

ボケて返す対話型エージェントの基礎検討

平成29年度 卒業論文

日本大学 文理学部 情報科学科 宮田研究室

鈴木 奨

概要

現代社会において対話型エージェントは、看護やショッピングなどの生活における様々なシーンへと普及し始めており、今後多くのユーザが利用すると考えられる。しかし、対話型エージェントとの会話は無機質なものになりがちであり、このようなエージェントに親しみを抱けないユーザは少なくないと思われる。この問題を解決するために、エージェントが対話の中でユーモアを交えることで、ユーザはエージェントに対して親しみを持てるのではないかという仮説に基づき、我々は、ユーザの入力に対し、ユーモアのある聞き間違いをして聞き返すボケをするエージェントを提案する。例えばこれは、“学会について教えて？”とエージェントに対して質問をすると、“え、合体？”と、エージェントが“学会”を“合体”と聞き間違いをして、聞き返すというものである。これによって、エージェントが対話の中でユーモアを交えられるようになり、ユーザはエージェントに対して親しみを感じられると考えられる。プロトタイプシステムを用いた検証実験の結果、提案手法によって生成したボケに対し、ユーザは一定のユーモアを感じられることが確認できた。また、我々の仮説通り、エージェントに対して一定の親しみを感じられることも確認できた。本稿の貢献は次の通りである。

- ユーザがエージェントと対話を行うシーンにおいて、エージェントがユーザの発言の一部に対してユーモアのある聞き間違いをして聞き返すボケをするシステムを提案したこと。
- 上記提案のプロトタイプシステムを構築し、ユーザ実験を行って有効性を検証したこと。

第1章 序論

1.1 研究の背景

現代社会において対話型エージェントは、看護やショッピングなどの我々の生活における様々なシーンへと浸透し始めており、将来的にはユーザの作業の自動化や、ユーザの友人になるといった活躍が期待される。しかし、その一方でエージェントとの無機質な対話に親しみを感じることができないユーザも少なからずいると思われる。例えば、リビングルームにコミュニケーションロボットが導入されたシーンを考えたとき、ロボットが常に堅苦しい話ばかりしていたら、家族の一員にはなりにくいだろう。あるいは、介護施設で高齢者の話し相手になるロボットが、冗談1つも言わず、生真面目な会話ばかりしていたら、ロボットが施設の人気者になることは難しいだろう。このような問題を解決するためには、エージェントとユーザの間に親和的な関係を築くことが必要であると言える。また、人同士の親和的な関係の形成には、笑いが欠かせないとされており [1]、ユーモアと親しみの関係性が窺える。日本国外においても、エージェントによるユーモア表現を試みている研究事例は散見される [2]。

1.2 研究の目的

本研究では、エージェントが対話の中でユーモアを交えられるようにすることで、ユーザがエージェントに対して親しみを持つことができるようになることを最終的な目的とする。このとき、エージェントが生成するユーモアはユーザが受容できるものにし、対話中にユーザが納得感を得られるようにする。さらに、日常生活の様々なシーンでの活用や、同一ユーザの複数回利用により生じる飽きを考慮し、ユーザからの入力できる内容を制限しないようにする。本稿では上記の条件を満たす対話型エージェントのプロトタイプシステムの構築、および検証実験からその効果を確認することを目的とする。

1.3 本論文の構成

本論文の構成は次のとおりである。

2章では、対話型エージェントを様々な分野に活用した研究事例や、対話型エージェントによるユーモア表現を試みている研究事例について紹介する。

3章では、対話型エージェントにユーモアを取り入れる際に生じる問題点について述べ、それらを踏まえたうえで本研究における課題を設定する。

4章では、学術的・実践的観点からボケ生成手法について検討し、ボケて返す対話型エージェントの提案を行う。

5章では、ボケ生成を行う際に参考にする3つのスコアの計算方法を中心に、ボケて返す対話型エージェントの実装方法について具体的に述べる。

6章では、行ったユーザ実験の目的や手順について述べ、実験結果から得られた知見についても述べる。

最後に7章にて、本論文の結論を述べる。

第2章 ユーモア表現を行うエージェント に関する研究事例

本章では、ユーモア表現を行うエージェントに関する研究事例について述べる。2.1節では、対話型エージェントに関する研究事例について紹介する。2.2節では、ユーモア表現を行うエージェントについて紹介する。また、ユーモア表現を行うエージェントは、大きく分けて、一方的にユーモア表現を行うエージェントと対話によってユーモア表現を行うエージェントに分けられる。これらは2.2.1節、2.2.2節においてそれぞれ紹介する。

2.1 対話型エージェントに関する研究事例

本節では、対話型エージェントに関する研究事例について述べる。近年では対話型エージェントを対象とした研究は多岐にわたり、様々なシーンでの活用方法が検討されている[3][4][5][6][7][8][9][10][11]。

[3]、[4]では、一人暮らしの高齢者に向けた対話型エージェントの研究が行われている。[3]は、エージェントと対話を行うことでコミュニケーション不足を解決しようとする試みである。[4]は、エージェントによる発話に対する、認知症患者であるユーザの反応から日々の健康状態・認知状態を把握することを目的としている。

[5]、[6]では、1人のユーザに対し、複数のエージェントが対話を行うシステムを提案している。[5]では、ユーザとエージェントによる擬似的な対話をしながらの鑑賞活動を実現する、鑑賞支援システムを提案している。実験により、2種類のエージェントを用意することで、人間による対話型鑑賞に近い話題の展開が可能であることが示唆されている。[6]では、ユーザ1人に対しエージェントを2つ用意し、三者間の対話システムを提案している。実験により、三者間の対話システムは、ユーザが感じるエージェントへの親しみや、対話の雑談らしきが増すことが示されている。

[7]、[8]では、エージェントをゲーム分野に利用する研究が行われている。[7]は対話によるコミュニケーションを必要とする人狼ゲームを題材とし、若者の対話教育のための支援環境として、擬人化エージェントを用いた手法を提案している。[8]は、GPS機能付き携帯電話を端末として、エージェントによる現実感のある世界の表現を試みている。著者らはこれを様々な分野に応用が可能であるとした上で、特にオンラインゲームなどで応用できると述べている。

[9]、[10]、[11]は、それぞれプレゼンテーション、車の運転、余暇の充実化についてユーザを支援するエージェントを提案している。[9]はエージェントが、エージェント作成者の代理として任意のユーザと会話することができる、分身エージェントの実現手法を提案している。[10]はドライバの退屈や眠気回避させる手段として、車から得られるデータを利用した会話型CGエージェントの提案をしている。[11]はスケジュール帳を情報源として、ユーザに余暇の過ごし方を提案するエージェントについて提案している。

