任意の話題を持つユーザ発話に対する 係り受けと用例を利用した応答文の生成

Leveraging Dependency Relations and Sentence Examples in Web-scale Corpus for Open-domain Utterance Generation

杉山 弘晃
Hiroaki Sugiyama

NTT コミュニケーション科学基礎研究所
NTT Communication Science Laboratories
sugiyama.hiroaki@lab.ntt.co.jp, <http://www.kecl.ntt.co.jp/icl/ce/member/sugiyama/>

目黒 豊美
Toyomi Meguro

(同 上)
meguro.toyomi@lab.ntt.co.jp

東中 竜一郎
Ryuichiro Higashinaka

NTT メディアインテリジェンス研究所
NTT Media Intelligence Laboratories
higashinaka.ryuichiro@lab.ntt.co.jp

南 泰浩
Yasuhiro Minami

NTT コミュニケーション科学基礎研究所*1
NTT Communication Science Laboratories
minami.yasuhiro@lab.ntt.co.jp

keywords: utterance generation, conversation, microblogs

Summary

The development of open-domain conversational systems is difficult since user utterances are too flexible for such systems to respond properly. To address this flexibility, previous research on conversational systems has selected system utterances from web articles based on word-level similarity with user utterances; however, the generated utterances, which originally appeared in different contexts from the conversation, are likely to contain irrelevant information with respect to the input user utterance. To leverage the variety of web corpus in order to respond to the flexibility and suppress the irrelevant information simultaneously, we propose an approach that generates system utterances with two strongly related phrase pairs: one that composes the user utterance and another that has a dependency relation to the former. By retrieving the latter one from the web, our approach can generate system utterances that are related to the topics of user utterances. We examined the effectiveness of our approach with following two experiments. The first experiment, which examined the appropriateness of response utterances, showed that our proposed approach significantly outperformed other retrieval and rule-based approaches. The second one was a chat experiment with people, which showed that our approach demonstrated almost equal performance to a rule-based approach and outperformed other retrieval-based approaches.

1. は じ め に

対話システムを実現する上で、ユーザと雑談する機能は、エンタテインメントやカウンセリング目的のみならず、タスクの達成においても重要な要素である。Takeuchiらは、タスク指向型対話においても、ユーザは対話中にタスクとは関係のない雑談的な発話をすることがあると報告している [Takeuchi 07]。また、Bickmoreらは、タスク中にユーザと雑談をすることが、より信頼の置ける対話システムの構築に有用であると指摘している [Bickmore 01]。すなわち、スムーズにタスクを達成したり、使い勝手の良い対話システムを実現するためには、対話システムはこうした雑談型発話を適切に扱える必要がある。

こうした背景もあり、近年、雑談を行う対話システム

の研究が増加している [Wallace 04, Shibata 09, Ritter 11, Wong 12, Meguro 10]。従来、雑談対話システムを構築する場合、ユーザの発話に含まれる単語をトリガーとしてあらかじめ記述しておいたシステム発話を入力する、ルールベースのアプローチを用いる場合が多い。代表的なルールベース対話システムとして A.L.I.C.E. [Wallace 04] などが知られており、雑談対話システムの国際的なコンテストであるローブナー賞*2において優秀な成績を収めている。しかし、ルールベースの欠点として、対応可能な話題の拡充が難しいことがあげられる。雑談対話に現れる話題は極めて多様であり、かつ日々増えていくという特徴を持つ。このようなユーザ発話に対して、適切なシステム発話を入力するルールを手で構築し続けるには、莫大なコストが必要となる。

*1 現所属は電気通信大学。

*2 <http://www.loebner.net/Prize/loebner-prize.html>

システムが対応可能な話題を自動的に拡充する試みとして、マイクロブログや Web の文章からユーザ発話に類似した文を検索し、システム発話として出力する、検索ベースのアプローチが提案されている [Shibata 09, Ritter 11]。Web では利用者が非常に多くの文を投稿し続けているため、多様かつ新しい話題を持つユーザ発話に対しても、このアプローチはユーザ発話に関連した内容のシステム発話を出力することができる。また、検索によって得られる文は実際に人が記述したものであるため、文法的な誤りが入り込みにくく、かつ発話のバリエーションが豊富であるという利点がある。これらの検索ベースのアプローチが持つ特徴は、雑談対話システムを構築する上で有用である。しかし、検索によって得られる文は、書かれた状況に応じた固有の文脈を持つため、内容の理解に必要な情報が省略されていたり、不要な情報が含まれている場合がある。これらをそのままシステム発話として出力すると、ユーザにとって理解困難な発話となる問題がある。

本研究では、Web コーパスの持つ話題の豊富さと不要な情報の抑制を両立するため、ユーザ発話に含まれる話題と、Web コーパスから検索したこれに関連する別の話題を組み合わせ、発話を生成する合成ベースの発話生成手法を提案する。ユーザ発話から話題を表す部分を抜き出し、この部分のみを用いて関連する話題を検索することで、ユーザ発話に無関係な情報の混入を抑制しつつ、Web コーパスの話題の豊富さを利用したシステム発話の生成が可能となる。本研究では、話題を表す表現として、係り受け関係を持つ任意のタイプの文節ペア（係り受けペアと呼ぶ）を用いる。係り受けペアには、主辞の品詞が名詞と述語のペアだけでなく、述語と述語ペア、名詞と名詞ペアなども含まれる。話題の表現に係り受けペアを用いる理由は、「お腹が」の 1 文節では「お腹が空いた」と「お腹が痛い」を区別できないように、単語や文節単体ではユーザ発話に含まれる話題を捉えきれない場合があるためである。係り受けペアで表すことで、よりユーザ発話の話題を正確に捉え、無関係な情報の混入を抑える効果が期待できる。また、2 つの係り受けペア間の関連性の指標として、各係り受けペアに含まれる文節間の係り受け関係の有無を用いる。係り受け関係は直接的な関連性を表すため、単純な共起関係で関連性を定義するよりも、ユーザ発話に関連する情報が検索されやすくなる。

提案するアプローチは、特定の話題に特化した知識に依らず、どのような話題を持つユーザ発話に対しても、無関係な話題を抑制しつつ、関連する話題を持つシステム発話を生成できる。本稿では、提案手法の有用性について、一問一答としての発話の適切さの側面と、対話システムへ組み込んだ場合の、文脈における発話の適切さの側面から検証する。

2. 関 連 研 究

Web コーパスからユーザ発話に類似した文を検索しシステム発話とする手法として、Shibata らは、Web からキーワード検索によって映画に関する文を収集しコーパスとする手法を提案している [Shibata 09]。特定の話題に特化したコーパスの構築が容易であり、それゆえ比較的深い話題にも対応できる一方、Web から検索された文は、対話中の 1 発話としては長すぎたり、発話として不自然な場合がある。

