一般論文

Peer-Reviewed

国会会議録のための音声から書き言葉への end-to-end 変換

三村　正人[†]・河原　達也[†]

従来の音声認識システムは，入力音声に現れるすべての単語を忠実に再現するように設計されているため，認識精度が高いときでも，人間にとって読みやすい文を出力するとは限らない．これに対して，本研究では，フィラーや言い誤りの削除，句読点や脱落した助詞の挿入，また口語的な表現の修正など，適宜必要な編集を行いながら，音声から直接可読性の高い書き言葉スタイルの文を出力する新しい音声認識のアプローチについて述べる．我々はこのアプローチを単一のニューラルネットワークを用いた音声から書き言葉への end-to-end 変換として定式化する．また，音声に忠実な書き起こしを疑似的に復元し，end-to-end モデルの学習を補助する手法と，句読点位置を手がかりとした新しい音声区分化手法も併せて提案する．700 時間の衆議院審議音声を用いた評価実験により，提案手法は音声認識とテキストベースの話し言葉スタイル変換を組み合わせたカスケード型のアプローチより高精度かつ高速に書き言葉を生成できることを示す．さらに，国会会議録作成時に編集者が行う修正作業を分類・整理し，これらについて提案システムの達成度と誤り傾向の分析を行う．

キーワード：end-to-end 音声認識，話し言葉スタイル変換，整形，国会会議録

End-to-End Generation of Written-style Transcript of Speech from Parliamentary Meetings

Masato Mimura[†] and Tatsuya Kawahara[†]

Because conventional automatic speech recognition (ASR) systems are designed to faithfully reproduce utterances word-by-word, their outputs are not necessarily easy to read even when they have few speech recognition errors. To address this issue, we propose a novel ASR approach that outputs readable and clean text directly from speech by removing fillers and disfluent regeons, substituting colloquial expressions with formal ones, insertintg punctuation and recovering omitted particles, and performing other types of appropriate corrections. We formalize this approach as an end-to-end generation of written-style text from speech using a single neural network. We also propose a method to guide the training of this end-to-end model using automatically generated faithful transcripts, as well as a novel speech segmentation strategy based on online punctuation detection. An evaluation using 700 hours of Japanese Parliamentary speech data demonstrates that the proposed direct approach

[†] 京都大学情報学研究科, School of Informatics, Kyoto University

successfully generates clean transcripts suitable for human consumption more accurately at a faster decoding speed than the conventional cascade approach. We also provide an in-depth analysis on the types of edits performed by professional human editors to create the official written records of Japanese Parliamentary meetings, and evaluate the level of achievement of the proposed system in terms of each of the edit types.

Key Words: *End-to-End Speech Recognition, Speaking Style Transformation, Parliamentary Report*

1　はじめに

会議や講義，プレゼンテーションなどの音声を自動で書き起こし，アーカイブ構築に用いることは，音声認識の重要な応用の一つである．その際，真に使いやすいアーカイブを構築するためには，単に音声認識誤りを最小化するだけでなく，システム出力の可読性も考慮する必要がある．従来の音声認識システムは，発話中のすべての単語を忠実に再現するように設計されているため，認識結果は必ずしも読みやすいものとはならない．自発的な発話はフィラーや言い誤りを含むだけでなく，流暢に話されたときでも，通常非文法的であり，書き言葉に相応しくない口語特有の表現も多い．また，文の区切りが明確でなく，通常の音声認識では句読点は付与されない．したがって，音声認識結果や忠実な書き起こしを元に可読性の高い文書を作成するためには，人手による相当量の修正が必要となる (Jones et al. 2003).

音声認識結果の可読性を改善するために，話し言葉から書き言葉への自動変換の研究が数多く行われてきた．例えば，非流暢な区間の検出と削除 (Liu et al. 2006; Yeh and Wu 2006)，句読点挿入 (Paulik et al. 2008; Gravano et al. 2009; Akita and Kawahara 2011)，あるいはより一般的な話し言葉スタイル変換 (spoken style transformation = SST) (Hori et al. 2013; Shitaoka et al. 2004; Neubig et al. 2012; Sproat and Jaitly 2017) などの研究が挙げられる．これらの既存研究では，雑音のある通信路モデルや CRF (conditional randomfield)，SVM (support vector machine)，ディープニューラルネットワークなどの機械学習モデルを用いて，書き起こしから書き言葉へのテキストベースの変換が行われる．したがって，自動整形は音声認識の後処理として行われることが多く，音声認識誤りに起因する性能の低下が避けられない問題があった．また，これらのテキストベースの手法では，モデルの教師つき学習に書き言葉テキストと話し言葉テキストのペアデータを用いるため，音声に忠実な書き起こしを新たに作成する必要がある．通常コスト面の制約から大量の書き起こしは利用できないため，カバーできる音響的・言語的現象に限りがある．

これに対して，本研究では，熟練した編集者が音声を聞き取りながら同時に記録文書に適した書き言葉を作成するときのように，フィラーや言い誤りの削除，句読点や脱落した助詞の挿入，

また口語的な表現の修正など，適宜必要な編集を行いながら，音声から直接可読性の高い書き言葉スタイルの文を直接出力する新しい音声認識のアプローチを提案する．このアプローチでは，忠実な書き起こしをターゲットとする従来の音声認識モデルとは異なり，Transformer (Vaswani et al. 2017) に基づく sequence-to-sequence モデルを音声と書き言葉のペアを用いて end-to-end (e2e) に写像を最適化する．また，推論時には音声から書き言葉を直接推論する．したがって，このアプローチは，上記のようなテキストベースの自動整形を用いたカスケード型アプローチの欠点を回避できる強みを持つ．特に，書き言葉予測では，修正の対象となるような非流暢な区間ほど認識誤りが生じやすい問題があるため，音声認識結果を用いないことは，大きな改善をもたらす可能性がある．さらに，提案法は，入力中の音響的な情報に基づいて修正・編集を行うことができる (Liu et al. 2006; Neubig et al. 2012)．また，新たに忠実な書き起こしを作成する必要がないため，教師つき学習におけるデータスパースネスの問題も回避できる．本論文では，特に国会の審議音声から会議録テキストを生成するタスクに焦点を当て，提案手法の詳細な評価と分析を行う．

本論文の構成は以下の通りである．2 章では，本研究で衆議院審議音声を用いることの意義を明らかにした上で，国会会議録で行われる編集作業の分類・整理を行う．3 章では，本研究の基盤となる e2e 音声認識のための手法を概説する．4 章で音声から書き言葉を e2e で予測するための提案手法について述べた後，5 章でその実験的評価とシステム出力の詳細な分析を行う．6 章で結論を述べる．

2　データセット

国会の審議音声における忠実な書き起こしと会議録テキストの例を図 1 に示す．この例では，冒頭のフィラーと末尾の句末表現が削除され，話し言葉特有の「シンギュラリティーってのは」という表現が，より改まった「シンギュラリティーというのは」という語句に改められている．また，主語と修飾句の間に読点を挿入することで読みやすさの改善が図られている．

本研究では，以下の理由から衆議院の音声と会議録から構成した「衆議院審議音声コーパス」を用いる．

一つは，国会会議録作成の効率化と高精度化のためである．我々は衆議院審議音声の自動書

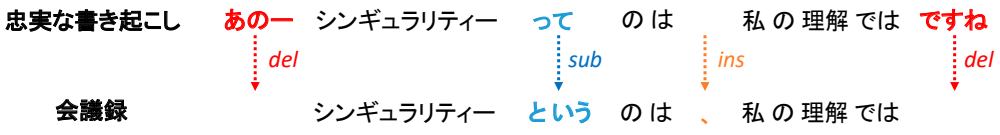

図 1　国会審議音声における忠実な書き起こしと会議録テキストのペアの例

き起こしシステムを開発しており，2011 年度から衆議院のすべての会議の会議録[1] の作成支援のために用いられている (秋田 他 2010; Kawahara 2012, 2021; Akita et al. 2009)．現行のシステムでは，まず DNN-HMM ハイブリッド型の音声認識システム (Mohamed et al. 2012; Hinton et al. 2012) を用いて音声を忠実に書き起こし，次にこの認識結果を元に編集者が必要な修正を施すことにより，最終的な会議録テキストが作成される．したがって，この工程において，音声認識モデルが直接書き言葉スタイルの文を出力することができれば，会議録作成のためのトータルな作業コストの軽減が期待される．また，本研究で用いる Transformer (Vaswani et al. 2017) などの e2e モデルの推論速度は，ハイブリッドシステムに比べて数十倍のオーダで速い．衆議院の会議の長さは，例えば本研究で評価に用いた 2015 年度のデータセットでは，平均で 3.8 時間，最大で 12 時間にも及ぶ．また，同時に多数の会議が開催される一方，並列処理のための計算機を十分確保することは困難であり，修正元となるドラフトを可能な限り迅速に作成することが求められる．したがって，処理速度は実用上重要な評価項目と言える．

次に，国会の審議音声と会議録のペアが，話し言葉と書き言葉の関係を知る上で理想的なデータであるためである．国会の審議音声，特に本会議以外の専門委員会では，話者が用意された原稿を読み上げるのでなく，自発的な発話を行うことが多い．したがって，次節で詳しく見るように，実際に行われた発話と整形済みの会議録テキストには相当程度の相違がある．また，国会会議録の性格上，編集において発話の意図が変わりかねないようなパラフレーズ等は一切許されないため，書き言葉と話し言葉の差異のみに焦点を当てることができる．国会会議録は，熟練した職業的編集者により非常に厳格かつ一貫したルールに則って作成されるため，任意性が低く，機械学習のターゲットとしても適切である．

さらには，信頼度の高い音声と会議録の大規模なペアデータが利用できるためである．話し言葉の整形の研究では，音声の書き起こしと書き言葉のペアデータを用いたテキストベースのアプローチが主であった．したがって，書き起こし作成のコストを考えると，学習データのサイズが限られていることが多かった．一方，本研究では，大量の音声-書き言葉のペアを用いて，人手による書き起こしを介することなく，直接音声から書き言葉への変換を実現することを目的とする．

本研究では 2015 年度に行われた第 189 回国会の会議から構築したデータセットを用いた．各モデルの学習データには，2015 年 6 月までに行われた 14 の本会議と 194 の委員会，計 208 会議から収集した 708 時間の音声を用いた．提案手法の評価と分析には，次節で述べる 5 会議，20 時間のデータを用いた．

[1] https://www.shugiin.go.jp/internet/itdb_kaigiroku.nsf/html/kaigiroku/kaigi_l.htm

2.1 会議録テキストにおける編集作業の分類

2015 年度に行われた第 189 会国会の一部の会議に対して音声の忠実な書き起こしを作成し，会議録と書き起こしで異なる箇所をアノテーションした．このアノテーションをもとに，会議録作成時にどのような編集・修正作業が行われているかを分析した．2015 年 7 月に行われた会議のうち，農林水産委員会第 19 号（話者ターン数 229，異なり話者数 22，5.5 時間）[2]，内閣委員会第 18 号（話者ターン数 201，異なり話者数 21，3.2 時間）[3]，厚生労働委員会第 29 号（話者ターン数 174，異なり話者数 17，4.1 時間）[4]，消費者問題に関する特別委員会第 4 号（話者ターン数 147，異なり話者数 27，3.1 時間）[5]，東日本大震災復興特別委員会第 5 号（話者ターン数 197，異なり話者数 26，4.4 時間）[6]の 5 つの会議を用いた．

編集者が行う修正を，語句の削除操作によるもの，置換操作によるもの，挿入操作によるものの 3 つに大別した上で，以下の項目 **A** から **M** のように分類した．なお，それぞれの例で，中括弧 {} で囲まれた箇所は削除を，小括弧 () で囲まれた箇所は挿入を表す．また，修正箇所は連続する場合が多いため，それぞれの例に句読点挿入など他の項目の修正箇所も含まれる場合がある．

2.1.1 削除

A. フィラーの削除　フィラー以外の機能を持たない形態を持つ語，すなわち「あー」「あのー」「いー」「うー」「えー」「えーと」「おー」「まあ」「んー」，およびそれらの変種（「えっと」など）を，この「フィラーの削除」に分類した．

例：「{ えー } アメリカ議会におきまして」,「{ まあ } しかし { あのー }(、)」

また，複数のフィラーが連続した箇所は，まとめて一つのフィラーとカウントした．

例：「{ まあ、あのー、おー } 法律の審議でございますので (、)」,「{ えー、ま } 大体もう { あの } 議論は出尽くしたところもありますけれども (，)」

B. 句末表現の削除　口語表現に特有の句末表現は削除する．

例：「正式な場におきまして { ですね }(、) 」,「自民党は { ね }(、) 本当に人情に厚くて { ね }(、)」

C. 言い誤りや繰り返しの前半部分 (reparandum) の削除　言い誤りにおいて，後半の言い直された部分のみを残して前半を削除する．言い誤り部にはフィラーが伴うことも多い．同じ語句の繰り返しも同様に削除する．

[2] https://www.shugiin.go.jp/internet/itdb_kaigirokua.nsf/html/kaigirokua/000918920150625019.htm
[3] https://www.shugiin.go.jp/internet/itdb_kaigirokua.nsf/html/kaigirokua/000218920150708018.htm
[4] https://www.shugiin.go.jp/internet/itdb_kaigirokua.nsf/html/kaigirokua/009718920150708029.htm
[5] https://www.shugiin.go.jp/internet/itdb_kaigirokua.nsf/html/kaigirokua/019718920150709004.htm
[6] https://www.shugiin.go.jp/internet/itdb_kaigirokua.nsf/html/kaigirokua/024218920150709005.htm

例：「{ 総理もこれまでああー }(大臣もこれまで)」，「{ 質問あの } (決議) に基づきまして (、)」

D. その他　「やはり」「もう」などの単語が間投詞的に用いられる例や，文脈上必要のない主語や指示語の削除が主である．これらを削除することで簡潔な文を作る．

例：「{ やはり } 事業を利用する組合員である農業者の利益 (、)」，「そこは残念だと { こう } 思っております (。)」

2.1.2　置換

E. 助詞の修正　言い誤りとは言えないが，助詞の用法が文法上誤りであるとき，正しい助詞に置換する．特に多い修正は，「が」から「は」(19 回)，「が」から「を」(19 回)，「の」から「に」(12 回)，「は」から「が」(12 回）への置換であった．

