Vol.2018-SLP-125 No.1
2018/12/10

部分文字再帰的ニューラルネットを使った end-to-end 音声認識の仮説修正

太刀岡 勇気[1,a)]

概要：End-to-end 音声認識は，その構成が単純なことから一般的になりつつある．しかし，語彙外単語に遭遇することが，従来の音響モデルと言語モデルの両方を使うハイブリッド手法よりも頻繁になる．とりわけ，単語に基づく end-to-end システムは学習データに現れなかった単語を出力することはできない．この問題に対処するため，文字単位の end-to-end システムが提案されているものの，ノイズの影響を受けやすく，出力される単語が必ずしも言語的に正しいものでなくなるという問題がある．これはデコード処理時に辞書や言語モデルといった言語制約を欠いているためである．ゆえにスペル誤りのような誤りが起こりやすくなる．自然言語処理の分野では，スペル誤りを修正するため，部分文字再帰的ニューラルネットワーク (scRNN) が提案されている．scRNN は，単語内の文字の個数を入力とし，単語 ID を出力とするものである．scRNN は置換誤りのみに焦点を当てているため，これを音声認識に適用する際には，拡張が必要となる．ここでは，挿入・置換誤りを考慮するため，connectionist temporal classification の空白記号に似た空白単語記号と単語結合を導入する．騒音下音声認識と大語彙音声認識の 2 つの異なる音声認識タスクにおいて，提案の拡張を用いた scRNN が単語誤り率を改善することを示す．

Yuuki Tachioka[1,a)]

1. はじめに

従来の音声認識は，音響モデル・言語モデル・発音辞書といった複数のモデルを組み合わせて認識を行うハイブリッド手法がよく使われている [1]．深層神経回路網 (deep neural network; DNN) の研究の進展により，この手法を簡略化する様々な end-to-end(E2E) 型の手法をとるシステムが提案されている [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12]．E2E システムの利点はモデリングが簡単なこと，デコードが速いことと辞書を必要としないことである．これは音響特徴量からシンボルへ直接的に変換する．シンボルとしては，音素 [3], [4], [5], [8]，文字 [2], [3], [6], [7], [11], [12]，単語 [9], [10] などが使われる．

文字単位あるいは単語単位の E2E システムはより単純で，言語モデルを使った付加的なデコーディングを必要としない [2], [3]．これはモデル構築のコストを小さいものにするため，異なる言語への適用が簡単になるという利点がある [13]．しかしながら，データのスパース性により，E2E は学習データに存在しない語彙外 (out-of-vocabulary; OOV) 単語に弱いという欠点がある．利用可能な音声データの量は，書きつけられた文書の量よりもはるかに少ないため，OOV 単語は従来のハイブリッド手法よりも頻繁に現れる．単語単位の E2E システムはそれ以外の手法よりもデコードが速いが，OOV 単語を出力することができないので，文字単位の E2E システムと組み合わせて使う手法も提案されている [14]．文字単位の E2E システムではこの問題を避けることはできるが，言語の制約が単語単位のものに比べて弱いため，ノイズによって誤りやすい．例えば，5 節に示すように，スペル誤りのような誤りが頻発する．

自然言語処理の分野では，神経機械翻訳 (neural machine translation; NMT) が広く使われている [15]．もしパラレルコーパスが用意できるなら，神経回路網により，いかなる対応 (翻訳) も学習することができる．音声認識の仮説を入力，正解ラベルを出力として，NMT モデルを文対文の変換として学習すればよい．この翻訳では，単語単位を全体的に変えてしまうが，多くの音声認識誤りは局所的なので，必ずしももこの手法が適切でないと考えられる．後節の実験においても，NMT がうまく動かないことを示す．

[1] デンソーアイティーラボラトリ
東京都渋谷区渋谷 2-15-1 渋谷クロスタワー 28F 150-0002
[a)] ytachioka@d-itlab.co.jp

