

# 発話者を考慮した学習に基づく対話システムの検討

河東 宗祐<sup>†</sup> 酒井 哲也<sup>†</sup>

<sup>†</sup> 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: <sup>†</sup>sow@suou.waseda.jp, <sup>††</sup>tetsuyasakai@acm.org

**あらまし** 近年、テキストや音声を入力とした対話システムが身近になっている。対話システムの構築に対してRNN (Recurrent Neural Network) を用いるなど様々なアプローチが研究されている。本論文では、対話データセットの発話者を考慮することで、発話者の感情ベクトルの生成を試みた。発話者を限定し学習したRNN Search モデルのEncoder 部を用いて、感情ベクトルを定義する。データセットとしては映画のキャラクター情報を含んだ台詞のコーパスを用いる。本研究の実験では、感情を表すベクトルの生成として有意な結果は得られなかったが、返答者1人に対する対話データが少ない時、返答者をクラスタリングする必要があるなどの今後の課題が明らかになった。

**キーワード** 対話システム, RNN

## 1 はじめに

近年、テキストや音声を入力として何らかのフィードバックが得られるような対話システムが非常に身近になってきている。例を出すと、Apple のSiri<sup>(注1)</sup> やMicrosoft のりんな<sup>(注2)</sup> などがあげられる。Siri では、ユーザーの音声を入力として会議のスケジュールを設定できたり、りんなでは、チャットにおいて女子高生を模した人工知能が返信を返してくれるというものである。また、Google のInbox<sup>(注3)</sup> というメールサービスにおいて、受信メールに対する返信文をシステムが自動生成するという機能が試みられているが、これもある種の対話システムである。

対話システムが身近になっている要因として、機械学習による人手の労力削減が考えられる。マイクロブログなどの普及により、学習のために用いることができるインターネット上の取得可能な対話データが増加している。

そのような背景の中、機械学習を用いたデータドリブンな対話システムの研究が盛んであるが、今後更に対話システムに求められるものは何であろうか。

### 1.1 対話システムの分類と歴史

Siri やりんなの例を見てもわかるように、実用的なものから、暇を潰せるようなものまで人工知能を用いた対話システムの用途は様々である。分類を考えると、Siri は音声を入力としていて、りんなではテキストを入力としている。その他にも、対話システムにおける入出力にはテキストや音声や画像、更にはロボットのアームの駆動などが考えられる。また、りんなのように対話における明確な目標がないものなどもあり、目標の有無、対話のドメインなどによっても分類することができる。対話システムにおける分類を表1に示す。また、本研究の対話システムの種類は 下線 を引いて示した。

表1 対話システムの分類

入力の型式	<u>テキスト</u> , 音声, 画像, etc
入力の型式	<u>テキスト</u> , 音声, 画像, etc
目標の有無	タスク指向型, <u>非タスク指向型</u>
対話のドメイン	<u>オープン</u> , クローズド

テキストを入出力とするような対話システムの歴史をさかのぼると、ユーザとの複数回の会話を想定した初期の対話システムとしてWeizenbaum のELIZA<sup>[9]</sup> など様々な対話システムが研究されてきた。対話のドメインと達成すべき目標の有無・種類に関する例をあげると、Siri のシステムとしての目標はユーザーの日常をサポートすること全般である。また、りんなのように対話のドメインを限定しないものをオープンドメインといい、目標を持たないようなものを非タスク指向型というが、上にあげたELIZA<sup>[9]</sup> もこれに該当する。

### 1.2 対話システムの構築のためのアプローチ

対話システムの構築のためのアプローチを考える。先に述べたELIZA<sup>[9]</sup> は、システムが従う規則を定めていき対話システムを構築するルールベースな対話システムである。定められた規則に従って返答をする。このような、ルールベースな対話システムをオープンドメインで実用に近づけるには、より多くの規則を追加していく必要がある。一方、機械学習によるデータドリブンなアプローチがある。機械学習とは、十分な量の入力と出力の既存のデータをもとに、システム内部のパラメータを調節することで入力に対して尤もらしい出力をするシステムを構築するものである。対話システムの構築において、機械学習が用いられるようになった理由は、機械学習に用いることができる公開されている対話データが増加しているからだと考えられる。スマートフォンなどの普及によって、マイクロブログなどのサービスのユーザーが増加している。これに伴い、データドリブンなアプローチによる対話システムの構築を試みる研究が多くなされている。テキストを入出力とするような対話システムにおいて、対話データの単語の系列を扱うためRNN

(注1) : <http://www.apple.com/jp/ios/siri/>

(注2) : <http://rinna.jp/rinna/>

(注3) : <https://www.google.com/inbox/>

(注4) (Recurrent Neural Network) を用いた研究が盛んである。Vinyals ら [8], Shang ら [6], Sordoni ら [7] の研究などがある (2.1 節)。

