

# 非タスク指向型対話システムにおける 共感誘発型発話手法に関する検討

宅和 晃志<sup>1,a)</sup> 吉川 大弘<sup>2,b)</sup> ジメネス フェリックス<sup>2,c)</sup> 古橋 武<sup>2,d)</sup>

**概要:** 近年、ユーザと自然な雑談を行うことを目指した非タスク指向型対話システムが注目されている。ユーザのシステムへの満足度を高めるためには、自然な対話を行うことに加え、ユーザの興味を惹きつける内容の発話を行うことが有効であると考えられる。そこで本稿では、「共感を得ることで笑いを誘う発話文」として、Twitter より「あるあるネタ」を表すツイートを取得し、発話文として用いる手法を提案する。「あるあるネタ」とは、聴衆の共感を得ることで笑いを誘う演芸における手法の一つである。一往復対話の評価実験の結果、提案手法による発話文が、雑談対話 API および通常のツイートと比較し、最も好まれることが確認できた。

## 1. はじめに

近年、人間と自然な対話を行うことを目指した対話システムが注目されている。NTT ドコモ社の「しゃべってコンシェル」<sup>\*1</sup> や、Softbank 社の「Pepper」<sup>\*2</sup>、Apple 社の「Siri」<sup>\*3</sup> など、実用化されているものも多数存在している。

対話システムは、タスク指向型対話システムと非タスク指向型対話システムの 2 種類に大きく分類される。タスク指向型対話システムは、ユーザの質問や要求などに対して指定された情報を提供するなど、特定のタスクの達成を目的とする対話システムである。例えば、天気案内システム [1] や観光案内システム [2] などが挙げられる。一方、非タスク指向型対話システムは、特定のタスクの達成を目的とせず、ユーザとの日常的な雑談を行うことでユーザを楽しませることを目指す対話システムである。雑談においては、想定される話題の範囲が極端に広いため、実用的なレベルでの発話生成はまだ難しく、現在も盛んに研究が行われている [3-8]。

古くから存在する非タスク指向型対話システムとして、ELIZA[3] や A.L.I.C.E[4] などのルールベース型のシステムがある。しかし、大規模なルールを手手で記述すること

は、時間的、人的コストが極めて大きい。そこで近年、統計的な手法を用いた非タスク指向型対話システムが提案されている [5][6]。上述のルールベース型の手法と統計的な手法は排他的なものではなく、ルールベース型の手法と統計的な手法を状況に応じて使い分けることで、より自然で適切な発話を行うことができるとされている [7]。

以上のように、非タスク指向型対話システムに関して、多様な手法やシステムがこれまでに提案されてきている。しかし、実際に非タスク指向型対話システムが用いられる状況においては、単に雑談を行うだけでは不十分だと考えられる。長期的に使用される状況においては、自然な対話を行うこと以上に、ユーザの興味を惹きつける発話を行うことが重要だと考えられる。実際に、徳久ら [8] は、「非課題遂行対話ではタスクの効率性よりもむしろユーザを飽きさせないように盛り上げながら対話を続けることが求められる」と報告しており、対話の効率性よりも対話の盛り上がりの重要性を指摘している。

本稿では、「共感」に着目する。共感とは、「他者と喜怒哀楽の感情を共有することを指す」(Wikipedia より引用)、「他人の考え・行動に、全くそのとおりだと感ずること。同感。」(大辞林より引用)とされており、人間同士のコミュニケーションにおいて非常に重要な役割を果たしていると考えられる。そこで本稿では、特に後者の意味で「共感を得ることで笑いを誘う発話文」として、Twitter より「あるあるネタ」を表すツイートを取得し、発話文として用いる手法を提案する。「あるあるネタ」とは、「日常生活などで多くの人が経験しているような身の回りの些細なことを挙げたり、観客の共感を得ることで笑いを誘う演芸などの