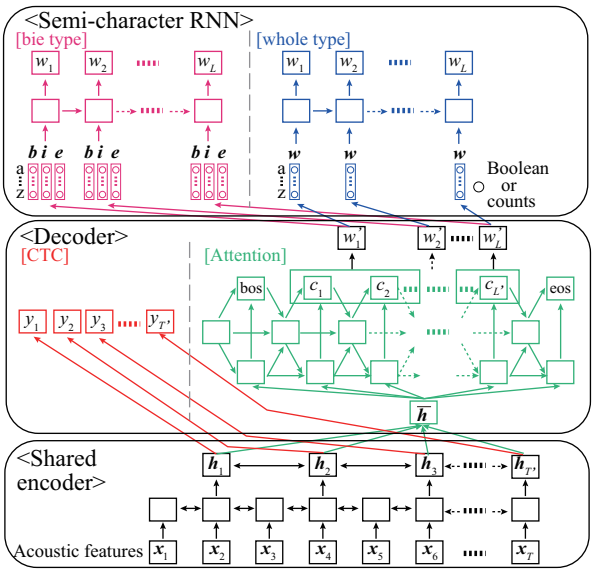

図 1 Structure of proposed end-to-end system with scRNN based correction on top of ASR system.

これに対して，スペル誤りを修正するため，同じく自然言語処理の分野で，部分文字リカレントニューラルネットワーク (semi-character recurrent neural network; scRNN) が提案された [16]．これは入れ替え誤りに焦点を当て，単語単位で局所的な誤りを修正できる．彼らの実験では，商用製品を含む先端的な手法で実験して，最も高いスペル誤り修正性能を示したとしている．

scRNN は単語対単語の変換のため置換誤りしか扱うことができない．音声認識の仮説修正に使うためには，1 対 1 の単語対応が得られない挿入・削除誤りに対応する必要がある．[16] の実験では，挿入・削除誤りはないと仮定しているが，音声認識ではそのような誤りが頻発する．scRNN を拡張してこの問題に対処するため，ここでは空白単語記号と単語結合を導入する．異なる観点からみると，音声認識の仮説修正ができるので，scRNN では識別学習的な効果を得られる [17] ということもできる．

本論は以下のように構成されている．2 節では，ここで採用した文字単位の E2E 手法を紹介し，3 節では scRNN について述べる．scRNN を音声認識に適用するため，4 節では，空白単語記号と単語結合を scRNN に導入する．5 節の実験では，提案法の有効性を 2 つの異なる音声認識コーパスに対して示す．第 1 は第 4 回 CHiME チャレンジで，限られた語彙での音声認識タスクである．第 2 は TED-LIUM コーパスで，こちらは大語彙連続音声認識タスクである．これらの実験により，NMT はほとんど動かないのに対して，提案の拡張をした scRNN では単語正解率と誤り率に改善が見られた．

2. End-to-end 音声認識

ここでは，最新の文字単位の E2E システムをベースラインとして採用する．E2E システムは connectionist temporal classification (CTC) [2], [3], [5], [9], [10] と注意機構 (attention; ATT) [4], [6], [7], [8] の 2 種類に分類できる．図 1 の下 2 段は，共有されたエンコーダ・デコーダである．ここで使った E2E システムは CTC と ATT の両方を結合して使う [11], [12].

入力は T フレームにまたがる音響特徴量 $\boldsymbol{x}_1, ..., \boldsymbol{x}_T$ である．エンコーダは $\boldsymbol{x}_{1:T}$ を受け取り，T' 長の内部表現 $\boldsymbol{h}_{1:T'}$ に変換する．ここで，間引きの影響で，T' は必ずしも T とは一致しない．CTC の出力は，T' 長のラベル系列 $y_1, ..., y_{T'}$ である．CTC は，文字長とラベル長 T' の差異を補完するため，空白記号やラベル中の繰り返し文字を許容する．文字系列 $\boldsymbol{c}$ に対する CTC の出力確率は

$$P_{CTC}(\boldsymbol{c}|\boldsymbol{x}) = \sum_{\boldsymbol{y}\in\Psi(\boldsymbol{c})} \prod_{t'=1}^{T'} \sigma_{\boldsymbol{h}_{t'}(\boldsymbol{x})}(y_{t'}), \quad (1)$$

のように，ラベル間 y で独立であると仮定されている．ここで，Ψ は，文字系列 $\boldsymbol{c}$ を実現する T' 長のすべてのラベル系列 $\boldsymbol{y} = y_1, ..., y_{t'}, ..., y_{T'}$ の集合である．$\sigma_{\boldsymbol{h}_{t'}(\boldsymbol{x})}$ は，エンコーダの出力 $\boldsymbol{h}_{t'}(\boldsymbol{x})$ で条件付けされた，ラベル $y_{t'}$ に対するソフトマックス出力である．最終的に，CTC は空白記号と繰り返し文字を削除したのち，文字列を出力する．