2.2 ユーモア表現を行うエージェントに関する研究事例

以下は、いずれも笑いを通してユーザとエージェントのコミュニケーションをより良いものにするという試みである。

2.2.1 一方的にユーモア表現を行うエージェント

一方的なユーモア表現を行うエージェントの例として、[12], [13], [14], [15], [16], [17], [18]が挙げられる。

[12], [13], [14], [15]はエージェントによる一方的な発話によってユーモア表現を試みている。[12]では、エージェントがユーザに笑い感情を誘起させる手段として、大喜利が用いられている。機械学習的アプローチに基づいてユーザに笑い感情を誘起させるロボットインタラクションモデルの生成を行っており、データセットにはセンサデータによってユーザの笑い感情が誘起することが確認された会話文を用いている。[13]では、Webニュース記事本文の感情情報を用いて、おかしみの構造図に基づいた形式的なボケによる漫才台本を自動生成する手法を提案している。[14]では、ユーザの入力した単語の意味を利用することで、日常会話に応用できる置換型駄洒落を出力するシステムを構築している。[15]は、漫才形式の対話文の自動生成システムである。入力した文章から単語を選び、それを別の単語に置換してボケを生成することで、漫才形式の対話を生成している。[14], [15]は、ある単語を別の単語に置換することでエージェントによる笑いの実現を目指している。

[16], [17], [18]はエージェントによる発話以外の方法でユーモア表現を試みている。[16]は、一見関係の無さそうな単語間に、意味的なつながりを持たせたユーモア文である、Tom Swifitiesの生成を試みている。[17]は、各単語の頭文字のアルファベットを連結することで省略している名称を、略称そのままに各単語を置換することでユーモアを生み出す試みである。単語の置換時には、もとの単語群と新たに生成された単語群の関係が、意味は遠く、音は近くなるようにしている。[18]は、画像にユーモアのあるタイトルを付ける試みである。また、ユーモア文の生成には、地域ごとのトレンドも参照している。

2.2.2 対話によってユーモア表現を行うエージェント

ユーザと対話を行うエージェントの例として[19], [20], [21], [22], が挙げられる。[19]はユーザの対話継続欲求を向上させるために、単語間類似度を用いたユーモア発話の自動生成手法を提案している。[20], [21], [22]では、ユーザからの特定の入力に対して、あらかじめ設定された、ユーモアを含む応答文を返していると思われる。

第3章 研究課題

3.1 問題の定義

現代社会において対話型エージェントは、看護やショッピングなど様々な場面でみかけられるようになった。将来的には家庭や介護などの人間とのコミュニケーションが重要な場面での活躍も期待されており、現在でも様々な分野での活用法が研究されている [3][4][5][6][7][8][9][10][11]。

しかし、このような対話型エージェントは未だその対話の多くが無機質なものである。無機質な対話に親しみを持ってないユーザには、このようなエージェントは受け入れられないという問題が懸念される。例えば、リビングルームにコミュニケーションロボットが導入されたシーンを考えたとき、ロボットが常に堅苦しい話ばかりしていたら、家族の一員にはなりにくいだろう。あるいは、介護施設で高齢者の話し相手になるロボットが、冗談1つも言わず、生真面目な会話ばかりしていたら、ロボットが施設の人気者になることは難しいだろう。本章では、対話の中にユーモアを取り入れる際に生じる問題点について述べる。

3.2 対話中のユーモア表現における研究課題の設定

今後より普及していくと予想される対話型エージェントは、未だその対話の多くが無機質なものである。[1]では、親和的な関係の形成に笑いが欠かせないとされており、ユーモアと親密性の関係性が窺える。このことから、エージェントが対話の中にユーモアを交えられるようにすることで、ユーザはエージェントに対して親しみを持つことができると考えられる。笑いを通してユーザと良好なコミュニケーションを築こうとしている事例はいくつか存在する [12][13][14][15][16][17][18]。しかし、このような研究はエージェントによる一方的なボケをユーザに見せることで笑いの提供を試みているため、ユーザとエージェントの間で対話などの直接的なコミュニケーションは発生していない。

[19], [20], [21], [22]は対話中でのユーモアの表現を試みている。しかし、これらの事例では次のような問題が挙げられる。[19]では、ユーザからの入力に対し、エージェントが突飛な発言を行うことがある。例えば、ユーザからの「メガネをかけます。」という入力に対し、「中破メガネですか？」という出力を行なっている。このような出力の場合、ユーザは納得感が得られず、エージェントの発話をユーモアとして受容できないことがある。[20], [21], [22]については、ユーモア表現を行う場合、あらかじめユーザが入力する文と、それに対応する応答文が決められていると思われるため、ユーザが入力できる文の自由度が低く、すぐに飽きられてしまう可能性がある。

上記の問題をふまえ、ユーザがエージェントに対して親しみを感じられるようにするために、ユーザがエージェントと対話を行うシーンにおいて、下記を研究課題として設定する。

課題1：エージェントが対話の中でユーモアを交えられるようにする

課題2：ユーザが受容できるようなユーモアを含む応答文を生成する

課題3：ユーザが入力できる内容を制限しないようにする

第4章 ボケて返すエージェントの提案

4.1 アプローチ

3.2節で設定した課題を達成するために、学術的・実践的それぞれの観点に基づいてアプローチの検討を行う。

学術的観点では、認知科学研究者の多くがユーモア生起に不可欠と主張している「不適合の認知」に着目した [23]。不適合とは、典型的には期待と実際とのズレとして定義されている。この不適合理論には立場の異なる2つのモデルが存在しており、それぞれ「不適合モデル」、「不適合-解決モデル」と呼ばれている。例えば、漫才・落語・4コマ漫画のようなユーモアは、「不適合-解決モデル」で説明できる [24][25][26][27][28]。[28]ではこのモデルにおける不適合について、[29]を参考に、下記のように説明している。

Shultz & Horibe (1974) による不適合 - 解決モデルの検証実験で用いられた以下のジョークを例に、2モデルにおける不適合概念の相違について説明する。

母親：先生、すぐに来てください！赤ちゃんが万年筆を飲み込んでしまいました。

医師：すぐにそちらへ行きます。その間、何をしていたらいいかわかりますか？

母親：ペンを使います。

Shultz & Horibe (1974) によれば、このジョークでは、最後の母親の言葉が予測と異なる不可解なものであり、不適合を生じさせている。ここで、母親が医師の言葉を“万年筆の代わりに何を使うか”と解釈したと考えると、母親の言葉の理由が明らかになり、この不適合は解決される。このとき解決される“論理の欠如”としての不適合が、不適合 - 解決モデルにおける不適合である。

すなわち、「不適合-解決モデル」とは、通常はまったく異種であり、関連がないと思われる思考、概念、状況（不適合）が、納得できる方法で結び合わされる（解決される）ことでユーモアを生起させるとする理論である。

実践的観点では [30] で定義されているボケの作り方に着目した。これは、ある音から連想される言葉の中から、できるだけ意味に差がある2つの言葉を選び、それらをタイミングに合わせて使うというものである。

4.2 ボケて返すエージェントの提案

4.1節より、ユーザが単語を入力すると、あらかじめ用意した出力候補単語群の中から、入力した単語（以降、入力単語）に対し、意味が離れており、音が近く、ユーザが理解できる単語を聞き間違いとして聞き返すボケをするエージェントを提案する。これは、「不適合-解決モデル」に基づいてユーモアの生成を行うものである。エージェントは、ユーザ