<sup>1</sup> 名古屋大学 大学院 工学研究科 計算理工学専攻  
<sup>2</sup> 名古屋大学 大学院 工学研究科 情報・通信工学専攻

a) takuwa@cmlpx.cse.nagoya-u.ac.jp

b) yoshikawa@cse.nagoya-u.ac.jp

c) felix@cse.nagoya-u.ac.jp

d) furuhashi@cse.nagoya-u.ac.jp

\*1 [https://www.nttdocomo.co.jp/service/shabette\\_concier/](https://www.nttdocomo.co.jp/service/shabette_concier/)

\*2 <http://www.softbank.jp/robot/consumer/products/>

\*3 <http://www.apple.com/jp/ios/siri/>

手法のひとつ」(Wikipediaより引用)とされている。ユーザにとって共感できる発話を行うことで、ユーザの興味を惹きつけ、より好まれる発話を行うことができると考えられる。

## 2. 関連研究

Twitterより文を取得し、発話文として使用可能な文のみを大量に取得するための手法として、稲葉らの手法[9]がある。この手法では、代名詞や文頭・文末の品詞などの単語ベースでフィルタリングを行ったのち、文中の単語の重要度によりスコア付けを行い、スコアの高い発話のみを取得する。

同様の手法として、Higashinakaらの手法[10]がある。この手法では、稲葉らの手法と類似したルールにより単語ベースでのフィルタリングを行ったのち、更に構文ベースでのフィルタリングを行う。最後に、話題語に関連した発話文を抽出するために、話題語と共起しやすい単語を多く含む発話文のみを取得する。

本稿では、これらの手法を参考にして取得された発話文を「通常発話文」として、提案手法との比較を行う。

## 3. 提案手法

本稿では、現在の対話における「話題語」が取得できていることを前提とする。

### 3.1 あるある発話文の取得

提案手法ではまず、Twitterより「あるあるネタ」を表すツイートを取得する。Twitterの検索APIを用い、ハッシュタグに話題語を付与した「#+話題語+あるある」を含むツイートを取得する。話題語を「名古屋」とした場合には、「#名古屋あるある」を含むツイートが取得されることになる。また、ユーザ名が「話題語+あるある bot」であるアカウントのツイートも全て取得する。このアカウントは、その話題語に関する「あるあるネタ」を定期的にツイートしているbotである。

取得したツイートの前処理として、ハッシュタグの部分を削除しておく。また、ツイート内に改行を含むものはツイートごと除去を行い、残ったツイートを「あるある発話文」として取得する。

### 3.2 発話文の選択

提案手法では、ユーザの発話文に対し、取得されたあるある発話文の中から、適切な発話文を選択することで、対話の流れとして自然にあるある発話文を使用することを目指す。発話文の選択手法としては、稲葉らの手法[11]がある。この手法は、単語や文末表現の遷移などを実際の対話データより学習し、情報検索の分野で用いられるランキング手法を用いることで、ユーザの発話文に対して適切な発

話文を選択する手法である。提案手法では、ユーザの発話(話題)に対し、共感を誘う発話文を生成するため、単語の遷移よりもむしろ、ユーザの発話文に含まれる特徴的な単語に反応することが重要である。

そこで、ユーザの発話文に含まれる内容に適切に反応するための発話文の選択手法を提案する。以下では、初めに選択手法の基礎となるベース手法を説明したのち、4つの工夫点についてそれぞれ説明する。

#### 3.2.1 ベース手法

ユーザの発話文に含まれる内容に適切に反応するための最も単純な方法として、ユーザの発話文に含まれる単語との一致性を利用することが考えられる。ベース手法ではまず、ユーザの発話文に含まれる名詞と用言(形容詞、形容動詞、動詞)を形態素解析により抽出する。抽出された単語の集合を以下の式で表す。

$$\mathbf{W}_{\text{user}} = \{w_1, w_2, \dots\} \quad (1)$$

候補となるあるある発話文も同様に、それぞれの文から名詞と用言を抽出しておく。各あるある発話文  $aruaru_i$  に対する抽出された単語の集合を以下の式で表す。