例：「農業委員の方 { が }(は)(、) 任命制があるので (、)」，「制度自体の大きな枠組みの変更 { が }(を) 検討中だということでございました (。)」

F. 口語表現の修正　話し言葉に特有の表現を，書き言葉に相応しいより改まった語句に修正する．

例：「{ いろんな }(いろいろな) 声が { えー } あるということ」，「よく承知して { ます }(います){ けども }(けれども)(、)」

G. 語順の入れ替え　書き言葉として自然になるように，連続した語句の順序を入れ替える．ただし，近年はこの修正はほとんど行われていない．

例：「{ よくないと思うんです一方的すぎると }(一方的すぎると良くないと思うんです)(。)」，「{ 両方出し手と受け手と }(出し手と受け手と両方) 考えてやって」

ただし，別の語句を挟んだ距離の離れた入れ替えについては，この項目ではなく，「その他の削除」および「その他の挿入」としてカウントする．

H. 言い誤り　言い誤りがあったが，言い直されてはいない箇所について，正しい語句に修正する．単に読みが非流暢な箇所だけでなく，意味を考慮した正しい語句への修正も含む．

例：「{ 組長 }(首長) というのは最終的には」，「{ 農林さんしょう }(農林水産省) の元同僚の皆さんに」

I. その他　省略表現の補完や，意味を考慮した表現の補足などに相当するが，頻度は非常に少ない．

例：「第八条 { の四で }(四項の二号で)」，「{ これ }(農業関係は) 二十万人どころじゃないんです (。)」

2.1.3　挿入

J. 読点，**K. 句点**　句読点を適宜挿入することにより，読みやすさを改善する．

L. 助詞の復元　話し言葉では助詞の脱落が頻繁に発生する．脱落した箇所に正しい助詞を復元し，文法的な文を作る．特に多い修正は，「を」(604 回)，「は」(549 回)，「が」(204 回)，「に」(151 回) の挿入であった．

例：「アメリカの議会 (は) この法案の審議をめぐっては」，「ここにいらっしゃる議員 (を) 初め (、)」

M. その他　項目 **H** で述べた離れた位置の語順の入れ替えの一部であることが多い．

例：「それを { 私は } この場で議論しないのはおかしいと (私は) 思うんですよ (。)」，「{ 比較的 } まだそれでも (、) そういう方の数も (比較的) あるんですが (、)」

2.2　各修正項目の頻度

忠実な書き起こし（423,786 文字）に対する会議録（403.813 文字）の差異（編集距離）は，置換 14,480 (3.4%)，挿入 19,727 (4,7%)，脱落 39,700 (9.4%) の計 73,907 文字 (17.4%) に上った．図 2 に，各修正項目の会議内における割合を示す．

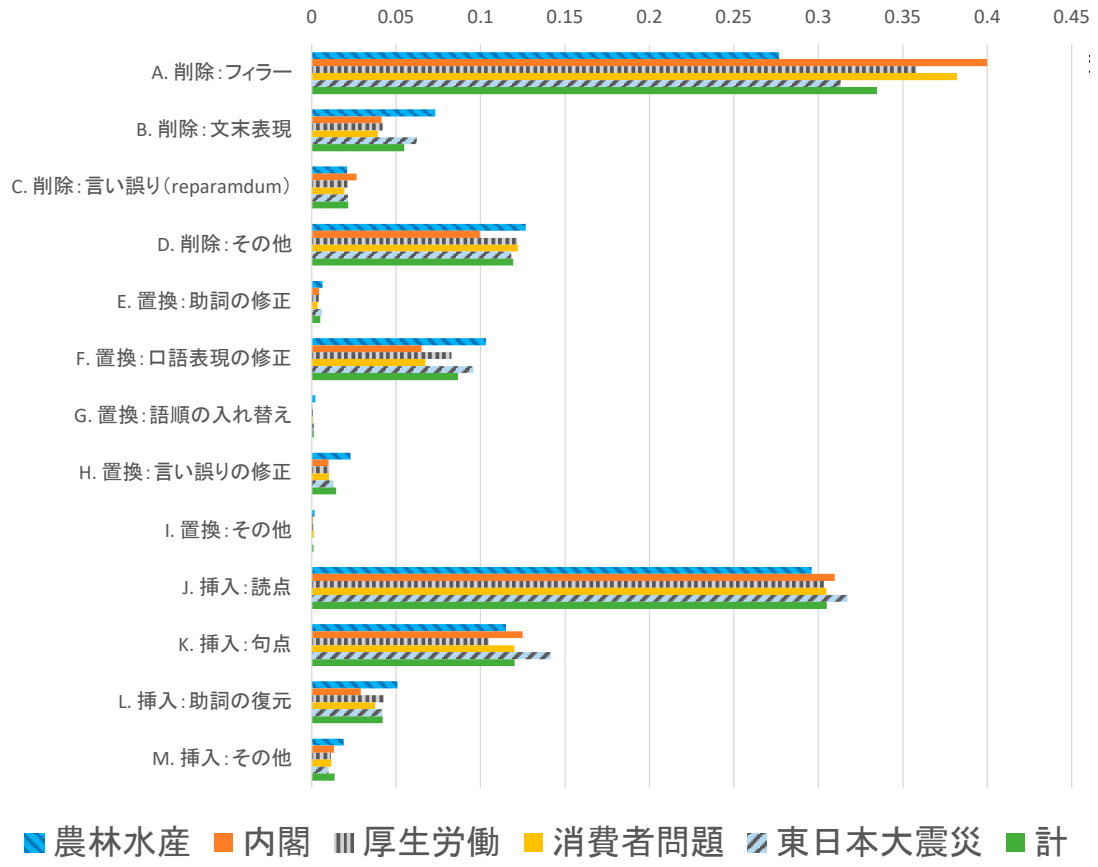

図 2　国会会議録における各編集の割合

句読点挿入を除けば，修正のほぼ半数はフィラーの除去であった．置換操作では口語表現の修正が最も多かったが，正しく行うためには，フィラーの削除より高度な規則を用いる必要があると考えられる．さらに難しいと考えられる助詞の挿入や言い誤りの削除なども相当回数出現した．一方，語順の入れ替えなど，単調なアライメントとならない修正はほとんど行われていない．

3 End-to-end 音声認識

本研究で用いる e2e 音声認識の手法について簡潔に述べる．e2e 音声認識では，単一のニューラルネットワークを用いて，入力音響特徴量系列から異なる長さのラベル系列を予測する．出力にはサブワード (Sennrich et al. 2016; Kudo and Richardson 2018) や文字など，書記素に基づく単位が用いられることが多く，直接単語系列を出力することも可能である (Audhkhasi et al. 2017; Soltau et al. 2017; Ueno et al. 2018)．HMM 音響モデル，統計的言語モデルおよび発音辞書から構成される従来のハイブリッド型システム (Hinton et al. 2012) に比べて単純な構造を持ち，サーチエラーが少ないことも加えて，一般に遥かに高速にデコードを行うことができる．

以下では，入力特徴量ベクトルの系列，または入力を畳み込みニューラルネットワークやフレームスタッキングと呼ばれる手法でサブサンプリングした系列を $\boldsymbol{X} = [\boldsymbol{x_1}, \boldsymbol{x_2}, ..., \boldsymbol{x_T}]$，ターゲットラベル系列を $Y = [y_1, y_2, ..., y_L]$ とする．

3.1 CTC 損失関数を用いたモデル

CTC(connectionist temporal classification) 損失関数に基づくモデル (Graves et al. 2006) では，ブランク (blank) と呼ばれる特殊なトークンを用いることで，系列長の異なる入力-ターゲット間の写像を行う．まず，単方向または双方向型の RNN (reccurent neural network) や次節で述べる Transformer に基づくエンコーダを用いて，入力特徴量系列 $\boldsymbol{X}$ を同じ長さの表現系列へと変換する．

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

この表現系列に線形変換 Linear() とソフトマックス関数 Softmax() を用いることで，時刻フレームごとの予測を行う．

$$\boldsymbol{O} = \mathrm{Softmax}(\mathrm{Linear}(\boldsymbol{H})) \tag{2}$$

ブランクを正解のラベル系列の任意の箇所に挿入することと同一ラベルの任意の回数の重複

を許すことで，与えられたラベル系列から入力と同じ長さを持つ拡張ラベル系列を作ることができる．この拡張ラベル系列の一つを $\boldsymbol{\pi}$ としたとき，$\boldsymbol{\pi}$ の条件付き確率は，

$$p(\boldsymbol{\pi}|\boldsymbol{X}) = \prod_{t=1}^{T} \boldsymbol{o}_{t,\pi_t} \tag{3}$$

で与えられる．$\boldsymbol{o}_{t,\pi_t}$ は，出力 $\boldsymbol{o}_t$ の拡張ラベル π_t に対応する次元の値を表す．ブランクと重複ラベルの削除を操作 β で表すとき，β によりラベル系列 Y へ縮退するすべてのアライメントパス $\boldsymbol{\pi}$ について上の条件付き確率の総和を取ることで，ラベル系列 Y の確率を計算する．

$$p(Y|\boldsymbol{X}) = \sum_{\boldsymbol{\pi} \in \beta^{-1}(Y)}^{T} p(\boldsymbol{\pi}|\boldsymbol{X}) \tag{4}$$

この確率の負の対数を CTC 損失関数と定義する．

$$loss_{CTC}(\boldsymbol{X}, Y) = -\log(p(Y|\boldsymbol{X})) \tag{5}$$

損失関数の性質から明らかなように，CTC に基づくモデルでは入力-ラベル系列間に単調なアライメントを仮定している．なお，推論時は，式 (3) に従ってラベル事後確率を計算し，各時刻で最大の確率を与えるラベルを選んだ上で，ブランクの除去と連続して出現した同一ラベルを一つにまとめることで認識結果を得る．

3.2 Transformer

Transformer (Vaswani et al. 2017) はエンコーダとデコーダサブネットワークから構成される seq2seq 型モデルの一種であり，最初機械翻訳のためのモデルとして提案されたが，後に音声認識を含む種々の音声アプリケーションでも再帰型ニューラルネットワーク (RNN) より高い性能を持つことが示された (Karita et al. 2019b; Dong et al. 2018; Karita et al. 2019a) ．Transformer に基づく音声認識モデルのアーキテクチャを図 3 に示す．

Transformer では，以下のようなマルチヘッドアテンション機構に基づき，各層の出力を順次計算する．

$$\text{Multihead}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{Concat}(head_1, head_2, ..., head_h)\boldsymbol{R}^o \tag{6}$$

$$head_i = \text{Attention}(\boldsymbol{Q}\boldsymbol{R}_i^Q, \boldsymbol{K}\boldsymbol{R}_i^K, \boldsymbol{V}\boldsymbol{R}_i^V) \tag{7}$$

$$\text{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{Softmax}(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_k}})\boldsymbol{V} \tag{8}$$

ここで，出力の次元数を d_{model}，ヘッド数を h として，$\boldsymbol{Q} \in \mathbb{R}^{t_q \times d_q}$，$\boldsymbol{K} \in \mathbb{R}^{t_k \times d_k}$，$\boldsymbol{V} \in \mathbb{R}^{t_v \times d_v}$，$\boldsymbol{R}_i^Q \in \mathbb{R}^{d_{model} \times d_q}$，$\boldsymbol{R}_i^K \in \mathbb{R}^{d_{model} \times d_k}$，$\boldsymbol{R}_i^V \in \mathbb{R}^{d_{model} \times d_v}$，$\boldsymbol{R}^O \in \mathbb{R}^{hd_v \times d_{model}}$，であり，通常 $t_k = t_v$，

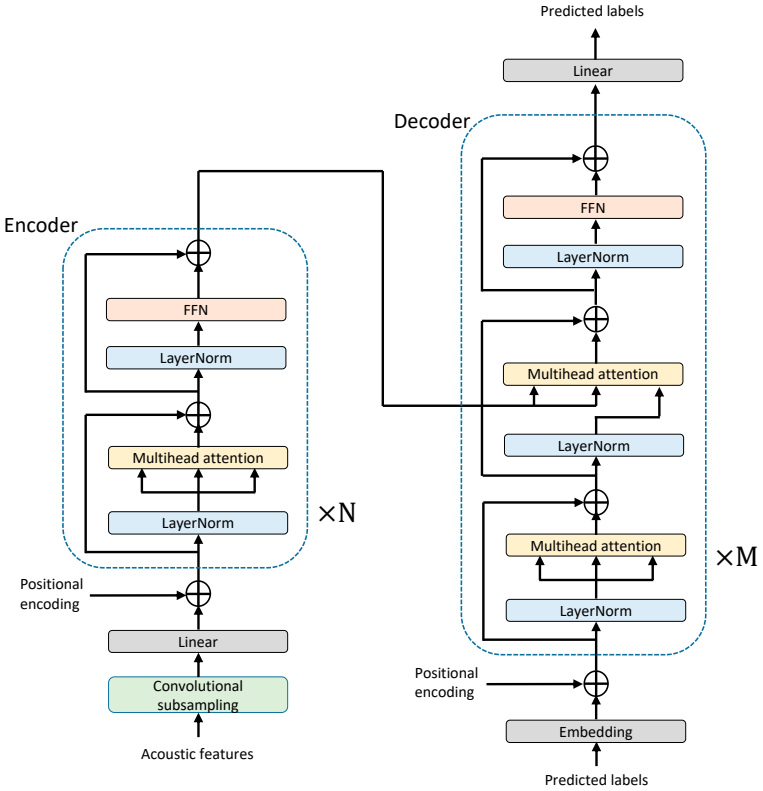

図 3　Transformer に基づく音声認識モデル

また $d_q = d_k = d_v = d_{model}/h$ である．このマルチヘッドアテンション機構を用いて，エンコーダの各層の出力 $\boldsymbol{X}_n^{enc}$ は，

$$\boldsymbol{A}_n^{enc} = \mathrm{LayerNorm}(\boldsymbol{X}_{n-1}^{enc}) \tag{9}$$

$$\boldsymbol{B}_n^{enc} = \boldsymbol{X}_{n-1}^{enc} + \mathrm{Multihead}(\boldsymbol{A}_n^{enc}, \boldsymbol{A}_n^{enc}, \boldsymbol{A}_n^{enc}) \tag{10}$$

$$\boldsymbol{C}_n^{enc} = \mathrm{LayerNorm}(\boldsymbol{B}_n^{enc}) \tag{11}$$

$$\boldsymbol{X}_n^{enc} = \boldsymbol{B}_n^{enc} + \mathrm{FFN}(\boldsymbol{C}_n^{enc}) \tag{12}$$