ATT の出力は，L' 長の文字系列 $c_1, ..., c_{L'}$ であり，文の始端記号 (bos) と文の終端記号 (eos) を含む．ATT の出力確率は，$c_0 = \text{bos}$ から始めて，

$$P_{ATT}(\boldsymbol{c}|\boldsymbol{x}) = \prod_{l'=1}^{L'} P(c_{l'}|\bar{\boldsymbol{h}}(\boldsymbol{x}), c_{0:(l'-1)}), \quad (2)$$

のように再帰形で表現される．ここで，$\bar{\boldsymbol{h}}$ は，エンコーダ出力 $\boldsymbol{h}_1, ..., \boldsymbol{h}_{T'}$ を束ねたものである．

これらの出力文字は空白で分離されているので，L 長の単語系列 $w'_1, ..., w'_L$ に変換できる．L' と L は，$L' = \sum_{l=1}^{L} |w'_l| + L - 1$ で関連付けられる．$|w'_l|$ は単語 w'_l 中の文字数である．このシステムでは，明示的な言語制約を使っていないので，出力単語は必ずしも言語的に正しくない．実際，E2E システムの性能は，よく単語誤り率 (word error rate; WER) ではなく，文字誤り率 (character error rate; CER) の観点で評価される．これは言語モデルを使う従来法と比較して，CER がおおむね同程度であっても WER は大幅に悪いということがあり得るからである．

CTC と ATT 両者の利点を活かすため，これらの 2 モデルの統合が提案されている [11]．参照文字系列 $\boldsymbol{c}^*$ に対するマルチタスク損失

$$\mathcal{L} = -\lambda \ln P_{CTC}(\boldsymbol{c}^*|\boldsymbol{x}) - (1-\lambda) \ln P_{ATT}(\boldsymbol{c}^*|\boldsymbol{x}) \quad (3)$$

を最小化する．CTC は前向き後ろ向きアルゴリズムによりモノトニックな整列が得られるが，ラベル間に独立性を仮定しているため，ラベル間の依存性を考慮することはできない．ATT はラベル間の依存性を考慮することはでき

るが，ATT モデルは CTC モデルよりも整列に対する制約が弱いので，ノイズの影響を受けやすい．CTC モデルの学習を補助タスクとして使うことで，CTC による整列がマルチタスク学習の収束を速めることができる．

3. 部分文字リカレントニューラルネットワーク (scRNN)

自然言語処理の分野では，スペル誤りを修正するため，scRNN が主にジャンブル誤りに焦点を当てて提案された．ジャンブル誤りは，単語の始めと終わりの文字が一定で，中間の文字のみが置換誤りを起こしているような誤りである．例えば，「characters」が「chraatcres」のようになる誤りである．これは比較的原型が推測しやすい誤りである [16]．これをモデル化するため，文字をはじめ，終わり，中間の 3 種に分けてそれぞれ別々に扱う．

図 1 の上段は scRNN を示す．これは音声認識の仮説の単語列 $w'_1, ..., w'_L$ から変換された単語中の文字数を受け取り，修正された単語系列 $w_1, ..., w_L$ を出力する．入力ベクトルの次元はアルファベットと記号を含む文字数の 3 倍である．入力 $\boldsymbol{b}$ と $\boldsymbol{e}$ は，ワンホットベクトルであり，$\boldsymbol{i}$ は疎なベクトルである．例えば，入力単語が「speech」の場合，それぞれのベクトルは $\boldsymbol{b} = \{s = 1\}$，$\boldsymbol{i} = \{c = 1, e = 2, p = 1\}$，$\boldsymbol{e} = \{h = 1\}$ のようになる．scRNN への入力はこれらの結合ベクトル $[\boldsymbol{b}; \boldsymbol{i}; \boldsymbol{e}]$ である．

文字数はノイズの影響を受けやすい．真偽変数は文字数よりも効果的な可能性がある．あいまい性が増加し識別性能が低下するものの，繰り返し文字や文字の脱落といったスペル誤りが頻発する状況においては，ノイズ耐性が改善するかもしれない．よって，5 節の実験では，文字数タイプと真偽値タイプの性能を比較することにする．真偽値タイプの入力ベクトルは，$\boldsymbol{b}$ と $\boldsymbol{e}$ は同一で，$\boldsymbol{i} = \{c = 1, e = 1, p = 1\}$ である．

