

# キャラクターを持った 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

瀧田 航平

# 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。このようなエージェントの会話相手としての価値や娯楽性を高める手段の1つとして、エージェントにキャラクター性を付与させることが考えられる。また、ユーモア会話を行うエージェントにキャラクター性を持たせることによって、ユーモアの幅を広げる効果やユーザの嗜好に合ったユーモアを提供する効果が生じ、サービスクオリティが向上することが期待できる。本稿では、ユーザの入力に対して、ボケて返すエージェントのプロトタイプシステムにキャラクター性を持たせることを提案する。これを実現するために、エージェントが特定の分野に偏った発言をすることで、ユーザはそのエージェントに対して当該分野に関するキャラクター性を感じやすくなるという仮説を立て、この仮説に基づくシステムの実装、実験を行なった。検証実験の結果、提案方式によるエージェントはキャラクター性が感じられることが確認できた。本稿の貢献は次のとおりである。

- ユーモアのある聞き間違いを行うシステムにキャラクター性を付与する方法を提案したこと。
- 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、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。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。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]、エージェントにユーモアの概念を導入することで、ユーザはエージェントに親しみを感じやすくなる可能性がある。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

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できるよう、ボケた返答を行う対話型エージェントがある。これは、認知科学の分野において支持されている不適合-解決モデルを参考に、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機能がある。例えば、“A について教えて?”とユーザが質問をすると、“え、B?”と、ユーモアのある聞き間違いをする。本稿では、このシステムに任意のキャラクターを持たせることを目指す。エージェントにキャラクターを持たせることによって、同一のシステムを使用しているエージェントでも異なった聞き間違いを行い、ユーモアの幅が広がるようになると考えられる。さらに、エージェントにユーザの好みや、様々な場所や場面に合わせたキャラクターを持たせることで、エージェントのサービスオリエンテーションの向上が期待される。そこで、キャラクターを持ったユーモアのある対話型エージェントのプロトタイプシステムを実現し、実験を行い、効果を検証することを本稿の目的とする。

## 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、エージェントにキャラクターを持たせ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と対話中のキャラクター創出におけ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やキャラクター付け手法について検討し、キャラクターを持ったボケて返す対話型エージェントの提案を行う。

5章では、ポケ生成を行う際に参考にする4つのスコアの計算方法を中心に、キャラクターを持ったポケで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、実験の目的や手順、実験結果から得られた知見について述べる。

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。2.3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。

## 2.3 エージェントにキャラクター性を持たせる研究事例

本節ではエージェントにキャラクター性を持たせた研究事例について述べる。



エージェントにキャラクターを持たせる取り組みの事例として、[23]が挙げられる。[23]では、話者のキャラクターに寄与する言語表現の基礎的分析を行っている。実験の結果、エージェントが発言する語彙や、単語の使用頻度にもキャラクターらしさが表れるという知見が得られている。

また、エージェントに性格付けを行うシステムやエージェントの発話にキャラクターを付与させるシステムがある [24][25]。エージェントに性格付けを行うシステムの例としては [24]が挙げられる。この研究では、性格付けにより、エージェントに対する“好ましさ”が変化することが報告されている。エージェントにキャラクターを付与させるシステムの例として [25]は、音声生成を前提にテキストレベルでのキャラクタ付与方法を提案し、ユーザが擬人化インタフェースに対して抱く印象の制御方法を提案している。

他にも、エージェントに固有の見た目を持たせることで、キャラクター性の創出を行うシステムも多く発表されている [26][27]。[26]では、ユーザとの対話時にスマートフォン上で3Dキャラクタを通すことでキャラクター性を表現するシステムの提案を行っている。[27]は、ユーザが写真共有システムに投稿した写真に特定の姿を付与することで、キャラクター性を持たせる試みである。このシステムにより、ユーザ間のコミュニケーションを促進させる効果が確認されている。

## 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いる。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とエージェントにキャラクターを持たせる際に生じる問題点について述べる。

### 3.1.1 対話中のユーモア表現における問題点

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、ユーモアと親しみの関係性が窺える。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。

### 3.1.2 対話中のキャラクター創出における問題点

対話型エージェントにキャラクターを持たせることで、様々な利用シーンにおいてサービスクオリティが向上すると考えられる。例えば、時代劇が好きなユーザと対話をするシーンにおいて、エージェントに侍キャラクターを持たせることで、ユーザの興味を引くことができる。また、複数のエージェントとの対話において、エージェントごとに異なる