$$\mathbf{W}_{\text{aruaru}_i} = \{w_1, w_2, \dots\} \quad (2)$$

そして、抽出された単語のうち、共通の単語を最も多く含むあるある発話文を選択する。すなわち、以下の式で表す共通単語数  $n$  が最大となるあるある発話文を選択する。

$$n = |\mathbf{W}_{\text{user}} \cap \mathbf{W}_{\text{aruaru}_i}| \quad (3)$$

$n$ が最大となるあるある発話文が複数ある場合は、文長(文字数)が最も短いものを選択する。これは、文が長ければ長いほど多くの情報を含む文であると考えられ、ユーザの発話に対して話題がずれてしまったり、不自然な応答になる可能性が高いため、発話文として使うには、より(余分な)情報の少ないものを使用するのが適切だと考えるためである。

本稿では、形態素解析器としてMeCab[12]を用いる。また、辞書にはmecab-ipadic-NEologd[13]を使用する。これは、多数のWeb上の言語資源から得た新語を追加することでカスタマイズしたMeCab用のシステム辞書であり、Twitterで使用されるような新語や固有表現に対応できると考えられる。

#### 3.2.2 工夫点1: 名詞(句)&用言の係り受けペアの抽出

提案手法では、名詞と用言を独立に扱わず、初めに名詞のみを  $\mathbf{W}'_{\text{user}}$ ,  $\mathbf{W}'_{\text{aruaru}_i}$  として抽出する。次に用言に関しては、名詞とその係り受け関係にあるペアとして扱う。名詞は形容詞により修飾されたり、動詞の主語や目的語、形容動詞の主語として用いられることが多く、文の主題になりやすい。そのため、係り受け関係にある用言はペアとして扱うことで、より発話文の内容に適切に反応できると

考えられる。

本稿では、係り受け解析器として CaboCha[14] を用いる。CaboCha を用いてユーザの発話文、および各あるある発話文から係り受け関係にある名詞&用言のペアを抽出する。ユーザの発話文、および各あるある発話文  $aruaru_i$  に対して、抽出される名詞&用言のペアの集合を以下の式で表す。

$$\mathbf{P}_{\text{user}} = \{p_1, p_2, \dots\} \quad (4)$$

$$\mathbf{P}_{\text{aruaru}_i} = \{p_1, p_2, \dots\} \quad (5)$$

ここで、用言についてはその原形を解析器より取得して抽出する。更に、単純に係り受け関係にある名詞に関してペアを抽出するのではなく、連続する名詞を名詞句としたペアも抽出する。たとえば、「カラオケオールで眠い」という文から抽出される名詞（句）&用言のペアは以下のようになる。

{カラオケ&眠い, オール&眠い, カラオケオール&眠い}

また、 $\mathbf{W}'_{\text{user}}$ ,  $\mathbf{W}'_{\text{aruaru}_i}$  に関しても同様に、名詞ではなく名詞（句）として抽出する。

### 3.2.3 工夫点 2: 要素の珍しさの考慮

提案手法では、ユーザの発話文とあるある発話文の共通の要素の数、つまり

$$n = |\mathbf{W}'_{\text{user}} \cap \mathbf{W}'_{\text{aruaru}_i}| + |\mathbf{P}_{\text{user}} \cap \mathbf{P}_{\text{aruaru}_i}| \quad (6)$$

によって選択を行うのではなく、各要素の珍しさをスコアとして考慮する。

ユーザの発話文に複数の名詞（句）&用言のペアが含まれる際には、より珍しいペアとの共通性を考慮することが適切だと考えられる。例えば、「歌が下手だけどカラオケ行きたい」というユーザの発話文からは、以下のペアが抽出される。

$$\mathbf{P}_{\text{user}} = \{ \text{歌\&下手}, \text{カラオケ\&行く} \} \quad (7)$$

ここで、両方のペアを要素に持つあるある発話文があればそれが最も適切ではあるが、どちらか一方のペアのみを持つあるある発話文しかない場合、より珍しいペアである「歌&下手」を要素に持つ発話文の方が、よりユーザの発話文の主題に反応できていると考えられる。