Web コーパスの構築に用いられることが多いブログや Wikipedia に比べ、より対話的な短い文を多数含む Web サービスに、Twitter がある。Twitter では、ユーザによって投稿文（ツイート）が多数投稿され続けており、返信機能（ここでは in-reply-to 機能と呼ぶ）を利用した対話的なやりとりも多数含まれる。Twitter から検索された文をシステム発話として利用する手法として、例えば稲葉らの手法がある [稲葉 14]。これは、与えられた話題語を含む文を Twitter から検索し、話題語に関する発話としての適切さを基準として文をリランキングすることで、話題語に適切に関連するシステム発話を得る手法である。彼らは、得られたシステム発話の 96% が、話題語に関連する適切な発話であったと報告している。また Higashinaka らは、稲葉らの手法を発展させ、与えられた話題語に対してシステム発話が含まべき話題語を別途推定しておき、これを用いて文を検索するとともに、機械学習を用いて不適切な文をフィルタする方法を提案している [Higashinaka 14]。これらの手法から得られる発話は、話題語のみが与えられた場合の発話としては適切なものの、ユーザ発話への応答に応用するには不十分である。その理由は、ユーザ発話には話題語以外の情報（話題に関する意見やその話題から展開する別の話題など）が含まれており、話題語のみから得られた発話は、これらの情報と合致しない情報を含む場合があるためである。さらに「お腹」1 単語では「お腹空いた」と「お腹痛い」を区別できないように、単語のみでは必ずしもユーザ発話の話題を捉えきれないという問題もある。

一方、Twitter の in-reply-to 機能を利用するアプローチとして、Ritter らは IR-status, IR-response, MT-chat の 3 つの手法を提案している [Ritter 11]。これらのアプローチでは、Twitter 中の in-reply-to 機能で対応付けられたツイートペアを、元ツイートを status、返信ツイートを response とする発話ペアとして利用する。このうち IR-status は、ユーザ発話と類似した status を検索し、この status に対応する response を、システム発話として出力するアプローチである。これは Twitter の in-reply-to 機能をシンプルに活かすアプローチであるものの、類似する status が見つからない場合や、status と response の関係が Twitter 上で観測できない文脈（実世界でのやりとりなど）を前提としている場合には、全く関係のない、

図1 発話生成の概要

ユーザにとって理解困難な response を、システム発話として出力してしまうという問題がある。この問題に対し、Bessho らは、ある IR-status ベースの対話システムが同時に複数のユーザと 1 対 1 で対話している状況を想定し、コーパス中の文との類似度が低いユーザ発話が入力された場合、その発話を他のユーザへのシステム発話として出力し、ユーザに適切な応答を生成させて元ユーザへのシステム発話へ流用する手法を提案している [Bessho 12]。IR-status の欠点を補う効果が期待できるものの、そのユーザ発話がどの程度他のユーザとの対話の文脈に合致するかは検証されていない。

IR-response は、status の代わりに response からユーザ発話との類似文を検索し、そのままシステム発話として出力するアプローチである。ユーザ発話と類似した response をシステム発話として出力するため、ユーザ発話と関係のない発話は出力されにくくなる。しかしながら、類似度が高すぎる文がコーパスに含まれていた場合、単なる入力文の繰り返しがシステム発話として出力されるという問題がある。MT-chat は、status と response の発話ペアをバイリンガルコーパスとみなして、機械翻訳の手法を用いてシステム発話を生成するアプローチである。IR-status のスパース性を緩和したものともみなせるが、代わりに IR-status の利点であった、実際のツイートをを用いることによる文法の適切さの保証がなくなり、文法的に不適切な文が出力されやすくなる点が問題である。

Ritter らは、MT-chat、IR-response、IR-status の順で、ユーザに好まれる理解しやすい発話を多く出力していたと報告している。多くの文法誤りを含むにも関わらず MT-chat が他手法よりも優れていた理由として、MT-chat では文を単語などのより細かいフレーズ（「potter」、「my birthday」など）に分解し、それぞれについて出力に含むべきフレーズ（「harry」、「happy birthday」など）を推定することで、ユーザ発話と関連するフレーズをシステム発話に導入できたためとしている。しかしながら、システム発話に含まれるフレーズは「my birthday」に対する「happy birthday」のような定型文か、「potter」に対する「harry」のような単純な共起語が多く、新たな話題の導入には不向きである。

システムが新しい話題を導入できない場合、ユーザは自ら話を広げなければならないため、ユーザが対話を苦

痛に感じてしまい、対話を継続する意欲が減退する恐れがある。この問題を防ぐには、システムは関連性の低い話題を抑制しつつ、関連する話題を適切に導入できる必要がある。しかし、任意の話題に対して、関連性の低い話題の抑制と関連する話題の導入を両立した例は知られていない。

3. 任意の話題を持つユーザ発話に対する発話文の生成

雑談で出現する任意の話題を持つユーザ発話に対して、Web コーパスを利用して関連する話題を持つシステム発話を出力するには、無関係な話題の抑制と、適切な話題の展開の両立が重要である。ユーザ発話から話題を表す表現を抽出する際、単語のみで話題を表現しようとしても、「お腹が空いた」と「お腹が痛い」を「お腹」1 文節では区別できないように、ユーザ発話の話題を捉えきれない場合があるため、単語よりも自由度の高い表現で話題を表現する必要がある。また、述語項構造に閉じた情報に基づいてコーパスを検索すると、展開する話題が狭い範囲に限定される可能性がある。そのため、述語と述語の間のような、一つの述語項構造に閉じない情報を利用する方法が必要と考えられる。

これらを踏まえ、本研究では、係り受け関係にある文節ペア（係り受けペアと呼ぶ）を用いて話題を表現し、ユーザ発話に含まれる係り受けペアと、これとコーパス内で係り受け関係にある係り受けペアを組み合わせ、発話文を生成するアプローチを提案する。係り受けペアを話題の単位として用いることで、単語のみを用いるよりも、ユーザ発話の話題を適切に表現することができる。また、直接的な関係性を表す係り受け関係に基づいて係り受けペア間の関連性を定義することで、単純な共起関係を用いるよりも、ユーザ発話に含まれる係り受けペアと関連性の高い係り受けペアを得ることができると考えられる。

図1は提案手法の概要を示している。例えば「東京に行きたいです」というユーザ発話について発話を生成する場合、まず、係り受けペア抽出部でユーザ発話に対して係り受け解析を行い、係り受けペア（「東京に 行きたいです」）を入力係り受けペアの集合 U として得る。次に、関連係り受けペア検索部では、得られた入力係り

受けペア $u \in U$ について、それぞれに関連する係り受けペアの集合 R_u を係り受けペアデータベースから検索する。本研究では、関連する係り受けペアの検索を、係り受けペア中の文節間の係り受けをたどることで実現する。例えば、入力係り受けペア中の「行きたいです」が係る文節を持つ係り受けペアとして「行ったら 登りたいな」を検索し、「登りたいな」に係る文節を持つ係り受けペアとして「東京タワーに 登りたいな」を得る。発話合成部では、入力係り受けペア u と u に関連する係り受けペア $r_u \in R_u$ のペア $p = \{u, r_u\}$ を組み合わせることで、「東京に行ったら東京タワーに登りたいですか?」のように、 p に対応する文を生成する。最後に、各 p について得られた文を発話選択部でランキングし、システム発話として出力する。この係り受け関係に基づく発話生成によって、特定の話題に関する知識を持たなくとも、ユーザ発話に関連する応答文を適切な文法表現と共に生成することができる。本章では、モジュールごとに詳細を説明する。