3 種類別々にモデリングした場合に加えて，単語全体の文字数をモデリングした場合も比較する．例えば，上記の例では，入力ベクトルを $\boldsymbol{w} = \{c = 1, e = 2, h = 1, p = 1, s = 1\}$ やその真偽値 $\boldsymbol{w} = \{c = 1, e = 1, h = 1, p = 1, s = 1\}$ のようにする．

4. 音声認識の仮説修正のための空白単語記号と単語結合の導入

図 2 では，置換・挿入と 2 種類の脱落の 4 種の誤りを示している．図 2 (substitution) の置換誤りに関しては，scRNN が直接的に使える．入力は音声認識仮説であり，出力は正解である．挿入と脱落誤りに関しては，scRNN では 1 対 1 の単語対応が必要なため，仮説と正解の間の単語長の差異を吸収する必要がある．最も簡単な対処法は挿入と削除誤りを無視する (ign) ことだが，単語の文脈を中断してしまう．

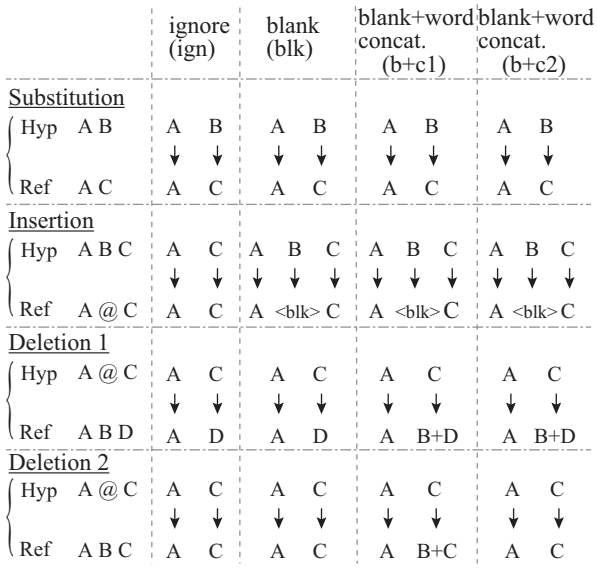

図 2 Four types of blank word symbols (blk) and word concatenation in training stage, where hypotheses (Hyp) and references (Ref) are aligned and @ is null symbol. Input and output to scRNN are Hyp and Ref, respectively.

そこで，CTC における空白記号に似た空白単語記号 (blk) を導入する．挿入・削除誤りにおいて 1 対 1 で単語対応させるためにこの記号を使う．図 2 (insertion) の挿入誤りは削除誤りよりも扱いやすい．正解における空白単語記号を，音声認識仮説中の入力単語と関連付ける．図中ブランク (blk) は，挿入誤りのみに対応しており，削除誤りは無視している．

削除誤りに対しては，空白単語記号はつかえない．テスト時に仮説中に削除誤りが起こっていることを検出することは難しいためである．正解テキストと音声認識仮説の間の 1 対 1 の単語対応を取るため，複数の単語を結合し，それらを 1 単語として扱う単語結合を行う．ここでは，2 種類の単語結合「b+c1」と「b+c2」を提案する．図 2 (deletion 1) では，削除の直後の単語が仮説と正解では一致していない．仮説中の “C” を，正解文中の “B D” に関連付けられる．これは「b+c1」「b+c2」の両者に共通している．文中に多量の削除誤りがある場合，結合単語の長さが長くなり，結合単語がほぼコーパス中に現れないことから性能が低下する．本報では，結合単語長の最大値を 2 とした．図 2 (deletion 2) では，削除の直後の単語が一致している．“B” は脱落している可能性があり，認識しがたい．ゆえに，“B” を無視することがよりよい選択肢になる可能性がある．これが「b+c2」である．この手法は文字単位の E2E と結合した単語単位の E2E[14] にも適応しうる．別の見方では，本手法により音声認識誤りを修正することができるので，識別学習的な効果を持つともいうことができる [17].

