Vol.2019-SLP-128 No.10
2019/7/20

双方向談話コンテキスト言語モデルに基づく反復リスコアリング

増村 亮[1,a)]　田中 智大[1]　斉藤 いつみ[1]　西田 京介[1]　大庭 隆伸[1]

概要：本稿では，会話や談話等の発話系列を扱う必要がある音声認識タスク向けの新たな言語モデルを提案する．このようなタスクで高精度な音声認識を実現するためには，言語モデルにおいて，発話境界を超えた言語コンテキストを精緻に捉えることが重要となる．これに対して，発話境界を超えた過去の言語コンテキストを利用するためのモデル化が近年検討されており，発話内の言語コンテキストのみを用いるモデル化と比較して，音声認識の改善に有効であることが報告されている．しかしながら，これまでの検討では，発話境界を超えた未来の言語コンテキストを利用するようなモデル化は検討されてこなかった．そこで本稿では，発話境界を超えた過去と未来の言語コンテキストを同時に考慮可能な双方向談話コンテキスト言語モデルを提案する．双方向談話コンテキスト言語モデルは，一般的な言語モデルと同様に，発話文に対する単語単位の自己回帰生成モデルとして表され，双方向の階層再帰型エンコーダデコーダを導入することで，対象の発話文以外の過去と未来の全ての発話文の系列情報も言語コンテキストとして考慮できる．さらに本稿では，会話や談話等の発話系列全体の音声認識仮説をインクリメンタルにリスコアリング行う双方向談話コンテキスト言語モデルによる音声認識の反復リスコアリングを提案する．日本語話し言葉コーパスを用いた音声認識実験において，双方向談話コンテキスト言語モデル，および反復リスコアリングの有効性を検証する．

キーワード：双方向談話コンテキスト言語モデル，反復リスコアリング，音声認識，日本語話し言葉コーパス

1. はじめに

談話や会話等の音声認識タスクでは，音声検索等の単発話の音声認識タスクとは大きく異なり，入力音声が発話系列として表されることが一般的である．この場合，性能の高い音声認識を実現するためには，言語モデルにおいて，発話内の言語コンテキストだけではなく，発話境界を超えた言語コンテキストまでを考慮して発話文予測を行うことが重要となる [1].

これまで，長距離の言語コンテキストを捉えるために，様々な言語モデルの検討がなされてきた．多くの音声認識システムで採用されている言語モデルは，スムージング付き n-gram 言語モデルであるが，考慮可能な言語コンテキストが直前の数単語のみに限られる点が課題として知られている．これに対して，近年最も注目を集めている言語モデルは，Long Short-Term Memory (LSTM) を含むリカレントニューラルネットワーク (RNN) に基づく言語モデルである．RNN を用いることで，発話文内のの長距離の言語コンテキストを柔軟に考慮することができ，スムージング付き n-gram 言語モデルよりも高い言語予測性能を示すことが知られている [2]．しかしながら，発話文を超えた言語コンテキストを明示的に捉えることができない点が課題として知られている．

これに対して、RNN を多段に用いることで，発話境界を超えた言語コンテキストの考慮を可能とする談話コンテキスト言語モデルが新たに注目を集めている．談話コンテキスト言語モデルは，自己回帰生成モデルと条件付き自己回帰生成モデルの両者で検討が進んできている．自己回帰生成モデルの場合は，発話文の生成確率を単語単位の自己回帰生成モデルとして計算する際に，発話文内の単語履歴だけでなく，過去の発話文も言語コンテキストを考慮する手法が検討されている [3–6]．一方，条件付き自己回帰生成モデルとしての談話コンテキスト言語モデルは，End-to-End 音声認識 [7–9] や End-to-End 機械翻訳 [10–12] において検討が進んできており，発話境界を超えた過去の言語コンテキストを考慮することで，発話内の情報のみを考慮する場合よりも高い性能が実現できることが報告されている．

[1] 日本電信電話株式会社 NTT メディアインテリジェンス研究所
[a)] ryou.masumura.ba@hco.ntt.co.jp

しかしながら，従来研究で検討されたモデル化では，対象発話よりも過去の発話文のみを言語コンテキストとしており，未来の発話文を言語コンテキストとして考慮することはなかった．これは言語モデルの役割に起因するものであり，一般的な自己回帰生成モデルでは，時系列順に単語や文を生成することを前提としているためである．一方で，未来の言語コンテキストの情報には，過去の発話文の言語予測を助けることが期待される．したがって，未来の言語コンテキストを観測可能な条件下であれば，積極的に利用するモデル化が求められる．