そこで、大規模なコーパスにおける各ペア  $p_i$  の頻度を  $N(p_i)$  として、以下の式で表されるスコアを定義する。

$$s(p_i) = \log \frac{\sum_j N(p_j)}{N(p_i)} \quad (8)$$

珍しいペア、つまり大規模コーパスにおいて出現頻度の少ないペアほど大きな値となる。この式は、単語の重み付けの手法としてよく使われる TF-IDF[15] における IDF 値を

参考に定義した。また、同様に各名詞（句）についても、名詞（句）の頻度を  $N(w'_i)$  として、以下の式でスコアを定義する。

$$s(w'_i) = \log \frac{\sum_j N(w'_j)}{N(w'_i)} \quad (9)$$

なお本稿では、約一年分（2016/5/18~2017/7/10）の Twitter データを大規模コーパスとして利用した。

ユーザの発話文とあるある発話文の共通する名詞（句）とペアそれぞれについてこのスコアを算出し、その和を各あるある発話文のスコアとする。つまり、各あるある発話文  $aruaru_i$  のスコアは以下の式で計算される。

$$s_{\text{aruaru}_i} = \sum_{w' \in \mathbf{W}'_{\text{user}} \cap \mathbf{W}'_{\text{aruaru}_i}} s(w') + \sum_{p \in \mathbf{P}_{\text{user}} \cap \mathbf{P}_{\text{aruaru}_i}} s(p) \quad (10)$$

提案手法では、このスコアが最も高いあるある発話文を選択する。同じスコアの発話文が複数ある場合は、文長が最も短いものを選択する。ただし、 $\mathbf{W}'_{\text{user}}$  と  $\mathbf{W}'_{\text{aruaru}_i}$  については、現在の話題語である名詞（句）をその集合から削除しておく。これは、現在の話題語が既にわかっており、さらにあるある発話文はその話題語をキーに取得されたものであるから、スコアとして加算する必要はないと考えたためである。

### 3.2.4 工夫点 3: 逸脱度合いの考慮

提案手法では更に、ユーザの発話文に対するあるある発話文の逸脱の程度を考慮する。ここでは、ユーザの発話文に含まれる内容に対し、各あるある発話文がどれだけ新しい内容を与えているかを逸脱の程度とし、各あるある発話文が持つ新しい名詞（句）および名詞（句）&用言のペアのスコアにより定量化する。以下に、スコアの逸脱度合いを算出する式を示す。

$$\bar{s}_{\text{aruaru}_i} = \sum_{w' \in \mathbf{W}'_{\text{aruaru}_i} \setminus \mathbf{W}'_{\text{user}}} s(w') + \sum_{p \in \mathbf{P}_{\text{aruaru}_i} \setminus \mathbf{P}_{\text{user}}} s(p) \quad (11)$$

ここで、 $\mathbf{A} \setminus \mathbf{B}$  は集合  $\mathbf{A}$  から集合  $\mathbf{B}$  を引いた差集合を表している。

逸脱度が大きなあるある発話文は、ユーザの発話文に対して新たな内容を多く含んでいることを意味する。逸脱度が過剰に大きければ、ユーザの発話文の内容から大きく逸脱した内容を含んでいると考えられるため、逸脱度がある閾値を超えたあるある発話文は、選択の候補より除外する。本稿では、この閾値を 100 とした。

また反対に、この逸脱度がゼロである発話文は、ユーザの発話文に対し、新しい内容を全く含まないことを意味する。つまり、ユーザの発話文に対してオウム返しになってしまうと考えられるため、選択の候補より除外する。

### 3.2.5 工夫点 4: あるある発話文の候補がない場合の対処

ここまで説明した提案手法では、ユーザの発話文に対し、応答となるあるある発話文の候補が(複数)取得されていることを前提としている。しかし実際には、ユーザの発話文より抽出される名詞(句)および名詞(句)&用言のペアと同じものを全く含まないあるある発話文、すなわち、 $s_{aruaru_i}$  がゼロとなるあるある発話しか取得できていない場合がある。