3.1 係り受けペア抽出部

係り受けペア抽出部では、入力されたユーザ発話から、発話の話題を表す係り受けペアを抽出する。本研究では、入力されたユーザ発話に含まれる文節ペアのうち、文節間で係り受け関係を持つものを全てを、入力係り受けペアの候補とする。ただし、係り受けペアを話題の表現として用いても、一方の文節が特定の意味を持たない単語（ストップワード）で構成される場合には、文節単体のみを用いた場合と同様の問題が発生する。そのためストップワードを含む係り受けペアを、入力係り受けペアの候補から除外する。本研究ではストップワードとして、代名詞「する、いう、なる、ある、いる」などの特定の意味を伴わず使われる補助的な動詞「こと、の」などの抽象名詞「昨日、来年」などの時間に関係する名詞を用いる。

加えて、本研究では係り受けペアを抽出する際、表 1 に示す文節情報を同時に抽出する。意味タイプのうち、評価表現は [小林 05] によって公開されている評価表現辞書^{*3}中の有無に基づいて判定し、それ以外は形態素解析器の出力を利用する。本研究では、形態素解析器は JTAG[Fuchi 98] をベースに、形態素解析器の辞書の固有名詞に Wikipedia^{*4} のエントリを追加したものを用い^{*5}、係り受け解析器は Jdep[Imamura 07] を用いる。

3.2 関連係り受けペア検索部

§ 1 係り受けペアデータベース

関連係り受けペア検索部では、係り受けペアデータベースを用いて入力係り受けペアに関連する係り受けペアを検索する。このデータベースは、表 1 に示す文節情報のい

表 1 文節情報。具体例には、文「ようやく/朝ごはん/食べ始めた」中の文節「食べ始めた」を利用した場合を示す。

文節情報の項目名	説明	具体例
主辞	主辞には、冠名詞（黒ヤギの「黒」など）を除いた、文節の最も先頭にある単語を利用する。標準形を抽出。	食べる
品詞	主辞の品詞。	動詞
文節 ID	文内の文節の出現順を表す ID（小さいほど文頭）。	3
意味タイプ	固有名詞、場所、動作、評価表現、指示語、質問、その他のいずれか 1 つ。	動作
文節表記	文節全体の表記文字列。	食べ始めた
文節標準形	文節表記の末尾を標準形に変換した文字列。	食べ始める
格情報	文節の格。格がない場合もしくは省略の場合は空白。	（空白）

図 2 文節情報の抽出とデータベースへの保存。文節情報の抽出は、「お腹も 空いたし」を例に図示する。

ずれかをクエリとして、一致する文節情報を持つ係り受けペアを検索できるようにしたデータベースである。図 2 に、コーパス文からの文節情報の抽出とデータベースへの保存手順を示す。まず、文のみが含まれたコーパスから重複文を取り除き、各文に一意的な文 ID を付与して文 ID 付与済みコーパスを構築する。これに含まれる文と文 ID について、3.1 節と同様の手順（ストップワードを含む係り受けペアの除去を除く）で係り受けペアと文節情報を抽出し、文 ID と合わせてデータベースへ保存する。なお、本研究では、データベースを構築するコーパスとして、新聞記事やブログなどに比べて口語的な話題や表現を多く含む、Twitter を用いる。

§ 2 データベースからの関連係り受けペアの検索

本節では、データベースからの関連係り受けペアの抽出手順を示す。まず、入力係り受けペア u に含まれる文節ペアの両方の主辞（主辞ペアと呼ぶ）をクエリとして、 u と同じ主辞ペアを持つ係り受けペアの集合 X をデータベースから検索する。次に、各係り受けペア $x \in X$ の文 ID と文節 ID をクエリとして、 x に含まれる文節と係

*3 http://www.syncha.org/open_dic/index.html

*4 <http://www.wikipedia.co.jp>

*5 <https://github.com/nabokov/mecab-dic-overdrive>

り受け関係にある全ての文節を検索し、これらを含む係り受けペアの集合 $Y = \{y : y \notin X\}$ を得る．さらに、 x から Y を得たのと同様に、 $y \in Y$ に含まれる文節と係り受け関係にある文節を含む係り受けペアの集合 $Z = \{z : z \notin X, Y\}$ を検索し、 Y, Z を、関係係り受けペアの集合 $R_u = \{y, z : y \in Z, z \in Z\}$ として得る．例えば「新宿で食べた」という入力係り受けペア u について「新宿で食べた ラーメンが 美味しかった」という文がコーパス内にあった場合を考える． x として「新宿で 食べた」が「食べた」を含む係り受けペア y として「食べた ラーメンが」が、 y 中の文節「ラーメンが」を含む係り受けペア z として「ラーメンが 美味しかった」が得られ、 y 「食べた ラーメンが」と z 「ラーメンが 美味しかった」が関係係り受けペアの集合 R_u に含まれる．この例では、 x 中の文節と係り受け関係にある文節の品詞が名詞であったが、これが動詞や形容詞の場合も、同様に検索される．例えば「お昼に カラオケに 行ってから [新宿で] 食べて 帰った」(「行ってから」と「新宿で」の双方から「食べた」へ係る)から「新宿で 食べた」の関係係り受けペアを検索する場合を考える．このとき、 x として「新宿で 食べた」が「食べた」を含む係り受けペア y として「行ってから 食べた」と「食べて 帰った」が、 y に含まれる文節「行ってから」を含む係り受けペア z として「カラオケに 行ってから」が得られ、関係係り受けペアの集合 R_u へは「行ってから 食べた」、「食べて 帰った」、「カラオケに 行ってから」が含まれる．なお「食べて 帰った」からは「帰った」を含む y 以外の係り受けペアが文中に存在しないため、 z は検索されない．

こうして得られた入力係り受けペアと関係係り受けペアの組 $p = \{u, r_u\} \in P$ について、含まれる主辞が同一のものをまとめ上げる．ただし、出現数が1回だったものは、ごく特定の文脈でのみ成立する組み合わせである可能性が高いため P から除外する．さらに、得られた係り受けペアの組に含まれる文節のうち、最も文節 ID が大きい文節（出現した文中で最も文末に近いもの）が事態表現（品詞が動詞、形容詞、形容動詞、事態性名詞（「推薦」や「綺麗」などの、行為や状態、出来事を表す名詞 [小町 10]）のもの）である組のみを残し、他の係り受けペアの組 ($p = \{u : \text{「新宿で 食べた」}, r_u : \text{「食べた ラーメンが」}\}$ など) を P から取り除く．これは、システム発話を生成する際、事態表現で終わる文のほうが適切な文となりやすいためである．なお、上記処理によって係り受けペアの組が見つからない場合には、入力係り受けペアに含まれる文節のいずれかの主辞（ペアではない）をクエリとして係り受けペアを検索し、得られた係り受けペアの集合を X として関係係り受けペアを検索する．

3.3 発話合成部

入力係り受けペアと検索された関係係り受けペアの組のうち、出現数が上位のものを組み合わせて、システム発話を生成する．しかし、これらの係り受けペアに含まれる文節を適切な順序と文法表現（接続に用いる助詞や機能表現など）で組み合わせることは、本研究で扱える意味タイプが簡易なものであることや、慣用的に自然と捉えられる言い回しの存在を考えると容易ではない．