そこで本稿では，発話境界を超えた過去と未来の言語コンテキストを同時に考慮可能な双方向談話コンテキスト言語モデルを提案する．双方向談話コンテキスト言語モデルは，これまでの言語モデルと同様に，発話文に対する単語単位の自己回帰生成モデルとして表され，双方向の階層再帰型エンコーダデコーダを用いることで対象の発話文以外の全ての発話文の系列情報も言語コンテキストとして考慮することが可能である．我々のアイデアは，双方向談話コンテキスト言語モデルを発話系列全体の音声認識仮説が与えられた状態でリスコアリングのために用いることである．これにより，未来の言語コンテキストを活用することが可能となる．さらに，双方向談話コンテキスト言語モデルの機能を最大限に活かすために，本稿では反復リスコアリングを導入する．反復リスコアリングでは，各発話文のリスコアリングを発話系列全体に対して複数回繰り返すことにより実現できる．これにより，リスコアリングにより改善された過去と未来の言語コンテキストを用いて，インクリメンタルにリスコアリングの機能を上げることが可能となる．

本稿では，日本語話し言葉コーパスを用いた音声認識実験において，発話境界を超えた過去と未来の言語コンテキストを同時に考慮することが有効であることを示す．また，反復リスコアリングによりさらなる改善効果が得られることを示す．

2. 関連研究

本節では，関連研究との位置づけについて述べる．

双方向言語モデル: 双方向談話コンテキスト言語モデルは双方向言語モデルの一種と見なせる．これまでの双方向言語モデルは，単語単位の生成確率を計算する際に，発話内の過去と未来の両者の単語系列を言語コンテキストとして考慮するモデル化を採用している [13–19]．したがって，これまでの双方向言語モデルは，単語単位の自己回帰生成モデルとしての機能は持っていない．これに対して，双方向談話コンテキスト言語モデルでは，発話境界を超えた過去と未来の言語コンテキストを利用するものの，発話内の未来の単語系列は言語コンテキストとして利用しないため，単語単位の自己回帰生成モデルとして表現できる．これにより，双方向談話コンテキスト言語モデルは，n-gram 言語モデルや RNN 言語モデルの自然な拡張として解釈を与えることができる．

ポストエディッティング: 双方向談話コンテキスト言語モデルを用いたリスコアリングはポストエディッティングに関連する．ポストエディッティングは，機械翻訳や音声認識において，一度デコーディングにより生成した結果を事後的に編集する処理であり，近年ニューラルネットワークを用いたモデル化が導入されている．これまでのポストエディッティングは，対象の発話文の仮説を言語コンテキストとして発話文の編集を行うことが一般的であり，対象の発話文以外の周囲の仮説の情報は利用していなかった [20–23]．これに対して，提案手法は，談話や会話等の複数文含むことを前提として，対象発話文の周囲の仮説情報を用いるポストエディッティングとみなすことができ，このような検討は，我々の知る限り本稿が初出である．また，反復的なポストエディティングはこれまでも提案されているが，周囲の仮説情報を反復的に更新して利用している点が過去の研究と異なる [24, 25]．

3. 双方向談話コンテキスト言語モデル

本節では，提案手法である双方向談話コンテキスト言語モデルについて，モデル定義，および学習方法の詳細を述べる．さらに，双方向談話コンテキスト言語モデルを用いた音声認識のリスコアリング方法について述べる．

3.1 モデル定義

双方向談話コンテキスト言語モデルは，談話内のある発話文の生成確率を計算するために，一般的な言語モデルと同様に単語単位の自己回帰生成モデルを構成するモデル化であり，その際に，全ての過去の発話文系列と全ての未来の発話文系列の情報も考慮する．ここでは，T 発話から構成される談話を $W^{1:T} = \{W^1, \cdots, W^T\}$ と定義する．この時，t 番目の発話文は単語系列 $W^t = \{w_1^t, \cdots, w_{N^t}^t\}$ として表され，N^t は t 番目の発話文に含まれる単語数を表す．双方向談話コンテキスト言語モデルでは，t 番目の発話文の生成確率を (1) 式の通り計算する．

$$P(W^t|W^{1:t-1}, W^{t+1:T}, \boldsymbol{\Theta}) = \prod_{n=1}^{N_t} P(w_n^t|w_1^t, \cdots, w_{n-1}^t, W^{1:t-1}, W^{t+1:T}; \boldsymbol{\Theta}) \quad (1)$$

ここで，$\boldsymbol{\Theta}$ はモデルパラメータを表す．

我々は，双方向談話コンテキスト言語モデルを階層再帰型エンコーダデコーダ [26–28] を過去の系列情報と未来の系列情報の両者を考慮できるように拡張することで構成する．具体的には，2 種類の発話レベルの RNN を用いることで，過去の発話系列と未来の発話系列をそれぞれ固定長