キャラクターを持たせることで、ユーザは幅広いジャンルの対話が可能になるため、よりユーザに受け入れられるようになると考えられる。

エージェントにキャラクターを持たせようとする事例はいくつか存在する [25][24]。

[25]では、キャラクターの形成に有用な特定の言語表現や言葉遣いを抽出し、ある文章をキャラクターを感じさせる文章へと変換させるシステムを提案している。そのため、任意のキャラクターの創出は可能であるが、一方的にテキストを表示するのみで、ユーザとの対話を行えていない。また、文章レベルでのキャラクター性を前提としており、評価実験の際に3-5節からなる150文字程度の文章を使用するという制約を用いているためユーザとエージェントとの自然な対話の中でキャラクター性を感じられるかは検証していない。

[24]では、エージェントの個性を、人間の性格分析法のひとつであるエゴグラムに基づいて性格付けを行っている。エゴグラムとは、人間の性格を、CP(Critical Parent):批判的な親の自我状態、NP(Nurturing Parent):養護的な親の自我状態、A(Adult):大人の自我状態、FC(Free Child):自由奔放な子供の自我状態、AC(Adapted Child):順応した子供の自我状態の5つの自我の状態に分けて分析する手法である。[24]は、この5つの自我状態はそれぞれ長所と短所が異なる性向をもつという特徴を生かし、それぞれの自我状態を特化させた5種類の異なる個性を実現している。しかし、この手法では5種類の特定のキャラクター性は実現できるが任意のキャラクター性の創出はしていない。

## 3.2 研究課題の設定

3.1.1節、3.1.2節の問題をふまえ、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**：ユーザが対話の中でエージェントにキャラクター性を感じることができる

**課題3**：任意のキャラクター性を実現できる

## 第4章 キャラクタ性を持ったボケて返す 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、ユーモア生起、キャラクタ性付け手法の2点に分けてアプローチの検討を行う。

### 4.1.1 ユーモア生起について

ユーモア生起について、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。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[28][29][30][31][32]。[32]ではこのモデルにおける不適合について、[30]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、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[33]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。

### 4.1.2 キャラクタ性付け手法について

キャラクタ性を持つ言語表現として、金水の役割語 [34] がよく知られている。役割語とは、ある特定の言葉遣いから特定の人物を思い浮かべることができる言葉遣いと定義したものである。例えば、A は老人のキャラクタ、B は上品な女性のキャラクタを表現したものである [34]。

A: 「そうじゃ、わしが知っている」

B: 「そうですわよ、私が存じておりますわ」

このように、同じ発話内容でも言葉遣いでキャラクタ像を想像できる。役割語を用いた研究の多くは文章表現を前提としている [25][24]。

しかし、本提案手法の元となる、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では、シチュエーションをユーザの入力した1単語の聞き間違いに限定している。そのため、本提案手法であるエージェントにキャラクタ性を持たせるシステムにおいても、1単語でキャラクタをユーザに連想させなければならない。一方、[23]では、エージェントが発言する語彙や、単語の使用頻度にもキャラクタらしさが表れるという知見がある。そこで、これらの考えをもとに、エージェントが特定の分野に偏った単語ばかりを出力すると、ユーザはそのエージェントに対してキャラクタ性を感じやすくなるという仮説を立てた。例えば、エージェントが旅行に関する単語ばかり出力すれば、ユーザはそのエージェントが旅行好きな性格であることを想像できる、という考えである。

## 4.2 キャラクタ性を持った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、あらかじめ指定したキャラクタ性にあっ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。さらに、任意のキャラクタ性を創出することによって、同一のシステムを使用しているエージェントでも出力単語が変化し、ユーモアの幅が広がるようになると考えられる。また、ユーザが任意に出力単語の分野を設定することで、ユーザはエージェントに任意のキャラクタ性を持たせることが可能になる。本提案手法のモデルを図4.1に示す。

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[33]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、出力単語を特定の分野に偏らせることで達成する。この手法により、ユーザがエージェントに対してキャラクタ性を感じることができる。課題3は、カテゴリ単語をユーザが任意に指定できることで達成する。この手法により、任意のキャラクタ性を創出することができる。