REF: Senate finance chairman Lloyd Bentsen D. Texas said he would speed up work on the package because of the crash
e2e: Then a finance chirman more fenseled the fecials said he would steed a work on a packed whe case in the proble
scRNN: Then a finance chairman more festive the fails said he would stood a work on a we case in the problem
NMT: Then she could board at the beginning to his government plans and when the policies was produced their their threat

図 3 Samples of error corrections on CHiME 4 evaluation set. One substitution error (chirman → chairman) and one insertion error ("a packed whe case in" → "a we case in") removed.

REF: You know I was already bargaining as a five year old child with doctor P to try to get out of doing these exercises unsuccessfully of course
e2e: You know I was already barganing at the five year old child with darker pretty to talk to get out of doing these ext eyes on successfully of course
scRNN: You know I was already bargaining at the five year old child with doctor pretty to talk to get out of doing these ex eyes on successfully of course
NMT: You know I was already married at the old number of my family dropped to me was going to get to some sense with these genomes of course on

図 4 Samples of error corrections on TED-LIUM test set. Two substitution errors (barganing → bargaining and darker → doctor) removed.

5. End-to-end 音声認識実験

5.1 実験条件

2 つのコーパスで提案法の有効性を検証した．1 つめのコーパスは，第 4 回 CHiME チャレンジ (CHiME 4) の 1ch トラックである．これは騒音下音声認識であり，音声強調手法は用いていない [18]．提案法は，E2E システムは騒音があるデータに影響を受けやすいので，騒音下音声認識においてより効果的であると考えられる．2 つめのコーパスは，TED-LIUM コーパスであり，これは大語彙連続音声認識タスクである [19]．学習・開発・評価セットが両者のコーパスに用意されている．

espnet ツールキット[*1]を利用して，音声認識仮説を得た [11], [12]．両コーパスに対して，付属のスクリプトを利用した．この E2E システムはいかなる言語モデルや辞書も利用していない[*2]．位置に基づく ATT を利用し，ユニット数を 320，λ を 0.5 とした．デコーディング時のビームサイズは 20 とした．共有されたデコーダーでは，上 2 層は長期短期記憶 (LSTM) とし，下層の出力を 2 フレームに 1 回入力した (すなわち $T' = T/4$)．

scRNN の語彙は，学習セットに現れた単語で構築したため，開発・評価セットには，OOV 単語がある条件である．特別な OOV 単語記号 (unk) を scRNN に追加した．scRNN は学習データにより，ミニバッチサイズ 256 で，650 ユニットの LSTM モデルをドロップアウト率 0.01 で学習した．エポック数は 15 とし，Adam [20] を使って，学習率を調整した．ここでは，[16] の著者らにより公開されているスクリプト[*3]を修正して用いた．2 つの入力ベクトルタイプ，真偽値と個数を比較した．入力文字タイプの次元は 50 であり，アルファベット (a–z) と記号 (ハイフン，コンマ，ピリオド等) から構成されている．3 つの異なるタイプの文字数 (bie) を入力とする scRNN と単語全体の数 (w) を入力にする scRNN を比較した．図 2 のように，空白単語記号と単語結合を 4 種の方法で導入した．"unk" と空白単語記号は削除してから評価した．

これに加えて，seq2seq ツールキット[*4]を使った NMT[21] と提案法を比較した．このモデルはあたかも音声認識仮説を原言語，正解文章を目的言語とする翻訳のように学習する．NMT は正解率基準 (acc) と bleu [22] 基準 (bleu) の 2 つの基準により学習した．

5.2 CHiME 4 チャレンジ (1ch track)

表 1 は，CHiME 4 開発セットに対する，単語正解率 (C)，置換率 (S)，削除率 (D)，挿入率 (I)，誤り率 (E)[%] である．単語正解率 (C) 以外は，低い数値が良い性能を示す．この場合，NMT が最も性能が低い．正解率基準のほうが bleu 基準よりも若干良好であった．脱落・挿入誤り数はほぼベースラインと同等でありながら，置換誤り数は顕著に増える．NMT は元の意味を保持していないようなありそうな文章を創り出してしまう．1 文当たりの平均単語長は，ベースライン (16.3 単語/文) と NMT(15.8) でほぼ等しい．scRNN (表 1 の 2–5 段目) は，ベースラインよりも良い性能を示している．すべての場合で，真偽値タイプ (·,b) はカウントタイプ (·,c) に劣った．文字数に対する

[*1] https://github.com/espnet/espnet から利用可能
[*2] 詳細な設定は [11] を参照されたい．
[*3] https://github.com/keisks/robsut-wrod-reocginiton から利用可能
[*4] https://github.com/google/seq2seq から利用可能

表 1 Evaluation of word correct (C), substitution (S), deletion (D), insertion (I), and error (E) rates [%] on CHiME 4 development set. Inputs were whole count (w) or separate count (bie). Input vector type was Boolean (b) or counts (c). scRNN has four types of inputs and outputs in Figure 2. CER of baseline was 33.5% (real) and 33.7% (simu).