### 1.3 対話システムによる自然な会話

不自然でない会話という意味での精度向上の先に求められるものを考える。現在、非タスク指向型でオープンドメインな対話システムにおいて、定量的な評価は難しくスタンダードな評価方法は確立されていない。非タスク指向型でオープンドメインなものの評価が難しい理由として、意味の異なる複数の返答が自然な返答の1つとして考えられてしまうことがあげられる。

ここで、多様な会話へのアプローチとして、人工知能が個人の性格をシミュレートし対話に多様性を持たせることができれば、より自然な会話を行う対話システムの構築につながるのではないかと考えた。本研究では、テキストを入出力とする対話システムのデータドリブンなアプローチによるシステム構築において、学習のための対話データの発話者に注目することとした。

## 2 関連研究

### 2.1 RNN を用いた対話システム

テキストを入出力とする対話システムの構築において、システムのパラメータの調節に機械学習を用いている研究があるが、近年、多く用いられているモデルに RNN を用いた Encoder-Decoder フレームワークがある。例をあげると、Vinyals らの Neural conversational model [8], Shang らの Neural Responding Machine [6] や Sordoni らの HRED (Hierarchical Recurrent Encoder-Decoder) [7] などがある。

Neural conversational model は、図 1 のように、RNN を用いて単語ベクトルの系列を入力として単語ベクトルの系列を生成するシステムである。

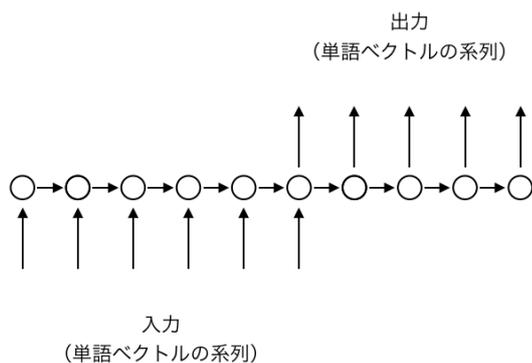


図 1 A neural conversational mode の概観 (Vinyals ら [8] の Fig. 1 をもとに作成)

図 2 のように RNN を 2 つ用いて 1 つを Encoder, もう 1 つを Decoder とし対話システムを構築したものが Neural Responding Machine である。Neural Responding Machine は、

機械翻訳に使われている RNN Encoder-Decoder モデル [2] や RNN Search モデル [1] を対話システムに利用したものである。異なる学習モデルを用いることで、多様な返答をする試みがされている。

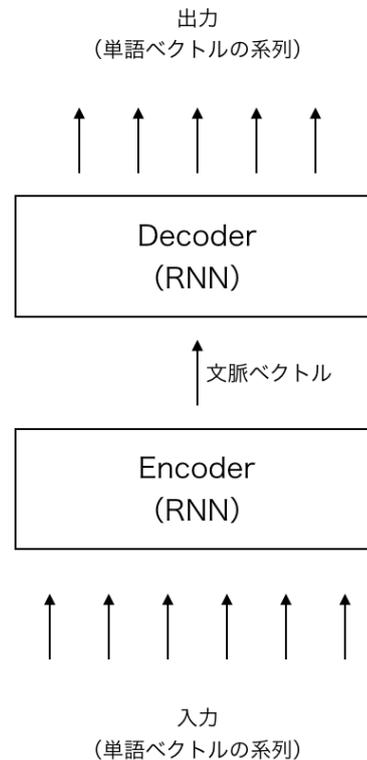


図 2 Neural Responding Machine の概観 (Shang ら [6] の Fig. 1 をもとに作成)

上記の Neural conversational model や Neural Responding Machine は会話の回数が 1 往復であるようなシステムであるが、複数回のテキストの入力に対応しているものが HRED [7] である。図 3 のように入力により Encoder が生成する 1 つ前の文脈ベクトルを、次の入力により生成する文脈ベクトルに出力している。

これらの対話システムにおいて発話者は考慮されていない。

### 2.2 RNN Search モデル

上記のようなデータドリブンなアプローチによる対話システムの構築に、統計的機械翻訳の技術が使われている。機械翻訳のために考えられたモデルであるが、RNN Encoder-Decoder モデル [2] を改良したものである RNN Search モデル [1] について説明する。

Encoder-Decoder フレームワークの Encoder 部の一般式を示す。入力系列  $X = (x_1, \dots, x_{T_x})$  を受け取った時に中間に生成する文脈ベクトル  $c$  を計算する。

$$c = Enc(x_1, \dots, x_{T_x})$$

RNN Encoder-Decoder モデルでは RNN の時刻  $t$  の隠れ層を  $h_t$  とすると、 $Enc(x_1, \dots, x_{T_x}) = h_T$  である。また、RNN