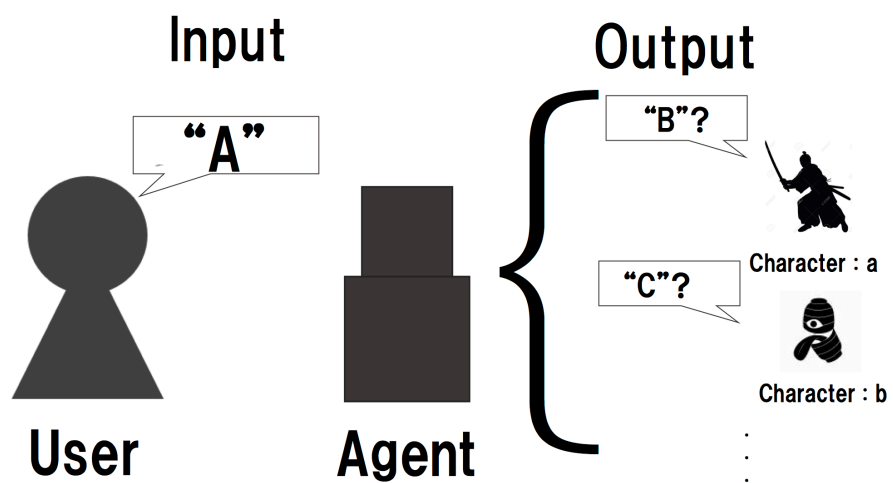


図 4.1: 提案モデル



## 第5章 キャラクタ性を持ったボケて返す 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、ユーザがあらかじめ、旅行、野球、妖怪といった特定の分野を象徴する単語（以降、カテゴリ単語）を決定する。このカテゴリ単語と概念距離が近く、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4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。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), どれくらいカテゴリ単語と意味が離れているかを表す  $s_c$  (Category Score) のそれぞれの算出方法と、その4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[35] 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に自然言語の単語をベクトル表現したものであり、word2vec[36]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。

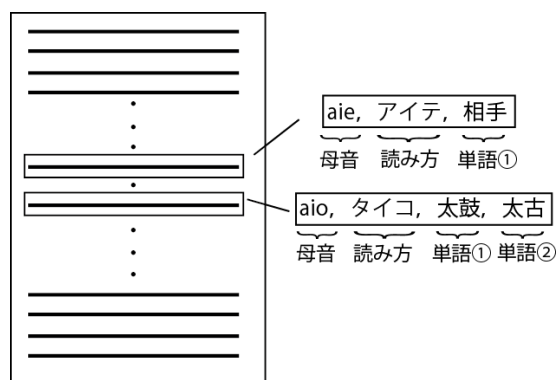


図 5.1: 読み方リスト

## 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( $s_s$ ,  $s_e$ ,  $s_f$ ,  $s_c$ ) 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 -1~1 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。

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

### 5.3.2 $s_e$ : 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。

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

$d_e$  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わかった。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いく。

表 5.2:  $s_e$  が高い例・低い例

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

### 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  $s_f$  は下記のようになる。

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

$f$  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。また、単語の出現回数はべき分布に従うため、ごく一部の単語の出現回数が極端に大きい。これらの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  $s_f$  とした。また、同様の理由から  $s_f$  の最大値を  $4.82$  とした。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、他の単語よりも圧倒的に出現回数の多かった単語群のうち、最も小さい値 ( $660610$  回) を持つ単語の  $s_f$  である。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いた。

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

表 5.3:  $s_f$  が高い例・低い例

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

### 5.3.4 $s_c$ : Category Score

ユーザの指定したカテゴリ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、言語モデルを用いて概念距離を測ることで  $s_c$  を算出した。カテゴリ単語と出力候補単語の概念距離を算出し、この値が大きいほど高い  $s_c$  (Category score) を与える。この  $s_c$  が高いほどカテゴリ単語と関連性の高い単語とする。  $s_c$  は下記のように計算される。

$$s_c = \frac{1}{1 + d_c} \quad (5.4)$$

$d_c$  はカテゴリ単語と出力候補単語の概念距離である。  $d_c$  の算出には gensim ライブラリ (バージョン 3.0.1) から word2vec の similarity メソッドを用いた。導き出される  $d_c$  の値域は -1~1 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。