のように計算される．ここで，FFN() は文献 (Vaswani et al. 2017) における Position-wise Feed-Forward Network コンポーネントと同一であり，ふたつの線形層 Linear_{FFN1}，Linear_{FFN2}，および ReLU 活性化関数 (Nair and Hinton 2010) を用いて，$\mathrm{FFN}(\boldsymbol{C}_n^{enc}) = \mathrm{Linear}_{FFN2}(\mathrm{ReLU}(\mathrm{Linear}_{FFN1}(\boldsymbol{C}_n^{enc})))$ と計算される．なお，主に正弦関数に基づく位置埋め込み $\boldsymbol{P}$ を用いて，$\boldsymbol{X}_0^{enc} = \boldsymbol{X} + \boldsymbol{P}$ と定義する．

一方，デコーダの各層では，デコーダの前層の出力 $\boldsymbol{X}_{m-1}^{dec}$ とエンコーダ出力 $\boldsymbol{X}_N^{enc}$ の二つを

用いて各層の出力を計算する．ただし，$\boldsymbol{X}_0^{dec} = \mathrm{Embedding}(Y) + \boldsymbol{P}$ と定義する．

$$\boldsymbol{A}_m^{dec} = \mathrm{LayerNorm}(\boldsymbol{X}_{m-1}^{dec}) \tag{13}$$

$$\boldsymbol{B}_m^{dec} = \boldsymbol{A}_{m-1}^{dec} + \mathrm{Multihead}(\boldsymbol{A}_m^{dec}, \boldsymbol{A}_m^{dec}, \boldsymbol{A}_m^{dec}) \tag{14}$$

$$\boldsymbol{C}_m^{dec} = \mathrm{LayerNorm}(\boldsymbol{B}_m^{dec}) \tag{15}$$

$$\boldsymbol{D}_m^{dec} = \boldsymbol{B}_m^{dec} + \mathrm{Multihead}(\boldsymbol{C}_m^{dec}, \boldsymbol{X}_N^{enc}, \boldsymbol{X}_N^{enc}) \tag{16}$$

$$\boldsymbol{E}_m^{dec} = \mathrm{LayerNorm}(\boldsymbol{D}_m^{dec}) \tag{17}$$

$$\boldsymbol{X}_m^{dec} = \boldsymbol{D}_m^{dec} + \mathrm{FFN}(\boldsymbol{E}_m^{dec}) \tag{18}$$

デコーダの最終層の出力 $\boldsymbol{X}_M^{dec}$ を用いて，ネットワークのラベル単位の予測は，

$$\boldsymbol{O} = \mathrm{Softmax}(\mathrm{Linear}(\boldsymbol{X}_M^{dec})) \tag{19}$$

で与えられる．この予測とターゲットラベルのクロスエントロピ (cross entropy = CE) として Transformer の損失関数を定義する．

$$loss_{CE}(\boldsymbol{X}, Y) = \sum_{l=1}^{L} \mathrm{onehot}(y_l) \log(\boldsymbol{o}_l) \tag{20}$$

ここで，$\mathrm{onehot}(y_l)$ は，トークン y_l に対応する次元のみが 1 であり，その他の次元が 0 であるような出力クラス数と同じサイズのベクトルとする．CTC とは異なり，Transformer では入力-ターゲット間のアライメントにどのような制約も仮定しておらず，各デコーダステップにおいて入力中の任意の箇所に注目することで，より柔軟な系列間の変換が行える．

4 提案手法

本研究では，音声を入力，書き言葉をターゲットに用いて単一のニューラルネットワークを e2e で最適化し，推論時にはこのネットワークを用いて音声から書き言葉を直接予測する．以降ではこのアプローチをダイレクト書き言葉予測，用いるモデルをダイレクトモデルなどと呼ぶ．このダイレクトモデルの利点は，音声認識とテキストベースの話し言葉スタイル変換 (SST) を組み合わせた従来のカスケード方式と比較したとき，以下のように要約できる．一つは，入力中の音響的な手がかりを用いて修正が行える点である．これにより，例えば，音声の非流暢な箇所を同定・削除したり，ポーズを句点に対応付けることが可能となる．次に，カスケードモデルで問題となる音声認識誤りに起因する精度低下を回避できる点である．修正の対象となるフィラーや言い誤りなどの非流暢な箇所では特に音声認識誤りが多いと考えられるため，書き言葉予測において音声認識結果を用いない意義は特に大きい．さらには，音声認識と SST が単

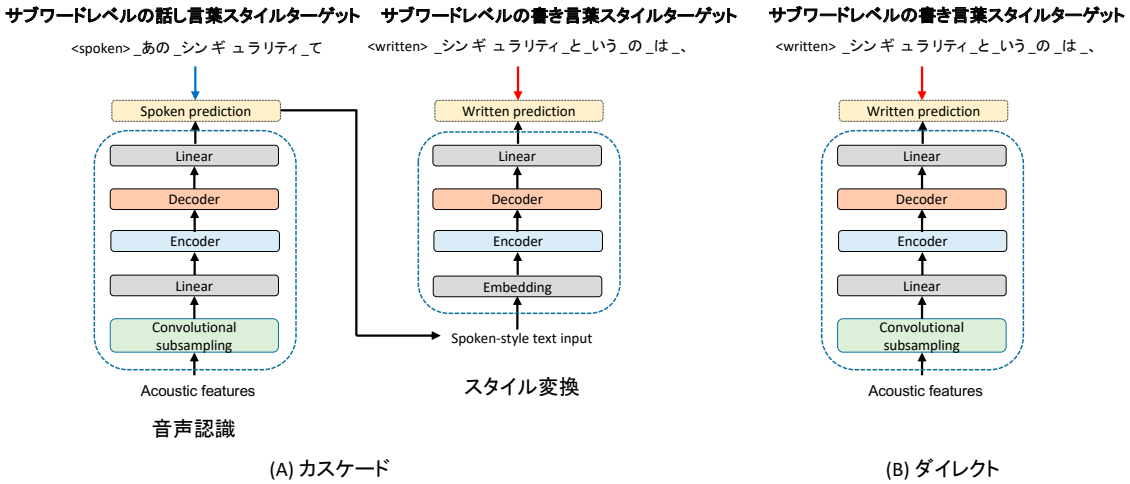

図 4　書き言葉生成のためのカスケード型アプローチ (A) とダイレクトアプローチ (B)

一のネットワークで同時に実現できるため，アーキテクチャが単純で扱いやすく，モデルサイズがコンパクトである上，推論速度もはるかに高速になる．

一方，このダイレクト変換は，入力音声中に対応する音響的なイベントを持たないラベルを挿入したり（助詞の復元や特に読点の挿入），対応するラベルを持たない音声区間を適切にスキップする必要があるため（フィラー等の除去），音声に忠実なラベルをターゲットとする従来の音声認識より難しいタスクであり，非常に柔軟なモデルが必要となる．本研究ではこのダイレクト書き言葉予測を音声翻訳 (Weiss et al. 2017) に近いタスクと考え，Transformer に基づくラベル同期型 seq2seq モデルを用いて実装する．ダイレクト方式とカスケード方式のアーキテクチャの違いを，図 4 に示す．以下の各節では，上記のような難しさを持つダイレクトアプローチのための改善法について述べる．

4.1　疑似的な書き起こしを用いた学習法の改善

2.2 節で示したように書き言葉のターゲットは音声との不一致が大きいため，音声から書き言葉へのダイレクトモデルは，通常の音声認識モデルより学習が難しいと考えられる．そこで，音声認識で用いるような忠実な書き起こしも援用することで，ダイレクトモデルの学習を補助することを考える．ただし，大規模な音声データに対して人手による書き起こしを作成するのはコスト面で現実的ではないため，4.1.1 節で会議録をもとに疑似的にこの忠実な書き起こしを自動復元する．4.1.2 節では，実際にモデルの学習に用いるための音声-書き言葉-疑似書き起こしの 3 つ組データの作成手順について述べる．4.1.3 節と 4.1.4 節で，具体的にこれらのデータを用いて書き言葉予測性能を改善する手法について述べる．

4.1.1 疑似的な書き起こしの作成

会議 m 内のある話者ターン s の会議録テキスト $W_{m,s}$ は利用可能であるが，その忠実な書き起こし $V_{m,s}$ をすべての (m, s) に対して人手で作成するのは膨大なコストが必要である．そのため，統計的機械翻訳の枠組みを用いて，自動で会議録テキスト $W_{m,s}$ から書き起こし $V_{m,s}$ を復元することを考える．一般に，書き言葉スタイルのテキスト W が与えられたとき，対応する話し言葉スタイルのテキスト V は，原理的には以下のベイズ則に基づいてデコードできる．

$$P(V|W) = P(V) \cdot \frac{P(W|V)}{P(W)} \tag{21}$$

しかし，実際には，例えばフィラーは任意の箇所に出現し得るなど，書き言葉から話し言葉への変換は本質的にランダムであり，会議録のテキストデータ W のみから話し言葉テキスト V を一意に復元することは非常に難しい．

そこで，V を上記の規則から直接デコードするのでなく，話し言葉スタイルの統計的言語モデル $P(V)$ を同様にベイズ則

$$P(V) = P(W) \cdot \frac{P(V|W)}{P(W|V)} \tag{22}$$

により推定し，得られた V の確率モデル，つまり話し言葉スタイルの言語モデル $P(V)$ を用いて音声認識を行うことにより，疑似的に V を復元することを提案する．言語モデル確率の変換は，実際には以下のように N-gram カウントの操作に基づいて行う．

$$Ngram(v_1^n) = Ngram(w_1^n) \cdot \frac{P(v|w)}{P(w|v)} \tag{23}$$

ここで，$Ngram(v_1^n)$ および $Ngram(w_1^n)$ は，話し言葉および書き言葉コーパスにおける各々の N-gram の出現回数を表す．また，v および w はそれぞれのスタイルにおける変換単位となるパターンを表す．例えば，フィラー「あのー」の挿入は，変換 $\{w = (w_{-1}, w_{+1}) \rightarrow v = (w_{-1}, \text{あのー}, w_{+1})\}$ として表される．これらのパターン間の変換規則を，ごく少量の会議録と書き起こしのパラレルデータを用いて獲得し，その確率 $P(v|w)$ および $P(w|v)$ を最尤推定により求める．また，データのスパース性に基づく影響を軽減するために，品詞情報に基づくスムージングも行う．この言語モデルスタイル変換のより詳細なアルゴリズムについては，(Akita and Kawahara 2007) を参照されたい．この手法の利点は，深層学習に基づくモデルのように大規模なペアデータを必要としない点と，小規模なペアデータから獲得された話し言葉に特定のパターンのみを変換の対象とするため，音声認識誤りを除けば，保持すべき内容語が削除されるなど，想定しない変換が行われない点である．

疑似的な書き起こし $\hat{V}_{m,s}$ の具体的な作成手順を以下に示す．

Step 1　会議 m の会議録全文のテキストデータ W_m を用いて，会議 m に依存した書き言葉スタイルの N-gram 言語モデル $P_m(W)$ を構築する．

Step 2　$P_m(W)$ に言語モデルスタイル変換を適用することにより，話し言葉スタイルの N-gram 言語モデル $P_m(V)$ を構築する．

Step 3　形態素解析から得られた各単語の発音を用いて発音辞書を構築する．この発音辞書と，他の書き起こしのある音声コーパスで事前に構築した HMM 音響モデル，および会議 m に依存した言語モデル $P_m(V)$ を用いて，この会議全体の音声データ $\boldsymbol{X}_m$ を認識する．$P_m(V)$ は，この会議に出現する単語と単語連接のみから構築したモデルであり，話題が $\boldsymbol{X}_m$ と完全に合致している上，フィラーや口語表現などの話し言葉特有の現象にも適切な確率を与えることができる．なお，言語モデル $P_m(V)$ で制約した探索空間でのみ音声認識を行う必要があるため，外部の学習データで獲得された内在的な言語モデル (McDermott et al. 2019) を包含する e2e 音声認識モデルではなく，モジュール性のあるハイブリッドシステムを用いてデコードを行う．また，Julius ツールキット (Lee et al. 2001) のショートポーズセグメンテーションアルゴリズムを用いてデコードとポーズに基づく音声の分割を同時に行う．ハイブリッドモデルを用いた音声認識はフレーム単位の予測に基づいて行われるため，デコードの過程ですべての単語にタイムスタンプが付与される．

Step 4　会議録の発言者のタグに従って，会議録テキスト W_m を話者ターン毎のテキスト $W_{m,s}$ に分割する．話者ターンの境界に特別なタグを挿入した上で，会議全体の認識結果と会議録テキストのアライメントを取得する．その上で，ターン境界タグに割り当てられた認識結果中の単語（実際には，主に長いポーズ）の時刻で音声を分割し，各話者ターンの音声 $\boldsymbol{X}_{m,s}$ を得る．

Step 5　$W_{m,s}$ を用いて，**Step 2** と同様に話者ターン s に依存した話し言葉スタイル言語モデル $P_{m,s}(V)$ を構築する．このモデルは会議全体から構築した $P_m(V)$ よりさらに強い制約を与える．$P_{m,s}(V)$ を用いて $\boldsymbol{X}_{m,s}$ を認識する．この結果得られた非常に正確な認識結果を，この話者ターンの疑似的な書き起こし $\hat{V}_{m,s}$ として用いる．

4.1.2　音声・疑似的書き起こし・会議録のアライメント

次節から述べる改善手法を用いてダイレクトモデルの学習を行うためには，扱いやすい長さのセグメントに分割した音声 $\boldsymbol{X}_{m,s,u}$ と，その疑似的な書き起こし $\hat{V}_{m,s,u}$ および会議録テキスト $W_{m,s,u}$ の組が必要である．この 3 つ組データ $(\boldsymbol{X}_{m,s,u}, \hat{V}_{m,s,u}, W_{m,s,u})$ を以下の手続きで作成する．