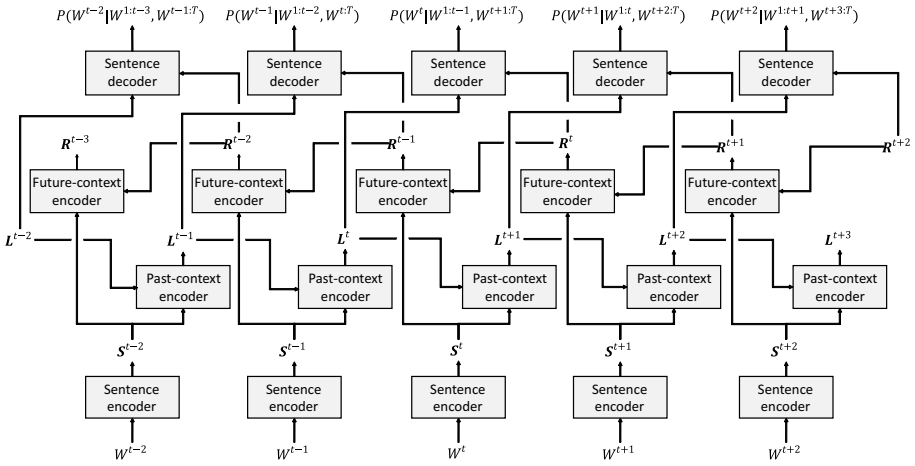

図 1 双方向談話コンテキスト言語モデルのモデル構造.

ベクトルに埋め込み，現在の発話文の生成確率を計算する際に利用する．

双方向談話コンテキスト言語モデルは，発話文デコーダ，過去コンテキストエンコーダ，未来コンテキストエンコーダ，そして発話文エンコーダの 4 つの部位によって構成される．図 1 に，双方向談話コンテキスト言語モデルのモデル構造を示す．t 番目の発話文における n 番目の単語の生成確率は (2) 式で表される．

$$P(w_n^t|w_1^t,\cdots,w_{n-1}^t,W^{1:t-1},W^{t+1:T};\boldsymbol{\Theta}) = \texttt{Decoder}(w_1^t,\cdots,w_{n-1}^t,\boldsymbol{L}^t,\boldsymbol{R}^t;\boldsymbol{\Theta}) \quad (2)$$

ここで，Decoder() は自己回帰生成モデルに対応する発話文デコーダである．$\boldsymbol{L}^t$ は $\boldsymbol{S}^{1:t-1}=\{\boldsymbol{S}^1,\cdots,\boldsymbol{S}^{t-1}\}$ を埋め込んだ固定長ベクトル，$\boldsymbol{R}^t$ は $\boldsymbol{S}^{t+1:T}=\{\boldsymbol{S}^{t+1},\cdots,\boldsymbol{S}^{T}\}$ を埋め込んだ固定長ベクトルを表し，それぞれ (3) 式，(4) 式の通り計算できる．

$$\begin{aligned}\boldsymbol{L}^t &= \texttt{PastEncoder}(\boldsymbol{S}^{1:t-1};\boldsymbol{\Theta}) \\ &= \texttt{PastEncoder}(\boldsymbol{S}^{t-1},\boldsymbol{L}^{t-1};\boldsymbol{\Theta})\end{aligned} \quad (3)$$

$$\begin{aligned}\boldsymbol{R}^t &= \texttt{FutureEncoder}(\boldsymbol{S}^{t+1:T};\boldsymbol{\Theta}) \\ &= \texttt{FutureEncoder}(\boldsymbol{S}^{t+1},\boldsymbol{R}^{t+1};\boldsymbol{\Theta})\end{aligned} \quad (4)$$

ここで，PastEncoder()，および FutureEncoder() は，それぞれ過去コンテキストエンコーダと未来コンテキストエンコーダを表す．それぞれ，1 つ前や 1 つの後のタイムステップの情報を再帰的に用いることで，長距離の情報を効率的に固定長ベクトルに埋め込むこと役割を担う．$\boldsymbol{S}^t$ は，発話文 W^t を埋め込んだ固定長ベクトルを表し，(5) 式の通り計算できる．

$$\boldsymbol{S}^t = \texttt{SentenceEncoder}(W^t;\boldsymbol{\Theta}), \quad (5)$$