また、出力単語をカテゴリ単語と一定以上の関連性を持たせるために、  $s_c$  が 0.3 未満の単語を出力候補単語から除く。この値は、いくつかのカテゴリ単語を設定し、出力候補単語とカテゴリ単語に関連性を感じられる  $s_c$  の最小値を試行の結果決めたものである。

表 5.4:  $s_c$  が高い例・低い例

カテゴリ単語	出力候補単語	$s_c$
妖怪	カッパ	0.400612
	ココナッツ	-0.099956
野球	ホームラン	0.560937
	宇宙	-0.0669442

## 5.4 各 Score 計算部の実装

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く。また、エージェントの返答がオウム返しに

表 5.5: 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

なってしまうと(表 5.5)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。
- カテゴリ単語との  $s_c$  が0.3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4つの  $\text{Score}(s_s, s_e, s_f, s_c)$  を算出し、それぞれ正規化処理、重み付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  $s_s, s_e, s_f, s_c$  共に0~1となる。すなわち  $s_t$  は下記のようになる。

$$s_t = w_s s_s + w_e s_e + w_f s_f + w_c s_c \quad (5.5)$$

$w_s, w_e, w_f, w_c$  は重み係数である(本稿では重み係数は全て1.0とした)。

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?”と出力するようにする。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。

## 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 WebAPI 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。さらに、シャープ社製のスマートフォン型ロボット, RoBoHoN[37] と本 API を用いて、音声による入出力も実現する。

表 5.6: json フォーマット

JSON key	データ型	データ例
result	文字列	success
		error
		NG
message	文字列	え、戦国時代?
url	文字列	<a href="https://ja.wikipedia.org/wiki/%e6%88%a6%e5%9b%bd%e6%99%82%e4%bb%a3">https://ja.wikipedia.org/wiki/%e6%88%a6%e5%9b%bd%e6%99%82%e4%bb%a3</a>
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、ユーザの発言の一部に対して聞き間違いボケをするエージェントにキャラクター性を創出する手法(4.1節参照)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、下記3点の Research Question を検証することを本実験の目的とする。

- RQ1:出力候補単語を特定のカテゴリに関連する語に限定することでエージェントにキャラクター性を付与することができるか。
- RQ2:特定のカテゴリに関する単語をユーモア表現に用いることでユーザが感じる面白さに変化があるのか。
- RQ3:キャラクター性を持たないエージェントと比べ、提案手法におけるエージェントに対してユーザはより親しさを感じられるか。

## 6.2 実験の概要

本実験の被験者は20代(男性9名、女性1名)の学生である。被験者は3パターン(カテゴリ指定を行うパターン1、パターン3については3種類のカテゴリを使用する)のシステムをランダムな順番で使用する。キャラクタ付けに利用するカテゴリは、多くの人にとって理解でき、馴染みがあるものがよいと考え、“旅行”、“スポーツ”、“お菓子”を選定した。3種類のカテゴリを使用するのは、設定したカテゴリ特有の影響が実験結果に表れる可能性を考慮し、その影響を少なくするためである。システムの仕様を聞いたことによる先入観をなくすため、被験者には現在どの仕様のプログラムが使われているのかは説明しない。使用する3パターンのシステムは、ユーザの入力に対し下記のような返答をする。

- パターン1:カテゴリスコアを使用するプログラム  
読み方リスト内のすべての単語から入力単語と頭文字の母音が一致し、かつ末尾の文字が一致する単語のうち、編集距離スコア・概念距離スコア・出現頻度スコア・カテゴリスコアの線形和が最も高い単語を返す。本稿の提案手法である。
- パターン2:カテゴリスコアを未使用のプログラム  
読み方リスト内のすべての単語から入力単語と頭文字の母音が一致し、かつ末尾の文字が一致する単語のうち、編集距離スコア・概念距離スコア・出現頻度スコアの線形和が最も高い単語を返す。
- パターン3:特定のカテゴリに関する単語の中からランダムで単語を出力するプログラム

読み方リスト内の一定以上のカテゴリスコアを持つ単語から、ランダムで単語を返す。

パターン2はカテゴリスコアを用いた影響を検証するため、パターン3は出力候補単語を特定のカテゴリに関連するという条件による影響を検証するために用いる。各パターンの出力例は表6.1に示す。なお、カテゴリ単語はP1-1, P3-1は“旅行”, P2-2, P3-2は“スポーツ”, P2-3, P3-3は“お菓子”を設定した。