(注4) : Recursive Neural Network の略語としても使われる

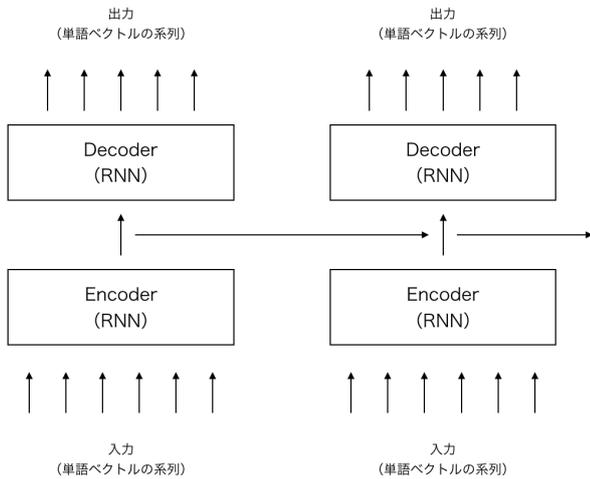


図3 HREDの概観 (Sordoniら [7] の Fig. 3 をもとに作成)

の隠れ層の計算において、LSTM や GRU など単純な RNN を改良したモデルがよく使われる。

RNN Search モデルでは Encoder 部の RNN に Bidirectional RNN を用いている。Bidirectional RNN では隠れ層の計算に入力系列  $X = (x_1, \dots, x_{T_x})$  を通常通り読み込んで計算した隠れ層  $(\vec{h}_1, \dots, \vec{h}_{T_x})$  と逆順に読み込んで計算した隠れ層  $(\vec{h}_1^r, \dots, \vec{h}_{T_x}^r)$  をそれぞれ計算する。そして、以下のように各時刻で通常の隠れ層と逆順の隠れ層をつなぎ合わせることで隠れ層を計算する。

$$h_t = [\vec{h}_t; \vec{h}_t^r]$$

Encoder-Decoder フレームワークの Decoder 部の一般式を示す。出力系列  $Y = (y_1, \dots, y_{T_y})$  の確率は文脈ベクトル  $c$  により次のように決まると考える。

$$p(y_t | \{y_1, \dots, y_{t-1}, X\}) = Dec(y_{t-1}, c)$$

$$p(Y) = \prod_{t=1}^{T_y} p(y_t | \{y_1, \dots, y_{t-1}, c\})$$

RNN Search モデルでは Decoder 部にも RNN を用いているが、隠れ層の計算において出力層の値も用いている。計算式は、Decoder の時刻  $t$  における隠れ層を  $s_t$  とすると

$$s_t = g(s_{t-1}, y_{t-1}, c_t)$$

である。

ここで、RNN Search モデルで取り入れられた attention モデルについて説明する。機械翻訳においては、翻訳前の文と翻訳後の文のどことどこが対応しているかを考え重み付けする。これは、長文における翻訳精度の向上につながっている。Encoder-Decoder フレームワークを用いた学習においても同様に考え、Encoder 部の RNN の隠れ層と Decoder 部の RNN の隠れ層から重みを計算している。Decoder 部の RNN の時刻  $i$ 、隠れ層  $s_i$ 、Encoder 部の RNN の時刻  $j$ 、隠れ層  $h_j$  と

して、Decoder 部の RNN の時刻  $i$  と Encoder 部の RNN の時刻  $j$  との関連性の重み  $\alpha_{ij}$  を以下のように計算する。

$$\alpha_{ij} = \frac{\exp(a_{ij})}{\sum_{k=1}^{T_x} \exp(a_{ik})}$$

$$a_{ij} = v_a^T \tanh(W_a s_{i-1} + W_a h_j)$$

$v_a$ ,  $W_a$ ,  $W_a$  はパラメータである。

上で求めた重み  $\alpha_{ij}$  を用いて、Decoder 部の RNN の時刻  $i$  の隠れ層  $s_i$  の計算に使われる文脈ベクトル  $c_i$  を次式で計算する。Encoder 部の RNN の時刻は  $j$ 。

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j$$

そして、Decoder 部の RNN の隠れ層  $s_i$ 、文脈ベクトル  $c_i$  を用いて時刻  $i$  の出力の確率を計算する。

$$p(y_i | \{y_1, \dots, y_{i-1}, X\}) = h(y_{i-1}, s_i, c_i)$$

### 2.3 NTCIR STC

NTCIR (NII Testbeds and Community for Information access Research)<sup>(注5)</sup> において、STC (Short Text Conversation)<sup>(注6)</sup> のタスクが設けられている [4]。これは、長期的には与えられたツイートに対して人間らしいリプライを機械的に生成し返す人工知能を目指すものだが、その前段階として、過去のツイートを検索して再利用するアプローチの有用性と限界を評価するものである。

## 3 提案と仮説

本研究の最終目的は、発話者をシミュレートする対話システムの構築である。本研究では、テキストを入出力とする対話システムにおいて、学習のための対話データの発話者に注目する。学習モデルとして、RNN を用いた Encoder-Decoder フレームワークである RNN Search モデル<sup>[1]</sup>を用いる。

ここで、その発話者が返答している会話データのみを学習データに使い対話システムを構築する方法が考えられる。しかし、ある返答者に限定した対話データをシステムの学習に十分なだけ獲得するのは難しい。現在、マイクロブログなどの普及により利用可能な会話データは増加しているとはいえ、1人の返答者に限定した会話データは多くはない。