ここで，SentenceEncoder() は発話文エンコーダを表す．

双方向談話コンテキスト言語モデルにおいて，過去コンテキストと未来コンテキストの両者を全く用いない場合，すなわち $\boldsymbol{L}^t$，および $\boldsymbol{R}^t$ の両者を t に関わらずゼロベクトルとする場合，RNN 言語モデルのモデル構造に一致する．また，$\boldsymbol{L}^t$ のみを用いる場合は，すなわち $\boldsymbol{R}^t$ を t にかかわらずゼロベクトルとする場合，階層再帰型エンコーダデコーダに基づく談話コンテキスト言語モデルのモデル構造に一致する．

3.2 詳細なモデル構造

双方向談話コンテキスト言語モデルの詳細なモデル構造を入力側から順番に説明する．

単語埋め込み層: 発話文エンコーダ，および発話文デコーダでは，単語を連続値ベクトルに変換して用いる．t 番目の発話文の n 番目の単語に対応する連続値ベクトルは (6) 式で表される．

$$\boldsymbol{w}_n^t = \texttt{EMBED}(w_n^t;\boldsymbol{\theta}_{\texttt{w}}) \quad (6)$$

ここで，EMBED() は単語を単語ベクトルに埋め込むための線形変換関数であり，$\boldsymbol{\theta}_{\texttt{w}}$ は学習可能なモデルパラメータを表す．

発話文エンコーダ: 発話文エンコーダでは，双方向 LSTM (BLSTM) と Self-Attetion 機構を用いることで，発話文の重要な言語コンテキストを固定長ベクトルに埋め込む．t 番目の発話文に対応する固定長ベクトルは (7) 式で表される．

$$\boldsymbol{S}^t = \texttt{SelfAttention}(\boldsymbol{U}^t;\boldsymbol{\theta}_{\texttt{s}}) \quad (7)$$

ここで，SelfAttention() は，入力されたベクトル系列に対して，各要素の重要度を考慮して重みづけて足し合わせることで固定長ベクトルに変換する関数であり，$\boldsymbol{\theta}_{\texttt{s}}$ は学習可能なモデルパラメータを表す [29]．$\boldsymbol{U}^t$ は (8) 式に従い，単語の連続値ベクトル系列から計算される．

$$\boldsymbol{U}^t = \texttt{BLSTM}(\boldsymbol{w}_1^t,\cdots,\boldsymbol{w}_{N^t}^t;\boldsymbol{\theta}_{\texttt{u}}) \quad (8)$$

ここで、BLSTM() は，BLSTM の機能を持つ関数であり，$\boldsymbol{\theta}_{\mathtt{u}}$ は学習可能なモデルパラメータを表す．

過去コンテキストエンコーダ: 過去コンテキストエンコーダでは，発話文単位の順方向 LSTM を用いることで，過去の発話文系列全体を固定長ベクトルに変換する．1 番目の発話文から $t-1$ 番目の発話文までを埋め込んだ固定長ベクトルは (9) 式で表される．

$$\begin{aligned}\boldsymbol{L}^t &= \overrightarrow{\mathtt{LSTM}}(\boldsymbol{S}^1, \cdots, \boldsymbol{S}^{t-1}; \boldsymbol{\theta}_{\mathtt{l}}) \\ &= \overrightarrow{\mathtt{LSTM}}(\boldsymbol{S}^{t-1}, \boldsymbol{L}^{t-1}; \boldsymbol{\theta}_{\mathtt{l}})\end{aligned} \tag{9}$$

ここで，$\overrightarrow{\mathtt{LSTM}}()$ は順方向 LSTM の機能を持つ関数であり，$\boldsymbol{\theta}_{\mathtt{l}}$ は学習可能なモデルパラメータを表す．

未来コンテキストエンコーダ: 未来コンテキストエンコーダでは，発話文単位の逆方向 LSTM を用いることで，未来の発話文系列全体を固定長ベクトルに変換する．T 番目の発話文から $t+1$ 番目の発話文までを埋め込んだ固定長ベクトルは (10) 式で表される．

$$\begin{aligned}\boldsymbol{R}^t &= \overleftarrow{\mathtt{LSTM}}(\boldsymbol{S}^{t+1}, \cdots, \boldsymbol{S}^T; \boldsymbol{\theta}_{\mathtt{r}}) \\ &= \overleftarrow{\mathtt{LSTM}}(\boldsymbol{S}^{t+1}, \boldsymbol{R}^{t+1}; \boldsymbol{\theta}_{\mathtt{r}})\end{aligned} \tag{10}$$

ここで，$\overleftarrow{\mathtt{LSTM}}()$ は逆方向 LSTM の機能を持つ関数であり，$\boldsymbol{\theta}_{\mathtt{r}}$ は学習可能なモデルパラメータを表す．