表 6.1: 各パターンの動作例

入力した文	出力例			
科学 について教えて？	P1-1 出力例 え、客？	P1-2 出力例 え、体育？	P1-3 出力例 え、焼く？	P2 の出力例 え、叩く？
	P3-1 の出力例 え、上陸？	P3-2 出力例 え、サッカーボール？	P3-3 出力例 え、オープン？	
スラム について教えて？	P1-1 出力例 え、住む？	P1-2 出力例 え、スタジアム？	P1-3 出力例 えっ、クリーム	P2 の出力例 え、含む？
	P3-1 の出力例 え、アトランタ？	P3-2 出力例 え、スタジアム？	P3-3 出力例 えっ、包装？	

### 6.3 実験の手順

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30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

実験に用いた単語例			
生徒	形式	グループ	最終
形態	複数	市内	区間
目的	効果	鉄道	国際

被験者にアンケート用紙を渡す。被験者は、各システムについてあらかじめこちらが用意していた30単語をキーボードで入力し、画面上でエージェントの出力を確認してもらう。被験者にはエージェントと雑談しているシチュエーションを想像してもらい、入力を“○○って知ってる？”，出力を“え、××？”という形式にすることで、実際の対話に近づける。被験者は各システムに対して全単語入力後に、アンケート用紙に記載されている3つの質問に5段階のリッカート尺度(5:とても感じた, 4:感じた, 3:普通, 2:感じなかった, 1:全く感じなかった)で答える。質問内容は次のとおりである。

- 質問 A. このエージェントにキャラクター性を感じましたか?
- 質問 B. 出力された返答はユーモアを感じましたか?
- 質問 C. エージェントに親しみを感じましたか?

## 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 6.1 のとおりである。質問 A のアンケート結果を図 6.1 に、質問 B のアンケート結果を図 6.2 に、質問 C のアンケート結果を図 6.3 に示す。パターン 1 に関するものを P1-1(旅行カテゴリ), P1-2(スポーツカテゴリ), P1-3(お菓子カテゴリ), パターン 2 に関するものを P2, パターン 3 に関するものを P3-1(旅行カテゴリ), P3-2(スポーツカテゴリ), P3-3(お菓子カテゴリ) とする。図 6.1 について、キャラクター性を感じたか? という問に対し、“とても感じた” または “感じた” と回答した被験者は、P1-1 で 30%, P1-2 で 80%, P1-3 で 40%, P2 で 10%, P3-1 で 10%, P3-2 で 20%, P3-3 で 20% であった。図 6.2 について、ユーモアを感じたか? という問に対し、“とても感じた” または “感じた” と回答した被験者は、P1-1 で 10%, P1-2 で 40%, P1-3 で 80%, P2 で 40%, P3-1 で 0%, P3-2 で 0%, P3-3 で 0% であった。図 6.3 について、親しみを感じたか? という問に対し、“とても感じた” または “感じた” と回答した被験者は、P1-1 で 40%, P1-2 で 30%, P1-3 で 50%, P2 で 50%, P3-1 で 20%, P3-2 で 10%, P3-3 で 10% であった。

### 6.4.1 キャラクター性の創出に関する考察

パターン 1 とパターン 2 の結果に対し、Wilcoxon の符号順位和検定を行うと、パターン 1 にスポーツカテゴリを用いた場合、質問 A において 5% 水準で有意差を確認できた。比較結果より、スポーツカテゴリを用いた場合、カテゴリスコアを追加したことによってエージェントにキャラクター性を持たせることができたと考えられる。これにより、ユーモア生成手法にカテゴリスコアを加えることでキャラクター性を持たせることができる可能性があることが示された。しかし、旅行カテゴリとお菓子カテゴリを用いた場合は有意差は確認できなかった。これは、被験者が出力単語からカテゴリ単語をうまく連想することができなかったため、出力単語の一貫性について感じるができなかったからであると考えられる。また、質問 B についていずれのカテゴリにおいても有意差が見られなかった。そのため、特定のカテゴリに関する単語をユーモア表現に用いても、ユーザが感じる面白さには好影響・悪影響ともにほぼ無いと考えられる。