Julius のショートポーズセグメンテーションの過程で，音声セグメント $\boldsymbol{X}_{m,s,u}$ の開始時刻と終了時刻が付与済みであるため，分割済み音声および疑似書き起こし $\hat{V}_{m,s,u}$ は容易に取得できる．また，$\hat{V}_{m,s,u}$ のすべての単語の出現時刻も付与されている．話者ターン s の疑似書き起こし $\hat{V}_{m,s}$ と会議録テキスト $W_{m,s}$ のアライメントを取得し，セグメント境界のポーズ（しきい値

200ms 以上のポーズ）が挿入誤りとして対応付けられた箇所で会議録テキスト $W_{m,s}$ を分割することで，セグメント u の会議録テキスト $W_{m,s,u}$ を取得する．

なお，上記の疑似的な書き起こしを作成した際のショートポーズベースの音声区分化は，各セグメント内の発話内容について考慮しないため，書き言葉予測に適した分割ではない可能性が高い．音声区分化の改善手法については，後の 4.3 章で述べる．

4.1.3 エンコーダのマルチタスク学習

e2e 音声認識において，エンコーダの役割は，雑音や話者性などに起因する入力中の局所的な変動成分を除去し，理想的には音素クラスのような言語的ユニットと直接対応付けられるようなより大域的な情報のみを抽出することであると考えられる (Baevski et al. 2020)．一方，単語やサブワードは一般に複数の音節から成り立ち，コンテクストによって発音も変化するため，これらの書記素に基づく出力単位を用いる e2e 音声認識では，入力-ラベル間の対応は単純なものとはならない (Audhkhasi et al. 2017; Soltau et al. 2017)．音声と異なる言語をターゲットとする e2e 音声翻訳 (Weiss et al. 2017) では，ラベルと音声の不一致はさらに著しい．そのため，これらのモデルでは，エンコーダの学習を補助するための改善手法が用いられることが多い．

その一つは，音声との対応がより単純な音素系列や，最終的な出力ユニットより少ないクラス数のサブワードまたは文字など，より低レベルのラベルをエンコーダの出力層または中間層に補助的なターゲットとして与えることである (Ueno et al. 2018; Higuchi et al. 2022; Sanabria and Metze 2018)．e2e 音声翻訳では，通常，元言語の書き起こしを用いたマルチタスク学習 (Weiss et al. 2017) やエンコーダの事前学習 (Bérard et al. 2018) が行われる．

もう一つは，seq2seq モデルにおいて，主タスクのクロスエントロピ損失の他に，2.2 節で述べた CTC 損失関数を用いた補助的なタスクを導入することである．CTC 損失関数は，入力-ターゲット間で単調アライメントの強い制約を与えるため，適切なラベルが与えられれば，クロスエントロピ損失のみより効率的にエンコーダの最適化を行うことができる (Kim et al. 2017; Karita et al. 2019b)．

以上を踏まえて，ダイレクト書き言葉予測のためのエンコーダのマルチタスク学習を提案する．この手法では，図 5 のエンコーダ部に示すように，書き言葉の予測を行うデコーダとは別に，エンコーダ出力からなるべく音声に忠実なラベルを予測する音声認識サブタスクを定義する．このサブタスクのターゲットとして，4.1.3 章で作成した疑似的な書き起こし $\hat{V}_{m,s,u}$ を用いる．また，モデル予測とこの補助ターゲット間の損失を，CTC 損失関数を用いて計算する．疑似的な書き起こし $\hat{V}_{m,s,u}$ は制約つき音声認識を用いて復元されたものであるため発話内容に忠実であり，アライメントの単調性も保証される．マルチタスク学習における損失関数は，3 章で定義した $loss_{CTC}$，$loss_{CE}$ とサブタスクの重み λ を用いて，

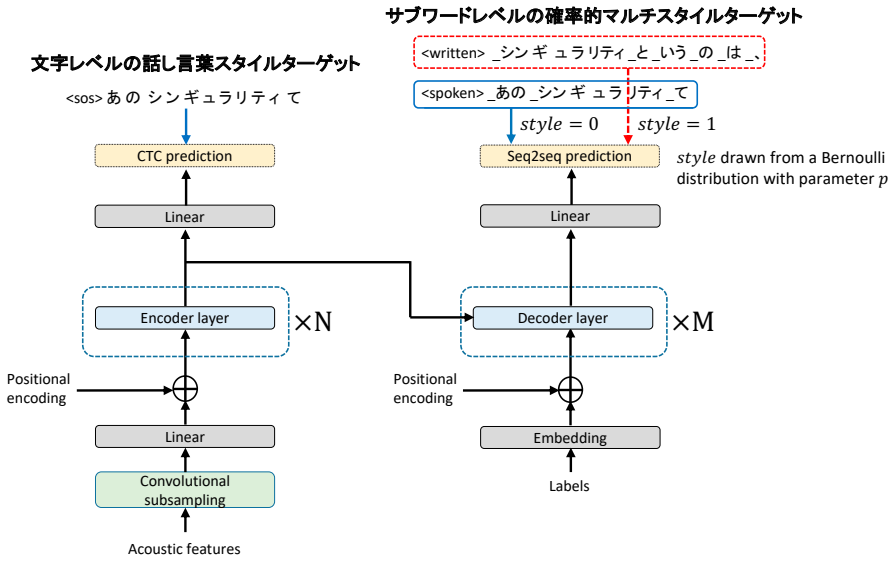

図 5　疑似的な書き起こしを用いたエンコーダのマルチタスク学習とデコーダのマルチスタイル学習

$$loss_{MTL}(\boldsymbol{X}, W, \hat{V}) = \lambda \cdot loss_{ctc}(\boldsymbol{X}, \hat{V}) + (1-\lambda) \cdot loss_{CE}(\boldsymbol{X}, W) \tag{24}$$

と定義する．

4.1.4　デコーダのマルチスタイル学習

前節で述べたマルチタスク学習では，書き起こしを用いて得られる損失はエンコーダ最上層に追加した線形層を介してエンコーダにのみ伝播し，主タスク（書き言葉予測）のための Transformer デコーダのパラメータ更新に利用されない．本節では，デコーダにも書き起こしテキストを与えて学習を補助するデコーダサイドのマルチスタイル学習を提案する．

このマルチスタイル学習は，複数言語の音声認識を単一のネットワークで行うマルチリンガル e2e 音声認識 (Watanabe et al. 2017) の枠組みを用いて行う．すなわち，日本語の書き言葉（会議録）と話し言葉（疑似書き起こし）を異なる言語とみなして，いずれかのラベルを確率的に選択し，各音声セグメントのターゲットに用いる．その際，両者で異なる文頭シンボル（‘<written>’ または ‘<spoken>’）を用いることで，文全体の予測を条件付ける．マルチリンガル学習では，低資源言語などの難しいタスクが比較的易しい英語などの音声認識により改善されることから，提案法も書き言葉の生成をより易しい音声認識タスクで補助することを目的とする．

このマルチスタイル学習の概要を図 5 のデコーダ部に示す．音声セグメント $\boldsymbol{X}_{m,s,u}$ ごとにパラメータ p のベルヌーイ分布に従う試行を行い，得られた確率変数 $style$ の値が 1 であれば会議録ラベル $W_{m,s,u}$ を，0 であれば書き起こしラベル $\hat{V}_{m,s,u}$ を選択して $\boldsymbol{X}_{m,s,u}$ のターゲットに用

いる．認識時は，開始ステップで ‘<written>’ タグを文頭シンボルとして与える．なお，アプリケーションによって音声に忠実な認識結果を得たいときは，‘<spoken>’ タグを用いる．

4.2 句読点位置を考慮した音声の自動区分化

e2e モデルを用いた音声認識や書き言葉の予測を行う上で，事前に評価に用いる各会議全体の音声データを扱いやすい長さの区間に分割しておく必要がある．その際，書き言葉の文に相当するような統語的・意味的まとまりを持った単位に分割することが理想的であるが，そのような発話境界位置のアノテーションは通常利用できない．

音声認識で主に用いられるショートポーズに基づく自動区分化では，200ms 程度のしきい値以上の無音が検出された箇所で，音声を分割する．しかし，自発的な発話において，ショートポーズの出現位置が文の区切りと一致するとは限らないため，音声の過分割を招き，予測のための重要なコンテクスト情報が失われる恐れがある．

一方，単にこのしきい値を大きくすると，信号対雑音比の低い音響条件下や話速の速い発話において，文の境界を超えるような不適切に長いセグメントが生成される傾向がある．e2e モデル，特にアテンション機構に基づくモデルでは，20 秒程度以上の長い発話に対して認識性能が顕著に低下することが知られており (Chiu et al. 2019; Pan et al. 2022)，単に長い単位へ分割するような戦略は望ましくない．また，評価環境ごとに適切なポーズ長のしきい値を決定することも困難である[7]．

以上のようなショートポーズによるセグメンテーションの問題を解決するために，句読点位置を手がかりとした音声の区分化手法を提案する．句読点は，ポーズとは異なり，発話スタイルや音響条件，話速等の話者性に関わらず出力側では安定した間隔で出現すると考えられる．そのため，句読点の出現位置を考慮することで，音声が極端な長さのセグメントに分割されるのを防ぐことができる．また，句読点に挟まれた比較的まとまった区間の情報が保持されるため，後続の書き言葉予測において一貫して十分な長さのコンテクストを用いることができる．

4.2.1 CTC モデルを用いたオンライン句読点検出

この音声区分化手法を，典型的には数時間程度の非常に長い音声を入力として，オンラインで句読点位置を予測しながら音声分割を行うタスクとして定式化する．句読点予測には，単方向型 RNN をエンコーダとする CTC モデルを用いる．このオンライン CTC モデルは，4.2.2 章でショートポーズセグメンテーションに基づいて作成した 3 つ組データ $(\boldsymbol{X}_{m,s,u}, \hat{V}_{m,s,u}, W_{m,s,u})$ を用いて学習する．すなわち，オンライン版のダイレクト書き言葉予測モデルとして構築する．

[7] 例えば，5.5 時間の農林水産委員会第 19 号を Julius のショートポーズセグメンテーションアルゴリズムを用いてポーズしきい値 200ms で区分化したとき，5,062 のセグメントに分割され，そのうち 2 秒未満の非常に短いセグメントは 1,309 であった．一方，やや長いしきい値 300ms では，セグメント数は 2,471 となり，うち 2 秒未満の短いセグメントは 340 と減少したが，20 秒以上の非常に長いセグメントが 166 と多数生成された．

なお，単方向型 RNN は Transformer より表現能力が低いため，次章の評価実験で見るように，書き起こしを用いたマルチタスク学習がモデルの収束のために必須となる．

この CTC モデルを用いて音声区分化を行うための手続きを Algorithm 1 に示す．このアルゴリズムでは，オンラインで時間同期に書き言葉予測を行いながら，句点または読点とポーズが共起した時刻で，音声を分割する．ポーズあるいは非音声区間はしきい値 N_{blank} 回以上のブランクの連続として検出できる (Yoshimura et al. 2020)．なお，句読点が挿入されたおおよその時刻は，対応する出力ノードの CTC スパイクにより知ることができるが，単方向エンコーダに基づくモデルでは，スパイクの位置は実際のラベルの出現時刻より一貫して遅れることが知られている．そのため，スパイクから一定フレーム数 (T_{margin}) さかのぼった時刻を実際の境界とする．境界を検出したエンコーダステップで RNN の状態をリセットした上で，再び時間同期の書き言葉予測を継続する．

Algorithm 1 PunctuationBasedSpeechSegmentation($\boldsymbol{X}, N_{blank}, T_{margin}, subsample\_rate$)

B : set of detected segmentation boundaries
$\boldsymbol{x}_t$: encoder input at t
$\boldsymbol{h}_t$: encoder output at t
$\boldsymbol{s}_t$: encoder state at t
$\hat{y}_t$: predicted label at t
$B \Leftarrow \{0\}$, $\boldsymbol{s}_0 = 0$, $blank\_count = 0$
$[\boldsymbol{x}_1.\boldsymbol{x}_2, ..., \boldsymbol{x}_{T/subsample\_rate}] = \text{Subsample}(\boldsymbol{X})$
for $t \in [1, 2, ..., T/subsample\_rate]$ **do**
  $\boldsymbol{h}_t, \boldsymbol{s}_t = \text{UnidirectionalRNN}(\boldsymbol{x}_t, \boldsymbol{s}_{t-1})$
  $\hat{y}_t = \text{argmax}(\text{Linear}(\boldsymbol{h}_t))$
  if $\hat{y}_t ==$ blank **then**
    $blank\_count += 1$
  else
    $previous\_nonblank\_token = \hat{y}_t$
    continue
  end if
  if $blank\_count > N_{blank}$ **then**
    if $previous\_nonblank\_token \in \{, , . \}$ **then**
      $\boldsymbol{s}_t = 0$
      $blank\_count = 0$
      $B \Leftarrow t \cdot subsample\_rate - T_{margin}$
    end if
  end if
end for
$B \Leftarrow T$
return B

句読点のみでなく，ポーズも考慮するのは，以下の理由による．CTC モデルにおいて，出力ノードのスパイクは当該トークンの周辺に存在するが，ハイブリッドモデルのように正確な出現時刻であることが保証されるわけではない．そのため，ポーズを伴わない句読点のスパイクは，実際には前後の他のサブワードの継続時間内に含まれる可能性が非常に高い．また，セグメント境界にポーズが存在しないと，文頭・文末の明確な手がかりがないため，一般に音声認識精度は低下する．さらに，ポーズと共起する句読点の予測は，そもそも信頼度が高いと考えられる．