の入力単語とエージェントが出力する単語の意味を離すことで不適合を作り、その不適合を、音が近いという関連性によって解決することでユーモアを生み出す。また、エージェントとの対話を、ユーザの発言に対してエージェントが聞き間違いとして聞き返すというシーンに設定する。これにより、[30]でボケの作り方として定義されているタイミングに合わせたボケの使用が可能になり、自然な文脈でボケることができるようになると思われる。さらに、出力する単語をユーザが理解できる単語にすることで、エージェントが突飛なボケをすることの防止を狙う。本提案手法のモデルを図4.1に示す。

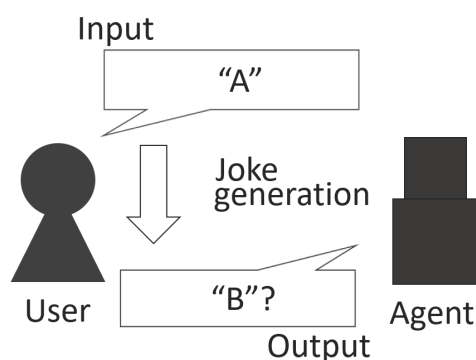


図 4.1: 提案モデル

上記の方式により、3.2節で述べた3つの研究課題が達成できると考えられる。課題1は、不適合-解決モデルと[30]を参考にボケ生成を行うことで達成する。この手法により、エージェントはユーザとの対話の中でユーモアを交えることができる。課題2は、あらかじめユーザがエージェントと対話をする際のシーンを設定し、ユーザに理解できる単語を使用したボケをすることで達成する。この手法により、自然な文脈で、突飛なボケを行わない、ユーザが受容できるユーモア表現をすることができる。課題3は、ユーザが入力を行った後に動的にボケを生成することで達成する。この手法により、あらかじめ入力に対する返答を結びつけておく必要がなくなるため、ユーザは自由に入力を行うことができる。

第5章 ボケて返すエージェントの実装

5.1 システムの全体像

提案システムはユーザの入力単語に対し、意味が離れていて、音が近く、ユーザが理解できる単語を聞き間違いとして聞き返す。システムはこれら3つの条件を Score として計算することで、出力単語を決定する。よって本章では、ユーザの入力単語に対して、どれくらい意味が離れているかを表す s_s (Semantic Score), どれくらい音が近いかを表す s_e (Edit distance Score), どれくらいユーザから理解されやすいかを表す s_f (Frequency Score) のそれぞれの算出方法と、その3つの Score をもとにボケを生成するまでの手順について述べる。実装は Python (バージョン 3.6.1) を用いて行う。

5.2 事前準備

Wikipedia 日本語記事全文 (取得日時 2017.10.20) を MeCab[31] を用いて形態素解析し、不要品詞を除去して分かち書きしたものをコーパスとする (960649 単語)。ここでの不要品詞とは IPA 品詞体系において、記号, 助詞, 助動詞, 接続詞, 副詞, 連体詞, 非自立, 代名詞, 接尾, 数, サ変・スルと分類されるものを指す。

また、コーパスを元に読み方リストと言語モデルの作成を行う。読み方リストとは、コーパス内での出現回数が 1000 回以上かつ日本語である単語の標準形, MeCab で取得した標準形の読み方 (カタカナ), 標準形の読み方をローマ字表記に変換したのち子音を削除して母音のみにしたもの, これら3つを単語ごとにまとめたリストである (図 5.1)。本研究における言語モデルとは、単語間の概念距離を計算できるよう自然言語の単語をベクトル表現したものであり、word2vec[32] を用いてコーパスから獲得する。このとき、次元数は 100, Window サイズは 15, 使用するライブラリは gensim (バージョン 3.0.1) とする。

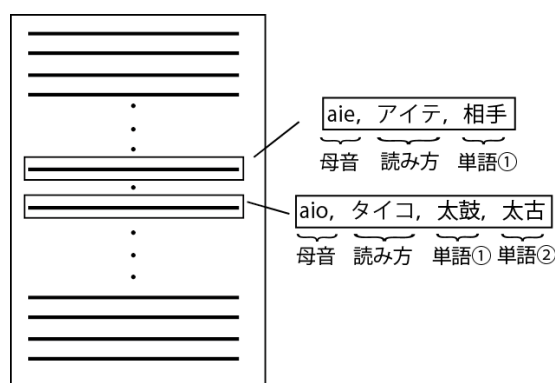


図 5.1: 読み方リスト

表 5.1: s_s が高い例・低い例

元の単語	出力候補単語	s_s
りんご	死後	1.243140
	イチゴ	0.619789
探偵	割合	1.293864
	犯罪	0.689766

5.3 各 Score 計算部の実装

本節ではユーザがシステムに単語を入力した際の、出力候補単語それぞれの各 Score (s_s , s_e , s_f) の計算方法について述べる。

5.3.1 s_s : Semantic Score

ユーザの入力単語と出力候補単語がどれくらい意味が離れているかを表す指標として、言語モデルを用いて概念距離を測ることで s_s を算出する。入力単語と出力候補単語それぞれとの概念距離を測り、その距離が遠いほど高い s_s を与える (表 5.1)。すなわち、 s_s は下記のようになる。

$$s_s = d_s \quad (5.1)$$

d_s は入力単語と出力候補単語の概念距離である。 d_s の算出には gensim ライブラリ (バージョン 3.0.1) で実装されている word2vec の similarity メソッドを用いる。similarity メソッドで導き出される類似度の値域は $-1 \sim 1$ であり、値が大きいほど単語間の概念距離が近いことを意味する。そのため、実際には得られる類似度の逆数を取得することで s_s を求める。また、このとき分母の値が 0 にならないように、similarity メソッドで導き出される類似度の値に 1 を加えてから逆数を取得する。

5.3.2 s_e : Edit distance Score

ユーザの入力単語と出力候補単語の音がどれくらい近いかを表す指標として、編集距離を測ることで s_e を算出する。編集距離とは 2 つの文字列がどの程度異なっているかを示す距離であり、1 文字の挿入・削除・置換によって一方の文字列をもう一方の文字列に変形するのに必要な手順の最小回数として定義される。入力単語と出力候補単語それぞれとの編集距離を測り、その距離が近いほど高い s_e を与える (表 5.2)。ただし、今回編集距離を測る際には、より音を近づけるため、発声において重要な要素である母音のみを参照する。すなわち、 s_e は下記のようになる。

$$s_e = \frac{1}{1 + d_e} \quad (5.2)$$

表 5.2: s_e が高い例・低い例

元の単語	出力候補単語	s_s
工場	向上	1.000000
	今週	0.333333
機会	未来	1.000000
	地方自治体	0.200000

表 5.3: s_f が高い例・低い例

単語	s_f
行こう (875205 回)	4.820000
共通 (41803 回)	4.621207
歪曲 (1012 回)	3.005181
鱗片 (1006 回)	3.002598

d_e は入力単語と出力候補単語の編集距離である。 d_e の算出には python-Levenshtein ライブラリ (バージョン 0.12.0) 内の distance メソッドを用いる。導き出される d_e の値域は $0 \sim \infty$ であり、値が大きいくほど単語間の編集距離が遠いことを意味する。