学習の際に、使用する対話データのある特定の返答者、つまり、対話データの出力の正解データを発言している人に限定することで、中間に生成されるベクトル、および、システムの出力は返答者毎にどのように変わるか検証する。学習のための対話データを1人の返答者に限定したシステム間において、類似性のある返答者の組み合わせが生じることを期待した。

(注5) : <http://research.nii.ac.jp/ntcir/>

(注6) : <http://ntcir12.noahlab.com.hk/stc.htm>

### 3.1 感情ベクトル

“文脈ベクトル”と表現されていた中間に生成されるベクトルを学習データのある返答者に限定した場合に、入力に対するある返答者の感情という意味で“感情ベクトル”と呼ぶこととする。学習のイメージを図4示す。全会話対を学習に用いて学習されたシステムの間で生成されるベクトルが“文脈ベクトル”と言われることと比較して、返答者 R のみの会話対を学習に用いたものが返答者 R の入力に対する感情ベクトルである、という意味を込めて打ち消し線を使っている。

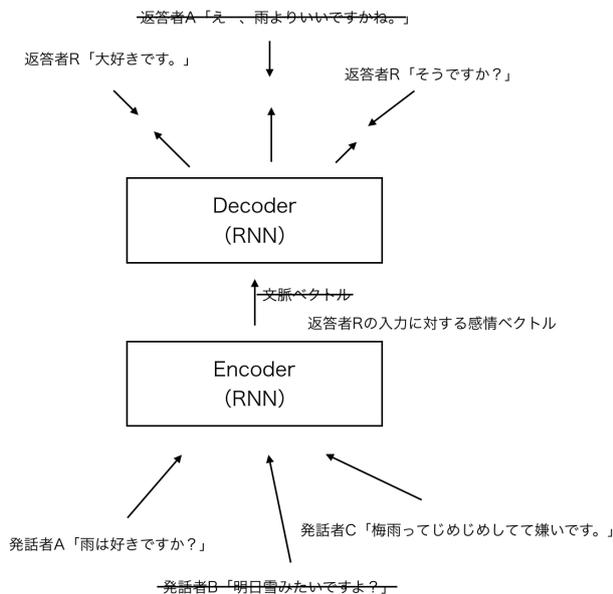


図4 感情ベクトルの学習のイメージ

理想として、3キャラクターに関してそれぞれ学習したシステムによる出力や感情ベクトルを考える。入力、出力を  $X$ ,  $Y$ , 中間生成される感情ベクトルを  $e$  として、3キャラクター毎に色を変えた理想のグラフを図5に示す。赤と青のように感情ベクトルの距離は遠いが出力は近くなっているものや、赤と緑のように感情ベクトルの距離は近いが出力は遠くなっているものなどのキャラクター間で、出力や感情ベクトルに類似性が現れることを期待した。

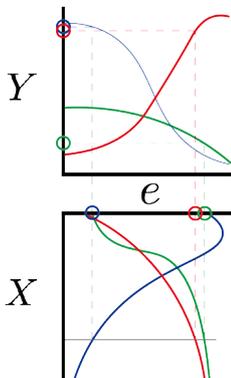


図5 現れて欲しい入力、感情ベクトル、出力に関する関係

返答者 R に限定して学習したシステムの Encoder 部を感情

Encoder として考え、入力に対する返答者 R の感情を表すベクトルを生成しているという仮説を立てた。

学習モデルとして用いた RNN Search モデルにおいて、感情 Encoder の生成する感情ベクトルを考える。返答者 R に注目して学習した RNN Search モデルの Encoder 部に用いた Bidirectional RNN の時刻  $t$  における隠れ層を  $h_t$  とした時、感情ベクトル  $e_R$  は次式で計算することとした。

$$e_R = \frac{\sum_{t=1}^{T_x} h_t}{T_x}$$

## 4 実 験

### 4.1 データセット

本研究は、文脈ベクトルを感情ベクトルと捉え発話者毎の感情 Encoder を学習することで、感情ベクトルの生成を試みるものである。より感情的である会話データとしてフィクションである映画のセリフのデータセットを用いた。データセットの言語は英語である。また、発話者毎の感情 Encoder を学習するため発話者に関するアノテーションが付加されている必要がある。また、本研究で用いる RNN Search モデルでは学習の際に単語系列の長さの上限値を設定している。映画のセリフのデータセットは Movie-Dic, Movie-Triples などがある [5] が、短い会話であり、且つ、キャラクターのメタデータが付加されている Cornell Movie-Dialogs Corpus<sup>[3] (注7)</sup> を用いた。

#### 4.1.1 Cornell Movie-Dialogs Corpus

Cornell Movie-Dialogs Corpus のデータ概要を表2示す。

表2 Cornell Movie-Dialogs Corpus のデータ概要

会話の数	220,579
発話者の数	9,035
単語の数	9M <sup>[5]</sup>

1つの会話データにセリフのリストがあり、奇数番目のセリフと偶数番目のセリフを発したユーザーが定められている。偶数番目のセリフを発したユーザーのユーザー ID を返答者 ID とする。