言語的知識を用いた音声区分化手法としては，最近，RNN-Transducer (Graves 2012) などの自己回帰型モデルを用いて音声認識と Endpointing を同時に行う手法が提案されているが (Chang et al. 2019; Mahadeokar et al. 2021)，これは音声検索のようにユーザ発話の終端が容易に検出できるタスクにおいて，システム応答の遅延を最小化することを目的とした手法である．一方，会議のような人間同士の話し言葉コミュニケーションでは，最適な発話の終端を通常決定できず，検出も難しい．また，オンライン音声認識技術を用いた遅延の最小化より書き起こし精度が重視されることが多い．したがって，本研究では，発話終端のアノテーションを必要とせず，代わりに書き言葉の句読点情報を用いた音声区分化と，後段のオフライン処理によるダイレクト書き言葉予測を独立して行うアプローチを用いる．

5 評価実験

提案手法を大規模な衆議院審議音声コーパスを用いて評価した．各モデルの学習データには，第 189 回国会において 2015 年 6 月までに行われた 14 の本会議と 194 の委員会，計 208 会議から収集した 708 時間の音声を用いた．評価データには，2 章で分析に用いた 5 つの会議，すなわち 2015 年 7 月に行われた農林水産委員会第 19 号（話者ターン数 229，異なり話者数 22，5.5 時間），内閣委員会第 18 号（話者ターン数 201，異なり話者数 21，3.2 時間），厚生労働委員会第 29 号（話者ターン数 174，異なり話者数 17，4.1 時間），消費者問題に関する特別委員会第 4 号（話者ターン数 147，異なり話者数 27，3.1 時間），東日本大震災復興特別委員会第 5 号（話者ターン数 197，異なり話者数 26，4.4 時間）を用いた．これらのデータについては，音声に忠実な書き起こしの作成と 2.1 節で述べた編集者による修正についてのアノテーションを人手により行っている．開発データには法務委員会第 12 号（話者ターン 75，異なり話者数 9，2.5 時間）を用いた．モデル学習や音声区分化における種々のハイパーパラメータは，この開発データにより決定した．

音響特徴量は，80 次元の対数メルフィルタバンク出力を用いた．音響分析のための分析窓幅は 25ms とし，フレームシフトは 10ms とした．学習データおよび評価データの音響特徴量の各次元は，学習データ全体の平均・標準偏差を用いて正規化した．すべてのテキストデータは，

ChaSen-2.4.4+UniDic-1.3.9 を用いて形態素へ分割したあと，byte pair encoding (BPE) (Sennrich et al. 2016) によりトークナイズを行った．BPE に基づくサブワードの異なり数は 10k とした．

e2e 音声認識とダイレクト書き言葉予測には，同一の構造を持つ Transformer に基づく seq2seq モデルを用いた．エンコーダには 12 層，デコーダには 6 層の Transformer を用いた．ヘッド数 h，角層の出力次元数 d_{model}，FFN の中間ノード数 d_{ff} は，それぞれ $h = 4$，$d_{model} = 256$，$d_{ff} = 2,408$ とした．入力音響特徴量系列は，二層の畳み込み層を用いて系列長が 1/4 となるようにサブサンプリングを行った上で，エンコーダへ入力した．各畳み込み層は，出力チャネル数 32，カーネルサイズ 3，ストライド 1 の二次元畳み込みニューラルネットワーク (convolutional neural network = CNN) (LeCun and Bengio 1995) とストライド 2 の二次元プーリング層により構成した．CNN 出力は，ReLU 関数 (Nair and Hinton 2010) を用いて非線形変換を行った．なお，比較のために CTC 損失関数を用いた e2e 音声認識およびダイレクト書き言葉予測モデルも構築した．CTC モデルでも，seq2seq モデルと同一の 12 層の Transformer エンコーダを用いた．

比較のためのカスケード方式（図 4 の (A)）では，上記の e2e 音声認識と，テキストベースで話し言葉から書き言葉への変換を行う SST モデルを組み合わせることで，音声から書き言葉の予測を行った．テキストベース SST モデルは，音声翻訳におけるカスケードモデル (Bentivogli et al. 2021) の例にしたがって，Transformer により実装した．エンコーダとデコーダの構成は，上記の e2e 音声認識およびダイレクトモデルと同一とした．ただし，これらのモデルにおけるサブサンプリング層の代わりに，サブワードのための埋め込み層を用いた．なお，e2e 音声認識と SST モデルは，4.2 章で述べた疑似的書き起こしを用いた準教師つき学習 (lightly-supervised training)(Lamel et al. 2001) の枠組みにより構築した．すなわち，音声認識は音響特徴量を入力・疑似書き起こしをターゲットとして，また SST モデルは疑似書き起こしを入力・会議録テキストをターゲットとして，それぞれ学習した．推論時は，すべての Transformer モデルでビーム幅 6 のビームサーチを行った．一方，CTC モデルでは一般にビームサーチの効果が低いことから，文献 (Graves et al. 2006) の greedy search アルゴリズムを用いてデコーディングを行った．

e2e 音声認識およびダイレクトモデルの学習時は，適応的 Specaugment (Park et al. 2020) による動的データ拡張を行った．マスク確率等はすべて文献 (Park et al. 2020) の設定に従った．モデルは Adam オプティマイザー (Kingma and Ba 2015) を用いて最適化した．すべてのモデルで (Dong et al. 2018) と同様の学習率のスケジューリングを行った．すなわち，ステップ数 n における学習率 $lrate(n)$ を，

$$lrate(n) = k \cdot d_{model}^{-0.5} \cdot \min(n^{-0.5}, n \cdot warmup\_n^{-1.5}) \tag{25}$$

とし，特に $warmup\_n = 25,000$，$k = 4.0$ とした．エンコーダマルチタスク学習（4.1.3 章）において，音声認識サブタスクの重みは $\lambda = 0.1$ とした．また，先行研究の知見を踏まえて (Ueno et al. 2018; Sanabria and Metze 2018)，サブタスクのターゲットには，サブワードでなく，より

クラス数が少なくクラスあたりの学習事例数が多いと期待される文字の系列を用いた．異なり文字数は 2,840 であった．デコーダのマルチスタイル学習において，スタイル選択に用いるベルヌーイ分布のパラメータ p は 0.5 とした．すべてのモデルでサイズ 120 のミニバッチを用いた誤差逆伝播法に基づく学習を 100 エポック分行い，最終的なネットワークは，開発セットに対して最も低い誤り率を与えた 10 のチェックポイントを平均することで構築した．

提案モデルの実用上の信頼度を評価するために，参考のため，現行の審議音声自動書き起こしシステムで運用実績のあるハイブリッド音声認識システムの性能とも比較を行う．ただし，ハイブリッドシステムでは大規模なテキストデータから構築した統計的言語モデルおよび発音辞書が利用できる大きな利点があるが，音響モデルが以下に述べるように単純なフィードフォワード型ニューラルネットワークにより構成されているため，認識性能面で Transformer を用いた e2e モデルと公平に比較できない点に留意する．このハイブリッドシステムにおいて，音響モデルは，9k クラスの triphone 状態 (senone) を識別する 7 層フィードフォワードニューラルネットワークと HMM から構成する DNN-HMM (Hinton et al. 2012) モデルを用いた．音響モデルは Kaldi ツールキット (Povey et al. 2011) を用いて学習した．公平のため，この音響モデルの学習には e2e モデルと同じ 2015 年度の 700 時間のみを用いた．言語モデルは 2006 年度から 2015 年度までの衆議院会議録テキストを用いて学習した書き言葉 trigram モデルに話し言葉スタイル変換 (Akita and Kawahara 2007) を適用することで構築した．デコードには Julius を用いた．Julius のデコードでは，4.1.1 章の疑似的な書き起こしの作成時と同様に，ショートポーズセグメンテーションアルゴリズム（ポーズ長のしきい値 200ms）を用いて音声区分化と音声認識を同時に行った．

4.1.1 章で提案した疑似的な書き起こしの作成では，2003 年度に行われた一部の会議（主に予算委員会）の書き起こし 666K 単語と対応する会議録のパラレルデータを用いて変換規則 $P(W|V)$ および $P(V|W)$ を学習した．この変換モデルを会議録の各話者ターンの文のみから構築した書き言葉言語モデルに適用することで，制約付き音声認識に用いる話者ターンの内容および発話スタイルにともにマッチした強い言語モデルを構築した．一方，音響モデルには，4.1.1 節と同様の手法で作成した 2009 年から 2011 年度の疑似書き起こしを用いて学習した GMM-HMM モデルを用いた．

評価データの音声区分化は，5.4 節の比較実験以外では，CTC に基づく自動区分化手法（4.2 章）を用いて行った．学習データは，4.1.2 章のアライメントの過程で得られたショートポーズベースの分割をそのまま用いたものと，5.4 節で詳しく述べる句読点ベースの手法を用いたものの二通りを構築したが，5.4 節の比較実験以外では，句読点ベースで分割したデータを用いた．

推論時の実行時間の計測には，CPU は AMD-Epyc7262，GPU は NVidia-RTX-3090（24G メモリ）を用いた．

5.1　ベースライン音声認識モデルの評価

提案法である書き言葉予測モデルの評価を行う前に，最初に，疑似書き起こしを用いた準教師付き学習により構築したベースライン e2e 音声認識モデルの性能を評価する．ここで，通常の音声認識モデルとしての評価は，人手で作成した音声に忠実な書き起こしに対するシステム出力の誤り率を用いて行う．表 1 に，忠実な書き起こしに対する CTC および Transformer に基づく e2e 音声認識モデルの文字誤り率および推論時間の実時間ファクタを示す．

いずれの e2e モデルも，CTC で 9.7%，Transformer で 9.1%と低い誤り率を達成した．CPU の推論速度では，Transformer で実時間の 0.09 倍，CTC で実時間の 0.009 倍の高速な推論が可能であった．また，GPU 上では推論はさらに高速となった．CTC と Transformer の比較では，速度では劣るものの，Transformer がやや低い誤り率を示した．さらに，4.1.4 節で述べたデコーダのマルチスタイル学習を行うことで，誤り率が絶対値で 0.9 ポイントと大幅に改善した（“Transformer 音声認識 + デコーダマルチスタイル学習”）．これは，準教師付き学習で用いたラベルに含まれる認識誤りの影響が，誤りを含まない会議録ターゲットを用いることで軽減したためである可能性がある．マルチスタイル学習はダイレクト書き言葉予測の改善を目的とした手法であるが，<spoken> 文頭タグを用いた音声認識モデルとしての性能も向上したことから，準教師付き学習の改善法としても有効であると考えられる．このことは，アプリケーションによってはなるべく忠実な書き起こしが必要になるため，重要な知見であると言える．

参考のため，現行の書き起こしシステムで用いられているハイブリッドシステムの結果も示す．いずれの e2e モデルも，音声認識性能と推論速度の両面でこのハイブリッドシステムを上回っており，書き起こしシステムにおいて実用上問題のない性能を達成できていることがわかる．

次に，これらの音声認識モデルの会議録テキストに対する文字誤り率を，表 2 に示す．ただし，音声認識モデルは句読点を一切出力することができないため，公平のため，この表では句読点は誤り率の算出から除外した．会議録に対する誤り率では，忠実な書き起こしに対する誤り率に比べて，いずれのモデルでも特に挿入誤りが顕著に増加した．ハイブリッドシステムで

手法	誤り種別				実時間ファクタ	
	置換	脱落	挿入	合計	CPU	GPU
ハイブリッドシステム (参考)	4.3	3.2	3.2	10.7	2.45	—
CTC 音声認識	3.0	**2.7**	3.6	9.3	**0.009**	**0.003**
Transformer 音声認識	2.8	3.0	3.2	9.1	0.09	0.04
Transformer 音声認識 + デコーダマルチスタイル学習	**2.7**	2.8	**2.7**	**8.2**	0.09	0.04

表 1　音声認識モデルの忠実な書き起こしに対する文字誤り率 (%) および処理時間の実時間ファクタ

手法	誤り種別				実時間ファクタ	
	置換	脱落	挿入	合計	CPU	GPU
ハイブリッドシステム（参考）	4.5	2.7	14.1	21.3	2.45	—
＋ 語彙的フィラーの削除	4.1	3.1	6.6	13.8	2.45	—
CTC 音声認識	3.3	2.9	15.2	21.4	**0.009**	**0.003**
Transformer 音声認識	3.0	2.6	14.2	19.7	0.09	0.04
＋ Transformer スタイル変換	3.0	3.8	**3.2**	9.9	0.18	0.08
Transformer 音声認識 + デコーダマルチスタイル学習	**2.9**	**2.3**	13.6	18.8	0.09	0.04
＋ Transformer スタイル変換	3.0	3.4	3.3	**9.7**	0.18	0.08

表 2 音声認識モデルの会議録に対する文字誤り率 (%) および処理時間の実時間ファクタ（句読点は除外した）

は，発音辞書中の語彙的なフィラーを除去することで，誤り率が絶対値で 7.5 ポイント改善した．この結果は，会議録における修正のほぼ半数がフィラーの除去であるという 2 章の分析と符合している．

会議録に対する誤り率においても，Transformer に基づく seq2seq モデルが CTC より高い性能を示した．さらに，Transformer に基づくテキストベース SST を後処理として用いることにより（カスケード方式の書き言葉予測），挿入誤りが大幅に削減された．テキストベース SST によるこの改善幅がハイブリッドモデルにおける語彙的フィラーの除去より顕著に大きいことから，Transformer に基づく系列間変換により，単純なルールベースの手法より高度な編集操作が可能であることがわかる．このカスケード方式の書き言葉予測において，初段の e2e 音声認識モデルとしてマルチスタイルモデルを用いることにより，さらに誤り率は改善し，全体で最も低い誤り率 (9.7%) を達成した．

以上のように，準教師付き学習で構築したモデルが全般に高い水準の音声認識精度と書き言葉予測精度を達成したことから，4.1.1 節の手法で十分信頼性のある書き起こしが作成可能であることがわかる．なお，学習データについては正解の書き起こしが存在しないため，評価データにおいて同一手法で疑似書き起こしを作成し，精度を評価したところ，人手書き起こしに対する誤り率は，置換誤りが 1.5%，脱落誤りが 2.8%，挿入誤りが 2.6%，計 6.9%となり，表 1 における一般的な言語モデルを用いたハイブリッドシステムの誤り率 (10.7%) よりはるかに高い精度となった．