Env.	real					simu				
	C	S	D	I	E	C	S	D	I	E
baseline	41.9	49.6	**8.5**	6.6	64.7	44.0	48.1	**7.9**	8.1	64.0
ign(w,b)	41.8	49.6	**8.5**	6.6	64.7	43.8	48.3	**7.9**	8.1	64.3
ign(w,c)	43.4	48.1	8.6	6.6	63.3	45.6	46.4	8.0	8.1	62.5
ign(bie,b)	43.6	47.8	8.6	6.6	63.1	45.7	46.3	8.0	8.1	62.4
ign(bie,c)	43.9	47.5	8.6	6.6	62.8	45.8	46.2	8.0	8.2	62.3
blk(w,b)	41.7	48.9	9.4	6.2	64.5	43.9	47.4	8.7	7.7	63.8
blk(w,c)	43.4	46.3	10.4	6.0	62.6	45.6	44.9	9.5	7.3	61.8
blk(bie,b)	43.7	46.2	10.1	6.0	62.3	45.7	44.9	9.4	7.4	61.8
blk(bie,c)	43.7	45.4	10.9	**5.7**	**62.0**	45.8	**44.0**	10.3	**7.0**	**61.3**
b+c1(bie,b)	43.7	46.4	10.0	6.4	62.8	45.8	44.9	9.3	7.9	62.1
b+c1(bie,c)	**44.0**	45.9	10.1	6.4	62.4	**46.0**	44.7	9.3	7.7	61.8
b+c2(bie,b)	43.6	46.0	10.4	6.3	62.7	45.7	44.4	9.9	7.7	62.0
b+c2(bie,c)	43.8	**45.2**	10.9	6.0	62.2	45.8	**44.0**	10.2	7.4	61.6
NMT(acc)	9.6	80.3	10.1	6.2	96.5	11.1	79.0	9.8	7.4	96.3
NMT(bleu)	9.2	80.9	9.9	6.4	97.2	10.6	80.1	9.2	7.9	97.2

ノイズの影響は限定的である．空白単語記号を導入した scRNN(blk) は，削除・挿入誤りを無視した scRNN(ign) を上回る性能を示した．これは，ign は，“unk” を出力する場合以外は，削除・挿入誤り数を減らすことが基本的にできないからである．単語結合 (b+c1) が最も良い単語正解率を得たが，全体的には blk が最良の WER を得た．残念ながら b+c2 は blk に劣った．相対誤り低減率は，挿入誤りに対して 15.7%，置換誤りに対して 9.3%，単語誤り率に対して 4.4%であった．

表 2 は，CHiME 4 評価セットに対する同種の評価である．傾向は同様である．NMT が最も悪く，scRNN は効果的で，空白単語記号の導入は有効である．b+c1 は最良の正解率を得た一方，blk は最良の WER を得た．相対値で単語誤り率を 3.7–3.9%改善した．図 3 は，結果のサンプルであり，スペル誤りのような誤り “chierman” や “fenseled” が E2E の仮説中に見られる．この場合，1 つの置換誤りと 1 つの挿入誤りを取り除くことができている．

5.3 TED-LIUM

表 3 は，TED-LIUM コーパスに対する同種の評価を示している．NMT の性能はベースラインに比べてとても悪い．置換誤り数は 3 倍多く，削除・挿入誤り数は 2–3 倍ベースラインより多い．

それ以外の傾向は，CHiME コーパスで観測されたものと類似である．scRNN は WER を相対値で 1.2–1.5%改善した．TED-LIUM は CHiME コーパスに比べて単語数が多く，scRNN の OOV 単語数が多いため，scRNN にとってより難しいタスクであるが，依然として scRNN による仮説修正は効果的である．図 4 は結果のサンプルであり，“barganing” は正しくない単語であるがこれが scRNN により修正されている．

表 2 Evaluation on CHiME 4 evaluation set. CER of baseline was 44.1% (real) and 41.7% (simu).