また、試行錯誤の過程で、入力単語との編集距離が同じである単語の中でも聞き間違いに聞こえる単語とそうでない単語があることが分かった。例えば、“戦闘機”という入力単語に対し、“倦怠期”、“関東”という単語の各母音のみを考慮した編集距離はともに 2 であるが、前者は聞き間違いに聞こえ、後者はそうでないと感じられる人が多かった。多くの試行を行った結果、入力単語と出力単語の頭文字の母音と末尾の文字が一致する場合は、聞き間違いに聞こえやすいということが分かった。この知見に基づき、 s_e が高い場合でも、入力単語と先頭の文字の母音・末尾の文字が異なる単語は出力候補単語から除く。

5.3.3 s_f : Frequency Score

出力候補単語がどれくらいユーザから理解されやすいかを表す指標として、Wikipedia コーパス内での各単語の出現回数を数えることで s_f を算出する。各出力候補単語の出現回数が多いほど高い s_f を与える (表 5.3)。すなわち s_f は下記ようになる。

$$s_f = \log f \quad (5.3)$$

f は出力候補単語のコーパス内での出現回数である。このとき、取りうる f の値域は $1000 \sim 3362984$ となる。下限が 1000 となる理由については後述する。ごく一部の極端に出現回数の多い単語が最終的な総合 Score に与える影響が大きくなりすぎないように、出現回数の対数をとったものを s_f とする。また、同様の理由から s_f の最大値を 4.82 と

する。この値は、コーパス内からランダムに選んだ1000単語を対象に出現回数の調査を行い、99.5パーセンタイル値である単語の出現回数に基づき決定した。 s_f の値が4.82となる出現回数は、660610回である。

一方、極端に出現回数が少ない単語はユーザに理解されない可能性が高いと考え、出現回数が1000回を下回る単語は出力候補単語から除く。

上記より、 s_f の値域は3~4.82となる(出現回数1000回の単語の s_f は3である)。

5.4 出力単語決定部の実装

人名や地名などは、ごく一部の有名なもの以外はユーザに理解されない可能性が高いと考え、MeCabで人名・地域と判定される単語は出力候補単語から除く。さらに、同様の理由から日本語以外も出力候補単語から除いた。また、エージェントの返答がオウム返しになってしまうと(表5.4)ユーモアとして受容されない可能性が考えられる。そのため、入力単語を単にひらがな、あるいはカタカナ表記にしたものは出力候補単語から除く。

上記より、ユーザからの入力単語に対する出力候補単語となる条件は下記の通りである。

- 頭文字の母音と末尾の文字が入力単語と一致する。
- コーパス内での出現回数が1000回以上である。
- MeCabで人名・地域と判定されない。
- 日本語である。
- 入力単語を、ひらがな・カタカナ表記にしたものではない。

ユーザの入力単語に対し、上記の条件を満たすコーパス内の出力候補単語の3つのScore(s_s , s_e , s_f)を算出し、それぞれ正規化処理、重みづけを行ってから合算したものを、出力候補単語それぞれの s_t (Total Score)とする。正規化は、最小値が0、最大値が1となるように行う。よって値域は s_s , s_e , s_f 共に0~1となる。すなわち s_t は下記のようになる。

$$s_t = w_s s_s + w_e s_e + w_f s_f \quad (5.4)$$

w_s , w_e , w_f は重み係数である(本稿では重み係数は全て1.0とする)。

全ての出力候補単語の s_t を算出し、 s_t が最も大きい値となる単語を出力単語とする。また、入力単語に対し、条件を満たす出力候補単語がない場合や、標準形がMeCabで判別できない、言語モデルに登録されていないといった場合には、ボケを正しく生成することができない。そのため、ボケを正しく生成できた場合と明確に区別するために、ボケを正しく生成できない場合には、“なあに?”と出力する。ここで口語的な表現を用いた理由は、ボケを正しく生成できなかった場合にも、エージェントが不自然な応答をしないようにするためである。

表 5.4: 返答がオウム返しになっている例

入力単語	出力単語
人参	にんじん
りんご	リンゴ

5.5 インタフェースの実装

構築したボケて返すエージェントを汎用的に利用するため、アプリケーションに Web API を設ける。外部アプリケーションから get/post でデータを受け取ると、生成されたボケ単語を URL エンコードして返す。レスポンスフォーマットには json を利用し、生成されたボケ単語以外にも、システムが正常に動いたかを示す result, 生成された単語の wikipedia 記事の url, 生成されたボケの s_t を示す score も返す。wikipedia 記事の url はディスプレイを用いてボケを視覚的に説明する際に使用する。json フォーマットは表 5.5 に示す。さらに、シャープ社製のスマートフォン型ロボット, RoBoHoN[33] と本 API を用いて、音声による入出力も実現する。

表 5.5: json フォーマット

JSON key	データ型	データ例
		success
result	文字列	error NG
message	文字列	え、戦国時代?
url	文字列	https://ja.wikipedia.org/wiki/%e6%88%a6%e5%9b%bd%e6%99%82%e4%bb%a3
score	文字列	0.76(0.00~1.00 の実数値)

第6章 評価実験

6.1 実験の目的

本研究は、ユーザがエージェントと対話をするシーンにおいて、エージェントが対話の中でユーモアを交えられるようにすることで、ユーザは親しみを持つことができるという仮説(3.2節参照)に基づき、ユーザの発言の一部に対して聞き間違ふボケをするエージェントを提案するものである。提案方式の有効性を確認するためには、実際にユーザに提案方式を取り入れたエージェントと対話を行ってもらい、検証を行うことが必要である。そこで本稿では、提案方式が生成したボケに対してユーモアを感じるか、それによってユーザはエージェントに対して親しみやすくなるのかを、対話を模した実験を行うことにより検証する。また、提案方式はボケを生成するにあたって、概念距離・編集距離・出現回数という3つの要素を参考にしている。これら3つの要素が面白さとどのように関連しているかも検証することで、提案方式によるユーモア生成を詳細に分析する。

6.2 実験の概要

本実験の被験者は20代の学生15名(男14, 女1)である。実験は下記のような、Baselineと、出力単語を選定する方法が異なる3つの提案方式(P1, P2, P3)を用いて行った。

- Baseline
入力単語に対して、コーパス内の出現回数が1000回以上である単語をランダムに返す。このとき s_s , s_e , s_f はすべて考慮しない。3つのScoreによる生成されるボケへの影響を調べるため、すべて考慮しないこの方式を、本研究におけるBaselineとする。
- P1
入力単語に対して最も s_s が高くなる出力候補単語を返す。出力候補単語は、コーパス内での出現回数が1000回以上である単語とする。すなわち、P1は概念距離を考慮して出力単語を決定する方式である。
- P2
入力単語に対して最も s_e が高くなる出力候補単語を返す。出力候補単語は、コーパス内での出現回数が1000回以上である単語とする。すなわち、P2は編集距離を考慮して出力単語を決定する方式である。
- P3
入力単語に対して s_t が最も高くなる出力候補単語を返す。出力候補単語は、コーパス内での出現回数が1000回以上である単語とする。すなわち、P3は概念距離、編集距離、出現回数、全てを考慮して出力単語を決定する方式である。