#### 4.1.2 学習データの作成

学習データとしてある返答者に限定した会話データの対を作成する。返答者があるユーザー (ID:  $r$ ) に注目したとする。Cornell Movie-Dialogs Corpus の会話データにおいて、偶数番目のセリフを話しているユーザーの ID が  $r$  な会話データを抽出し、奇数番目のセリフを入力データ、偶数番目のセリフを出力の正解データとした。

また、事前にセリフを単語に分割するためのスクリプトとして、`tokenize.perl`<sup>(注8)</sup> を使用した。単語分割した後の平均の単語系列の長さは約 13.5 であった。また、Cornell Movie-Dialogs Corpus 内には空行のセリフがあるため、空行は <SILENT> で置換した。

(注7) : [http://www.cs.cornell.edu/~cristian/Cornell\\_Movie-Dialogs\\_Corpus.html](http://www.cs.cornell.edu/~cristian/Cornell_Movie-Dialogs_Corpus.html)

(注8) : <https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl>

### 4.1.3 返答者の選定

RNN Search モデルを用いてパラメータを学習する返答者を4人選んだ。選出方法は上記の方法で抽出される会話データがより多い発話者を4人選んだ。抽出された会話対の数と各キャラクターのメタデータを表3に示す。

表3 選定したキャラクターのメタデータ

会話対の数	ユーザー ID	名前	性別	映画タイトル
391	4460	MASON	m	chill factor (1999)
338	4421	REGGIE	m	charade (1963)
335	8677	JOHN	m	u-turn (1973)
325	1323	ROB	m	high fidelity (2000)

## 4.2 実験方法

### 4.2.1 パラメータ

本研究で用いたパラメータと Neural Responding Machine (NRM)<sup>[6]</sup> で用いられたパラメータを表4に示す。Neural Responding Machine では、入力データセットと出力データセットで別の辞書を用いており、語彙数 40,000 での出力単語系列のカバー率は入力・出力データセットがそれぞれ 97.8%, 96.2% である。本研究では、入力データセット、出力データセット同じ辞書を用い、語彙数 2,500 で全単語系列のカバー率は 90.6% であった。パラメータの学習はバッチ方式の確立的勾配法を用いて行った。

表4 本研究と関連研究のパラメータ

parameter	this study	NRM
vocabulary size (input)	2,500	40,000
vocabulary size (output)	2,500	40,000
embedding dimension (encoder)	180	620
embedding dimension (decoder)	180	620
hidden state dimension (encoder)	250	1,000
hidden state dimension (decoder)	250	1,000

### 4.2.2 キャラクター間の類似度

返答者を限定して学習したそれぞれのシステムにもかかわらず、出力が似るようなキャラクターの組み合わせが存在するのかわ。また、出力が似ていなくても、中間生成されるベクトルが似るようなキャラクターの組み合わせが存在するのかわ。これらを検証するため、各返答者毎に学習したシステムに全データセットの中からランダムに抽出したサンプル 100 個を入力として、キャラクター間の類似度を出力文と感情ベクトルに関して計算した。4キャラクターの組み合わせ 6 通りを比較した。出力文の類似度の計算には Jaccard 係数を用いた。また、感情ベクトルの類似度の計算にはコサイン類似度を用いた。

### 4.2.3 サンプル入力文に対する出力と感情ベクトル

ある返答者に限定して学習したシステムが、その返答者の実際のセリフに対して、他のシステムを比べて特異な感情ベクトルを生成するのか。これを検証するため、サンプル入力として 8 個の文を入力した。これらは各返答者の実際のセリフを 1 人 2 個ずつ恣意的に抽出したものである。各サンプル入力文と実際にその文をセリフとしている返答者を表5に示す。表中の“.”

が前の単語と離れているのは、学習の際に“.”を1つの単語として辞書に登録してしまったことを意味するためそのままとした。“,”や“t”も同様である。

表5 サンプル入力文

サンプル番号	ユーザー ID	文
1	4460	Where do you want this stuff ?
2	4460	Take your gun !
3	4421	What is it ?
4	4421	Oh , you should see your face .
5	8677	Can I help you , sir ?
6	8677	Just banged my head . It was an accident .
7	1323	Why would they care ?
8	1323	I don 't want you talking to our customers like that again .