Env.	real					simu				
	C	S	D	I	E	C	S	D	I	E
baseline	31.5	59.2	**9.2**	9.6	78.1	34.8	57.1	**8.1**	10.0	75.2
ign(bie,c)	**33.0**	57.6	9.3	9.7	76.7	**36.5**	55.4	8.2	10.1	73.6
blk1(bie,c)	32.9	**55.4**	11.7	8.3	**75.3**	36.2	53.0	10.8	**8.6**	**72.4**
blk2(bie,c)	**33.0**	56.1	11.0	9.2	76.2	36.4	53.7	9.9	9.6	73.2
blk3(bie,c)	32.8	55.5	11.7	8.8	76.0	36.1	**52.9**	11.0	9.0	72.9
NMT(acc)	8.8	80.8	10.4	**8.0**	99.3	9.6	80.9	9.5	9.0	99.4

表 3 Evaluation on TED-LIUM development (dev) and test set. CER of baseline was 12.8% (dev) and 12.4% (test).

Set	dev					test				
	C	S	D	I	E	C	S	D	I	E
baseline	78.5	17.7	**3.8**	4.3	25.8	78.8	16.9	**4.2**	3.4	24.5
ign(bie,b)	78.2	17.9	**3.8**	4.2	26.0	78.8	17.0	**4.2**	3.4	24.6
ign(bie,c)	**78.6**	17.6	**3.8**	4.3	25.7	**79.0**	16.7	**4.2**	3.4	24.3
blk(w,b)	74.5	20.9	4.7	3.9	29.4	75.4	19.4	5.1	3.1	27.6
blk(w,c)	77.4	17.4	5.2	**3.8**	26.4	77.8	16.6	5.6	**3.0**	25.2
blk(bie,b)	78.2	16.9	5.0	**3.8**	25.6	78.6	15.8	5.6	**3.0**	24.5
blk(bie,c)	78.4	**16.3**	5.3	**3.8**	**25.4**	78.8	**15.5**	5.7	**3.0**	**24.2**
b+c1(bie,b)	78.2	16.8	5.0	4.0	25.9	78.6	16.0	5.4	3.3	24.7
b+c1(bie,c)	78.5	16.4	5.1	3.9	25.5	78.9	15.6	5.6	3.3	24.4
b+c2(bie,b)	78.2	16.9	4.9	4.0	25.8	78.6	16.0	5.4	3.3	24.7
b+c2(bie,c)	78.4	**16.3**	5.3	4.0	25.6	78.8	**15.5**	5.7	3.2	24.4
NMT(acc)	34.6	53.6	11.7	11.7	77.0	41.2	46.4	12.4	11.6	70.4
NMT(bleu)	34.0	54.2	11.8	12.0	77.9	40.2	47.8	12.0	11.6	71.4

6. まとめ

E2E 音声認識システムは騒音の影響を受けやすい．とりわけ，文字単位の E2E は明示的な言語制約を使っていないため，スペル誤りのような誤りを出力することがある．これらの誤りを修正するため，スペル誤りの修正を目的とした scRNN を音声認識の問題に適用した．scRNN は置換誤りにのみ焦点を当てているため，直接的な適用は困難である．削除・挿入誤りを扱うため，空白単語記号と単語結合を導入した．2 種の音声認識タスクの実験により，両タスクに対して，我々の拡張を入れた scRNN はベースラインの性能を改善した一方，NMT は全く改善が見られなかった．とりわけ，騒音下音声認識タスクにおいて，提案の scRNN は WER を相対値で 4%改善した．