### 6.4.2 ユーモア生起に関する考察

パターン1とパターン3の結果に対し、Wilcoxonの符号順位和検定を行うと、スポーツカテゴリとお菓子カテゴリを用いた場合、質問Bにおいて有意差を確認できた。これにより、スポーツカテゴリとお菓子カテゴリを用いた場合、出力単語を特定のカテゴリに関連するという条件を加えても提案手法でユーモアを生み出すことができていると考えられる。しかし、旅行カテゴリを用いた場合は有意差は確認できなかった。これは、旅行カテゴリの出力候補単語内で入力単語と上手く不適合が作り出せなかったためと考えられる。

### 6.4.3 カテゴリスコアを用いた影響に関する考察

パターン1とパターン2の結果に対し、Wilcoxonの符号順位和検定を行うと、質問Cにおいて、いずれのカテゴリにおいても有意差は見られなかった。これにより、カテゴリスコアの有無によらず、ユーザがエージェントに感じる親しみは変化しなかったと言える。

### 6.4.4 実験のまとめ

6.4.1項、6.4.2項、6.4.3項より、6.1節で立てた Research Question について以下のことが導出される。

- RQ1について、カテゴリによっては、特定のカテゴリに関する単語を出力するようになったことで、ユーザはエージェントにキャラクター性を感じるようになった。
- RQ2について、特定のカテゴリに関する単語を出力するようにしても、ユーザがエージェントに感じる面白さには好影響も悪影響も生じない。
- RQ3について、カテゴリスコアを追加しても、ユーザがエージェントに感じる親しみには変化が見られない。

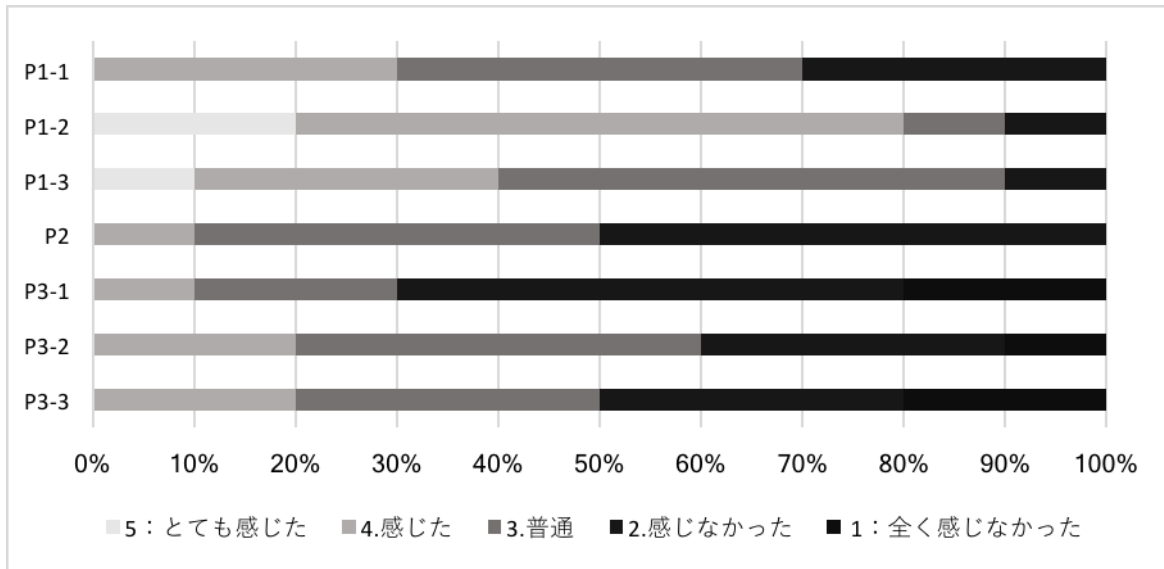


図 6.1: 質問 A の回答 (N=10)