そこで本研究では、検索された関係係り受けペアの、もととも含まれていた文での使われ方（用例）を、できるだけそのまま利用して発話を生成する．まず、入力係り受けペア $u \in U$ と関係係り受けペア $r_u \in R_u$ の組 $p = \{u, r_u\}$ ごとに、各係り受けペアに含まれる文節表記を、文節 ID に基づいて文節が現れた順に接続し、 u に対応する出力文字列の候補 $\hat{s}_p \in \hat{S}_u$ として得る．なお、入力係り受けペアと関係係り受けペアで重複する文節表記は削除する．こうして各 p から得られた出力文字列候補 $\hat{s}_p \in \hat{S}_u$ について、完全に同一の文字列をまとめ上げる．このうち、出現数が上位 N 個の出力文字列候補 $\hat{s}_p$ を、システム発話に用いる出力文字列 $s_p \in S_u$ として得る．本研究では、 N を 10 とする．得られた文字列は、係り受けペア間の関係（順接など）を適切に表現する助詞や、その文字列が出現する上で自然と捉えられる文法表現が補われたものになると考えられる．

次に、出力文字列 $s_p \in S_u$ の文末表現を発話文に利用しやすいように変換し、システム発話候補 $o_p \in O_u$ を得る．出力文字列は、元々は文の途中に含まれていた場合が多いため、その文末表現は必ずしも発話文として適切なものにはなっていない．そのため本研究では、疑問詞が含まれている場合は疑問文（「??ですか？」など）、それ以外は平叙文（「??ですね」など）とする、簡易な文末変換を行うことで、システム発話を得る．ただし、「なんで」、「何」のように理由や名称を問う単語については、汎用性の高い「どんなのですか？」という質問を代わりに出力する．これは、今回用いたコーパスでは、これらの疑問詞が「なんでこうもお腹くかなあ」などの自問に用いられている場合が多く、この文から発話を生成すると「なんでお腹が空くんですか？」のような、他者への質問としては自明すぎて不自然な発話となったためである．

また、入力係り受けペアが固有名詞を含み、かつ検索された関係係り受けペアの数がしきい値 t 以下であった場合、上記手法での発話生成は困難であるため、固有名詞を「<固有名詞>」ってあまり知らないんですけどどういのですか？」というテンプレートに当てはめて発話を生成することとする．なお本研究では、 t を 5 とする．

3.4 発話選択部

発話選択部では、ある入力係り受けペア u とその関係係り受けペア r_u の組 $p = \{u, r_u\}$ から得られたシステム

発話候補 $o_p \in O_u$ について,

$$\text{score}(o_p; u) = \frac{1}{\log(n(u))} \frac{n(r_u; R_u)}{n(u)} \cdot n(s_p), (1)$$

に従って並べ替える．ここで $n(u)$ は入力係り受けペア u の主辞ペアと同一の主辞ペアを持つ係り受けペアのコーパス中の出現数, $n(r_u; R_u)$ は関連係り受けペア集合 R_u のうち, r_u と同一の主辞ペアを持つ係り受けペアの出現数, $n(s_p)$ は, o_p の元となる, 文末表現変換前の出力文字列 s_p のコーパス中の出現数を示す．第 1 項の $\frac{1}{\log(n(u))}$ は, 入力係り受けペア u の入力発話中における顕著性を表し, 注目しやすい話題についてシステム発話を生成するようにする工夫である．第 2 項は入力係り受けペア u の出現数に対する関連係り受けペア r_u の割合を表しており, これが大きいほど, u と関連が強い話題がシステム発話に含まれると考えられる．第 3 項は, 合成された文字列の妥当性を表す．顕著性が高い話題に対して関連が強い話題を検索できても, その話題があまりに一般的でない場合, ユーザはその話題を唐突に感じるだろう．また, Twitter がスラングやタイプミスを多分に含むことを考えると, 用例ベースであっても, 不適切な発話が生成される可能性がある．第 3 項を導入することで, より一般的な表現が選ばれやすくなるため, こうした不適切性が高い発話を抑制する効果が期待できる．

4. 実験 1: 一問一答での評価

提案手法は, 入力された発話の話題に応じて関連する話題を含む発話を生成する手法である．本研究ではまず, あらかじめ用意した入力文に対して, 提案手法を含む複数の手法を用いて応答文を出力し, それらを人手で評価し比較することで, 一問一答での適切さを基準として提案手法の有効性を検証する．

4.1 評価方法

提案システムおよび比較するシステム(後述)では, 入力文が 1 つ以上の話題を含むことを仮定している．我々は, 1 つ以上の明確な話題を含む文を集めるため, まず我々が収集した雑談対話コーパスと Twitter コーパスという 2 種類のコーパスから, 入力文の候補を大量に収集する．雑談対話コーパスは, のべ 360 名以上の話者から, 1 対 1 のテキストチャット形式による雑談を, 計 3680 対話, 約 13 万文収集したものである．これに, 目黒ら [Meguro 10] によって定義された対話行為を付与し, 自己開示に関する対話行為(自分についての事実や経験などを話した発話)もしくは情報提供に関する対話行為を, 入力文の候補として抽出した．Twitter コーパスから話題を含む文を容易に集める方法として, 話題となりうる単語を含む文を Twitter から検索し, そのうち非文でないものをルールで抽出する, 稲葉らの方法 [稲葉 14] がある．本研究ではこれを参考に, 話題となりうる単語 (Google trends

表 3 評価基準

評価項目名	説明
(1) 応答文の自然さ	応答文として直感的に違和感を感じないか
(2) 日本語としての正しさ	日本語としての理解できる文法か
(3) 応答する話題	入力文中の話題のうち, 適切な話題に応答しているか
(4) 発話間の矛盾	入力文と矛盾していないか
(5) 話題の関連性	入力文に関連する話題について発話しているか
(6) 話題の広がり	話を広げる情報が含まれているか
(7) 話の続けやすさ	システム発話の後に発話を続けやすいか

2012 in Japan^{*6}の各カテゴリで 10 位以上の単語のうち, 「Xperia acro HD」などのように, 空白を含まないもの)を含むおよそ 1 億 5 千万ツイートを入力文の候補として抽出した．収集した入力文候補について, 筆者ら以外の 1 名のアノテータが, 5 段階の Likert 尺度で内容の明瞭性を付与した．そのうち, 最良値の 5 (内容は省略なく明確に記述されている)を得た文から, コーパスごとに 150 文をランダムに選び入力文とした．

次に, 抽出した計 300 文に対し, 以下の 5 システムに 5 つずつ発話を生成させ, 応答文を得た．ただし, システムが生成した発話が 4 つ以下の場合には, 生成できた発話のうち, 発話選択部の並べ替え順序が最下位の発話を, システム発話の総数が 5 つになるまで追加した．また, 300 文中 20 文は, 「すずしくなってきましたね」のようにストップワード以外の文節を 1 つしか含まなかったため, 提案手法が入力係り受けペアを抽出できず, システム発話を生成できなかった．そのため実験では, 残りの 280 文を入力文として評価した．生成された発話を, どのシステムが生成した発話かわからないように, 入力文ごとにランダムに並び替え, これに筆者以外の 3 名のアノテータが分担して, 5 段階の Likert 尺度で評価値を付与した．表 3 に, 各評価項目の項目名とアノテータへ教示した説明文を示す．