表 6.1: 被験者への質問一覧

Q1	出力された返答にユーモアを感じたか (5:とても感じた ~ 1:全く感じなかった)
Q2	出力された返答は意外だと感じたか (5:とても感じた ~ 1:全く感じなかった)
Q3	エージェントのボケに納得感は得られたか (5:とても納得できた ~ 1:全く納得できなかった)
Q4	エージェントに親しみを感じたか (5:とても感じた ~ 1:全く感じなかった)

6.3 実験の手順

被験者には 6.2 節に示した 4 つのパターンをそれぞれ使用してもらい、アンケートに 5 段階のリッカート尺度で回答してもらった。被験者への質問と選択肢を表 6.1 に示す。

被験者には、各パターンを使用してそれぞれ、あらかじめ実験者が用意した 10 個の単語を入力し、出力を確認してもらった。このとき用意した単語については、コーパス内の出現回数が 1000 回以上である単語からランダムに選ばれたものを使用した。実験に用いた単語の一部を表 6.2 に示す。

表 6.2: 実験に用いた単語例

解読	撤回	余剰	課金
精製	メモリー	帰省	表彰式
本物	正しい	心理学者	類似
ミニカー	ボタン	合併	不向き
地底	定期的	個人	骸骨

実際のエージェントとの対話を行うシーンに近づけるために、入出力の形式に工夫を行った。入力は、「“単語”について教えて?」とすることで、ユーザからエージェントへの質問を行う形式にした。出力は、「え、“単語”?」とし、ユーザからの質問をエージェントが聞き間違えたかのような形式にした。

次に順序効果を相殺するために、被験者が使用するパターンが変わるたびに入力する単語の順番をランダムに入れ替え、被験者ごとに使用するパターンの順番についてもランダムに入れ替えるという工夫も行った。このとき、システムの仕様を聞いたことによって生じる先入観をなくすため、各パターンの出力単語の生成方法は被験者には伝えなかった。

さらに、被験者が変わると、被験者が入力する 10 個の単語をランダムに選び直すという工夫も行った。これにより、実験全体に用いる単語の数が増え、入力された各単語が実験結果に与える影響を少なくできると考えられる。

なお、入力についてはキーボードを用い、出力については画面にテキストを表示した。Q1~Q3 については 1 回の入出力が終わるごとに回答してもらい、Q4 については、各方式に対して 10 回の入出力を終えるたびに回答してもらった。

表 6.3: 各パターンの動作例

入力した文	Baseline 出力例	P1 出力例	P2 出力例	P3 出力例
魚について教えて？	え、王政？	え、さいたまスーパーアリーナ？	え、あだ名？	え、刀？
不便について教えて？	え、脈？	え、出身？	えっ、右舷？	え、主演？
投稿者について教えて？	え、指標？	え、冒険者？	え、功労者？	え、労働者？

6.4 実験結果・考察

実験時に被験者が入力した文とその単語に対する各パターンの出力例は表 6.3 のとおりである。Q1 の回答結果を図 6.1 に示す。“出力された返答にユーモアを感じたか”という質問に対し、“とても感じた”または“感じた”と回答した被験者は、Baseline では 12 %，P1 では 22 %，P2 では 50 %，P3 では 44 %であった。Q2 の回答結果を図 6.2 に示す。“出力された返答は意外だと感じたか”という質問に対し、“とても感じた”または“思う”と回答した被験者は、Baseline では 41 %，P1 では 36 %，P2 では 53 %，P3 では 49 %であった。Q3 の回答結果を図 6.3 に示す。“エージェントのボケに納得感は得られたか”という質問に対し、“とても納得できた”または“納得できた”と回答した被験者は、Baseline では 5 %，P1 では 14 %，P2 では 48 %，P3 では 50 %であった。Q4 の回答結果を図 6.4 に示す。“エージェントに親しみを感じたか”という質問に対し、“とても感じた”または“思う”と回答した被験者は、Baseline では 7 %，P1 では 7 %，P2 では 60 %，P3 では 53 %であった。

6.4.1 Semantic Score を考慮した効果に関する考察

P1 と Baseline を比較する。P1 と Baseline の各質問に対する回答結果に対し、それぞれ Wilcoxon の符号順位検定を行うと、Q1、Q3、Q4 において 1 %水準で有意差を確認できた。ここから、P1 は Baseline よりもユーモアの感じやすさ、ボケに対しての納得感、親しみの感じやすさにおいて優位なアプローチであることが示唆される。また、P1 は、意外性については Baseline と同程度であった。以上より、ユーザにユーモア・納得感・親しみを感じさせるために、ボケの生成時に概念距離を考慮することは効果があったと判断できる。

6.4.2 Edit distance Score を考慮した効果に関する考察

P2 と Baseline を比較する。P2 と Baseline の各質問に対する回答結果に対し、それぞれ Wilcoxon の符号順位検定を行うと、全ての質問において 1 %水準で有意差を確認できた。ここから、P2 は Baseline よりもユーモアの感じやすさ、ボケに対しての意外性、ボケに対しての納得感、親しみの感じやすさにおいて優位なアプローチであることが

示唆される。以上より、ユーザにユーモア・意外性・納得感・親しみを感じさせるために、ボケの生成時に編集距離を考慮することは効果があったと判断できる。

6.4.3 全ての Score を考慮した効果に関する考察

P3 と Baseline を比較する。P3 と Baseline の各質問に対する回答結果に対し、それぞれ Wilcoxon の符号順位検定を行うと、全ての質問において 1%水準で有意差を確認できた。ここから、P3 は Baseline よりもユーモアの感じやすさ、ボケに対しての意外性、ボケに対しての納得感、親しみの感じやすさにおいて優位なアプローチであることが示唆される。以上より、ユーザにユーモア・意外性・納得感・親しみを感じさせるために、ボケの生成時に編集距離・概念距離・出現回数の 3 要素を考慮することは効果があったと判断できる。

次に、P3 と P1 を比較する。P3 と P1 の各質問に対する回答結果に対し、それぞれ Wilcoxon の符号順位検定を行うと、全ての質問において 1%水準で有意差を確認できた。ここから、P3 は P1 よりもユーモアの感じやすさ、ボケに対しての意外性、ボケに対しての納得感、親しみの感じやすさにおいて優位なアプローチであることが示唆される。以上より、ユーザにユーモア・意外性・納得感・親しみを感じさせるためには、概念距離だけでなく、編集距離・出現回数も考慮した方がより効果があったと判断できる。

