DEIM Forum 2018 G5-3

単語レベルと文字レベルの情報を用いた日本語対話システムの試作

村田　憲俊†　　酒井　哲也†

† 早稲田大学基幹理工学部情報理工学科　〒 169–8555 東京都新宿区大久保 3-4-1
E-mail: †muratacnttto@suou.waseda.jp, ††tetsuyasakai@acm.org

あらまし　近年の深層学習の発達に伴い，雑談のための非タスク指向対話システムの実用化が始まっている．従来の対話システムの研究では英語を対象としたものが主流であったが，日本語対話システムには日本語特有の問題がある．本研究では，テキストの単語分割 (word segmentation) 手法に着目し，日本語テキストを単語レベルに分割した情報と文字レベルに分割した情報の両方を用いた非タスク指向日本語対話システムを試作し，評価を行う．機械翻訳の評価指標 BLEU による評価のもとでは，提案システムは文字レベルの情報のみを用いたシステムおよび単語レベルの情報のみを用いたシステムを平均的に上回り，前者に対する差は統計的に有意であった ($p \approx 0.0000$)．また，単語レベルのシステムのほうが文字レベルのシステムよりも有効であり ($p = 0.0014$)，提案システムと単語レベルのシステムの間には有意差は認められなかった．

キーワード　対話システム，seq2seq，テキストマイニング

1. はじめに

近年，AI スピーカーの家庭への普及や，音声アシスト機能の付いたアプリケーションが数多くサービスとして展開されている．それに伴い，対話システムのさらなる発展の需要が高まっている．開発されている対話システムは，タスク指向のものと非タスク指向のものに大別される．雑談は会話の約 60 %を占めると言われており [1]，非タスク指向の対話システムは会話を行う上で非常に重要である．それらの対話システムは，自然言語処理を元とした技術により構築されている．一般に，対話システムがユーザーからの入力に対して返答する手法は 2 種類に大別される．1 つ目は自然言語処理の手法から得られた情報を元に，人手でルールを作成し，ルールベースで返答を生成するものである．2 つ目は機械学習の手法も用いて，実際の会話データから返答を生成する，非ルールベースの手法である．ルールベースなものには ELIZA [2] がある．現在，非ルールベースの手法は deep learning といった機械学習の手法の発展と，ルールベースでは表現しきれない多様な回答を実現することの需要に伴い，数多くの手法が提案されている．しかし，その多くは英語を対象として評価されており，日本語を扱った場合の技術課題が必ずしも明らかになっていない．日本語の対話システムを構築する上で，英語にはない課題のひとつに単語分割 (word segmentation) がある．日本語は英語と異なり，文中の単語が空白等で区切られていないため，英語で用いられている手法をそのまま利用することができない．よって，日本語での対話システムの開発には，文を文字レベルで分割し利用する方法と，分かち書きを行い単語レベルで分割し利用する方法 [3] が考えられる．

本稿では，ニューラルネットを用いた非タスク指向の非ルールベースの日本語対話システムを作成する．日本語における大きな課題である単語分割の問題に着目した，新たな日本語対話システム手法の試作，評価を行う．この手法では，文字レベルと単語レベルの両方の分割から得られた情報を元に文の生成を行う．

2. 関連研究

本節では，本研究に関連する研究について説明をする．

2.1 Sequence-to-Sequence

Sequence-to-Sequence は再帰ニューラルネットワーク (Recurrent Neural Network; RNN) である．文 (sentence) のような可変長の文字列をシーケンスデータとして扱う．可変長のシーケンスデータを入力とし，可変長のシーケンスデータを出力することができるニューラルネットワークモデルである．しかし，このモデルには時間方向に層が深くなるにつれて，学習が上手く進まなくなる勾配消失問題があった．この問題への対策として，Long Short-Term Memory (LSTM) [4] を用いたモデルがあり，勾配消失問題を緩和し，性能が向上することが知られている [5]．LSTM を用いた場合，長さ T_i の入力シーケンス $\{x_1, x_2, ..., x_{T_i}\}$ に対して，長さ T_o の出力シーケンス $\{h_1, h_2, ..., h_{T_o}\}$ を得るための処理は以下のように示される．