表5の各文に対して、各返答者の感情ベクトル及び出力文が生成される。文毎の感情ベクトルを主成分分析を用いて2次元に次元圧縮しプロットしたものの図を結果に示す。

## 4.3 結果と考察

### 4.3.1 キャラクター間の類似度

ランダムに 100 個抽出した入力文に対する各キャラクター間毎の出力文の Jaccard 係数のグラフを図6に感情ベクトルのコサイン類似度を図7に示す。4桁の数字はキャラクターのユーザー ID である。

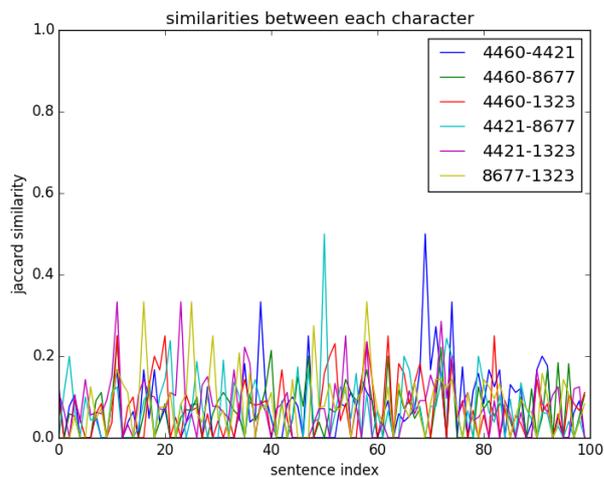


図6 各キャラクター間の出力文の Jaccard 係数

図6において、高い類似度を示すキャラクターの組み合わせは見られなかった。返答者を限定して学習したそれぞれのシステムにもかかわらず、出力が似るようなキャラクターの組み合わせが存在するとは言えない。次に、図7において、出力が似ていなくても、中間生成されるベクトルが似るようなキャラクターの組み合わせが存在するとは言えない。本実験では4キャラクターに対するシステムの学習を行ったが、更に数を増やしてキャラクター毎の比較を試みたい。

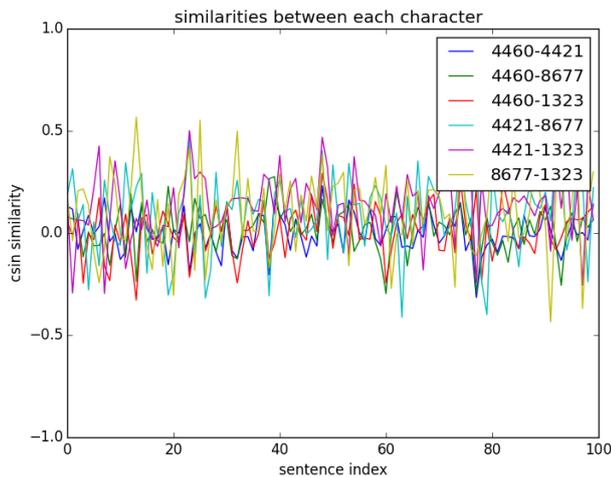


図7 各キャラクター間の感情ベクトルのコサイン類似度

#### 4.3.2 サンプル入力文に対する出力と感情ベクトル

サンプル番号1の入力文において、出力された各返答者の文を表6に、各返答者の感情ベクトルを2次元に次元圧縮プロットした図を図8に示す。同様に他のサンプル番号の入力文においても順に表7、図9、表8、図10、表9、図11、表10、図12、表11、図13、表12、図14、表13、図15に示す。各表においてUNKは辞書にない単語。また、表12のユーザーID1323の出力文は論文掲載に不適切なため非表示とした。

入力に対する各出力文を見ると、キャラクターと実際のセリフの組み合わせ(例えば、サンプル番号1, 2に対するユーザーID4460)は会話として不自然でない。しかし、それら以外は不自然な会話対となっている。400弱の会話対では学習データとして少なく対話システムを構築できないことがわかる。また、中間生成された感情ベクトルにおいても同様に、キャラクターと実際のセリフの組み合わせの感情ベクトルだけが他の感情ベクトルと異なる振る舞いを見せるかと思われたが、図8から図15を見るとそうになっていない。ほとんどの図において各ユーザーが似たような位置関係になっている。ある返答者に限定して学習したシステムが、その返答者の実際のセリフに対して、他のシステムを比べて特異な感情ベクトルを生成するとは言えない。

少し細かく見るために図8から図15を相対的な配置関係に注目すると、図10のみがユーザーID8677とユーザーID4421を結んだ線を境にしてユーザーID4460を反対側にユーザーID1323があることがわかる。サンプル番号3の入力は“What is it?”である。各返答者に限定して抽出したデータセットを見ると“What”で始まり“?”で終わる入力文が散見される。学習データ内の入力文と似た入力文を得ると、多少システム固有な感情ベクトルを生成している可能性がある。

表6 サンプル番号1に対する各キャラクターの出力文  
入力文：Where do you want this stuff ?

ユーザー ID	出力文
4460	UNK in the back .
4421	The man with the Oh . about a man with the same and the field should start down anything anything 't that ?
8677	All I have in the world .
1323	Jesus , Rob . But this As long as you don 't don 't . UNK it .