しかし、P3 と P2 を比較したところ、どの質問の回答結果に対しても有意差が見られなかった。しかし、6.4.1 項にあるように、ボケの生成時に概念距離を考慮することが、よりユーモアを感じづらくする要因であるとは考えにくい。よって、今後さらに被験者の人数を増やして追加実験を行い、よりユーモアを感じさせやすくできる要素を検証していきたい。

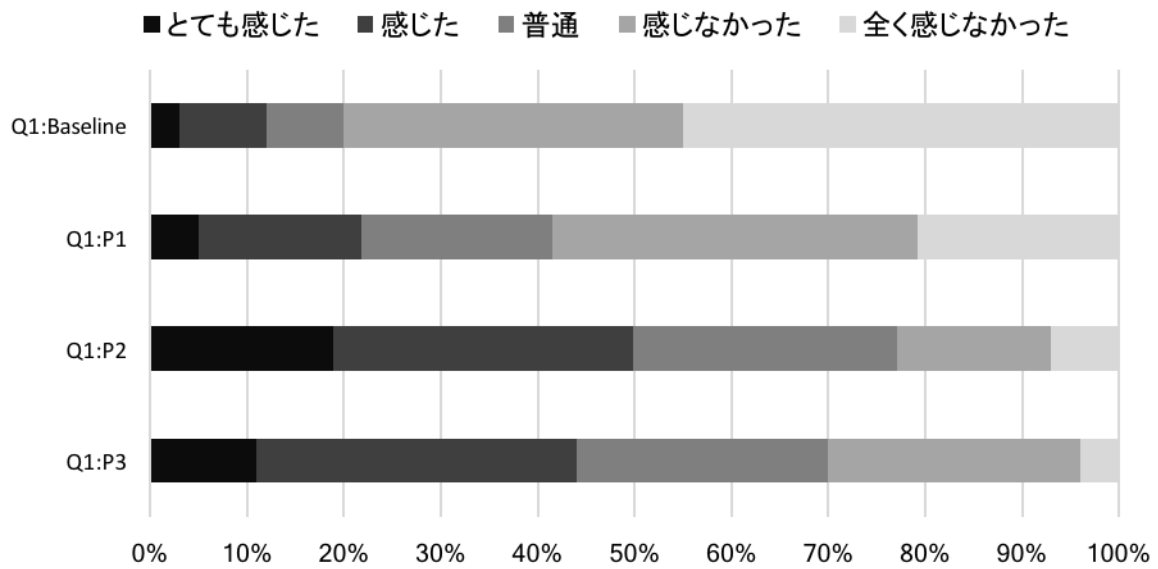


図 6.1: Q1 の回答 (N=15)

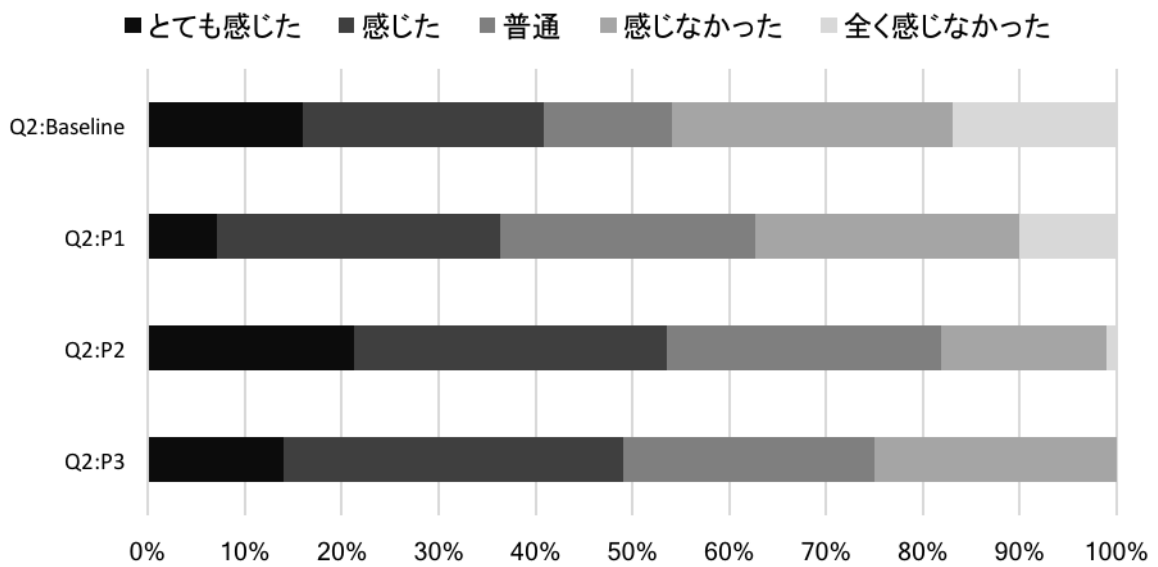


図 6.2: Q2 の回答 (N=15)

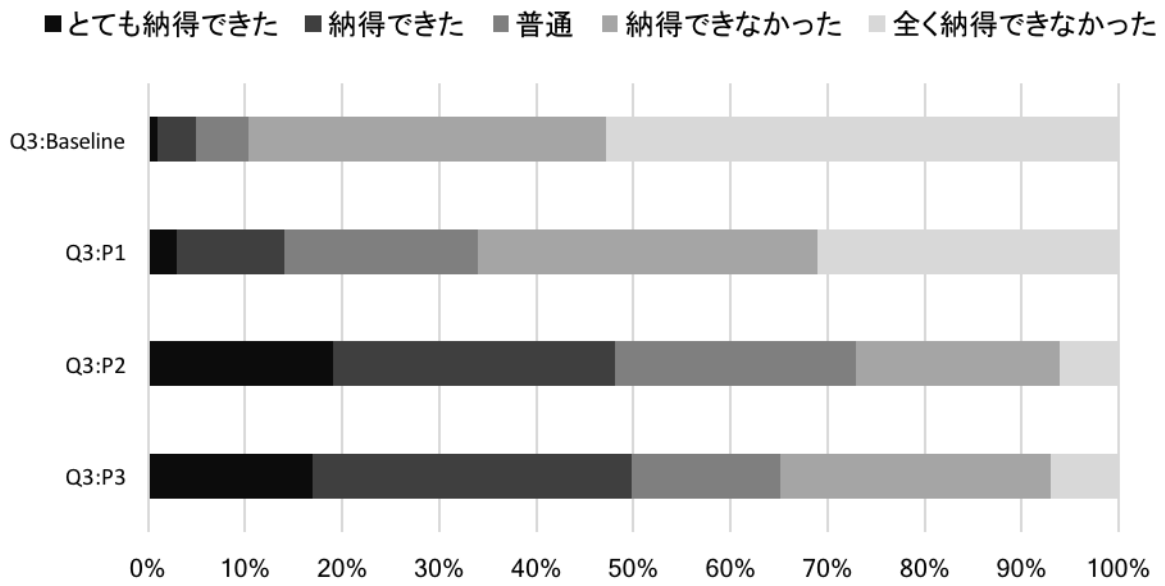


図 6.3: Q3 の回答 (N=15)