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \quad (1)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \quad (2)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \quad (3)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (4)$$

$$h_t = o_t \odot \tanh(c_t) \quad (5)$$

σ はシグモイド関数，tanh はハイパボリックタンジェントである．W_* は i_t, f_t, o_t, c_t, h_t を算出するための重み行列，b_* はバイアスベクトルである．LSTM の学習では，W_* と b_* が学習により最適化されるパラメータである．

Sequence-to-Sequence を言語モデルとして利用するためには，更に拡張が必要となる．まず，入出力ベクトルの次元数 (語彙数) と処理系内部での次元数を合わせるために，以下の式が

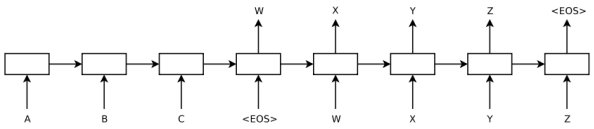

図 1　ニューラルネットワーク対話モデル（文献 [6] より引用）

導入される．

$$x_t = W_{x'x} x'_t \tag{6}$$

$$y_t = softmax(W_{hy} h_t) \tag{7}$$

入力シーケンス (単語列) は $\{x'_1, x'_2, ..., x'_{T_i}\}$ で，x'_* は語彙数と同じ次元数のベクトルである．語彙を表現するために，該当するインデックスの次元の値のみが 1 となり，それ以外の値は 0 になるような one-hot ベクトルになっている．出力シーケンスは $\{y_1, y_2, ..., y_{T_o}\}$ で，y_* は，時系列における各時刻の予測単語の確率分布ベクトルである．softmax はソフトマックス関数であり，出力値を正規化し，確率分布として扱うことができる．

上記の言語モデルの学習では，入力シーケンスは単語列に該当し，$\{x'_1(=\langle BGN\rangle), x'_2, ..., x'_{T_i}\}$ を入力することにより，それぞれの次の単語が何であるかを順番に予測していく．$\langle BGN\rangle$ は文頭を表現するためのシンボル単語である．言語モデルの場合，期待される出力単語列は $\{y'_1 = x'_2, y'_2 = x'_3, ..., y'_{T_i} = \langle END\rangle\}$ となる．これは，入力単語の次の単語が何であるかを予測していることになる．$\langle END\rangle$ は文末を表現するためのシンボル単語である．$\langle BGN\rangle$ や $\langle END\rangle$ といった特殊なシンボル単語は使用される語彙として事前に one-hot ベクトルの次元として組み込まれる．出力単語列 y'_* は予測出力単語の確率分布を表すベクトルのため，期待される出力単語列の出力確率は以下のような同時確率となる．

$$p(y'_1, y'_2, ..., y'_T | x'_1, x'_2, ..., x'_T) = \prod_{t=1}^{T} p(y'_t | c_{t-1}, h_{t-1}, x'_t) \tag{8}$$

このとき，初期状態 c_0, h_0 は零ベクトルとする．

2.2　対話モデル

Sutskever らが提案した seq2seq [6] を元に，Vinyals らは，対話データを用いるように一部拡張をし，ニューラルネットワークを用いた対話モデルとして提案した [7]．以下に，その提案手法での拡張部分の詳細を述べる．

対話モデルの構造の概要は 2.1 の LSTM を用いた言語モデルと同様である．対話モデルの構造について図 1 に示す．

拡張部分は，主に以下の 2 点である．

I　発話者の切り替えを示すためのシンボル文字の導入

II　学習時に予測する出力単語列を返答文のみとする

I は，対話には発言者と返答者がいるが，入力単語列で発話者の切り替えを表現するために発言終了のシンボル単語 $\langle EOS\rangle$ を導入する．入力単語列中にこのシンボル単語が出現した場合，それは発話者の切り替えを意味し，返答者が発話を開始する．