5.2 カスケードモデルとダイレクトモデルの比較

本節では，音声からの書き言葉予測において，カスケードモデルと提案法であるダイレクトモデルの比較を行う．システム出力の会議録に対する誤り率を表 3 に示す．これらのモデルは

手法	誤り種別				実時間ファクタ	
	置換	脱落	挿入	合計	CPU	GPU
Transformer カスケード	3.3	4.5	3.5	11.4	0.18	0.08
Transformer カスケード + デコーダマルチスタイル学習	3.3	4.1	3.6	11.0	0.18	0.08
+ 外部言語モデル統合	3.3	4.3	3.5	11.0	0.22	0.10
CTC ダイレクト	(did not converge)					
CTC ダイレクト + エンコーダマルチタスク学習	**2.4**	4.2	3.2	9.8	**0.009**	**0.003**
単方向型 CTC ダイレクト	(did not converge)					
単方向型 CTC ダイレクト + エンコーダマルチタスク学習	3.6	5.2	3.5	12.3	0.03	0.01
Transformer ダイレクト	2.8	3.6	3.2	9.6	0.09	0.04
Transformer ダイレクト + エンコーダマルチタスク学習	**2.1**	2.8	3.2	8.2	0.09	0.04
Transformer ダイレクト + デコーダマルチスタイル学習	2.2	2.9	3.0	8.2	0.09	0.04
Transformer ダイレクト + 両方	**2.1**	3.0	**2.7**	**7.8**	0.09	0.04
+ 外部言語モデル統合	2.2	**2.3**	4.6	9.1	0.12	0.06

表 3　ダイレクト書き言葉生成モデルの評価（数値は会議録に対する文字誤り率 (%) および処理時間の実時間ファクタ）

句読点を含めた書き言葉の予測を行うため，表 2 と異なり，句読点も誤り率の算出に用いた．カスケードモデルは，表 2 の Trasnformer 音声認識と SST の組み合わせの結果と同一である．

Transformer に基づくダイレクトモデル（“Transformer ダイレクト”）は，初段の音声認識においてマルチスタイルモデルを用いた最良のカスケードモデル（“Transformer カスケード + デコーダマルチスタイル学習”）より，絶対値で 1.4 ポイント低い誤り率を実現した．また，このダイレクトモデルは，カスケードモデルの 2 倍の速度で書き言葉を出力できた．このことから，4 章の冒頭に述べた性能と速度の両面におけるダイレクトモデルの優位性が確かめられた．

一方，CTC に基づくダイレクトモデル（“CTC ダイレクト”）は，書き言葉ターゲットだけでは 100 エポックまで学習が収束せず，意味のある結果を出力するに至らなかった．この結果から，音声認識に加えて多くの削除・置換・挿入操作を行う必要のある書き言葉予測タスクにおいて，Transformer に基づく seq2seq モデルが CTC に基づくモデルより適していることがわかる．また，このことは，表 1 において Transformer と CTC モデルが音声認識としてはほぼ同等の性能を示したことと対照的であり，書き言葉予測は通常の音声認識とは明確に異なる性質を持つタスクであることがわかる．

5.3　疑似的書き起こしを用いた学習法の効果

次に，書き言葉のダイレクト予測において，疑似的書き起こしを用いた改善手法（4.1.3 節，4.1.4 節）の評価を行う．

疑似書き起こしを用いたエンコーダのマルチタスク学習（4.1.3 節）（表 3 の “Transformer ダイレクト + エンコーダマルチタスク学習”）では，書き言葉のみを用いて学習したモデル（表 3 の “Transformer ダイレクト”）に比べて，1.4 ポイントと大幅に誤り率が改善した．また，CTC に基づくダイレクトモデルは，このマルチタスク学習を用いることで学習が収束し，9.8%と妥当な性能を示すに至った．ただし，依然マルチタスク学習を用いない Transformer モデルの性能に及ばなかった．また，表 3 の “単方向型 CTC ダイレクト” および “単方向型 CTC ダイレクト + エンコーダマルチタスク学習” の行に，4.2 章で提案した音声区分化手法に用いるオンライン CTC モデルの書き言葉予測性能を示す．表からわかるように，このオンラインモデルも，エンコーダのマルチタスク学習により初めて収束した．これらの結果から，音声に忠実なラベルを併用することが，書き言葉予測性能の改善において非常に重要な役割を果たすことがわかる．また，人手による正解ラベルでなく，統計的機械翻訳に基づいて自動生成されたラベルによりこれらの大幅な改善が得られたことは，特に重要な点である．

このマルチタスク学習におけるターゲットラベルの種類の影響を表 4 に示す．ターゲットとして疑似書き起こしを用いる提案法において，サブワードレベルのラベルを用いるより文字ラベルを用いる方が有意に性能が高かった．これは，クラスあたりの学習事例数が多いトークンを用いることで，マルチタスク学習の効率が改善されたためと考えられる．また，サブタスクのターゲットとして書き言葉を用いたとき，性能はむしろ大幅に低下した．このことは，CTC に基づくダイレクトモデルが書き言葉ターゲットのみでは収束しなかった結果と併せて，CTC 損失関数が書き言葉のようなターゲットを扱うのに適さないことを示している．この結果から，ダイレクト書き言葉予測において，単に CTC によるマルチタスク学習 (Karita et al. 2019a) が有効なのではなく，音声に忠実なラベルを用いることが性能改善にとって本質的であったことがわかる．

次に，デコーダのマルチスタイル学習（4.1.4 節）の効果を評価する．デコーダのターゲットとして疑似書き起こしと書き言葉ターゲットを確率的に併用したマルチスタイル学習により，絶対値で 1.4 ポイントの改善が得られた（表 3 の “Transformer ダイレクト + デコーダマルチスタイル学習”）．さらに，エンコーダのマルチタスク学習とデコーダのマルチスタイル学習を同

サブタスクのターゲット	誤り種別			
	置換	脱落	挿入	合計
疑似的な書き起こし，文字	**2.1**	**2.8**	3.2	**8.2**
疑似的な書き起こし，サブワード	2.6	3.2	**3.0**	8.7
書き言葉，文字	2.9	4.2	4.2	11.3

表 4 マルチタスク学習におけるサブタスク・ターゲットの影響（数値は会議録に対する文字誤り率 (%)）

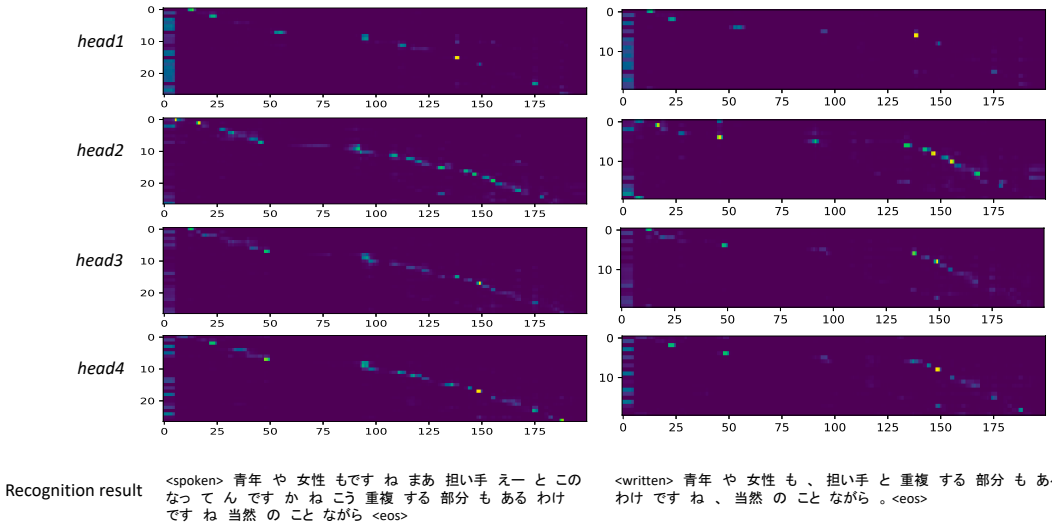

図 6　マルチスタイルモデルの出力とデコーダ最上層のアテンション重みの例．左が <spoken> 文頭タグで条件づけた忠実な音声認識出力．右が <written> 文頭タグを用いた書き言葉出力．正解は，忠実な書き起こしが「青年や女性もですね担い手えーとこのなんていうんですかねこう重複する部分もあるわけですね当然のことながら」，会議録が「青年や女性も担い手と重複する部分もあるわけですね、当然のことながら。」

時に用いることで，誤り率はさらに 0.4 ポイント減少した（表 3 の “Transformer ダイレクト + 両方”）．

書き言葉予測におけるダイレクトモデルの振る舞いを理解するために，マルチスタイルモデルにおいて`<spoken>`文頭タグを与えて e2e 音声認識モデルとして動作したときと，`<written>`タグを与えてダイレクト書き言葉予測モデルとして動作したときのデコーダ最上層のクロスアテンション重みの例を図 6 に示す．右列のアテンション重みを見ると，100 から 200 フレームあたり[8]までの「ですねまあ」，400 フレームから 520 フレーム辺りまでの「この何て言うんですかね」の二つの非流暢な領域を適切にスキップしている様子がわかる．また，それらに囲まれた助詞「と」，および「担い手」を正しく出力した．

5.4　言語モデル統合の効果

e2e 音声認識では，大規模な言語資源を用いて構築したニューラル言語モデルを推論時に統合する shallowfusion (Gulcehre et al. 2015; Chorowski and Jaitly 2017) などの外部言語モデル統合が広く精度改善のために用いられる．書き言葉のダイレクト生成においても，会議録テキストはペアデータよりはるかに容易に入手できるため，大規模な書き言葉言語モデルが構

[8] 図中のアテンション重みの横軸はサブサンプリング後のエンコーダステップであることに注意する．

築可能である．本節では，2006 年度から 2015 年度までの会議録テキスト（28M 単語）を用いた外部言語モデルの効果を，カスケードモデルとダイレクトモデルにおいて評価する．統合手法には shallow fusion を用いた．言語モデルは 12 層の Transformer を用いて実装した．ヘッド数 h，出力次元数 d_{model}，FFN の中間ノード数 d_{ff} は，それぞれ $h = 4$，$d_{model} = 256$，$d_{ff} = 2,408$ とした．各デコーディングステップ l における統合スコアは，スタイル変換モデルあるいはダイレクトモデルの出力確率 p_{am} と言語モデルの出力 p_{lm}，および言語重み λ を用いて，$\log p(y_l|\boldsymbol{X}, y_1, y_2, ..., y_{l-1}) + \lambda \log p_{lm}(y_l|y_1, y_2, ..., y_{l-1})$ と計算した．カスケードモデルとダイレクトモデルに言語モデル統合を用いたときの会議録に対する誤り率を表 3 の “外部言語モデル統合” の行に示す．大規模言語モデルの統合は，カスケードモデルでは効果がなく，ダイレクトモデルでは挿入誤りが有意に増え，むしろ精度の低下をもたらした．書き言葉予測は音声に忠実でないラベルの出力も行うように学習されるが，外部言語モデルを用いることでより音声と無関係な不適切なラベルの挿入が促進された可能性がある．

5.5 句読点を考慮した音声区分化の効果

句読点を手がかりとした音声区分化手法（4.2 章）を評価する．区分化の基準としては，連続ブランクとして検出されたポーズと句読点が共起したときに分割する手法（提案法），ポーズと文末シンボル (<eos>) が共起したときに分割する手法，ポーズのみを用いる手法の 3 つを比較する．Algorithm1 において，ポーズ検出に用いる連続 blank 数のしきい値 (N_{blank}) を変化させたときの開発セットに対する誤り率の推移を図 7 に示す．マージン T_{margin} は各しきい値で別途調整した[9]．各データ点の近傍に各手法としきい値の組み合わせで生成されたセグメントの数を併せて示す．なお，書き言葉予測には，共通してエンコーダマルチタスク学習で構築した Transformer を用いて行った．

この結果から，句読点を手がかりに区分化を行う提案法が，ブランク数 15 程度まで安定して高い精度を与えることがわかる．ポーズのみに基づく手法は，提案法より全体に誤り率が高く，またしきい値依存性も高かった．提案法が最も低い誤り率を与えた設定 ($N_{blank} = 10$) では，音声は 1,228 のセグメントに分割されたが，ポーズに基づく手法では，$N_{blank} = 25$ でこれと同程度のセグメント数となった．しかし，誤り率の比較では，後者は 15.7%と大幅に提案法 (7.3%) を上回った．セグメント長の標準偏差は，前者で 4.95 秒であったのに対して，後者で 7.06 秒となり，分割されたセグメント長に大きなばらつきがあった．また，文末シンボル<eos>を用いた手法は，文や話者の境界とは無関係に，一貫して 20 から 25 秒程度の長いセグメントを生成する傾向があり，学習データ中にこれらの長さのセグメントがほとんど出現しないため，性能も低かった．

[9] $N_{blank} = 5, 10, 15, 20, 25$ に対して，それぞれ $T_{margin} = 20, 30, 40, 40, 40$ を用いた．ただし，T_{margin} の影響は全体に軽微であった．

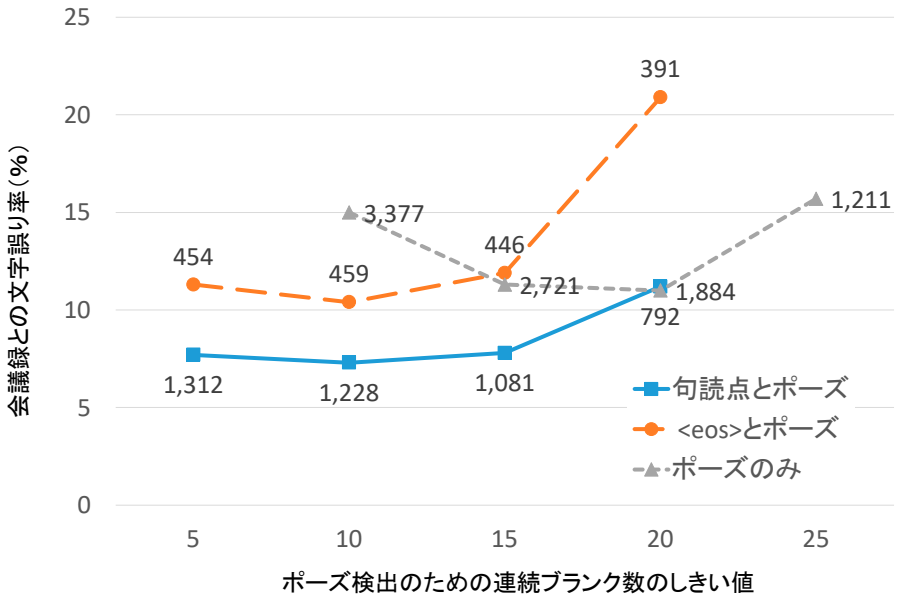

図 7　CTC に基づく区分化手法におけるポーズ検出のためのブランク数のしきい値の影響（数値は開発セットに対する書き言葉生成精度 (%)．各データ点に生成されたセグメント数を付す．）