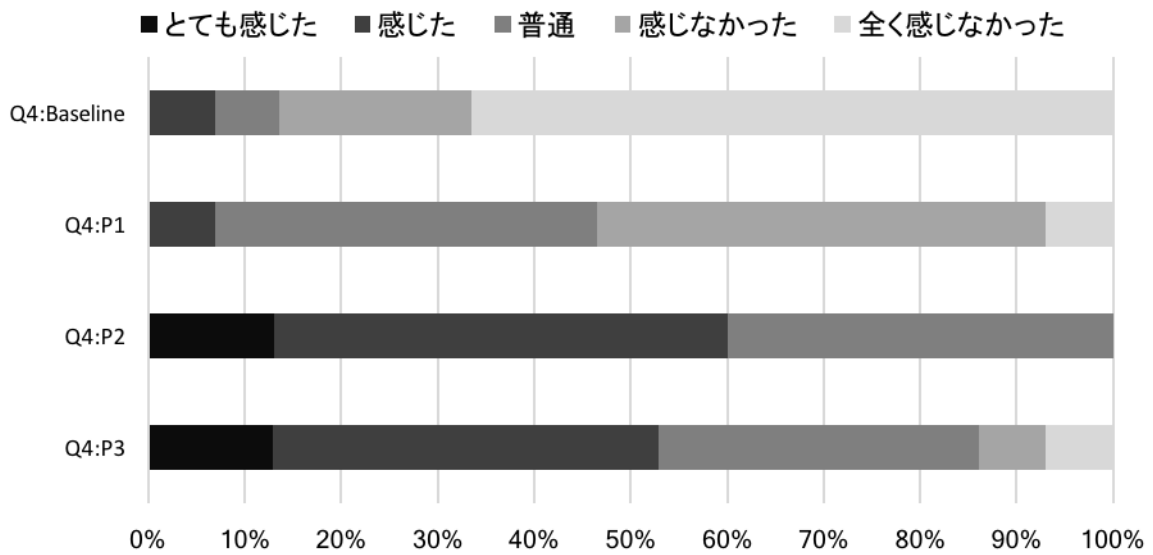


図 6.4: Q4 の回答 (N=15)

第7章 結論

本稿は、普及した対話型エージェントとの無機質な対話に親しみを持てず、ユーザがエージェントを受け入れられないという問題の解消を狙ったものである。この問題を解決するためにユーモアと親しみの関係性に着目し、エージェントにユーモアのある対話をさせることで、ユーザは親しみを持つことができるのではないかという仮説を立てた。この仮説に基づき、ユーモアを作成する手法として、認知科学領域で支持されている不適合-解決モデルや [30] のボケの作り方を参考に、ボケて返す対話型エージェントの試作を行った。これは、ユーザが単語を入力すると、その入力単語に対し、意味が離れており、音が近く、ユーザが理解できる単語を、聞き間違いとして聞き返すボケを行うエージェントである。実験の結果、ユーモアと親しみの関連性の観点では、エージェントにユーモアのある対話をさせることで、ユーザは親しみを持つ傾向にあった。ボケ生成手法の観点では、一定の納得感・意外性を持つボケに対して、ユーザはユーモアを感じる傾向にあった。加えて、一定の納得感を得られたことから、提案手法は突飛な回答は行わず、ユーザにとって受け入れやすいボケを生成できたと考えられる。また、東京ゲームショウ 2017 [34] にて本提案の展示を行った。その際、多くのユーザを笑わせることができたことから、提案システムは一般ユーザにも十分な受容性があると考えられる。

提案システムの改善案としては、生成されるボケの面白さの向上を図ることが第一に考えられる。この問題は最終的な Score 算出時の重み係数の値変更によって、より面白いボケが生成されること期待される。また、単語のみならず、文の入力に対してもボケて返すことができるエージェントを考案することで、より様々なシーンで応用することが可能になると考えられる。文の入力に対するボケの作り方として、例えば、入力文の中から聞き間違えると一番面白い単語を特定し、その単語を聞き間違えて返事をするといった方法や、ボケる単語を選定する際に文脈を考慮するといった方法が考えられる。さらに、今回は日本語に限定した実装・実験しか行っていないが、他国語への適応も行うことで、より多くのユーザに受容されることが期待される。現在、ごく一部の処理（母音・子音処理等）のみ日本語を前提としているが、基本的には言語非依存のアルゴリズムであるため、比較的少ない手間で他国語対応は実現可能であると考えている。

本研究の期待される活用法として、ユーザとの良好なコミュニケーションがパフォーマンスや継続利用率の向上につながる場面での活用が期待される。例えば、NTT レゾナント社の教えて goo [35] にはユーザからの質問を AI が自動で応答する機能がある。このような質問掲示板で、AI がボケた返答をしてユーザに親しみを抱かせるといった活用法が考えられる。

参考文献

- [1] 井上宏. 「笑い学」研究について. 笑い学研究, No. 9, pp. 3–15, 2002.
- [2] Graeme Ritchie. Can computers create humor? In *AI MAGAZINE*, Vol. 30, pp. 71–81. Association for the Advancement of Artificial Intelligence, 2009.
- [3] 武田風太, 鈴木慎太郎, 矢島敬士. 対話型キャラクタエージェントを用いたコミュニケーションによる高齢者の見守りシステム支援. 情報処理学会第 78 回全国大会 2016, 第 1 巻, pp. 401–402, 2016.
- [4] 野中裕子, 酒井洋一, 安田清, 林佑樹, 中野有紀子. 語りかけエージェントとの会話における韻律情報の分析. 情報処理学会第 75 回全国大会講演論文集 2013, 第 1 巻, pp. 179–180, 2013.
- [5] 小倉拓人, 徳永隼人, 松村冬子, 原田実. 進行役と鑑賞者のエージェントを用いた対話型鑑賞システム. じんもんこん 2015 論文集, 第 2015 巻, pp. 43–50, 2015.
- [6] 藤堂祐樹, 西村良太, 山本一公, 中川聖一. 複数の対話エージェントを用いた音声対話システムの分析と評価. 情報処理学会研究報告ヒューマンコンピュータインタラクション (HCI) , 第 2013-HCI-151 巻, pp. 1–7, 2013.
- [7] 片上大輔, 小林優, 大澤博隆, 稲葉通将, 篠田孝佑, 鳥海不二夫. 擬人化エージェントを用いた人狼対戦システムの開発. ゲームプログラミングワークショップ 2014 論文集, 第 2014 巻, pp. 133–134, 2014.
- [8] 松原和也, 林勇輔, 水久保勇記, 鶴見悠子, 垂水浩幸. 携帯電話向けインタラクティブエージェントと npc への応用. 情報処理学会研究報告エンタテインメントコンピューティング (EC) , 第 2005 巻, pp. 71–78, 2005.
- [9] 久保田秀和, 黒橋禎夫, 西田豊明. 知識カードを用いた分身エージェント. 電子通信情報学会論文誌, No. 8, pp. 600–607, 2003.
- [10] 廣瀬有紹, 加藤誠巳. ドライバの退屈・眠気回避を目的とした会話型 cg エージェントに関する検討. 情報処理学会第 65 回全国大会講演論文集, 第 2003 巻, pp. 411–412, 2003.