II は，学習時に予測する出力単語列は返答文のみであり，発言文に関しては無視をするということである．相手の発言文を $\{x_{i*}\}$ とし，それに対するシステムの返答文を $\{x_{o*}\}$ とする．入力単語列は以上のことから $\{x_{i*}, \langle EOS\rangle, x_{o*}, \langle EOS\rangle\}$ となる．これに対して，対話モデルでは $\{\square, ..., \square, x_{o*}, \langle EOS\rangle\}$ を予測すればよく，$\square$ に関しては評価をしない．

3.　提 案 手 法

本節では，ニューラルネットワークを用いた，対話モデルにおける日本語での新しい手法について述べる．英語では文が空白で区切られているため問題にはならないが，日本語の場合には入力単語列を得る際に，何をもって単語と見なすかについて明らかにする必要がある．ここでは，単語レベルの情報と文字レベルの情報を併用し，双方のメリットを活かすことを狙った提案手法について述べる．3.1 で提案手法の概要について説明する．3.2 では，提案手法とするニューラルネットワークによる対話モデルの詳細について説明する．

3.1　概　　要

2. において，関連研究としてニューラルネットワークを用いた対話モデルについて述べた．しかし，これらの研究は，英語などの分かち書きされた文を対象にすることを前提としている．よって，日本語でこの対話モデルをそのまま使用することはできず，日本語の文を何らかの方法で分割をする必要がある．分割の手法は主に以下の 2 つがある．

A　入力文字列の 1 文字ずつを単語として扱う文字レベルでの分割

B　入力文字列を分かち書きし，それぞれを単語として扱う単語レベルでの分割

これらの分割手法であるが，それぞれについて以下のようなメリットとデメリットがある．

A のメリットは，文の文法的な乱れに対して強いことである．一方，単語の区切りの情報を活用できないというデメリットがある．

B のメリットは，形態素解析により取得した単語の情報を利用できることである．一方，形態素解析は辞書や学習コーパスに依存するため，特に文法的な乱れや一般的でない表記が多い文に対しては，必ずしも適切な単語が抽出できないというデメリットがある．

以上の両方のメリットを得るために，提案手法では以下の手順の対話モデルを構築する．

（1）　発言文と返答文をシーケンスとして扱うために，分割を行う．このとき，文字レベルでの分割をしたシーケンスと単語レベルでの分割をしたシーケンスの両方を用意する．