このような場合には、ユーザの発話文をその話題語から最も連想しやすい用言を伴った話題語&用言のペアから成る発話文に置き換えて選択を行う。例えば、「カラオケ」を話題語とするユーザの発話文に対して、発話文と同じ名詞(「カラオケ」以外)や同じペアを持つあるある発話文の候補がない場合、「カラオケ」という話題語から最も連想しやすい用言を取得する。取得する方法として、大規模コーパスにおける係り受け関係にある話題語&用言のペアの頻度をカウントし、最もペアとなる回数が多い用言を用いる。「カラオケ」という話題語に対し、「行く」が最もペアとなる回数が多い用言であったとすると、ユーザの発話文を「カラオケ行く」と変換して扱うことで、本節で述べた工夫点に基づき、あるある発話文を選択する。

この方法を用いることで、候補となるあるある発話文がない場合でも、その話題語から連想するのに最も妥当な(無難な)あるある発話文を選択することができる。

## 4. 実験

通常発話文と比べたあるある発話文の良さ、および提案手法によるあるある発話文の選択手法の有効性を確認するため、実験を行った。また、比較のための対話システムとして、NTTドコモの雑談対話API<sup>\*4</sup>を用いた。

ある話題語に関するユーザの発話文に対して、各手法によるシステムの発話文を取得することで、一往復の対話を100個作成した。これらの一往復の対話を評価者に見てもらい、応答として好むかどうかの点数付けをしてもらった。詳しい実験方法を以下に説明する。

### 4.1 実験方法

#### 4.1.1 話題語の設定

今回の実験では、ユーザが自らの趣味に関する発話を行う状況を想定して、話題語を趣味として設定した。Wikipedia全文より、「趣味は○○」、「○○が趣味」などのパターンによりマッチングを行い、趣味を取得した(例えば芸能人を解説した記事からその人の趣味が取得される)のち、最も登場回数が多い100個の趣味を取得した。

#### 4.1.2 あるある発話文の取得

各話題語について、Twitterよりあるある発話文を取得

した。ハッシュタグを用いて検索した期間は、2012年～2016年の五年間とした。

また、同じ話題語として扱えるものは、よりあるある発話文数の多い趣味へと統合した。例えば、「映画鑑賞」、「映画観賞」、「映画」の3つの話題語に関するあるある発話文は、「映画」に関するあるある発話文として統合した。

#### 4.1.3 入力発話文の選択

あるある発話文の取得数が多い上位10個の話題語(カラオケ、野球、サッカー、水泳、ゲーム、麻雀、バイク、ギター、テニス、アニメ)に関する通常発話文を入力発話文として取得した。具体的には、各話題語を含む通常発話文を約一年分(2016/5/18～2017/7/10)のTwitterデータより取得し、以下の基準を全て満たす発話文からランダムに選ばれた10文ずつを、入力文(ユーザ発話文)として取得した。基準を満たすか否かの判定は、著者以外の一名にお願いした。

- その話題語に関することを言っている
- 単文で意味がわかる
- Twitter用語を含まない
- 含まれる話題語がその人にとっての趣味だとみなせる
- 他の話題語が含まれる場合は、その話題語が文のメインとなっていない
- 既に取得されている文と似ていない文(入力文に多様性を持たせるため)
- 疑問文でない

#### 4.1.4 システム発話文の生成

提案手法の有効性の検証を行うため、表1に示す4つの手法を設定した。

表1 設定した手法

手法	使用する発話文	発話文の選択手法
手法1	NTTドコモの雑談対話APIの応答をそのまま使用	
手法2	通常発話文	ベース手法
手法3	あるある発話文	ベース手法
手法4(提案手法)	あるある発話文	工夫を加えた手法

手法1と手法4を比較することで従来手法と提案手法の違いを、手法2と手法3を比較することで通常発話文とあるある発話文を用いることの違いを、手法3と手法4を比較することで3.2.1で示したベース手法と提案手法による選択方法の違いを確認することを目的としている。