4.2 比較システム

提案手法の有用性を検証するため, 以下の 5 システムを比較した．

- (1) 提案手法 係り受けペアデータベースを構築するため, 評価に用いるツイートとは異なる期間で, 約 10 億ツイートを収集した．ツイート特有の表現や形態素解析に失敗した表現が係り受けペアデータベースに混入することを避けるため, Twitter のリツイートを表す「RT」が含まれる文, 返信を表す「@」で始まる文, URL が含まれている文, 「【】」などの括弧を含む文, 2 文字以上の単語が 3 回以上繰り返されている文(「ねえねえねえねえ」など), 10 文字以下の文を取り除いた．最終的に得られた 1 億 2 千万ツイートを係り受けペアデータベースの構築に用いた．
- (2) 文節間の係り受けを利用した発話生成 (単語ベース)

^{*6} <https://www.google.co.jp/trends/topcharts#date=2012>

Sugiyama ら [Sugiyama 13] によって提案された, ユーザとの対話から焦点語を抽出し, 焦点語が含まれる文節と係り受け関係にある文節の主辞を, 焦点語と共にテンプレートに代入することで, 発話を生成する手法に基づくシステムである. Sugiyama らは, この手法が, ユーザが容易に発話を続けられるシステム発話の生成に有用であったと述べている. この手法は, 係り受け関係を話題間の関連性に利用する点で提案手法と類似した手法であり, 係り受けペアではなく単語を話題の単位とする点で異なる. 3.1 節で示したように, 単語のみではユーザ発話の内容を十分に捉えられない場合があるものの, 構造が単純なため理解が容易な文が生成されやすい手法である. 係り受け関係を持つ単語の検索には, 提案手法と同じコーパスを用いた.

- (3) **IR-status** Twitter から入力文に類似したツイート (status) を検索し, in-reply-to 機能で対応付けられた返信ツイート (response) をシステムの発話文として出力するシステムであり, 2 章で説明した [Ritter 11] の IR-Status と同一のものである. 入力文と似た意味を持つ status が存在する場合は, 適切に話題を遷移させる発話が得られると考えられる. 提案手法および評価用コーパスとは違う期間に収集したツイートのうち, in-reply-to で紐づいた約 2 億 4 千万ペアを抽出し, さらにリツイート (「RT」), URL, ハッシュタグ (「#」) を含むペアを取り除いた約 4500 万ペアでシステムを構築した.
- (4) **IR-response** [Ritter 11] の IR-Response と同一のものである. IR-status に比べてユーザ発話の内容からかけ離れた発話が出力されにくいという利点を持つ一方, 入力文とほぼ同一の, 単なる繰り返しと感じられる発話出力されうという欠点を持つ. システム構築には IR-status と同一のコーパスを用いた.
- (5) **ルールベース** 筆者らではないルール作成者 1 名が, 2ヶ月半をかけて構築したシステムである. ルールは Artificial Intelligence Markup Language (AIML)[Wallace 04] と呼ばれるマークアップ言語によって記述されており, 各ルールは入力文との一致を調べるパターンと, それに紐づいた出力文のペアから成る. 記述したルールの総数は 149,300 個である. またルールの検索には, 入力文の先頭単語から合致するルールを検索して対応する発話を出力する, 先頭一致優先型検索を用いた. このルールベースのアプローチは, ロープナー賞で複数回優勝しているシステムで採用されているアプローチである [Wallace 04]. そのため, 他の統計的に構築した 4 システムに対する有力なベースラインになると考えられる.

表 4 雑談対話コーパスにおける 1-best の平均値. (5 が最良. Bold 体は最大の評価値を表す. 有意差は提案手法との差異を表す. *: $p < .1$, **: $p < .05$)

	提案手法	単語	IR-st.	IR-re.	ルール
(1) 応答文の自然さ	3.21	3.32	2.11*	2.90*	3.90**
(2) 日本語としての正しさ	4.06	4.15	3.13**	4.09	4.52**
(3) 応答する話題	3.37	3.46	2.12**	3.14	3.86**
(4) 発話間の矛盾	3.36	3.48	2.28**	3.30	4.00**
(5) 話題の関連性	3.49	3.54	2.29**	3.38	3.93**
(6) 話題の広がり	2.96	3.10	2.10**	2.86	3.41**
(7) 話の続けやすさ	3.12	3.25	2.10**	2.83*	3.51**

表 5 雑談コーパスにおける 5-best の最大値の平均値. (5 が最良. Bold 体は最大の評価値を表す. 有意差は提案手法との差異を表す. *: $p < .1$, **: $p < .05$)

	提案手法	単語	IR-st.	IR-re.	ルール
(1) 応答文の自然さ	4.06	3.98	3.54**	4.01	4.46**
(2) 日本語としての正しさ	4.56	4.47	3.76**	4.23**	4.76
(3) 応答する話題	4.01	4.06	3.41**	3.85	4.32**
(4) 発話間の矛盾	3.99	3.94	3.43**	3.93	4.34**
(5) 話題の関連性	4.03	4.06	3.42**	3.96	4.27*
(6) 話題の広がり	3.57	3.65	3.16**	3.29**	3.40
(7) 話の続けやすさ	3.69	3.65	2.93**	3.30**	3.51

4.3 結果と分析

表 4, 表 5 に, 雑談対話コーパスにおける 1-best, 5-best の評価値をそれぞれ示す. 有意性の検定にはウィルコクソンの符号順位検定を用いた. 雑談コーパスはルールベース手法の構築に用いたコーパスと同一であることから, ルールベースシステムの評価値が有意に他システムの評価値を上回っていた. すなわち, closed データにおいては, ルールベース手法が適切に動作していることを示している. ルールベース以外の統計的手法の中では, 提案手法と単語ベースの手法が, 何ターンも継続して対話を続ける上で重要な (7) 話の続けやすさと, 直感的な自然

表 6 Twitter コーパスにおける 1-best の平均値. (5 が最良. Bold 体は最大の評価値を表す. 有意差は提案手法との差異を表す. *: $p < .1$, **: $p < .05$)

	提案手法	単語	IR-st.	IR-re.	ルール
(1) 応答文の自然さ	3.35	3.30	2.11**	2.73**	2.83**
(2) 日本語としての正しさ	4.34	4.23	2.92**	3.84**	4.17
(3) 応答する話題	3.42	3.37	2.12**	2.84**	2.81**
(4) 発話間の矛盾	3.43	3.37	2.12**	2.97**	2.78**
(5) 話題の関連性	3.53	3.37	2.18**	3.05**	2.73**
(6) 話題の広がり	3.18	3.23	2.12**	2.59**	2.63**
(7) 話の続けやすさ	3.37	3.36	2.06**	2.49**	2.69**

表 7 Twitter コーパスにおける 5-best の最大値の平均. (5 が最良. Bold 体は最大の評価値を表す. 有意差は提案手法との差異を表す. *: $p < .1$, **: $p < .05$)