発話文デコーダ: 発話文デコーダは，自己回帰生成モデルに対応し，RNN 言語モデルの拡張モデルとして表現できる．t 番目の発話の n 単語目の単語の生成確率は (11) 式に従い計算できる．

$$\begin{aligned}&P(w_n^t | w_1^t, \cdots, w_{n-1}^t, W^{1:t-1}, W^{t+1:T}; \boldsymbol{\Theta}) \\ &\qquad = \mathtt{SOFTMAX}(\boldsymbol{v}_n^t; \boldsymbol{\theta}_{\mathtt{o}})\end{aligned} \tag{11}$$

ここで，SOFTMAX() は Softmax アクティベーションを含む線形変換関数であり，$\boldsymbol{\theta}_{\mathtt{o}}$ は学習可能なモデルパラメータを表す．$\boldsymbol{v}_n^t$ はコンテキスト情報に対応し，(12)-(13) 式によって得られる．

$$\boldsymbol{v}_n^t = \overrightarrow{\mathtt{LSTM}}(\boldsymbol{c}_{n-1}^t, \boldsymbol{v}_{n-1}^t; \boldsymbol{\theta}_{\mathtt{v}}) \tag{12}$$

$$\boldsymbol{c}_{n-1}^t = [{\boldsymbol{w}_{n-1}^t}^\top, {\boldsymbol{L}^t}^\top, {\boldsymbol{R}^t}^\top]^\top \tag{13}$$

ここで，$\boldsymbol{\theta}_{\mathtt{v}}$ は学習可能なモデルパラメータを表す．このように，発話内の直前の単語の連続値ベクトルとともに，発話境界を超えた過去と未来の言語コンテキストを含んだ固定長ベクトルを入力することで，様々な言語コンテキストを同時に考慮する．

3.3 最適化

談話コンテキスト言語モデルのモデルパラメータの最適化について述べる．ここで，モデルパラメータ $\boldsymbol{\Theta}$ は $\{\boldsymbol{\theta}_{\mathtt{w}}, \boldsymbol{\theta}_{\mathtt{s}}, \boldsymbol{\theta}_{\mathtt{u}}, \boldsymbol{\theta}_{\mathtt{l}}, \boldsymbol{\theta}_{\mathtt{r}}, \boldsymbol{\theta}_{\mathtt{v}}, \boldsymbol{\theta}_{\mathtt{o}}\}$ に対応する．このモデルパラメータの最適化は (14) 式に違う．

$$\hat{\boldsymbol{\Theta}} = \underset{\boldsymbol{\Theta}}{\operatorname{argmin}} - \sum_{W^{1:T} \in \mathcal{D}} \sum_{t=1}^{T} \log P(W^t | W^{1:t-1}, W^{t+1:T}; \boldsymbol{\Theta}) \tag{14}$$

ここで，$\mathcal{D}$ は複数の談話データを含む学習データセットである．この学習は，談話単位のデータでミニバッチを構成することにより，ミニバッチ確率的勾配法により学習できる．

Algorithm 1 音声認識のための反復リスコアリング．

Input: $\boldsymbol{\Lambda}^{1:T} = \{\boldsymbol{\Lambda}^1, \cdots, \boldsymbol{\Lambda}^T\}$, $\boldsymbol{\Theta}$, μ
Output: $\hat{\mathcal{W}}^{1:T} = \{\hat{W}^1, \cdots, \hat{W}^T\}$

```
for t = 1 to T do
    Ŵ^t = argmax_{W^{t,m}} log P(X^t | W^{t,m}) + log P(W^{t,m})
end for
for i = 1 to I do
    for t = 1 to T do
        Ŵ^t = argmax_{W^{t,m}} log P(X^t | W^{t,m}) + μ log P(W^{t,m})
              +(1 − μ) log P(W^{t,m} | Ŵ^{1:t−1}, Ŵ^{t+1:T}; Θ)
    end for
end for
return Ŵ^{1:T} = {Ŵ^1, ··· , Ŵ^T}
```

3.4 音声認識のための反復リスコアリング

双方向談話コンテキスト言語モデルを音声認識に適用する方法について述べる．ここでは，音声認識デコーダによって出力された発話系列の N-best リストを $\boldsymbol{\Lambda}^{1:T} = \{\boldsymbol{\Lambda}^1, \cdots, \boldsymbol{\Lambda}^T\}$ と定義する．ここで，t 番目の発話の N-best リストは $\boldsymbol{\Lambda}^t = \{(W^{t,1}, \boldsymbol{S}(W^{t,1})), \cdots, (W^{t,M}, \boldsymbol{S}(W^{t,M}))\}$ として表され，M は N-best の仮説数，$W^{t,m}$ は t 番目の発話の m 番目の仮説を表す．なお，音響モデルと言語モデルを用いた音声認識システムを用いる場合，入力音声を $\boldsymbol{X}^t$ とすると，$\boldsymbol{S}(W^{t,m})$ は，デコーディング時の音響モデルスコア $\log P(\boldsymbol{X}^t | W^{t,m})$，および言語モデルスコア $\log P(\boldsymbol{W}^{t,m})$ に対応する．