（2）　文字レベルと単語レベルのシーケンスそれぞれに対応した対話モデルを作成し，学習を行う．

（3）　それぞれの対話モデルで返答文の生成を行う．

（4）　それぞれの対話モデルから生成された文を文字レベルで分割する．

（5）　それぞれに生成された返答文を入力とし，最終的な返答文を生成する複合対話モデルを作成し，学習を行う．

（6）　返答文の生成を行う．

以上の提案モデルの概略図を図 2 に示す．

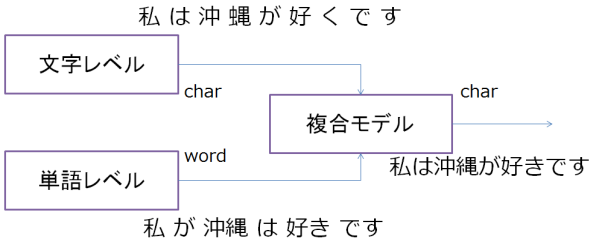

図 2　単語レベルと文字レベルを利用した提案モデルの概要

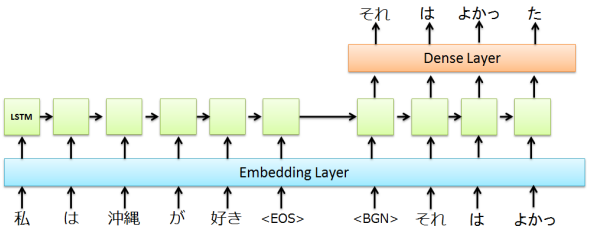

図 3　本研究でのニューラルネットワーク対話モデルの概要

3.2　文字レベル・単語レベル複合対話モデル

以下では，提案手法の各手順について詳細を説明する．

3.2.1　単語分割とシーケンスの整形

本研究では単語分割について，文字レベルと単語レベルの 2 種類の情報を使用する．まず，それぞれの単語分割について例を示す．「私は夏休みに沖縄に行きました」という文があったとする．また，単語分割を行った後のそれぞれの単語について，文字レベルでは $\{x_{c1}, x_{c2}, ..., x_{ct}\}$ とし，単語レベルでは $\{x_{w1}, x_{w2}, ..., x_{wt}\}$ とする．t は各文の文長である．この場合，文字レベルでは，{ 私, は, 夏, 休, み, に, 沖, 縄, に, 行, き, ま, し, た } となる．単語レベルでは形態素解析を行い，{ 私, は, 夏休み, に, 沖縄, に, 行き, まし, た } といったようになる．

以上の手順を行った後に，ニューラルネットワークで学習を行うために 2. で行われていたことを含め拡張を行う．まず，文字列に $\langle BGN \rangle, \langle EOS \rangle$ を導入をする．$\langle BGN \rangle$ は文頭である x_{c0} として追加をする．$\langle EOS \rangle$ は文末である x_{ct+1} として追加をする．次に，ニューラルネットワーク対話モデルは任意の長さのシーケンスを扱うことができるが，実装が非常に複雑になるため文長を一定にする．そのためにシンボル文字 $\langle PAD \rangle$ を導入する．まず文字列長 T を設定する．本研究では $T = 70$ とした．その後，データセットで内の各文を文字列長 T に合わせる．各文の文字列長 t が $t < T$ の場合，$t = T$ になるように，$\langle PAD \rangle$ を文字列として末尾に追加する．$t > T$ の場合，$t = T$ になるように，文字列の末尾を削除する．上記の例の単語レベルで $T = 15$ とすると $\{\langle BGN \rangle$, 私, は, 夏休み, に, 沖縄, に, 行き, まし, た, $\langle EOS \rangle, \langle PAD \rangle, \langle PAD \rangle, \langle PAD \rangle, \langle PAD \rangle\}$ となる．

3.2.2　文字レベルと単語レベルの対話モデルの作成

文字レベルと単語レベルの対話モデルには，2. で用いられているモデルを拡張したものを使用する．そのために前工程で作成したシーケンスの各単語をベクトルに変換する．文字レベルで用いられる語彙全体を $X_c \ni x_{c*}$ とする．また，単語レベルで用いられる語彙全体を $X_w \ni x_{w*}$ とする．これらの語彙にはシンボル文字も含まれる．文字レベルでの語彙のベクトル化においては，まず X_c の各要素にインデックスを付ける．その後，x_{c*} をそれぞれ次元が X_c と同じ大きさにし，該当インデックスの位置のみ 1 となり，それ以外が 0 のベクトル（one-hot ベクトル）にする．単語レベルでも X_w を使い，同様の処理を行う．対話モデル内部ではこのベクトルを embedding layer [8] に入れ固定長のベクトルに変換する．

単語のベクトル化が終了後，ニューラルネットワーク対話モデルの学習を行わせる．ここでは，文字レベルと単語レベルは語彙が異なるのみのため，文字レベルの場合のみを述べる．モデルは，発言文の解釈を行う *Encoder* と返答文の生成を行う *Decoder* に分けることができる．*Encoder* では，発言文のベクトルを入力とし，計算を行い内部状態を *Decoder* に共有する．*Decoder* では，*Encoder* の内部状態を受け取り，学習の際は返答文のベクトルを入力とする．$\{x_{c1} = \langle BGN \rangle, x_{c2}, ..., x_{cT-1}\}$ を *Decoder* への入力とし，$\{x_{c2}, ..., x_{cT}\}$ を正解として学習を行う．これは，1 時刻前で出力した文字を入力として，現時刻の出力を予測することに当たる．このとき用いたニューラルネットワーク対話モデルの概要を図 3 に示す．