	提案手法	単語	IR-st.	IR-re.	ルール
(1) 応答文の自然さ	4.12	3.83**	3.64**	3.88**	3.14**
(2) 日本語としての正しさ	4.59	4.38**	3.88**	4.31**	4.21**
(3) 応答する話題	3.93	3.76*	3.40**	3.76	3.04**
(4) 発話間の矛盾	4.07	3.74**	3.44**	3.90	3.05**
(5) 話題の関連性	4.08	3.79**	3.51**	3.94	2.99**
(6) 話題の広がり	3.65	3.65	3.05**	3.36**	2.74**
(7) 話の続けやすさ	3.87	3.67*	2.94**	3.32**	2.66**

表 8 生成された発話例

例の種類	入力文	生成した発話	システム
提案手法が良い例	ウォーキングではなく、例えば都会の中で電車に乗らずに、歩いて新しいスポットを探すのが好きです。	都会の中でも自然を感じられるよね	提案手法
		都会に行きたいです	単語
		おふう ('・ '・)	IR-status
		電車乗るのだいすきです	IR-response
		ウォーキング流行ってますし楽しいですよ	ルール
IR-response が良い例	テザリングに限界を感じてきてる	無課金の限界を感じるよね	提案手法
		限界に近いな	単語
		今日は学校行った?	IR-status
		テザリングマサキも限界感じてますやっぱ WiMAX がイチバン	IR-response
		テザリングに興味ありますか?	ルール

さを表す (1) 応答文の自然さにおいて、他の検索ベースの手法を有意に上回っていた。

表 6 は Twitter コーパスに対する 1-best の評価値を示している。ほぼ全ての評価基準において、提案システムと単語ベースシステムが有意に他手法を上回っていた。雑談コーパスでの結果と異なり、ルールベース手法が (2) 理解のしやすさを除いて提案システムを下回っている。この理由として、Twitter コーパスから得られた入力文には、雑談コーパスに現れない固有名詞を含む、複雑な構造を持つ文が含まれていたことが挙げられる。ルールは、「天一食べた」のような単純な発話に対しては、「美味しかったですか?」のような定型パターンで返答することで、我々が予想していたよりも固有名詞のバリエーションの弱さを補っていた。しかし、複文などより複雑な構造を持つ発話に対しては、先頭一致では適切なルールを発見することが難しく、評価が低くなっていた。

表 4, 表 5, 表 6 において、単語ベースのシステムと提案システムとの差が小さかった理由として、以下の 2 点が考えられる。1 つは、話題の明瞭性が高い文を入力文としたことで、「秋刀魚が好きですね」のような、単語のみで話題を表現可能な文が入力文に多く含まれていたことである。この場合、単語のみからでも「秋刀魚 食べたいです」のように、入力文の話題に合致した文を出力することができる。しかも、単語の組み合わせのみで発話を構成するため、シンプルで理解しやすい文となる。一方、提案手法は、入力された文節ペアの出現数が少ない場合でも、極力文節ペアに基づいて文を合成するように設計されている。この特性は、ユーザの話題を適切に捉えるためには有用である。一方で、検索によって得られる関連係り受けペアが少ない場合には、ごく特定の文脈で出現する話題の出現数が相対的に大きくなる。これらから生成されたシステム発話は、ユーザ発話との関連性がわかりにくいものが含まれるため、評価が低くなっているものと考えられる。もう 1 つの理由は、アドホックな基準でのリランキングである。本研究では、提案手法と単語ベースの手法において、複数生成された発話候補をリランキングすることで出力するシステム発話を選択している。しかし、このリランキングが適切に働い

ていないため、理解困難な文が上位にランクされている場合があった。この問題は単語ベースシステムにおいても起こりうるが、提案システムのほうが文生成の自由度が大きいため、より大きく影響したと考えられる。雑談対話コーパスでの 1-best の平均値 (表 4) と 5-best の最大値の平均値 (表 5) を比較すると、どちらも有意ではないものの、1-best の平均値では単語ベースのほうが良い評価を得る傾向にあり、逆に 5-best の最大値の平均値では提案手法のほうが上回る傾向が見られた。この結果は、単語ベースではリランキングの悪影響が小さいという仮説を支持している。

表 7 は、Twitter コーパスにおける 5-best の最大値 (評価値の合計値を基準とする) の平均値を示す。表 7 において、提案システムが単語ベースシステムを含む他手法にほとんどの指標で上回っていた。また、この 5-best に採用された文の平均順位は、提案 3.87, 単語 3.16, IR-status 3.59, IR-response 3.95, ルール 1.91 であった。ここからも、提案手法が持つ大きな自由度によって、単語ベースシステムよりも評価値の高い発話を生成可能であることと、単語ベースではリランキングの悪影響が小さいことが示されている。

検索ベースの IR-status と IR-response を比較すると、全ての指標において、IR-response が IR-status よりも優れていた。これは、IR-status は (6) 話題の広がり性に優れ、一方で IR-response は (5) 意味の関連性や (4) 発話間の矛盾の少なさにおいて優れるもののオウム返しの発話が多くなるという、我々の予想とは大きく異なる結果であった。IR-status の (2) 理解のしやすさが他のシステムよりも極端に低いことから、本研究で用いた比較的大きなコーパスサイズに関わらず、適切な発話を検索できず、入力発話と無関係なツイートをシステム発話として生成してしまったためと考えられる。

表 8 に、各システムが出力したシステム発話文の例を示す。提案手法が良い例では、提案手法は「都会の中」という入力係り受けペアから「都会の中でも自然を感じられる」という、ユーザ発話に関連する話題を含む発話を、適切な副助詞「でも」とともに生成していた。一方、単語単位の係り受けでは「都会」のみを用いていたため、

図3 対話システムの構成図

ユーザ発話全体とは関連の薄い、「都会に行きたいです」という発話を生成していた。検索ベースの IR-status と IR-response を比べると、IR-status は、ほとんどユーザ発話と関係のない理解困難な相槌を生成していた。一方 IR-response は、ユーザ発話と関連があるものの、ユーザ発話に含まれる単語で構成される発話が生成されていたため、新しい話題を導入できていなかった。ルールは「ウォーキング」という単語に対応した発話を生成していたが、ユーザ発話では「ウォーキングではなく」と否定系で用いられていたため、生成された発話はユーザ発話の話題とずれた発話となっていた。IR-response が良い例では、IR-response は「WiMAX がイチバン」のように、関連する話題が含まれた発話を検索できていた。これは、ユーザ発話の話題の「テザリング」が一般的ではなかったことから、繰り返しとなる発話が見つからなかったためと考えられる。これに対し、提案手法と単語単位の係り受けは、ユーザ発話の話題として「テザリング」ではなく「限界」を用いた発話が生成されていた。これは「テザリング」の出現数が少なすぎたことで「テザリング」の関連係り受けペアを検索できず、より顕著性の低い「限界」を用いた発話しか得られなかったことによる。またこちらでも、IR-status はユーザ発話と全く関係のない発話を生成していた。こうした発話が、IR-status の評価値の低さにつながったものと考えられる。ルールは「テザリング」に対応した発話を生成することができたものの、応答文としてはやや不自然な発話であった。話題を表す単語が一致していても、そのみでは、適切な応答文の生成は困難であると考えられる。