次に，学習・評価データに対する区分化手法の組み合わせについて評価を行った．ポーズに基づく手法には，学習・評価データとも，Julius のショートポーズセグメンテーションアルゴリズムを用いた．しきい値は音声認識で一般に用いられる 200ms とした．句読点に基づく手法としては，学習データでは，ショートポーズセグメンテーションで得られたセグメント境界時刻のうち，会議録とのアライメントで句読点に対応付けられた時刻でのみ，区分化を行った[10]．評価データは CTC セグメンテーション手法により分割した．パラメータは，開発データを用いた図 1 の実験で得られた最適な値を用いた ($N_{blank} = 10$，$T_{margin} = 30$)．

表 5 に，学習・評価時における区分化手法の 4 通りの組み合わせに対する評価データの誤り率を示す．学習・評価時のいずれも句読点を考慮することで，いずれもショートポーズで分割したときより絶対値で 1.6 ポイントと大きな改善が見られた．また，学習データをポーズ，評価データを句読点で区分化したとき，脱落誤りが顕著であった．ショートポーズセグメンテーションではごく短いセグメントの占める割合が高く，評価時に比較的長いセグメントを正しく認識できなかったためと考えられる．

以上から，ダイレクト書き言葉生成において，句読点検出に基づく区分化手法は，ショート

[10] 評価時との整合性を考慮すると，学習データ自体も CTC で分割することでさらに性能が改善する可能性があるが，モデルの再学習のコストが大きいため今回はこのルールベースの区分化のみを評価した．

学習データ／評価データ	ショートポーズ (17.973)	句読点を考慮した CTC 区分化 (10,083)
ショートポーズ (616,047)	9.8	22.2
句読点を考慮 (442,816)	10.2	**8.2**

表 5 学習・評価データに対する区分化手法の比較（数値は会議録テキストに対する文字誤り率 (%)，括弧内は分割後のセグメント数）

ポーズよりも一貫して高い性能を与えることがわかった．学習時と評価時の整合性の面でも，句読点という明確で音響環境等の条件に依存しない基準を用いることが重要であると考えられる．

なお，表 3（“単方向型 CTC ダイレクト + エンコーダマルチタスク学習”）に示すように，区分化に用いた単方向 CTC 自体の書き言葉予測の性能は，未来の情報が使えないため，Transformer に基づくその他のモデルに比べて高いとは言えず，特に脱落誤りが多かった．ただし，句読点の適合率のみに注目すると，句点が 0.934，読点で 0.811 と高い水準であった．

5.6 誤りの分析

本節では，提案システムの出力においてどのような修正がどの程度の達成度で実現できたかを評価する．図 8 に，句読点以外の修正項目について，カスケードモデルおよびダイレクトモデルが正しく行った修正の数と，編集者が行った修正に対する再現率を示す．カスケードモデルにはマルチスタイルモデルとテキストベース SST を組み合わせたモデル（表 3 の “Transformer カスケード + デコーダマルチスタイル学習”）を，ダイレクトモデルはエンコーダマルチタスク学習とデコーダマルチスタイル学習の両方を用いて構築したモデル（表 3 の “Transformer ダイレクト + 両方”）を用いた．

削除操作では，語彙的なフィラーと句末表現の削除は，いずれのモデルも高い再現率を示した．一方，言い直しにおける言い誤り箇所の削除では，カスケードモデルが 52.4%，ダイレクトモデルが 79.9%と，性能に大きな差が見られた．このことから，非流暢な言い誤り箇所 (reparundum) の同定に音響的な情報が役に立つこと，その上で単語連接としてより自然な言い直し部分のみが保持されたことが示唆される．また，言い誤りでは認識誤りが多く，カスケードモデルでは後段の SST でリカバーできなかった可能性が高い．“その他の削除” の項目では，どちらのモデルも相対的に性能が低かったが，文脈上不要な主語や指示語の削除など，高度な判断を必要とする例が多いためと考えられる．また，「やはり」など，間投詞以外の用例が多い単語の削除は失敗することが多かった．

置換操作では，助詞の修正や言い誤りの修正は，非常に低い再現率となった．これらの修正は，出現頻度も低く，意味や常識に基づいた修正が必要であるため，seq2seq モデルのみでは原理的に行えない例も多かった．助詞の修正では，特に「が」から「は」への修正，「が」から

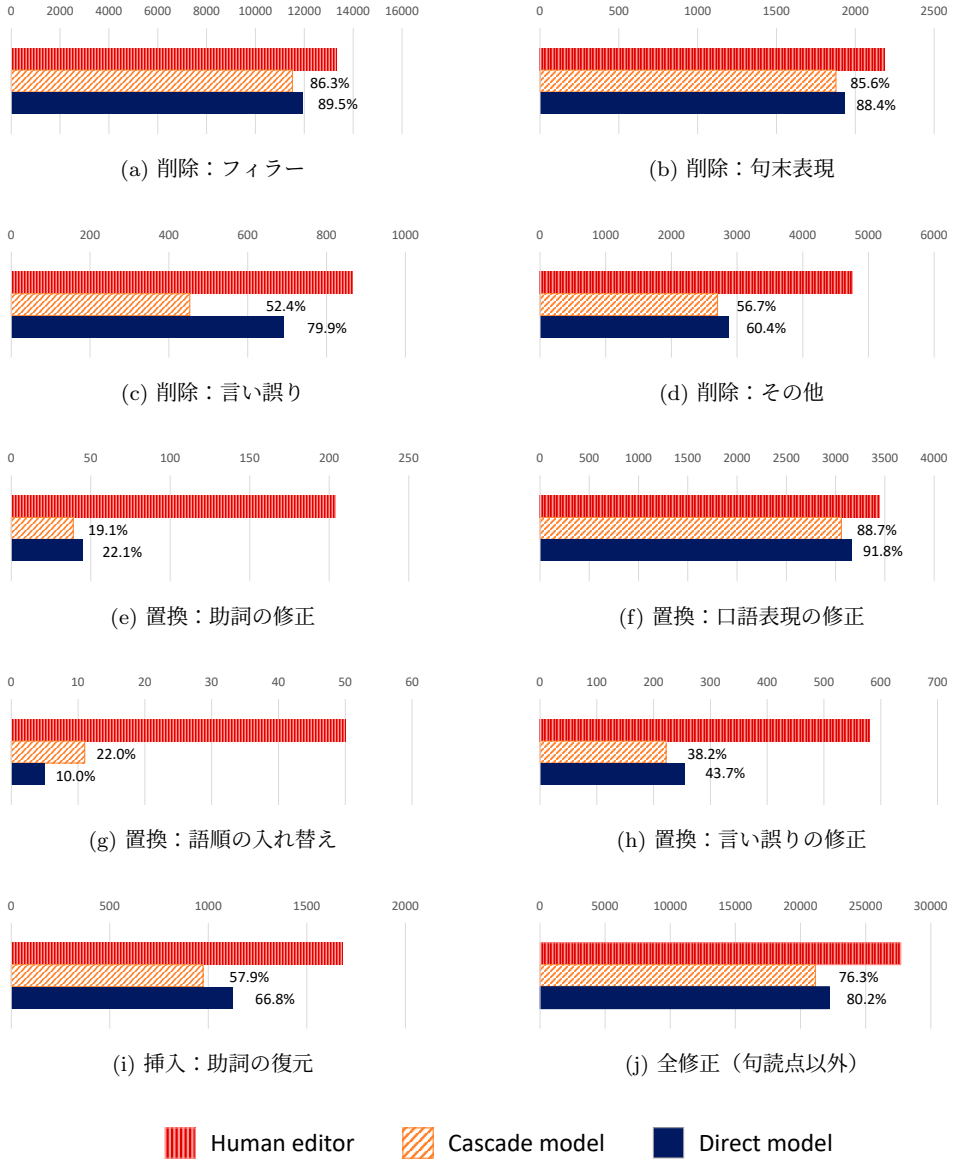

図 8　各修正項目においてカスケードモデルおよびダイレクトモデルが正しく行った修正の数と，編集者が行った修正に対する再現率

「を」への修正において，実際の発音に忠実に認識される誤りが多かった．また，語順の入れ替えについても，ほとんどの例で正しく修正できなかった．音響的な情報を用いないカスケードモデルが，例えば「今審議がなされている」を「審議が今なされている」に，「大きな僕は矛盾だと思います」を「僕は大きな矛盾だと思います」になど，比較的単純な例で正しく修正でき

手法	読点			句点		
	再現率	適合率	F 値	再現率	適合率	F 値
カスケードモデル	**0.79**	**0.74**	**0.76**	0.90	0.81	0.86
ダイレクトモデル	**0.79**	0.73	**0.76**	**0.94**	**0.84**	**0.88**

表 6　句読点挿入の性能

たことから，この項目ではカスケードモデルがダイレクトモデルより高い再現率となった．一方，口語表現の修正は，「いろんな」から「いろいろな」へ，「やつ」から「もの」へなど，定型的な言い換えに帰着できる例が多いため，どちらのモデルも非常に高い再現率となった．

挿入操作では，助詞の挿入は妥当な水準で再現可能であった．カスケードモデルでは，助詞の脱落の前後で認識誤りが多く，ダイレクトモデルとの性能差が大きかった．

以上のように，ほぼすべての修正項目において，ダイレクトモデルがカスケードモデルより有意に高い性能を示した．特に，言い誤りの削除や助詞の復元などの高度な修正で性能差が大きかった．全体として，ダイレクトモデルは編集者が行った編集作業のうち 80.2%を再現することができた．

最後に，システム出力における句読点挿入の性能を評価する．表 6 に，カスケードモデルとダイレクトモデルを用いた句読点挿入における再現率，適合率および F 値を示す．読点挿入の性能では，カスケードモデルとダイレクトモデルの性能差は見られなかった．句点挿入では，ダイレクトモデルが有意に高い性能を示した．これは句点の方がよりポーズとの相関が高いため，音響的な情報が特に有効であったためと考えられる．

図 9 に，人手による書き起こし，会議録，マルチスタイルモデルによる音声認識結果，カスケードモデルおよびダイレクトモデルの出力の例を示す．この例から，音声認識結果は非常に高い精度であっても文として読みにくく，提案手法により可読性が改善されることが確認できる．また，ダイレクトモデルでは，「これは」における助詞の復元や，「反映されることが」の箇所で言い誤りのみ削除されるなど，高度な修正が正しく行われていることがわかる．一方，冒頭の「やはり」の削除に失敗するなど，課題も見える．

6　結論

本研究では，音声から読みやすい書き言葉スタイルの生成を行うタスクにおいて，音声認識とテキストベースのスタイル変換を組み合わせたカスケード方式の問題を解決するために，e2e 音声翻訳の枠組みを用いて，音声から書き言葉を直接生成する新しいアプローチを提案した．

書き起こし	えーこのやはりこの農協が地域の農業者と力を合わせてですねえー農業所得の増大に向けて適切に事業運営を行ってやはりこの担い手の意見が反映させることはあされることが必要不可欠であるということでえーこの義務づけをいたしましたであの一一方でですねこの一年齢や性別についてはえーこのお一青年層やですね女性の方これ生産やはんで大きな役割果たしておりますので
会議録	農協が地域の農業者と力を合わせて農業所得の増大に向けて適切に事業運営を行っていく、担い手の意見が反映されることが必要不可欠であるということで、この義務づけをいたしました。一方で、年齢や性別については、青年層や女性の方、これは生産や販売で大きな役割を果たしておりますので、
音声認識	えーこのやはりこの農協が地域の農業者と力を合わせてですねえー農業所得の増大に向けて適切に事業運営を行っていくやはりこの担い手の意見が反映させることはされることが必要不可欠であるということでえーこの義務づけをいたしましたであの一一方でですねあの一年齢や性別についてはえーこのお一成年層やですね女性の方これ生産販売販売で大きな役割を果たしておりますので
カスケード整形文出力	農協が地域の農業者と力を合わせて農業所得の増大に向けて適切に事業運営を行っていく、やはり担い手の意見が反映させることが必要不可欠であるということで、この義務づけをいたしました。一方で、性別や年齢については、成年年齢層や女性の方、生産、販売で大きな役割を果たしておりますので
End-to-end整形文出力	やはり農協が地域の農業者と力を合わせて農業所得の増大に向けて適切に事業運営を行っていく。やはり担い手の意見が反映されることが必要不可欠であるということで、この義務づけをいたしました。一方で、年齢や性別については青年層や女性の方、これは生産、販売で大きな役割を果たしておりますので、

図 9　書き起こし，会議録，システム出力の例

700 時間の大規模な衆議院審議音声を用いた評価実験により，提案法であるダイレクトモデルはカスケード方式より高い精度で会議録文書を再現できることを示し，書き言葉の生成において音響情報を用いること，初段の音声認識における誤りを回避することの意義を明らかにした．また，非自己回帰型の CTC モデルは音声認識では Transformer と同等の性能を持つ一方，書き言葉のダイレクト予測タスクでは学習が収束せず，意味のある文を出力するに至らないことを実験的に示し，書き言葉予測は音声認識とは明確に異なる難しさのタスクであり，Transformer に基づく柔軟な seq2seq 変換を用いることが必須であることを明らかにした．

さらに，学習データの書き言葉（会議録）から自動で近似的な書き起こしを生成する手法と，これを用いてダイレクトモデルの学習を補助する 2 つの手法を提案し，実験により Transformer に基づくダイレクトモデルの性能をさらに向上できることを示した．会議のような話し言葉コミュニケーションでは，音声検索などのタスクのように明確な発話終端の情報が利用できないことから，書き言葉の句読点情報を用いた新しい音声区分化手法も併せて提案し，ショートポーズに基づく区分化と比較することで有効性を示した．これらの提案手法による予測結果を編集者が行った修正と比較することにより，提案モデルは編集者による修正の 80%以上を再現できること，特に助詞の復元や言い誤りの除去などの編集においてカスケード方式より大幅に高い性能が得られることを示した．

一方，いくつかの修正項目では低い再現率となったため，これらの改善が課題といえる．意味や常識に基づいて行われる助詞の修正や復元などは，学習事例数を増やすことや，大規模言

語モデルに基づくタギング (Malmi et al. 2019) などの事後的な編集で改善できる可能性がある．

欧州議会 (Díaz-Munío et al. 2021) やアイスランド議会 (Steingrímsson et al. 2020) など，他言語の議会音声コーパスを用いて提案手法の一般性を評価することも重要な課題である．また，講義やプレゼンテーション (Maekawa 2003) など，自動整形の潜在的な需要のあるタスクは多いが，これらのデータでは通常大規模かつ信頼度の高い書き言葉のアノテーションは利用できないため，国会会議録を用いて学習したダイレクト整形文予測モデルのドメイン適応や転移学習も今後検討すべき方向性の一つである．

参考文献

Akita, Y. and Kawahara, T. (2007). "Topic-independent Speaking-style Transformation of Language Model for Spontaneous Speech Recognition." In *ICASSP*, Vol. 4, pp. 33–36.