#### 4.1.5 評価者による評価

入力文(ユーザ発話文)と出力文(システム発話文)の一往復の対話に対し、評価者による評価を行った。評価者として、著者以外の男子大学生10名にお願いした。各評価者は、入力文に対する4つの手法の出力文を見て、各出力文に点数付けを行った。点数付けの基準は、各評価者の感性に基づき、好むか好まないかを-3～+3点で評価した。

<sup>\*4</sup> [https://dev.smt.docomo.ne.jp/?p=docs.api.page&api\\_name=dialogue&p\\_name=api\\_usage\\_scenario](https://dev.smt.docomo.ne.jp/?p=docs.api.page&api_name=dialogue&p_name=api_usage_scenario)

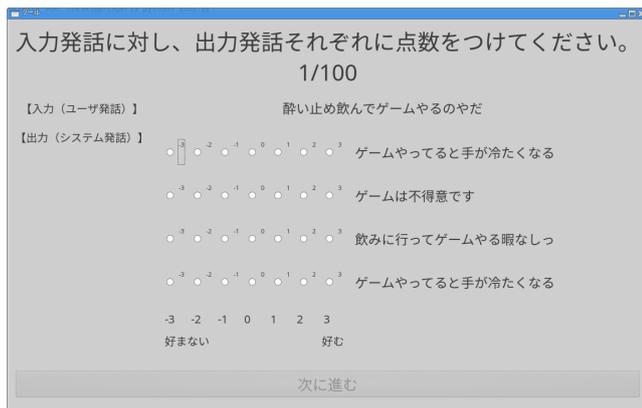


図 1 評価用ツール

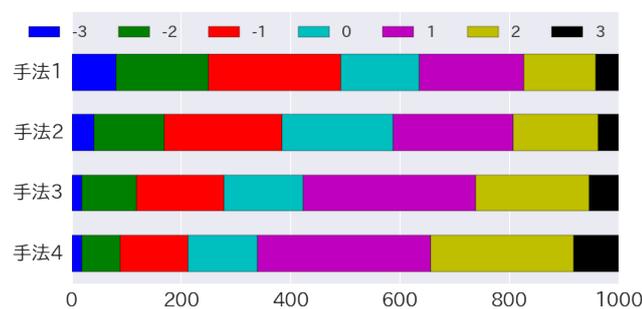


図 2 各手法の点数内訳

評価時に使用した評価用ツールの画面を図 1 に示す。入力文の提示順、および各入力文に対する出力文の順番は、評価者・入力文ごとにランダムに提示されるようになっており、評価者は各出力文がどの手法により生成されたものであるかはわからない。

#### 4.2 実験結果

1000 個の評価（10 名の評価者の 100 個ずつの評価）の点数内訳を図 2 に示す。

図 2 より、手法 1 から手法 4 にかけて、低い点数の割合が減り、高い点数の割合が増えており、提案手法である手法 4 では、7 割弱の出力文で“好む”と評価されていることがわかる。また、各評価者が各手法に与えた点数の平均値を図 3 に示す。

図 3 より、どの評価者も手法 1 から手法 4 にかけて高い点数をつける傾向にあることがわかる。全評価者の平均点をみると、手法 1 から手法 4 にかけてそれぞれ 0.2~0.3 点の差があることがわかる。

好まれる手法に差があるかの検定を符号検定により行った。4 群の比較を行うため、ボンフェローニの方法により多重比較を考慮し、 $\alpha' = 0.0083 = 0.05/6 (= {}_4C_2)$  を有意水準とした。検定の結果、全ての手法間で有意差が確認された。これにより、従来手法である API よりも提案手法が有意に好まれたこと、通常発話文よりもあるある発話文が有意に好まれたこと、ベース手法よりも提案手法により工