3.2.3　複合対話モデルの作成

文字レベルと単語レベルの対話モデルから生成された文から得られる情報を結合して，最終的な返答を生成するモデルの作成を行う．このモデルでは文字レベルと単語レベルから生成された文書をそれぞれ一度日本語に戻し，文字レベルで分割をしなおす．再度分割した 2 つの文を複合対話モデルに入力をし返答を出力する．

文字レベルの対話モデルから得られた文字列を $\{y_{c1}, ..., y_{cT}\}$ とする．また，単語レベルの対話モデルから得られた文字列を $\{y_{w1}, ..., y_{wT}\}$ とする．複合対話モデルでは，2 つの文を入力するために再度文字レベルに分割しなおす．その文字列をそれぞれ x'_{c*}, x'_{w*} とする．

複合対話モデルのニューラルネットワークの構造は 3.2.2 のモデルを結合したものと考えることができる．複合対話モデルでは，x'_{c*}, x'_{w*} それぞれに対して *Encoder* と *Decoder* を設ける．そして，それぞれの LSTM layer からの出力を共通の dense layer で受け，$softmax$ 関数により出力 y'_* を決定する．

以上のモデルの複合対話モデルを図 4 に示す．

4.　評価・実験

提案手法の性能を評価するために比較実験を行った．比較実験では，同一のニューラルネットワーク構造を持つ，文字レベルでの対話モデル (char-model)，単語レベルでの対話モデル (word-model)，2 つを複合した提案モデル (compound-model) について比較を行った．これによって，この提案モデルがどのような特徴を持つかを分析した．4.1 では評価に使用したデータセットの説明，4.2 では評価を行う上で設定をした各種数値等の説明，4.3 では実験結果の説明を行う．

4.1　データセット

対話の学習データとして，Twitter のツイートを利用した．

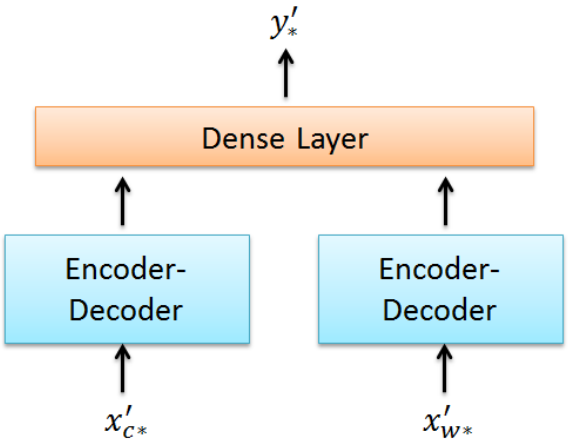

図 4　複合対話モデル

Twitter は現在，日本国内で幅広く利用されており，数多くのジャンルの会話が行われていることが期待される．また，ツイートは日本語の文法的な乱れやネットスラングが含まれていることが多いため，精度の高い分かち書きが困難である．そのため，文字レベルの対話モデルと単語レベルの対話モデルのそれぞれの特徴を活かせる可能性がある．本研究では任意のツイートと，そのツイートに対するリプライというペアを対話データとして扱った．学習データ全体は 2014/01/01 から 2014/12/31 までに取得された 1,000,000 (1M) ツイートを対象としており，500,000 ペアの対話のデータを用意した．

評価を行うために，上記の学習データを $train, test, val$ の 3 つのデータセットに分割した．分割をする際，今回のデータは 1 年間のデータを取得しているため，日付に対してランダムに抽出を行った．これは，例えば年始の挨拶のようなツイートが $train, test, val$ のいずれかに偏るような状況を避けるためである．$train$ はモデルの学習のために用いる．$test$ はモデルの評価のために用いる．val はモデルが学習をする際に，モデルが $train$ に対して過学習を行わないようにするために，epoch ごとにモデルを検証し，学習を制御するために用いる．各データセットのサイズについて，表 1 に示す．