音響モデルと言語モデルを用いた音声認識システムを用いる場合の反復リスコアリングは，Algorithm 1 に従う．1 行目から 3 行目の処理において，デコーディング時のスコアが最大の仮説を初期の音声認識結果に決定する．4 行目から 8 行目の処理において，発話ごとに双方向談話コンテキスト言語モデル，およびデコーディング時のスコアをもとにリスコアリングを行う．なお，μ はリスコアリングのハイパーパラメータを表す．反復リスコアリングにおいて，過去の言語コンテキストはリスコアリングにより更新された仮説を逐次利用することになる．I は反復リスコアリングの反復回数に対応し，$I \geq 2$ の場合に，リスコアリングにより更新された仮説を未来の言語コンテキストとし

表 2　パープレキシティ (PPL)，および単語誤り率 (WER %) の実験結果．

	t 番目の発話に対する過去コンテキスト	t 番目の発話に対する未来コンテキスト	評価データ 1		評価データ 2		評価データ 3	
			PPL	WER%	PPL	WER%	PPL	WER%
ベースライン	-	-	61.86	14.64	63.85	11.52	63.69	12.70
LSTM 言語モデル	-	-	43.47	13.85	46.02	10.81	46.17	11.96
談話コンテキスト言語モデル	$1:t-1$	-	37.75	13.30	38.87	10.15	38.70	11.15
談話コンテキスト言語モデル	-	$t+1:T$	39.79	13.44	41.65	10.42	41.58	11.37
談話コンテキスト言語モデル	$1:t-1$	$t+1:T$	**36.05**	**13.02**	**36.85**	**9.92**	**36.60**	**10.92**

表 1　実験データ．

	講話数	発話数	総単語数
学習データ	3,181	413,240	8,749,094
開発データ	33	4,166	87,568
評価データ 1	10	1,272	30,195
評価データ 2	10	1,292	31,008
評価データ 3	10	1,385	21,053

表 3　反復リスコアリング時の単語誤り率 (%)．

	評価データ 1	評価データ 2	評価データ 3
I=1	13.02	9.92	10.92
I=2	**12.90**	**9.81**	10.85
I=3	**12.90**	**9.81**	**10.80**
Oracle	12.75	9.70	10.64

て利用できることになる．I の値を増やすことにより，より音声認識誤りを含まない情報を言語コンテキストとして利用することができ，リスコアリングによる音声認識性能の改善効果を高めることが期待できる．

4. 評価実験

評価実験では，日本語話し言葉コーパスを用いた談話単位の音声認識タスクにより，提案手法の有効性を評価した．我々は，日本語話し言葉コーパスを学習データ，開発データ，および 3 種類の評価データに分割した．各談話は，無音区間の時間長により自動的に発話単位に分割して用いた．各データの詳細を表 1 に示す．

音声認識実験を行うために，本稿では重み付け有限状態トランスデューサに基づくデコーダを準備した．デコーディングに用いる音響モデルには，メル周波数フィルタバンク出力 40 次元および，そのデルタ成分，デルタデルタ成分を入力音響特徴量とし，3072 の共有トライフォン状態に対する事後確率を出力とする畳み込みニューラルネットワークを準備した．また，デコーディングに用いる言語モデルには，階層 Ptiman-Yor 3-gram 言語モデルを準備した．

4.1 比較手法

提案手法の有効性を評価するために，我々は，LSTM 言語モデル，および 3 種類の談話コンテキスト言語モデルを構築した．LSTM 言語モデルは，650 ユニットの単語埋め込み層，650 ユニットの 1 層 LSTM，語彙サイズ数に対応したユニットを持つ Softmax 層により構成した．その際，単語分散表現の出力に対して，30%のドロップアウトを適用し，LSTM に対して 60%の変分ドロップアウト [30] を適用した．談話コンテキスト言語モデルは，過去の言語コンテキストのみを考慮する場合，未来の言語コンテキストのみを考慮する場合，過去と未来の言語コンテキストを考慮する場合の 3 種類のモデルを構築した．その際，650 次元の単語埋め込み層を導入し，発話文エンコーダ，未来コンテキストエンコーダ，過去コンテキストエンコーダ，発話文デコーダでは，それぞれ 650 ユニットの 1 層 LSTM を導入した．また，単語分散表現の出力に対して，30%のドロップアウトを適用し，発話文デコーダの LSTM に対して 60%の変分ドロップアウト，その他の LSTM に対して 20%の変分ドロップアウトを適用した．各モデルの学習には，ミニバッチ確率的勾配法を用いた．初期の学習率を 0.5 とし，開発データに対するロスを監視することで学習率を調整し，アーリーストッピングにより開発データに対するロスが最小となった際のモデルパラメータを用いた．音声認識のリスコアリングの際は，デコーディング時のの音声認識仮説の 100-best に対して処理を行った．