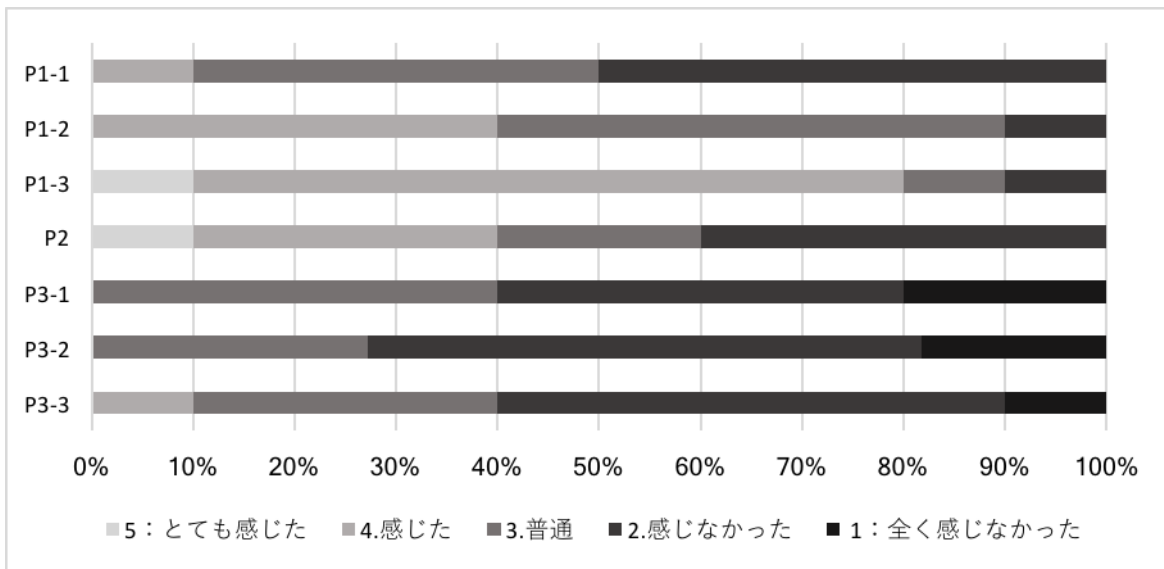


図 6.2: 質問 B の回答 (N=10)

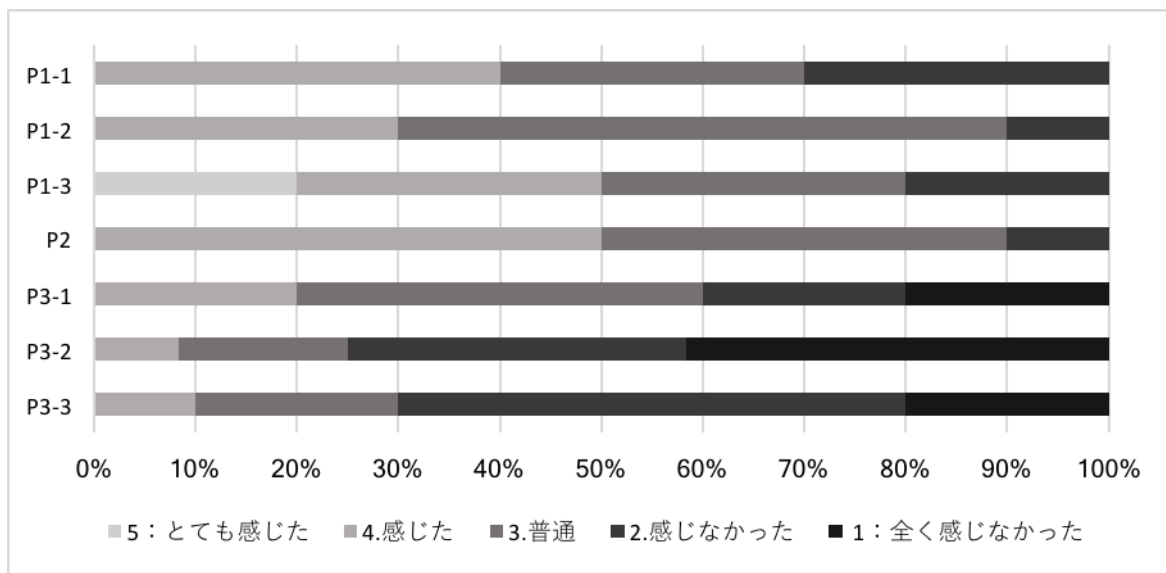


図 6.3: 質問 C の回答 (N=10)