5. 実験 2: 対話システムへ組み込んだ場合の評価

ここでは、提案手法を対話システムへ組み込んだ場合の有用性について検証する。提案手法は挨拶や質問への応答については対象としていないため、システムへの質問以外への応答を行うモジュールへ各手法を組み込み、対話システムへ応用した場合の有用性を検証する。

5.1 対話システム

図3に対話システムの構成図を示す。この対話システムは、[Higashinaka 03] や [Sugiyama 13] と同様に、発話理解部、対話制御部、発話生成部、発話決定部から構成される。発話理解部では、入力されたユーザ発話について、[Meguro 10] で定義されるユーザ対話行為の推定、入力係り受けペアの抽出、システムへの質問か否かの分類を行う。対話制御部では、推定されたユーザ対話行為に基づいて、システムが出力すべき対話行為（ここでは上位5つ）を推定する。この推定は、雑談コーパスに人手で付与した対話行為に基づき、[Sugiyama 13] と同様の手続きで線形 SVM を用いて行う。発話生成部では、ユーザ発話がシステムへの質問であるか否かに従い、発話生成方法を切り替える。ユーザ発話がシステムへの質問と分類された場合は、システムへの質問とその回答を収集した、PersonDataBase [杉山 14] を用いてシステム発話を生成する。一方、システムへの質問でなかった場合は、4.2 節に示した手法のうち、単語ベース手法を除く4手法のいずれかを用いて、N-best のシステム発話候補を出力する。ただし、提案手法は挨拶などへの応答は想定していないため、システム対話行為が挨拶、共感、非共感、相槌、フィラーのいずれかであった場合にのみ「こんにちは」などの定型句を出力する。発話決定部では、対話の流れを大きく逸脱するような発話を抑制するため、各手法が出力したシステム発話候補の対話行為を推定し、対話制御部で推定されたシステム対話行為と異なる対話行為を持つものを取り除く。残された N-best のシステム発話候補のうち、最上位のものを、システム発話として出力する。

5.2 実験設定

実験参加者は日本語話者 28 人（男女比はほぼ同数。20 代?60 代で 30 代、40 代が全体の 7 割。著者は含まない）で、テキストチャットによる対話を計 8 回（4 システムと 2 回ずつ）行った。用いたシステムは、4.2 節に示した手法のうち、単語ベース手法を除く 4 手法（ルール、提案手法、IR-status、IR-response）を用いたシステムである。収集された対話数は、それぞれのシステムについて 56 であった。対話時間は 8 分で、1 対話に含まれる、ユーザ

図 4 各対話システムの評価値．有意差はウィルコクソンの符号順位検定を用いて提案手法との差の有無を検定．

*: $p < 0.05$

表 9 評価基準

評価項目名	説明
(1) 対話の流れ	対話の流れは自然かの自然さ
(2) 文法の正しさ	日本語としての理解できるか
(3) 1 ターンの自然さ	1 ターンごとに見た時、発話は適切か
(4) 意味の通った応答	あなたの発話に対してシステムは意味の通った応答をしているか
(5) 有用性	対話自体が有用だったか
(6) 容易さ	次の発話を容易に考えられるか
(7) 多様性	システムの発話は多様であったか
(8) あなたの対話参加意欲	あなたの対話への参加意欲は高かったか
(9) システムの対話参加意欲	あなたが感じる、システムの対話への参加意欲は高かったか
(10) また対話したいか	この対話システムとまた対話してみたいか

とシステムの発話を合わせた発話数は約 23.2 であった．各対話は、ユーザから話しかけることで対話が開始される．各対話後に、表 9 に示す各評価項目について、5 段階評価を行った．有意性の検定は、提案手法との差異について、ウィルコクソンの符号順位検定を用いて行った．

5.3 結果と分析

図 4 に、各システムの評価値を示す．全体的にルールベースシステムが最も高い評価値を得ており、特に (2) 文法の正しさと (3) 1 ターンの適切さにおいて、提案手法を有意に上回っている．(2) 文法の正しさはルールベースが最も得意とするところであり、高い評価を得ることは予想通りである．(3) 1 ターンの適切さについては、実験参加者の対話システムに対する期待値があまり高くなく、ルールで扱いやすい短いユーザ発話が多かったためと考えられる．加えて、ルールベースシステムで対応するパターンが見つからない時には「どうなのでしょう」という定型文が出力され、これが何度も繰り返されるとユーザ評価に悪影響をおよぼすと予想していたが、比較的短い対話でシステムが切り替わったことで、飽きられずに受け入れられたと考えられる．一方、統計的手法の中で

は、提案手法が最も良い評価を得ていた．検索ベースの 2 手法は、項目によって得手不得手があるものの、それらのうち高い評価値を得た手法よりも、提案手法は同等以上の評価を得ていた．各項目を調べると、1 ターンの適切さは、実験 1 の雑談対話と同様に、ルール、提案手法、IR-response、IR-status の順に並んでいることが確認できる．他の評価項目も概ね同様の順で並んでいることから、一問一答の適切さは、対話実験の評価にある程度関連するものと考えられる．また、(6) 発話を考える容易さと (8) ユーザの対話への意欲については、提案手法が検索ベースの 2 手法のどちらに対しても有意に良い評価を得た．IR-response ではオウム返しが出力されてしまうことで、自分で話題を広げる必要があることから (6) 容易さが低下し、また表面的に回答している印象が強まったため、(9) システムの意欲とともに (8) ユーザの対話意欲も低下したと考えられる．一方、提案手法は、(7) 多様性と (9) システム意欲についてルールと同等以上の値を得た．ユーザ発話に関連する話題を積極的に出力することで、ユーザの対話への参加意欲を喚起できる可能性が示されている．

検索ベースの 2 手法内で比較すると、IR-status は対話への意欲と多様性以外の項目で IR-response よりも低い評価であった．文法や 1 ターンの適切さが低いことを考えると、理解困難な発話が多かったと推測される．しかし、多様性とあなたの/システムの意欲については、IR-response を上回っており、IR-response の弱みである、話題の展開性の欠如がこの結果に影響していると考えられる．

6. ま と め

本研究では、雑談対話において任意の話題を持つユーザ発話に対する応答文を適切に生成するため、入力発話に含まれる係り受け関係を持つ文節ペア（係り受けペア）と、それに関連を持つ係り受けペアを Twitter 中の係り受け関係に基づいて抽出し、これらをコーパス中の用例と

組み合わせることで発話を生成する合成ベースの手法を提案した。人同士の対話や Twitter から収集した文に対して生成した応答文を評価する実験の結果、提案手法が従来の検索ベースの手法を上回り、Twitter コーパスにおける 5-best の最大値では、単語単位で係り受け関係を利用する手法を上回った。また、実際の対話システムへ組み込んだ場合においても、作りこまれたルールによる手法には劣っていたものの、検索ベースの手法よりも高いユーザ評価を得る傾向が見られ、特に対話の継続性と、ユーザの対話への意欲の喚起に有用であることが示された。