参考文献

[1] Kita, K., Kawabata, T. and Hanazawa, T.: HMM Continuous Speech Recognition Using Stochastic Language Models, *Proceedings of ICASSP*, Vol. 1, pp. 581–584 (1990).
[2] Graves, A. and Jaitly, N.: Towards End-to-end Speech Recognition with Recurrent Neural Networks, *Proceedings of the 31st International Conference on Machine Learning*, pp. 1764–1772 (2014).
[3] Miao, Y., Gowayyed, M. and Metze, F.: EESEN: End-to-end Speech Recognition Using Deep RNN Models and WFST-based Decoding, *Proceedings of ASRU*, pp. 167–174 (2015).
[4] Chorowski, J., Bahdanau, D., Serdyuk, D., Cho, K. and Bengio, Y.: Attention-based Models for Speech Recognition, *Proceedings of NIPS*, pp. 577–585 (2015).
[5] Senior, A., Sak, H., de Chaumont Quitry, F., Sainath, T. and Rao, K.: Acoustic Modeling with CD-CTC-SMBR LSTM RNNs, *Proceedings of ASRU*, pp. 604–609 (2015).
[6] Bahdanau, D., Chorowski, J., Serdyuk, D., Brakel, P. and Bengio, Y.: End-to-end Attention-based Large Vocabulary Speech Recognition, *Proceedings of ICASSP*, pp. 4945–4949 (2016).
[7] Lu, L., Zhang, X. and Renals, S.: On Training the Recurrent Neural Network Encoder-decoder for Large Vocabulary End-to-end Speech Recognition, *Proceedings of ICASSP*, pp. 5060–5064 (2016).
[8] Prabhavalkar, R., Sainath, T., Li, B., Rao, K. and Jaitly, N.: An Analysis of "Attention" in Sequence-to-sequence Models, *Proceedings of INTERSPEECH*, pp. 3702–3706 (2017).
[9] Audhkhasi, K., Ramabhadran, B., Saon, G., Picheny, M. and Nahamoo, D.: Direct Acoustics-to-word Models for English Conversational Speech Recognition, *Proceedings of INTERSPEECH*, pp. 959–963 (2017).
[10] Soltau, H., Liao, H. and Sak, H.: Neural Speech Recognizer: Acoustic-to-word LSTM Model for Large Vocabulary Speech Recognition, *Proceedings of INTERSPEECH*, pp. 3707–3711 (2017).
[11] Kim, S., Hori, T. and Watanabe, S.: Joint CTC-attention Based End-to-end Speech Recognition Using Multi-task Learning, *Proceedings of ICASSP*, pp. 4835–4839 (2017).
[12] Watanabe, S., Hori, T., Kim, S., Hershey, J. R. and Hayashi, T.: 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).
[13] Watanabe, S., Hori, T. and Hershey, J. R.: Language Independent End-to-end Architecture for Joint Language and Speech Recognition, *Proceedings of ASRU*, pp. 265–271 (2017).
[14] Li, J., Ye, G., Zhao, R., Droppo, J. and Gong, Y.: Acoustic-to-word Model without OOV, *Proceedings of ASRU*, pp. 111–117 (2017).
[15] Luong, M.-T., Pham, H. and Manning, C. D.: Effective Approaches to Attention-based Neural Machine Translation, *Proceedings of EMNLP*, pp. 1412–1421 (2015).
[16] Sakaguchi, K., Duh, K., Post, M. and Van Durme, B.: Robsut Wrod Reocginiton via Semi-Character Recurrent Neural Network (Authors intentionally jumbled the title), *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 3281–3287 (2017).
[17] Tachioka, Y. and Watanabe, S.: Discriminative Method for Recurrent Neural Network Language Models, *Proceedings of ICASSP*, pp. 5386–5390 (2015).
[18] Vincent, E., Watanabe, S., Nugraha, A. A., Barker, J. and Marxer, R.: An Analysis of Environment, Microphone and Data Simulation Mismatches in Robust Speech Recognition, *Computer Speech and Language*, Vol. 46, pp. 535–557 (2016).
[19] Rousseau, A., Deléglise, P. and Estève, Y.: Enhancing the TED-LIUM Corpus with Selected Data for Language Modeling and More TED Talks, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)* (2014).
[20] Kingma, D. and Ba, L.: Adam: A Method for Stochastic Optimization, *Proceedings of ICLR*, (2015).
[21] Britz, D., Goldie, A., Luong, T. and Le, Q.: Massive Exploration of Neural Machine Translation Architectures, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 1442–1451 (2017).
[22] Papineni, K., Roukos, S., Ward, T. and Zhu, W.-J.: BLEU: A Method for Automatic Evaluation of Machine Translation, *Proceedings of the 40th Annual Meeting of Association for Computational Linguistics (ACL)*, pp. 311–318 (2002).