## 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たず、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、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、ユーモアのある返答を行うエージェントのプロトタイプシステムを構築した。このプロトタイプシステムには、入力単語が同一であると常に同じ返答を行ってしまい、ユーザに飽きられてしまう可能性があるという問題点があった。そこで、このプロトタイプシステムにキャラクターを持たせることで、ユーモア表現の幅の広がりやサービスクオリティの向上を目指した。キャラクターの創出を実現するために、エージェントが特定の分野に偏った発言をすることで、ユーザはそのエージェントに対して当該分野に関するキャラクターを感じやすくなるという仮説を立て、この仮説に基づくシステムの実装を行った。

実験の結果、カテゴリによってはエージェントにキャラクターを持たせられたことが確認された。さらに別のカテゴリを試すことで、設定カテゴリによる影響が明確になってくるであろう。例えば、本稿ではカテゴリ単語を”旅行”，”スポーツ”，”お菓子”の3パターンで検証を行ったが、カテゴリ単語をユーザの趣味嗜好に合わせた単語や、場所や状況に合わせた単語に設定することで、ユーザが感じる親密度や満足感が向上することが考えられる。また、東京ゲームショウ 2017[38]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。

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 [39]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利用法が考えられる。ユーザの好みに合わせたキャラクターを持たせることで、ユーザはエージェントに対して親近感を感じ、質問しにくいことについても気軽に尋ねることができるようになると考えられる。



## 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.
- [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, Vol. J86-D1, 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 2017/11/24).
- [21] Softbank. : Papper, <https://www.softbank.jp/robot/> (last visited on 2017/10/13).
- [22] Google inc. : Google assistant, <https://assistant.google.com/> (last visited on 2017/10/13).
- [23] 宮崎千明, 東中竜一郎, 牧野俊朗, 松尾義博, 佐藤理史. 話者のキャラクター性に寄与する言語表現の基礎的分析. 言語処理学会 第 20 回年次大会 発表論文集, 第 2014 巻, pp. 232–235, 2014.

- [24] 倉本到, 安田淳志, 山本景子, 水口充, 辻野嘉宏. 対話エージェントへの「個性」の付与:意思決定支援システムに対する影響. 情報処理学会インタラクシオン, 第2012巻, 2012.
- [25] 沈睿, 菊池英明, 太田克己, 三田村健. 音声生成を前提としたテキストレベルでのキャラクター付与. 情報処理学会論文誌, Vol. 53, No. 4, pp. 1269–1276, 2012.
- [26] 山本大介, 大浦圭一郎, 西村良太, 打矢隆弘, 内匠逸. スマートフォン単体で動作する音声対話3dエージェント「スマートメイちゃん」の開発. 情報処理学会インタラクシオン, 第2012巻, 2012.
- [27] 吉野孝, 松尾知哉. コミュニケーション支援のためのキャラクターを作成した写真を使用した写真共有システムの開発と評価. 映像情報メディア学会誌, 第67巻, pp. 64–73, 2013.
- [28] S. Coulson and R. F. Williams. Hemispheric asymmetries and joke comprehension. *Neuropsychologia*, Vol. 43, No. 1, pp. 128–141, 2005.
- [29] 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.
- [30] 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.
- [31] J. M. Suls. Cognitive processes in humor appreciation. In *In Handbook of Humor Research*, pp. 39–57. Springer-Verlag New York, 1983.
- [32] 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.
- [33] 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.
- [34] 金水敏. ヴァーチャル日本語 役割語の謎. 岩波書店, 1999.
- [35] Mecab: Yet another part-of-speech and morphological analyzer, <http://taku910.github.io/mecab/> (last visited on 2017/11/24).
- [36] 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.
- [37] ロボホン開発環境, <https://robohon.com/sdk/> (last visited on 2018/1/25).

- 
- [38] 東京ゲームショウ 2017, <http://expo.nikkeibp.co.jp/tgs/2017/> (last visited on 2017/11/24).
- [39] 教えて goo, <https://oshiete.goo.ne.jp> (last visited on 2014/4/1).

# 研究業績

## 査読付き国内会議

- (1) 呉健朗, 鈴木奨, 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提案, 情報処理学会インタラクシオン 2018 (2018年3月発表予定).
  - (2) 瀧田航平, 鈴木奨, 呉健朗, 堀越和, 中辻真, 宮田章裕: キャラクター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, 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).
  - (2) 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, キャラクター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).
-