Interspeech*, 2023, pp. 1369–1373.

[37] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Proc. Interspeech*, 2019, pp. 2613–2617.

[38] S. Watanabe, T. Hori, S. Karita, T. Hayashi *et al.*, "Espnet: End-to-end speech processing toolkit," in *Proc. Interspeech*, 2018, pp. 2207–2211.

[39] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *Proc. ICASSP*, 2015, pp. 5206–5210.

[40] K. Maekawa, "Corpus of spontaneous Japanese: Its design and evaluation," in *ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, 2003.

[41] A. Kurematsu, K. Takeda, Y. Sagisaka, S. Katagiri, H. Kuwabara, and K. Shikano, "Atr japanese speech database as a tool of speech recognition and synthesis," *Speech Communication*, vol. 9, no. 4, pp. 357–363, 1990.