- [11] 近藤亮太, 畑中聖二, 加藤誠巳. 余暇の過ごし方を提案する会話型エージェントシステムに関する検討. 情報処理学会第 69 回全国大会講演論文集, 第 2007 巻, pp. 543–544, 2007.
- [12] 伊勢崎隆司, 小林明美, 望月崇由, 山田智弘. 笑い感情を誘起するロボットインタラクションの検討. 情報処理学会研究報告グループウェアとネットワークサービス (GN), 第 2017-GN-100 巻, pp. 1–5, 2017.
- [13] 真下遼, 梅谷智弘, 北村達也, 灘本明代. 文の感情を考慮した漫才ロボット台本自動生成手法の提案. In *DEIM Forum*, 2015.
- [14] 中谷仁, 岡夏樹. ロボットの日常会話におけるユーモア生成の試み. 人口知能学会 2009 年全国大会論文集, 2009.
- [15] 吉田裕介, 萩原将文. 漫才形式の対話文自動生成システム. 日本感性工学会論文誌, Vol. 11, No. 2, pp. 265–272, 2012.
- [16] G. Lessard and M. Levison. Computational modelling of linguistic humour: Tom swifties. In *Association for Literary and Linguistic Computing (ALLC) and the Association for Computers and the Humanities (ACH)*, pp. 175–178, 1992.
- [17] O. Stock and C. Strapparava. Getting serious about the development of computational humor. In *International Joint Conference on Artificial Intelligence (IJCAI)*, Vol. 3, 2003.
- [18] Dongkeon Lee, Seung-Ho Han, Kyo-Joong Oh, and Ho-Jin Choi. A temporal community contexts based funny joke generation. In *Mobile Data Management (MDM)*, 2017.
- [19] 藤倉将平, 小川義人, 菊池英明. ユーモア発話の自動生成における単語間類似度導入によるユーモア受容性の向上. HAI シンポジウム, 2014.
- [20] Apple inc. : Siri, <https://www.apple.com/ios/siri/> (last visited on 2018/1/10).
- [21] Softbank. : Papper, <https://www.softbank.jp/robot/> (last visited on 2018/1/10).
- [22] Google inc. : Google assistant, <https://assistant.google.com/> (last visited on 2018/1/10).
- [23] R. A. Martin. *The Psychology of Humor*. Elsevier Academic Press, 2007.

- [24] S. Coulson and R. F. Williams. Hemispheric asymmetries and joke comprehension. *Neuropsychologia*, Vol. 43, No. 1, pp. 128–141, 2005.
- [25] A. C. Samson, O. Hempelmann, C. F. Huber, and S. Zysset. Neural substrates of incongruity-resolution and nonsense humor. *Neuropsychologia*, Vol. 47, No. 4, pp. 1023–1033, 2009.
- [26] T. R. Shultz. The role of incongruity and resolution in children's appreciation of cartoon humor. *Experimental Child Psychology*, Vol. 13, No. 3, pp. 456–477, 1972.
- [27] J. M. Suls. Cognitive processes in humor appreciation. In *In Handbook of Humor Research*, pp. 39–57. Springer-Verlag New York, 1983.
- [28] 伊藤大幸. ユーモアの生起過程における論理的不適合及び構造的不適合の役割. 認知科学, 第 17 巻, pp. 297–312, 2010.
- [29] T.R. Shultz and F. Horibe. Development of the appreciation of verbal jokes. In *Developmental Psychology*, Vol. 10, pp. 13–20, 1974.
- [30] 織田正吉, 野村雅昭. シャレ・ダジャレ学事始 (ことはじめ). 笑い学研究, 第 32 巻, pp. 237–242, 1999.
- [31] Mecab: Yet another part-of-speech and morphological analyzer, <http://taku910.github.io/mecab/> (last visited on 2018/1/10).
- [32] Tomas Mikolov, Kai Chen, Greg Corred, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *In Proceedings of Workshop at ICLR*, 2013.
- [33] ロボホン開発環境, <https://robohon.com/sdk/> (last visited on 2018/1/25).
- [34] 東京ゲームショウ 2017, <http://expo.nikkeibp.co.jp/tgs/2017/> (last visited on 2018/1/10).
- [35] 教えて goo, <https://oshiete.goo.ne.jp> (last visited on 2018/1/10).

研究業績

査読付き国内会議

- (1) 瀧田航平, 鈴木奨, 呉健朗, 堀越和, 中辻真, 宮田章裕: キャラクタ性を持ったボケて返す対話型エージェントの基礎検討, 情報処理学会グループウェアとネットワークサービスワークショップ 2017 論文集, Vol.2017, pp. 1-6 (2017年11月).
-

研究会・シンポジウム

- (1) 呉健朗, 鈴木奨, 瀧田航平, 堀越和, 中辻真, 宮田章裕: 対話型エージェントにおけるボケるインタラクションの実装, 情報処理学会インタラクション 2018 (2018年3月発表予定).
 - (2) 長岡大二, 中原涼太, 小林舞子, 鈴木奨, 呉健朗, 宮田章裕: 文脈を考慮してボケるエージェントの基礎検討, 情報処理学会インタラクション 2018 (2018年3月発表予定).
 - (3) 呉健朗, 鈴木奨, 瀧田航平, 中原涼太, 中辻真, 宮田章裕: ボケて返す対話型エージェントの実装, サイバースペースと仮想都市研究会第20回シンポジウム (2017年12月).
 - (4) 鈴木奨, 呉健朗, 瀧田航平, 堀越和, 中辻真, 宮田章裕: ボケて返す対話型エージェントの基礎検討, 情報処理学会研究報告 (GN) グループウェアとネットワークサービス, Vol.2017-GN-102, No.3, pp.1-6 (2017年5月).
-

メディア掲載

- (1) AI、笑いを学ぶ, 国際イベントニュース, 第25号16面 (2017年10月10日).
 - (2) 「電話番号教えて」「え、ケンカ番長?」日大生が作った“ボケるAI”, ITmedia, <http://www.itmedia.co.jp/news/articles/1709/21/news093/> (2017年9月21日)
 - (3) 「ボケるAI」初公開。スゴイ技術で、あえて間違える【TGS2017】, 日経トレンドイネット, <http://trendy.nikkeibp.co.jp/atcl/pickup/15/1003590/092101200/> (2017年9月21日).
-

展示

- (1) ボケるAI, 東京ゲームショウ 2017 (2017年9月21~22日).
-

受賞

- (1) 情報処理学会グループウェアとネットワークサービスワークショップ 2017 ベストペーパー賞, キャラクタ性を持ったボケて返す対話型エージェントの基礎検討, 受賞者: 瀧田航平, 鈴木奨, 呉健朗, 堀越和, 中辻真, 宮田章裕 (2017年11月).
-