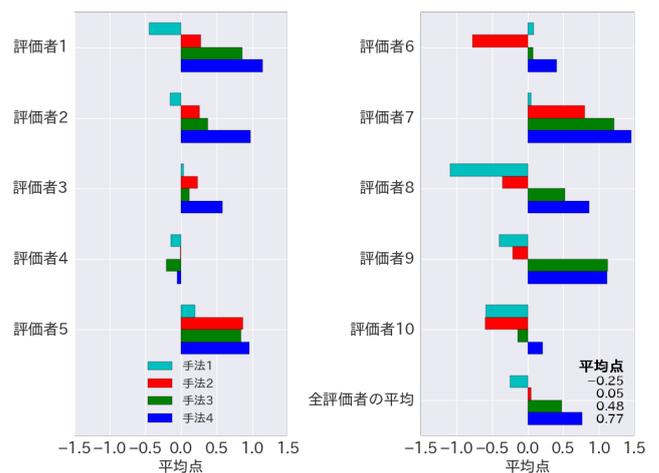


図 3 評価者ごとの平均点

夫を加えた選択手法が有意に好まれたことが確認できた。

### 5. アンケート結果

実験の直後に、評価者にアンケートを実施し、「あなたにとって、好む/好まないの評価基準になったものは何ですか?」という質問に対して自由記述で回答をもらった。記述結果を著者の判断で類型分けをして整理した結果を表 2 に示す。

表 2 アンケート結果の類型

類型	言及した人数	実際の記述
会話としてのつながりの自然さ	9 人	「会話の流れの自然さ」、「意味がわかるもの」など
面白さ、笑えるか	8 人	「返しの面白さ」、「ふふっと笑ってしまったもの」など
共感できるか	3 人	「共感できるもの」など
有益か	2 人	「有益さ」、「単なる繰り返しは×」

表 2 より、「会話としてのつながりの自然さ」および「面白さ、笑えるか」が最も大きな評価要因であったことがわかる。

雑談対話 API では、入力文とは全く関係ない発話を行うことが多かった。これは、適切な応答が困難だと判断された際の、新たな話題へと転換するための工夫だと考えられるが、他の手法と比較すると不自然なつながりとなっていたと考えられる。また、雑談対話 API は、特に面白い発話や笑える発話を目指しているわけではないため、その点においても他の手法と比較して低い評価となったと考えられる。

通常発話文では、Twitter から取得した文であることで、「今日はカラオケ行ってきた」などの発言者の近況報告を表す発話文の割合が多く、会話としてのつながりとしては不自然になりやすい傾向にあったと考えられる。一方、

Twitterにおいてはウケを狙ったツイートをする人が一定数いるため、それらの文が選択された際に、面白さの点においては雑談対話 API より好まれる発話を行っていたと考えられる。

あるある発話文では、「共感を得ることで笑いを誘う発話文」としてツイートされたあるあるネタが多いため、通常発話文と比較して、より多様な入力文に対して自然なつながりを維持できたと考えられる。また、笑いを誘うことを狙ってツイートされたものが多いため、面白さの点においてもより好まれる発話文であったと考えられる。更に、選択手法に工夫を加えた提案手法では、ユーザの発話文の内容に応じて適切に反応できたために、ベース手法による選択方法と比較して高い評価を得られたと考えられる。

## 6. おわりに

本稿では、ユーザの興味を惹くための発話文として「あるあるネタ」を表すツイートを Twitter より取得し、システム発話として使用する手法を提案した。提案手法ではまず、Twitter より「あるあるネタ」として発言されているツイートを取得する。次に、ユーザの発話文に含まれる名詞（句）および名詞（句）&用言の係り受けペアの珍しさを考慮した上で、出力する発話文を選択する。また、ユーザの発話文に対しての逸脱の度合いを定義し、逸脱度の大きなものやゼロとなるものは選択肢から除外する。更に、あるある発話文の候補が存在しない場合には、ユーザの発話文をその話題語から連想しやすい用言に変換して出力発話文の選択を行う。