4.2 実験結果

表 2 にパープレキシティ，および単語誤り率の実験結果を示す．なお，ここでは反復リスコアリングにおける反復回数は 1 としている．ベースラインは，デコーディング時の 1-best についての結果であり，3-gram 言語モデルを用いた場合の性能と見なせる．次に，LSTM 言語モデルを用いてリスコアリングを行うことにより，ベースラインよりも改善効果を示すことが見て取れる．これは，発話内の長距離言語コンテキストを考慮できたことに起因する改善と考えられ，リスコアリングにより音声認識性能を改善可能であることを示している．さらに，各談話コンテキスト言語モデルを導入することにより，LSTM 言語モデルを用いてリスコアリングを行った場合よりも性能改善できることが確認できる．これは，発話境界を超えた各情報が言語予測に有効であることを示している．未来のコンテキストと過去のコンテキストと比較すると，未来のコンテキストは性能改善への寄与が小さいことが見て取れる．これは，通常時系列順に発話系列が生成されるためであると考えられる．そして，未来のコンテキストと過去のコンテキストを

同時に用いることにより，最高性能を示していることが見て取れる．この結果から，発話境界を超えた未来のコンテキストも有効活用することが，音声認識性能の改善に有効であることが示された．

表 3 に，双方向談話コンテキスト言語モデルによる反復リスコアリングの反復回数を増やした場合の評価結果を示す．反復回数を増やすことにより音声認識性能の僅かな改善が確認された．最終的に，3 回反復させた場合の結果は，オラクルのコンテキストを用いた場合と同等を示し，反復リスコアリングの有効性が示された．

5. まとめ

本稿では，双方向談話コンテキスト言語モデル，および音声認識のための反復リスコアリングを提案した．提案手法の強み，発話境界を超えた過去と未来の言語コンテキストを柔軟に捉えて言語予測できる点である．評価実験から，未来の言語コンテキストを捉えることが音声認識性能の改善に有効であることを示した．