表 1　データセットのサイズ

種類	割合	サイズ
ALL	100	500,000
train	81	405,000
val	9	45,000
test	10	50,000

4.2 実験設定

以下，本研究での実験条件詳細について述べる．

まず，入力シーケンスである $\{x_{w*}\}$ を得るために，データセットの分かち書きが必要となるデータセットの分かち書きについては $Mecab$ [9] を使用した．文字列長 T については，テキストチャットは一般的に文字列長が短い傾向にあると考えられる．よって，長いものについてはチャット以外の用途で用いられると仮定し，$T = 70$ とした．シンボル文字 $\langle PAD \rangle$ は後ろ詰めで調整をした．単語レベルの語彙 X_w は，データセット全体を形態素解析し，低頻度語を除いた上で $|X_w| = 15000$ となるように設定した．文字レベルの語彙 X_c については，データセットにあるすべての文字を使用した．本研究で使用したデータセットでは $|X_c| = 5955$ となった．

ニューラルネットワーク部分についての説明を行う．embedding layer の出力次元数，LSTM layer の入力次元数は 250 とした．dense layer については各モデルの語彙数と同じである．学習の際の batch サイズは文字レベルのモデル，複合モデルでは 600，単語レベルのモデルでは 300 として学習を行った．学習の際，各 epoch ごとに val データセットを用いて，学習中のモデルの検証を行った．そのとき，val データセットへの損失が上昇した場合，$train$ に対して過学習をしていると考え，学習を止めるようにした．

4.3 実験結果の算出手法

本研究では，各手法を比較するために BLEU [10] [11] を使用する．BLEU は以下の式により計算される．

$$BLEU = BP * PREC \tag{9}$$

$$BP = \exp(\min(0, 1 - \frac{SBML}{SYSL})) \tag{10}$$

$$SBML = \sum_s BML(s) \tag{11}$$

$$= \sum_s \arg\min_{len(s^*)} |len(s) - len(s^*)| \tag{12}$$

$$SYSL = \sum_s len(s) \tag{13}$$

$$len(s) = \text{文 } s \text{ の長さ} \tag{14}$$

$$PREC = \exp(\frac{1}{2} \sum_{N \in \{1,2\}} lnPrec_N) \tag{15}$$

$$Prec_N = \frac{\sum_s \sum_{e \in gram_N(s)} Clip(c, s)}{\sum_s \sum_{e \in gram_N(s)} C(e, s)} \tag{16}$$

$$Clip(e, s) = \min(\max_{s^*} C(e, s^*), C(e, s)) \tag{17}$$

$$C(e, s) = s \text{ の中に含まれている単語 } e \text{ の個数} \tag{18}$$

今回の場合はそれぞれ，s はシステムの返答文，s^* はテストデータの返答文，$gram_N(s)$ は文 s から得られる N-gram の集合，e は文 s を形態素解析をした単語である．また，$N \in \{1, 2\}$ とした．すなわちユニグラムとバイグラムのみを考慮した．

4.4 実験結果

Twitter のデータから作成した対話のデータセットを用いて，提案手法の比較を行った．評価には $test$ データセットを使用した．baseline として，文字レベルでの対話モデルと単語レベルでの対話モデルを用いる．baseline と提案手法に対して，$test$ の発言への返答と，$test$ での実際の返答を用いて BLEU を算出する．

$test$ での BLEU の平均について表 2 に示す．

各モデルに対して BLEU を算出した結果，BLEU の平均値において baseline を僅かに上回った．4.5 でこれらの平均値の差の統計的検定結果を示す．

次に各モデルの $test$ での BLEU のヒストグラムを図 5 に示す．

表 2　各モデルの BLEU の比較

種類	BLEU
char-model(baseline)	0.113847
word-model(baseline)	0.117180
compound-model	0.118322