実験の結果、あるある発話文が通常発話文と比較して有意に好まれること、提案する選択手法がベース手法と比較して有意に好まれること、提案手法が従来手法である雑談対話 API と比較して有意に好まれることをそれぞれ確認した。

あるある発話文を用いることの問題点として、話題語によっては、十分な数のあるある発話文を取得することができないことが挙げられる。そこで今後、通常発話文の中にあるあるある発話文として使用できる発話文を取得する方法について検討する。また、提案手法はあくまで、共感による面白さにより、ユーザの興味を惹きつけ、より好まれる発話を行うための方法であり、一般的な繰り返し（複数往復）の対話には対応できない。そのため、雑談対話 API と組み合わせることで、通常の対話を行いつつ、適宜提案手法を用いるシステムを構築する予定である。

謝辞 本研究は、科学研究費補助金（基盤研究（B）, No.16H02889）の助成を受けたものです。

## 参考文献

- [1] Zue, V., Seneff, S., Glass, J. R., Polifroni, J., Pao, C., Hazen, T. J. and Hetherington, L.: JUPITER: a telephone-based conversational interface for weather information, *IEEE Transactions on speech and audio processing*, Vol. 8, No. 1, pp. 85–96 (2000).
- [2] 翠 輝久, 河原達也, 正司哲朗, 美濃導彦ほか: 質問応答・情報推薦機能を備えた音声による情報案内システム, 情報処理学会論文誌, Vol. 48, No. 12, pp. 3602–3611 (2007).
- [3] Weizenbaum, J.: ELIZA—a computer program for the study of natural language communication between man and machine, *Communications of the ACM*, Vol. 9, No. 1, pp. 36–45 (1966).
- [4] Wallace, R. S.: The anatomy of ALICE, *Parsing the Turing Test*, pp. 181–210 (2009).
- [5] Ritter, A., Cherry, C. and Dolan, W. B.: Data-driven response generation in social media, *Proceedings of the conference on empirical methods in natural language processing*, Association for Computational Linguistics, pp. 583–593 (2011).
- [6] Vinyals, O. and Le, Q.: A neural conversational model, *arXiv preprint arXiv:1506.05869* (2015).
- [7] 目黒豊美, 杉山弘晃, 東中竜一郎, 南 泰浩: ルールベース発話生成と統計的発話生成の融合に基づく対話システムの構築, 人工知能学会全国大会論文集, Vol. 28, pp. 1–4 (2014).
- [8] 徳久良子, 寺島立太: 非課題遂行対話に関する研究動向—雑談システムへの接近, 言語・音声理解と対話処理研究会, Vol. 46, pp. 15–20 (2006).
- [9] 稲葉通将, 神園彩香, 高橋健一: Twitter を用いた非タスク指向型対話システムのための発話候補文獲得, 人工知能学会論文誌, Vol. 29, No. 1, pp. 21–31 (2014).
- [10] Higashinaka, R., Kobayashi, N., Hirano, T., Miyazaki, C., Meguro, Toyomi and Makino, T. and Matsuo, Y.: Syntactic filtering and content-based retrieval of twitter sentences for the generation of system utterances in dialogue systems, *Situated Dialog in Speech-Based Human-Computer Interaction*, Springer, pp. 15–26 (2016).
- [11] 稲葉通将, 平井尚樹, 鳥海不二夫, 石井健一郎: 非タスク指向型対話エージェントのための統計的応答手法, 電子情報通信学会論文誌 D, Vol. 95, No. 6, pp. 1390–1400 (2012).
- [12] Kudo, T.: Mecab: Yet another part-of-speech and morphological analyzer (2005).
- [13] Toshinori, S.: Neologism dictionary based on the language resources on the Web for Mecab (2015).
- [14] 工藤 拓, 松本裕治: チャンキングの段階適用による日本語係り受け解析, 情報処理学会論文誌, Vol. 43, No. 6, pp. 1834–1842 (2002).
- [15] Sparck Jones, K.: A statistical interpretation of term specificity and its application in retrieval, *Journal of documentation*, Vol. 28, No. 1, pp. 11–21 (1972).