Akita, Y. and Kawahara, T. (2011). "Automatic Comma Insertion of Lecture Transcripts Based on Multiple Annotations." In *INTERSPEECH*, pp. 2889–2892.

Akita, Y., Mimura, M., and Kawahara, T. (2009). "Automatic Transcription System for Meetings of the Japanese National Congress." In *INTERSPEECH*, pp. 84–87.

秋田祐哉，三村正人，河原達也 (2010)．会議録作成支援のための国会審議の音声認識システム．電子情報通信学会論文誌, J93-D 巻, pp. 1736–1744. [Y. Akita et al. (2010). Automatic Transcription System for Creation of Meeting Records in the National Diet. The IEICE Transactions, J93-D, pp. 1736–1744.].

Audhkhasi, K., Ramabhadran, B., Saon, G., Picheny, M., and Nahamoo, D. (2017). "Direct Acoustics-to-Word Models for English Conversational Speech Recognition." In *INTERSPEECH*, pp. 959–963.

Baevski, A., Zhou, Y., Mohamed, A., and Auli, M. (2020). "wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations." In *Advances in Neural Information Processing Systems*, Vol. 33, pp. 12449–12460. Curran Associates, Inc.

Bentivogli, L., Cettolo, M., Gaido, M., Karakanta, A., Martinelli, A., Negri, M., and Turchi, M. (2021). "Cascade versus Direct Speech Translation: Do the Differences Still Make a Difference?" In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, Vol. 1, pp. 2873–2887.

Bérard, A., Besacier, L., Kocabiyikoglu, A. C., and Pietquin, O. (2018). "End-to-End Automatic Speech Translation of Audiobooks." In *ICASSP*, pp. 6224–6228.

Chang, S.-Y., Prabhavalkar, R., He, Y., Sainath, T. N., and Simko, G. (2019). "Joint Endpointing

and Decoding with End-to-End Models." In *ICASSP*, pp. 5626–5630.

Chiu, C.-C., Han, W., Zhang, Y., Pang, R., Kishchenko, S., Nguyen, P., Narayanan, A., Liao, H., Zhang, S., Kannan, A., Prabhavalkar, R., Chen, Z., Sainath, T., and Wu, Y. (2019). "A Comparison of End-to-End Models for Long-form Speech Recognition." In *ASRU*, pp. 889–896.

Chorowski, J. and Jaitly, N. (2017). "Towards Better Decoding and Language Model Integration in Sequence to Sequence Models." In *INTERSPEECH*, pp. 523–527.

Díaz-Munío, G. V. G., Silvestre-Cerdà, J.-A., Jorge, J., Pastor, A. G., Iranzo-Sánchez, J., Baquero-Arnal, P., Roselló, N., de Martos, A. P.-G., Civera, J., Sanchis, A., and Juan, A. (2021). "Europarl-ASR: A Large Corpus of Parliamentary Debates for Streaming ASR Benchmarking and Speech Data Filtering/Verbatimization." In *INTERSPEECH*, pp. 2695–2699.

Dong, L., Xu, S., and Xu, B. (2018). "Speech-Transformer: A No-Recurrence Sequence-to-Sequence Model for Speech Recognition." In *ICASSP*, pp. 5884–5888.

Gravano, A., Jansche, M., and Bacchiani, M. (2009). "Restoring Punctuation and Capitalization in Transcribed Speech." In *Proc. ICASSP*, pp. 4741–4744.

Graves, A. (2012). "Sequence Transduction with Recurrent Neural Networks." In *LCML*, pp. 4945–4949.

Graves, A., Fernandez, S., Gomez, F., and Schmidhuber, J. (2006). "Connectionist Temporal Classification : Labelling Unsegmented Sequence Data with Recurrent Neural Networks." In *Proceedings of the 23st International Conference on Machine Learning*, pp. 369–376.

Gulcehre, C., Firat, O., Xu, K., Cho, K., Barrault, L., Lin, H.-C., Bougares, F., Schwenk, H., and Bengio, Y. (2015). "On Using Monolingual Corpora in Neural Machine Translation." *arXiv preprint arXiv:1503.03535*.

Higuchi, Y., Karube, K., Ogawa, T., and Kobayashi, T. (2022). "Hierarchical Conditional End-to-End ASR with CTC and Multi-granular Subword Units." In *ICASSP*, pp. 7797–7801.

Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., and Kingsbury, B. (2012). "Deep Neural Networks for Acoustic Modeling in Speech Recognition." *IEEE Signal Processing Magazine*, **29** (6), pp. 82–97.

Hori, T., Willett, D., and Minami, Y. (2013). "Paraphrasing Spontaneous Speech using Weighted Finite-state Transducers." In *ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, pp. 219–222.

Jones, D., Wolf, F., Gibson, E., Williams, E., Fedorenko, E., Reynolds, D., and Zissman, M. (2003). "Measuring the Readability of Automatic Speech-to-Text Transcripts." In *Eurospeech*, pp. 1585–1588.

Karita, S., Chen, N., Hayashi, T., Hori, T., Inaguma, H., Jiang, Z., Someki, M., Enrique, N., Soplin, Y., Yamamoto, R., Wang, X., Watanabe, S., Yoshimura, T., and Zhang, W. (2019a). "A Comparative Study on Transformer vs RNN in Speech Applications." In *ASRU*, pp. 449–456.

Karita, S., Soplin, N. E. Y., Watanabe, S., Delcroix, M., Ogawa, A., and Nakatani, T. (2019b). "Improving Transformer-Based End-to-End Speech Recognition with Connectionist Temporal Classification and Language Model Integration." In *INTERSPEECH*, pp. 1408–1412.

Kawahara, T. (2012). "Transcription System using Automatic Speech Recognition for the Japanese Parliament (Diet)." In *AAAI/IAAI*, pp. 2224–2228.

Kawahara, T. (2021). "Captioning Software using Automatic Speech Recognition for Online Lectures." *The Journal of Professional Reporting and Transcription (Tiro),* No. 1.

Kim, S., Hori, T., and Watanabe, S. (2017). "Joint CTC-Attention based End-to-End Speech Recognition using Multi-task Learning." In *ICASSP*, pp. 4835–4839.

Kingma, D. P. and Ba, J. (2015). "Adam: A Method for Stochastic Optimization." In *ICLR*.

Kudo, T. and Richardson, J. (2018). "SentencePiece: A Simple and Language Independent Subword Tokenizer and Detokenizer for Neural Text Processing." In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Vol. 1, pp. 66–71.

Lamel, L., Gauvain, J., and Adda, G. (2001). "Investigating Lightly Supervised Acoustic Model Training." In *ICASSP*, Vol. 1, pp. 477–480.

LeCun, Y. and Bengio, Y. (1995). "Convolutional Networks for Images, Speech, and Time-series." In Arbib, M. A. (Ed.), *The Handbook of Brain Theory and Neural Networks*. Cambridge, MA, USA: MIT Press,.

Lee, A., Kawahara, T., and Shikano, K. (2001). "Julius: An Open Source Real-Time Large Vocabulary Recognition Engine." In *EUROSPEECH*, pp. 1691–1694.

Liu, Y., Shriberg, E., Stolcke, A., Hillard, D., Ostendorf, M., and Harper, M. (2006). "Enriching Speech Recognition with Automatic detection of Sentence Boundaries and Disfluencies." *IEEE Transactions on Audio, Speech & Language Processing.*, **14**, pp. 1526–1540.

Maekawa, K. (2003). "Corpus of Spontaneous Japanese: Its Design and Evaluation." In *Proceedings of ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, pp. 7–12.

Mahadeokar, J., Shangguan, Y., Le, D., Keren, G., Su, H., Le, T., Yeh, C.-F., Fuegen, C., and Seltzer, M. L. (2021). "Alignment Restricted Streaming Recurrent Neural Network Transducer." In *SLT*, pp. 52–59.

Malmi, E., Krause, S., Rothe, S., Mirylenka, D., and Severyn, A. (2019). "Encode, Tag, Realize:

High-Precision Text Editing." In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 5054–5065.

McDermott, E., Sak, H., and Variani, E. (2019). "A Density Ratio Approach to Language Model Fusion in End-to-End Automatic Speech Recognition." In *ASRU*, pp. 434–441.

Mohamed, A., Dahl, G., and Hinton, G. (2012). "Acoustic Modelling using Deep Belief Networks." *IEEE Transactions on Audio, Speech, & Language Processing.*, **20** (1), pp. 14–22.

Nair, V. and Hinton, G. E. (2010). "Rectified Linear Units Improve Restricted Boltzmann Machines." In *Proceedings of ICML*, pp. 807–814.

Neubig, G., Akita, Y., Mori, S., and Kawahara, T. (2012). "A Monotonic Statistical Machine Translation Approach to Speaking Style Transformation." In *Computer Speech and Language*, Vol. 26, pp. 349–370.

Pan, J., Lei, T., Kim, K., Han, K., and Watanabe, S. (2022). "SRU++: Pioneering Fast Recurrence with Attention for Speech Recognition." In *ICASSP*, pp. 7872–7876.

Park, D. S., Zhang, Y., Chiu, C.-C., Chen, Y., Li, B., Chan, W., Le, Q. V., and Wu, Y. (2020). "Specaugment on Large Scale Datasets." In *ICASSP*, pp. 6879–6883.

Paulik, M., Rao, S., Lane, I., Vogel, S., and Schultz, T. (2008). "Sentence Segmentation and Punctuation Recovery for Spoken Language Translation." In *ICASSP*, pp. 5105–5108.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., and Vesely, K. (2011). "The Kaldi speech recognition toolkit." In *Proc. ASRU*, pp. 1–4.

Sanabria, R. and Metze, F. (2018). "Hierarchical Multitask Learning With CTC." In *SLT*, pp. 485–490.

Sennrich, R., Haddow, B., and Birch, A. (2016). "Neural Machine Translation of Rare Words with Subword Units." In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Vol. 1, pp. 1715–1725.

Shitaoka, K., Nanjo, H., and Kawahara, T. (2004). "Automatic transformation of lecture transcription into document style using statistical framework." In *INTERSPEECH*, pp. 2169–2172.

Soltau, H., Liao, H., and Sak, H. (2017). "Neural speech recognizer: Acoustic-to-word LSTM model for large vocabulary speech recognition." In *Interspeech*, pp. 3707–3711.

Sproat, R. and Jaitly, N. (2017). "An RNN Model of Text Normalization." In *Interspeech*, pp. 754–757.

Steingrímsson, S., Barkarson, S., and Örnólfsson, G. T. (2020). "IGC-Parl: Icelandic Corpus of

Parliamentary Proceedings." In *LREC*, pp. 11–17.

Ueno, S., Inaguma, H., Mimura, M., and Kawahara, T. (2018). "Acoustic-to-word Attention-based Model Complemented with Character-Level CTC-based Model." In *ICASSP*, pp. 5804–5808.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). "Attention Is All You Need." In *31st Conference on Neural Information Processing Systems (NIPS 2017)*, Vol. 30, pp. 6000–6010.

Watanabe, S., Hori, T., and Hershey, J. R. (2017). "Language Independent End-to-End Architecture for Joint Language Identification and Speech Recognition." In *ASRU*, pp. 265–271.

Weiss, R. J., Chorowski, J., Jaitly, N., Wu, Y., and Chen, Z. (2017). "Sequence-to-Sequence Models Can Directly Translate Foreign Speech." In *Interspeech*, pp. 2625–2629.

Yeh, J. and Wu, C. (2006). "Edit Disfluency Detection and Correction using a Cleanup Language Model and an Alignment Model." *IEEE Transactions on Audio, Speech & Language Processing*, **14**, pp. 1574–1583.

Yoshimura, T., Hayashi, T., Takeda, K., and Watanabe, S. (2020). "End-to-End Automatic Speech Recognition Integrated with CTC-Based Voice Activity Detection." In *ICASSP*, pp. 6999–7003.

略歴

三村　正人：2000 年京都大学大学院情報学研究科修士課程修了．2022 年同博士課程修了．現在，京都大学情報学研究科特定研究員．IEEE，日本音響学会各会員．

河原　達也：1987 年京都大学工学部情報工学科卒業．1989 年同大学修士課程終了．京都大学工学部助手，同助教授，学術メディアセンター教授を経て，現在，情報学研究科教授．音声情報処理，特に音声認識及び対話システムに関する研究に従事．博士（工学）．IEEE Fellow，ISCA，APSIPA 各理事．情報処理学会，日本音響学会，電子情報通信学会，人工知能学会，言語処理学会各会員．日本学術会議連携会員．

（2022 年 5 月 10 日　受付）
（2022 年 8 月 30 日　再受付）
（2022 年 10 月 18 日　採録）