今後の課題として、生成された発話の適切なランキングが挙げられる。適切なランキングができれば、本研究で用いた基準では下位にランクされていた発話からも、より適切な文を出力できるようになる。また、利用する文節数の適切な調整も課題の一つである。ストップワードが連なる場合や「の」によって文節が連なる場合など、2 文節のみでは意味を限定しきれていない場合もあった。一方で、単語のみで十分に意味を捉えられる場合もあり、その場合は単語のみを利用したほうがスパース性を緩和でき、より安定して発話を生成できると考えられる。

また、ユーザ対話に対して無矛盾な発話生成も大きな課題である。実験では提案システムは比較的無矛盾な発話を生成できていたと評価されているが、あくまでも一問一答としての無矛盾性である。本来、複数ターンから成る対話を適切に継続させるには、複数ターンでの無矛盾性を適切に扱える必要がある。この機能は、従来より含意認識として取り組まれているものであるため、これらの知見を利用してシステムを構築し、より完成度の高いシステムへとつなげていきたい。

◇ 参 考 文 献 ◇

- [Bessho 12] Bessho, F., Harada, T., and Kuniyoshi, Y.: Dialog System Using Real-Time Crowdsourcing and Twitter Large-Scale Corpus, in *Proceedings of the 13th annual SIGdial Meeting on Discourse and Dialogue*, pp. 227–231 (2012)
- [Bickmore 01] Bickmore, T. and Cassell, J.: Relational Agents: A Model and Implementation of Building User Trust, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 396–403 (2001)
- [Fuchi 98] Fuchi, T. and Takagi, S.: Japanese morphological analyzer using word co-occurrence: JTAG, in *Proceedings of the 17th international conference on computational linguistics*, pp. 409–413 (1998)
- [Higashinaka 03] Higashinaka, R., Nakano, M., and Aikawa, K.: Corpus-based discourse understanding in spoken dialogue systems, in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, Vol. 1, pp. 240–247 (2003)
- [Higashinaka 14] Higashinaka, R., Kobayashi, N., Hirano, T., Miyazaki, C., Meguro, T., Makino, T., and Matsuo, Y.: Syntactic Filtering and Content-based Retrieval of Twitter Sentences for the Generation of System Utterances in Dialogue Systems, in *Proceedings of International Workshop Series on Spoken Dialogue Systems Technology*, pp. 113–123 (2014)
- [Imamura 07] Imamura, K., Kikui, G., and Yasuda, N.: Japanese Dependency Parsing Using Sequential Labeling for Semi-Spoken Language, in *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pp. 225–228 (2007)
- [Meguro 10] Meguro, T., Higashinaka, R., Minami, Y., and Dohsaka, K.: Controlling Listening-oriented Dialogue using Partially Observable Markov Decision Processes, in *Proceedings of the 23rd International Conference on Computational Linguistics*, pp. 761–769 (2010)
- [Ritter 11] Ritter, A., Cherry, C., and Dolan, W.: Data-Driven Response Generation in Social Media, in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pp. 583–593 (2011)
- [Shibata 09] Shibata, M., Nishiguchi, T., and Tomiura, Y.: Dialog System for Open-Ended Conversation Using Web Documents, *Informatica*, Vol. 33, pp. 277–284 (2009)
- [Shibata 12] Shibata, T. and Kurohashi, S.: Predicate-Argument Structure-Based Textual Entailment Recognition System Exploiting Wide-Coverage Lexical Knowledge, *ACM Transactions on Asian Language Information Processing*, Vol. 11, No. 4, pp. 16:1–16:23 (2012)
- [Sugiyama 13] Sugiyama, H., Meguro, T., Higashinaka, R., and Minami, Y.: Open-domain Utterance Generation for Conversational Dialogue Systems using Web-scale Dependency Structures, in *Proceedings of the 14th annual SIGdial Meeting on Discourse and Dialogue*, pp. 334–338 (2013)
- [Takeuchi 07] Takeuchi, S., Cincarek, T., Kawanami, H., Saruwatari, H., and Shikano, K.: Construction and optimization of a question and answer database for a real-environment speech-oriented guidance system, in *Proceedings of the International Committee for the Co-Ordination and Standardization of Speech Databases and Assessment Techniques*, pp. 149–154 (2007)
- [Wallace 04] Wallace, R.: The Anatomy of A.L.I.C.E., *ALICE Artificial Intelligence Foundation, Inc.* (2004)
- [Wong 12] Wong, W., Cavedon, L., Thangarajah, J., and Padgham, L.: Strategies for Mixed-Initiative Conversation Management using Question-Answer Pairs, in *Proceedings of the 24th International Conference on Computational Linguistics*, pp. 2821–2834 (2012)
- [稲葉 14] 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)
- [小町 10] 小町守, 飯田龍, 乾健太郎, 松本裕治: 名詞句の語彙統語パターンを用いた事態性名詞の項構造解析, *言語処理学会*, Vol. 17, No. 1, pp. 141–159 (2010)
- [小林 05] 小林のぞみ, 乾健太郎, 松本裕治, 立石健二, 福島俊一: 意見抽出のための評価表現の収集, *自然言語処理*, Vol. 12, No. 3, pp. 203–222 (2005)
- [杉山 14] 杉山弘晃, 目黒豊美, 東中竜一郎, 南泰浩: 対話システムのパーソナリティを問う質問に対する発話生成, *言語・音声理解と対話処理研究会 (SIG-SLUD-B303)*, pp. 33–38 (2014)

〔担当委員：難波 英嗣〕

2014 年 5 月 8 日 受理

著 者 紹 介

杉山 弘晃(正会員)

2007 年東京大学工学部機械情報工学科卒。2009 年同大知能機械情報学専攻修士課程終了。同年 NTT 入社。現在、奈良先端科学技術大学院大学情報科学研究科博士後期課程在学中。人と自然な対話を行う雑談対話システムの研究に従事。

目黒 豊美(正会員)

2006 年東北大学工学部卒業。2008 年東北大学大学院工学研究科博士課程前期修了。同年 NTT コミュニケーション科学基礎研究所入社。対話研究に従事。情報処理学会会員。

東中 竜一朗(正会員)

1999 年に慶應義塾大学環境情報学部卒業。2001 年に同大学大学院政策・メディア研究科修士課程。2008 年に博士課程修了。博士(学術)。2001 年に日本電信電話株式会社入社。現在、NTT メディアインテリジェンス研究所にて勤務。音声言語メディアプロジェクトにて、質問応答システム・音声対話システムの研究に従事。情報処理学会、電気情報通信学会、言語処理学会、各会員。2004 年 11 月より 2006 年 3 月までシェフィールド大学客員研究員。

南 泰浩

1986 年慶應義塾大学理工学部電気工学科卒業。1991 年同大大学院博士課程了。同年 NTT 入社。1999-2000 年 MIT 客員研究員。現在、電気通信大学情報システム学研究科教授。音声認識、対話処理の研究に従事。1993 年日本音響学会粟屋潔学術奨励賞。2003 年電子情報通信学会論文賞受賞。2005 年テレコムシステム技術賞受賞。2006 年情報処理学会創立 45 周年記念論文優秀論文賞受賞。2008 年テレコムシステム技術賞受賞。電気情報通信学会、情報処理学会、音響学会、IEEE、各会員。工博。