参考文献

[1] Rosenfeld, R.: Two decades of statistical language modeling: Where do we go from here?, *Proceedings of the IEEE*, Vol. 88, pp. 1270–1278 (2000).
[2] Sundermeyer, M., Ney, H. and Schluter, R.: From Feedforward to Recurrent LSTM Neural Networks for language models, *IEEE/ACM TASLP*, Vol. 23, No. 3, pp. 517–529 (2015).
[3] Lin, R., Liu, S., Yang, M., Li, M., Zhou, M. and Li, S.: Hierarchical Recurrent Neural Network for Document Modeling, *In Proc. EMNLP*, pp. 899–907 (2015).
[4] Wang, T. and Cho, K.: Larger-Context Language Modelling with Recurrent Neural Network, *In Proc. ACL*, pp. 1319–1329 (2016).
[5] Liu, B. and Lane, I.: Dialogue Context Language Modeling with Recurrent Neural Networks, *In Proc. ICASSP*, pp. 5715–5719 (2017).
[6] Masumura, R., Tanaka, T., Ando, A., Masataki, H. and Aono, Y.: Role Play Dialogue Aware Language Models based on Conditional Hierarchical Recurrent Encoder-Decoder, *In Proc. INTERSPEECH*, pp. 1259–1263 (2018).
[7] Kim, S. and Metze, F.: Dialog-Context Aware End-to-End Speech Recognition, *In Proc. SLT*, pp. 434–440 (2018).
[8] Pundak, G., Sainath, T. N., Prabhavalkar, R., Kannan, A. and Zhao, D.: Deep Context: End-to-end Contextual Speech Recognition, *In Proc. SLT* (2018).
[9] Masumura, R., Tanaka, T., Moriya, T., Shinohara, Y., Oba, T. and Aono, Y.: Large Context End-to-end Automatic Speech Recognition via Extension of Hierarchical Recurrent Encoder-decoder Models, *In Proc. ICASSP*, pp. 5661–5665 (2019).
[10] Jean, S., Lauly, S., Firat, O. and Cho, K.: Does Neural Machine Translation Benefit from Larger Context?, *arXiv preprint arXiv:1704.05135* (2017).
[11] Wang, L., Tu, Z., Way, A. and Liu, Q.: Exploiting Cross-Sentence Context for Neural Machine Translation, *In Proc. EMNLP*, pp. 2826–2831 (2017).
[12] Maruf, S. and Haffari, G.: Document Context Neural Machine Translation with Memory Networks, *In Proc. ACL*, pp. 1275–1284 (2018).
[13] Arisoy, E., Sethy, A., Ramabhadran, B. and Chen, S.: Bidirectional Recurrent Neural Network Language Models for Automatic Speech Recognition, *In Proc. ICASSP*, pp. 5421–5425 (2015).
[14] Mousa, A. E.-D. and Schuller, B.: Contextual Bidirectional Long Short-Term Memory Recurrent Neural Network Language Models: A Generative Approach to Sentiment Analysis, *In Proc. EACL*, pp. 1023–1032 (2017).
[15] Chen, X., Liu, X., Ragni, A., Wang, Y. and Gales, M. J. F.: Future Word Contexts in Neural Network Language Models, *In Proc. ASRU*, pp. 97–103 (2017).
[16] Chen, X., Ragni, A., Liu, X. and Gales, M. J. F.: Investigating Bidirectional Recurrent Neural Network Language Models for Speech Recognition, *In Proc. INTERSPEECH*, pp. 269–273 (2017).
[17] Peters, M. E., Ammar, W., Bhagavatula, C. and Power, R.: Semi-supervised sequence tagging with bidirectional language models, *In Proc. ACL*, pp. 1756–1765 (2017).
[18] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. and Zettlemoyer, L.: Deep contextualized word representations, *In Proc. NAACL-HLT*, pp. 2227–2237 (2018).
[19] Irie, K., Lei, Z., Deng, L., Schluter, R. and Ney, H.: Investigation on Estimation of Sentence Probability By Combining Forward Backward and Bi-directional LSTM-RNNs, *In Proc. INTERSPEECH*, pp. 392–395 (2018).
[20] Pal, S., Naskar, S. K., Vela, M. and van Genabith, J.: A Neural Network based Approach to Automatic Post-Editing, *In Proc. ACL*, pp. 281–286 (2016).
[21] Pal, S., Naskar, S. K., Vela, M., Liu, Q. and van Genabith, J.: Neural Automatic Post-Editing Using Prior Alignment and Reranking, *In Proc. EACL*, pp. 349–355 (2017).
[22] Junczys-Dowmunt, M. and Groundkiewicz, R.: An Exploration of Neural Sequence-to-Sequence Architectures for Automatic Post-Editing, *In Proc. IJCNLP*, pp. 120–129 (2017).
[23] Tanaka, T., Masumura, R., Masataki, H. and Aono, Y.: Neural Error Corrective Language Models for Automatic Speech Recognition, *In Proc. INTERSPEECH*, pp. 401–405 (2018).
[24] Novak, R., Auli, M. and Grangier, D.: Iterative Refinement for Machine Translation, *arXiv preprint arXiv:1610.06602* (2016).
[25] Lee, J., Mansimov, E. and Cho, K.: Deterministic Non-Autoregressive Neural Sequence Modeling by Iterative Refinement, *In Proc. EMNLP*, pp. 1173–1182 (2018).
[26] Sordoni, A., Bengio, Y., Vahabi, H., Lioma, C., Simonsen, J. G. and Nie, J.-Y.: A Hierarchical Recurrent Encoder-Decoder for Generative Context-Aware Query Suggestion, *In Proc. CIKM*, pp. 553–562 (2015).
[27] Serban, I. V., Sordoni, A., Bengio, Y., Courville, A. and Pineau, J.: Building End-to-End Dialogue Systems Using Generative Hierarchical Neural Network Models, *In Proc. AAAI*, pp. 3776–3783 (2016).
[28] Serban, I. V., Sordoni, A., Lowe, R., Charlin, L., Pineau, J., Courville, A. and Bengio, Y.: A Hierarchical Latent Variable Encoder-Decoder Model for Generating Dialogues, *In Proc. AAAI*, pp. 3295–3301 (2017).
[29] Lin, Z., Feng, M., dos Santos, C. N., Yu, M., Xiang, B., Zhou, B. and Bengio, Y.: A Structured Self-attentive Sentence Embedding, *In Proc. ICLR* (2017).
[30] Gal, Y. and Ghahramani, Z.: A theoretically grounded application of dropout in recurrent neural networks, *In Proc. NIPS*, pp. 1027–1035 (2016).