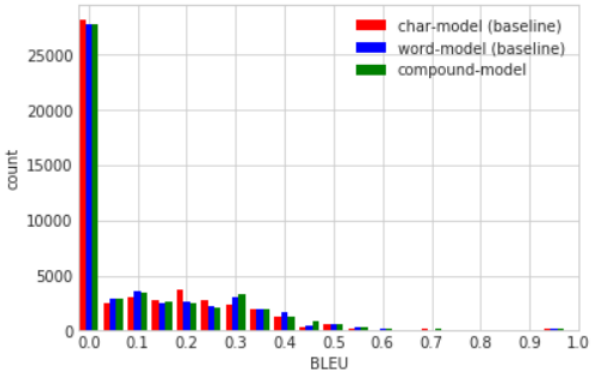

図 5　各手法の BLEU のヒストグラム

どのモデルも共通して，BLEU において 0 付近の値が非常に多くなっている．これはつまりモデルの出力と実際の人間の返答を比較した時，ユニグラムとバイグラムで一致するものが存在しないということが主な要因である．

各モデルが出力した例を表 3 に示す．

複合モデルでは，一部ではあるが文字レベルでのモデルと単語レベルでのモデルの折衷案のような返答を返していることが確認できた．

4.5　Tukey HSD 検定による統計的有意性

本節では上記の各システムにおける BLEU の平均値の差について統計的有意性を確かめるために，Tukey HSD 検定による各システム対の差に関する p 値を算出した．その値を表 4 に示す．

表 4 より，機械翻訳の評価指標 BLEU による評価のもとでは，複合モデルと文字レベルでのシステムの差は統計的に有意であった ($p \approx 0.0000$)．また，単語レベルでのシステムのほうが文字レベルでのシステムよりも有効であり ($p = 0.0014$)，複合システムと単語レベルでのシステムの間には有意差は認められなかった．このことから，提案手法である複合システムに主として貢献しているのは単語レベルのシステムのほうであると考えられる．

5.　考　　察

実験結果において，Twitter の対話データセットに対して，全体的に BLEU が低い数値となっていた．今回，任意のツイートとそれに対する reply を 1 つの対話として扱ったが，これ以前の対話の文脈についての情報がないため，発言からの返答を予測するのが困難であったと考えられる．実際，季節の挨拶や Twitter のフォローに関するやり取り等，返答がある程度定型化しているものについては比較的適切な回答が得られたが，それ以外の発話に対しては「それはそう思う」のように「それは...」などが多用された中身のない回答を返していることが多

表 3　各モデルの出力例

発言文	あけましておめでとうございます．今年もよろしくです!!
正解	おめでとうございます．こちらこそよろしくです
文字レベルモデル	おめでとう !!!!!!!!!!!!! ...
単語レベルモデル	こちらこそありがとうございます
複合モデル	おめでとうございます
発言文	おやすみなさい。今日も一日お疲れ様でした。
正解	おやすみー
文字レベルモデル	おやすみなさい。
単語レベルモデル	おやすみなさいませー
複合モデル	おやすみなさい
発言文	フォローありがとうございます！よろしくお願いします！
正解	こちらこそありがと〜よろしく！
文字レベルモデル	こちらこそよろしくお願いします！
単語レベルモデル	こちらこそよろしくです！
複合モデル	こちらこそよろしくお願いします！
発言文	誕生日おめでと !!
正解	おかちゃあああああああん !!!!! ありがとー !!!
文字レベルモデル	ありがとうございます !!!!!!!!!!!!! ...
単語レベルモデル	ありがとう !!!!!!!!!!!!! ...
複合モデル	ありがとう !!!!!!!!!!!!! ...
発言文	フロントガラス 12 万だってよ。保険きかないってよ。いやっほーい！
正解	きれそう w 思えば 16 日の海から負の連鎖は始まってたんだね…
文字レベルモデル	それは、それは、それでも、それを見てるのかな？
単語レベルモデル	俺も今日は 1 日で 1000 円で 1000 円で 1000000...
複合モデル	それは、それをしてるのかな？
発言文	あいふぉんのケースの中にクレカいれっぱじゃんまずすぎる
正解	
文字レベルモデル	それは、それは、それは、それは、それでも、それを見ているのは、それは、それを見ているのは、それは...
単語レベルモデル	のがいいですよね
複合モデル	それはいいですよね