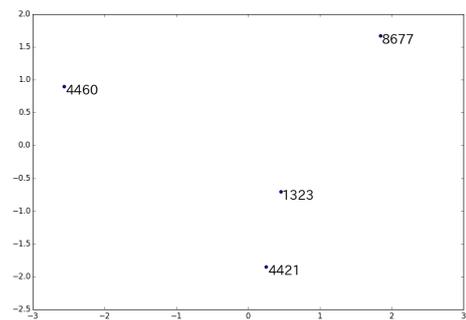


図8 サンプル番号1に対する各キャラクターの感情ベクトル

表7 サンプル番号2に対する各キャラクターの出力文  
入力文：Take your gun !

ユーザー ID	出力文
4460	Doesn 't work .
4421	It 's all right - I don 't think UNK does , either .
8677	I 'm her husband .
1323	I 'm just sick of thinking about it all the time .

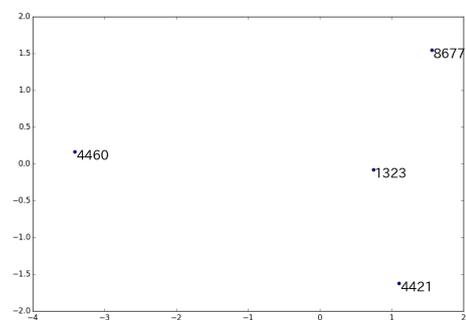


図9 サンプル番号2に対する各キャラクターの感情ベクトル

表 8 サンプル番号 3 に対する各キャラクターの出力文

入力文：What is it ?

ユーザー ID	出力文
4460	Well , UNK usually gives me a signed check for UNK , UNK ' for
4421	Open up .
8677	UNK giving you problems ?
1323	That 's what , you . It 's everybody 's a UNK actor in the

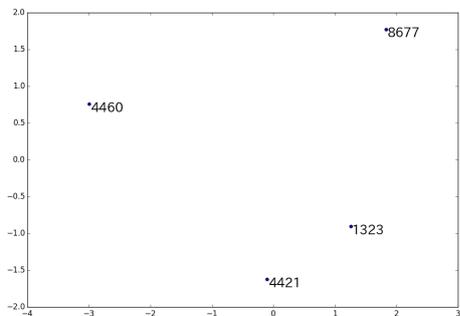


図 10 サンプル番号 3 に対する各キャラクターの感情ベクトル

表 9 サンプル番号 4 に対する各キャラクターの出力文

入力文：Oh , you should see your face .

ユーザー ID	出力文
4460	What ?
4421	What about it ?
8677	I told you . I knew it was there . Didn 't I tell you ?
1323	I 'm just sick of thinking about it all the time .

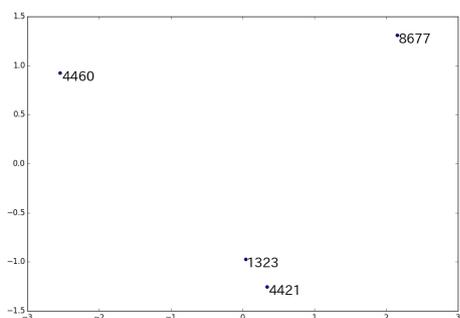


図 11 サンプル番号 4 に対する各キャラクターの感情ベクトル

表 10 サンプル番号 5 に対する各キャラクターの出力文

入力文：Can I help you , sir ?

ユーザー ID	出力文
4460	UNK .
4421	In money .
8677	I need a ticket .
1323	Hello , Rob . Remember me ? I 'm Ray . UNK .

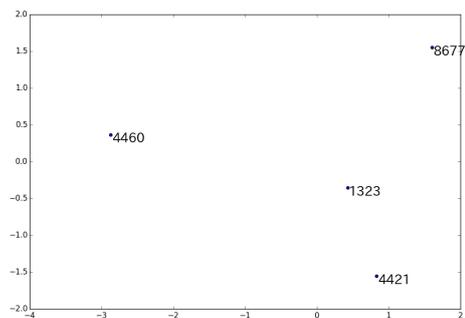


図 12 サンプル番号 5 に対する各キャラクターの感情ベクトル

表 11 サンプル番号 6 に対する各キャラクターの出力文

入力文：Just banged my head . It was an accident .

ユーザー ID	出力文
4460	<SILENT>
4421	Is that all ? Can I go now ?
8677	Another accident ? You got to be more careful .
1323	No way ! I 'd UNK his style . And I wouldn 't want to listen to all that stuff happening on the other side of the bedroom wall . I 'm way

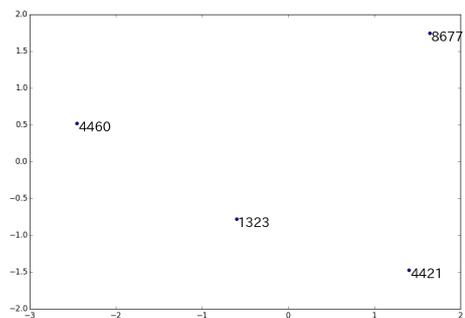


図 13 サンプル番号 6 に対する各キャラクターの感情ベクトル

表 12 サンプル番号 7 に対する各キャラクターの出力文

入力文：Why would they care ?

ユーザー ID	出力文
4460	<SILENT>
4421	Well , the idea 's right , anyway . Are you going to call a UNK ?
8677	We was born to die , boy . From the minute you take your first breath you got
1323	(not shown)

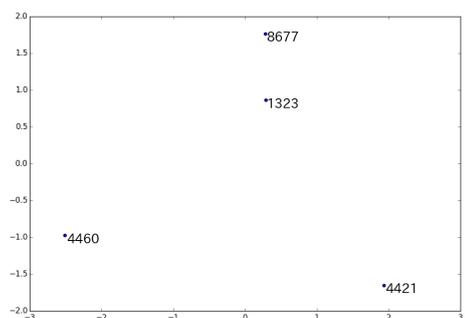


図 14 サンプル番号 7 に対する各キャラクターの感情ベクトル

表 13 サンプル番号 8 に対する各キャラクターの出力文

入力文: I don't want you talking to our customers like that again .

ユーザー ID	出力文
4460	... the master of his UNK .
4421	What do you 're Yes to ever with me . Think , UNK - written means we 're do ?
8677	Are I know you 're full of that .
1323	You don 't ever think about other people ?

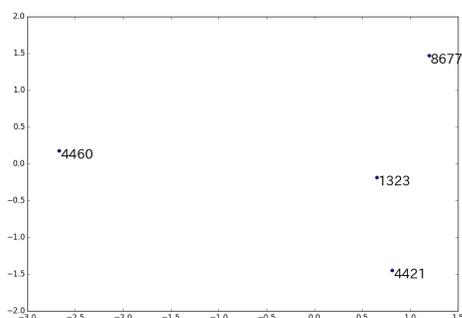


図 15 サンプル番号 8 に対する各キャラクターの感情ベクトル

## 5 結論と課題と展望

結論としては、返答者を限定したデータセットを用いて学習した RNN Search モデルにおいて、出力や Encoder 部を持ちいて生成された感情ベクトルに類似性が生じる返答者の組み合わせがあるとは言えなかった。

今後の課題を以下にまとめる。

学習に関する課題

- 4人だけでなくより多くの返答者毎の学習をして分析する

- データセット内で返答が多いキャラクターを選出したが、全キャラクターデータ数が 400 未満であるので、似たキャラクターをクラスタリングしデータ数を増やす

分析に関する課題

- 次元圧縮だけでなく、生成された感情ベクトルそのもの同士の距離など細かい比較をする

- システム使用時の入力文とどれほど似ている学習データ内の入力文が存在するかによって、中間生成される感情ベクトルがどれほど変わるのか確かめる必要がある。

今後の展望として、発話者に注目することで発話者毎の実用的な感情 Encoder が構築された後、それを用いた対話システムの構築のイメージを図 16 に示す。返答者毎の感情 Encoder を利用して更に 2 段階目の機械学習を試みるものである。

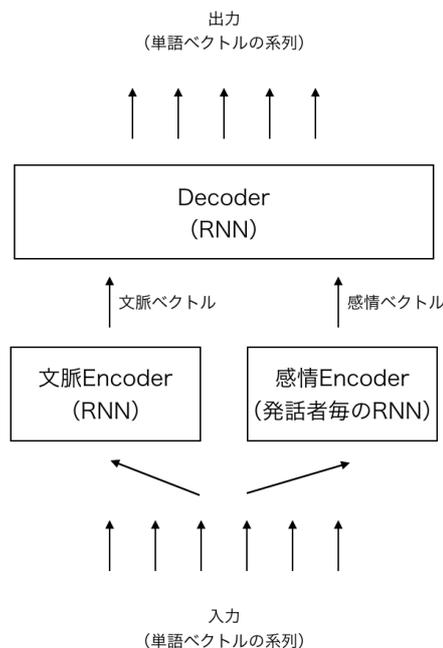


図 16 発話者を考慮した対話システムの概観

## 文 献

- [1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.
- [2] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.
- [3] Cristian Danescu-Niculescu-Mizil and Lillian Lee. Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011*, 2011.
- [4] Tetsuya Sakai, Lifeng Shang, Zhengdong Lu, and Hang Li. Topic Set Size Design with the Evaluation Measures for Short Text Conversation, AIRS 2015, LNCS 9460, pages 319–331, 2015.
- [5] Iulian Vlad Serban, Ryan Lowe, Laurent Charlin, and Joelle Pineau. A survey of available corpora for building data-driven dialogue systems. *arXiv preprint arXiv:1512.05742*, 2015.
- [6] Lifeng Shang, Zhengdong Lu, and Hang Li. Neural responding machine for short-text conversation. *arXiv preprint arXiv:1503.02364*, 2015.
- [7] Alessandro Sordani, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 553–562. ACM, 2015.
- [8] Oriol Vinyals and Quoc Le. A neural conversational model. *arXiv preprint arXiv:1506.05869*, 2015.
- [9] Joseph Weizenbaum. Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45, 1966.