表 4　各手法同士の BLEU 平均値の差に対する p 値

システム対	p 値
word-model vs char-model	0.001376
compound-model vs char-model	0.000008
compound-model vs word-model	0.453996

かった．今回の提案手法は，ベースとして，文字レベルの対話モデルと単語レベルの対話モデルからの出力を入力としている．そのため，提案手法の出力に関しても同様の内容に乏しい返答になってしまい，実際の返答と比較した BLEU での改善は若干にとどまったと考えられる．提案手法をさらに改善するためには，ニューラルネットワークに対して attention [12] を導入するなどの改善が必要であると考えられる．

また，本研究では対話システムの評価をするために，評価指標として BLEU を用いた．しかし，非タスク指向の対話システムを評価する上では BLEU は最適な評価指標であるとは言い

難い [13]．よって，適切な評価指標自体を検討する必要がある．

6. ま と め

本研究では，文字レベルと単語レベルの情報を併用したニューラルネットワークによる日本語対話システム手法の試作し，評価を行った．機械翻訳評価指標 BLEU により提案手法を文字レベルのみと単語レベルのみのモデルと比較した結果，提案システムは文字レベルの情報のみを用いたモデルを統計的に有意に上回ることが確認できた．一方、単語レベルの情報のみを用いたモデルとの差は統計的に有意ではなかった．今後，文字レベルと単語レベルのそれぞれに用いられているニューラルネットワーク対話モデルに attention などを導入し，生成する文の精度を上げる改善をする必要がある．

文　献

[1] 小磯 花絵, 石本 祐一, 菊池 英明, “大規模日常会話コーパスの構築に向けた取り組み : 会話収録法を中心に,” 言語・音声理解と対話処理研究会, 人工知能学会, Vol. 74, pp. 37–42, 2015.

[2] J. Weizenbaum, “ELIZA—a computer program for the study of natural language communication between man and machine,” Communications of the ACM (CACM), Vol. 13, No. 1, pp. 36–45, 1966.

[3] T. Fujita, W. Bai, C. Quan, “Long short-term memory networks for automatic generation of conversations,” IEEE SNPD 2017, 2017.

[4] S. Hochreiter, J. Schmidhuber, “Long Short-Term Memory,” Journal Neural Computation, Vol. 9, No. 8, pp. 1735–1780, 1997.

[5] A. Graves, “Generating Sequences With Recurrent Neural Networks,” arXiv preprint arXiv:1308.0850, 2014.

[6] I. Sutskever, O. Vinyals, Q. V. Le, “Sequence to Sequence Learning with Neural Networks,” NIPS 2014, pp. 3104–3112, 2014.

[7] O. Vinyals, Q. V. Le, “A Neural Conversational Model,” ICML 2015 Deep Learning Wrokshop, 2015.

[8] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, “Distributed representations of words and phrases and their compositionality,” NIPS’13 Proceedings of the 26th International Conference on Neural Information Processing Systems, Vol. 2, 2013.

[9] “MeCab: Yet Another Part-of-Speech and Morphological Analyzer,” http://taku910.github.io/mecab/.

[10] K. Papineni, S. Roukos, T. Ward, W. Zhu, “BLEU: a Method for Automatic Evaluation of Machine Translation,” the 40th Annual Meeting of the Association for Computational Linguistics (ACL), pp. 311–318, 2002.

[11] 酒井 哲也, “情報アクセス評価方法論 検索エンジンの進歩のために,” コロナ社, 2015.

[12] M.T. Luong, H. Pham, C. D. Manning, “Effective Approaches to Attention-based Neural Machine Translation,” Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1412–1421, 2015.

[13] C.W. Liu, R. Lowe, I. V. Serban, M. Noseworthy, L. Charlin, J. Pineau, “How NOT To Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation,” Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 2122–2132, 2016.

[14] Tetsuya Sakai, “Laboratory Experiments in Information Retrieval:Sample Sizes, Effect Sizes, and Statistical Power (to appear),” Springer, 2018.