

# シズルワードから想起される料理の雑学を話す対話システム

新名 和也<sup>†</sup> 嶋田 和孝<sup>†</sup>

<sup>†</sup>九州工業大学大学院 情報工学府 先端情報工学専攻 〒 820-8052 福岡県飯塚市川津 680-4

E-mail: †{k\_niina,shimada}@pluto.ai.kyutech.ac.jp

**あらまし** カーナビの様々な機能の1つに飲食店検索がある。現在の飲食店検索は、ユーザが具体的な料理名を与えることを前提としている。しかし、ユーザは「あっさりしたものが食べたい」のような抽象的な情報のみの要求を頻繁に行う可能性があり、現在のカーナビではこの要求に対応できない。本研究では、この問題の解決策として、食べ物のおいしさなどを表す「シズルワード」に着目し、シズルワードから想起される料理名の獲得を行う。また、近年カーナビのようなタスク指向型と雑談対話システムのような非タスク指向型を組み合わせ、満足度の向上を図る動きがある。しかし、非タスク指向型に関する研究で成果をあげている手法では、汎用的な文を生成しやすいため、ユーザが対話に飽きる危険性がある。本研究では、ユーザを楽しませる要素として雑学に注目し、Web ページから雑学を含む文の抽出を行う。さらに、提案する2つの手法を組み合わせた雑談対話システムの構築を行う。

**キーワード** シズルワード, 雑学, 雑談, 対話システム

## A Dialogue System with Miscellaneous Knowledge from Sizzle Words

Kazuya NIINA<sup>†</sup> and Kazutaka SHIMADA<sup>†</sup>

<sup>†</sup> Graduate School of Computer Science and System Engineering, Kyushu Institute of Technology  
680-4, Kawazu, Iizuka, Fukuoka, 820-8502 Japan

E-mail: †{k\_niina,shimada}@pluto.ai.kyutech.ac.jp

**Abstract** One of car navigation system's functions is restaurant search. The current restaurant search is based on the premise that a user gives it a cooking name. However, a user tends to give abstract information to the system. In this paper, we focus on sizzle words, and acquire cooking names from sizzle words. Also, there is a movement to combine task-oriented dialogue system and non-task-oriented dialogue system. Methods on recent non-task-oriented researches, such as Sequence-to-Sequence models, tend to generate generic responses. We extract a miscellaneous knowledge sentence from the Web, and create a dialogue system with miscellaneous knowledge from sizzle words.

**Key words** Sizzle Word, Miscellaneous Knowledge, Chat, Dialogue System

### 1. ま え が き

対話システムの一つに、カーナビゲーションシステム（カーナビ）がある。カーナビは、飲食店検索や目的地までの経路検索など、ユーザの様々な要求に答える必要がある。このようなユーザから要求されたタスクを遂行する対話システムはタスク指向型と呼ばれており、近年では Partially Observable Markov Decision Process (POMDP) を用いた研究 [1] や End-to-End を用いた研究 [2] が行われている。これらの研究や現在のカーナビでは、ユーザが具体的な情報を与えることを前提としているため、データベースもそれに沿って構築されている。しかし、ユーザは必ずしも具体的な情報を与えるとは限らない。例えば、カーナビに対してユーザが飲食店検索をしてほしいと思った時、「あっさりしたものが食べたい」のような発話をするこ

とは十分に考えられる。現在のカーナビでは、このような発話に対して適切に対応できない。本研究では、この問題への対応策として、「シズルワード」に着目する。シズルワードとは食べ物のおいしさなどを表現する単語のことで、「あっさり」などが該当する。このシズルワードは料理レシピのタイトル名などに使われることがあるため、料理レシピからシズルワードと関係ある料理名が獲得できると考えられる。よって、本研究では Web ページから獲得したレシピに基づき、シズルワードから想起される料理名を獲得する手法を提案する。

また、近年ではタスク指向型の対話システムと、雑談対話システムを代表とする非タスク指向型の対話システムを組み合わせることで、ユーザの満足度などの向上を図る動きが見られる。カーナビについても、雑談対話機能を追加することで、居眠り運転の防止など様々な点に貢献できると考えられる。非タ

スク指向型の対話システムにおいて、近年では Sequence-to-Sequence (Seq2Seq) モデルを用いた手法が一定の成果を挙げている [3] [4]. しかし、このモデルは汎用的な文 (「はい」や「そうですね」など、文脈に依存せず使用できる文) を生成しやすい傾向があるため、ユーザが対話に飽きてしまう危険性がある。ユーザを対話に飽きさせないためには、ユーザを楽しませる文を生成して対話に引き込む必要がある。よって、本研究では雑学を含んだ文 (雑学文) がユーザを楽しませる文であるとして、Wikipedia<sup>(注1)</sup> から雑学文を獲得する手法を提案する。

また、本研究はカーナビを想定しているため、提案する 2 つの手法を組み合わせた対話システムを構築する。ユーザからシズルワードを含む入力文を受け取った時、そのシズルワードから想起される料理の雑学文を出力する対話システムを構築する。

## 2. 関連研究

シズルワードに関する研究として、加藤ら [5] の研究がある。加藤らは、食品に対するシズルワードの使われ方がメディアによって異なるとして、Twitter<sup>(注2)</sup>、Web ページ、料理レシピ投稿 Web サイト CookPad<sup>(注3)</sup> の 3 つのメディアにおけるシズルワードと食品の関連性を分析している。

非タスク指向型の対話システムの研究では、稲葉ら [8] が Twitter からの自動独話文生成を行っている。Twitter から独話文として利用できるツイートを抽出し、意外性、ユーモア性、トピックにおける特殊性の 3 つの観点からスコアを計算し、順位づけを行っている。また、江頭ら [9] は、Web 上のニュース記事と Wikipedia の記事を知識データとして、Markov Decision Process (MDP) を用いた対話システムを構築している。挨拶や話題提示などの各応答文生成モジュールを作成し、MDP により獲得した対話戦略を用いて対話を行っている。

本研究では、Wikipedia を知識源として用いている。Wikipedia からシステムの発話文を抽出する研究として、太田ら [6] の研究がある。太田らは、意外性のある文は対話を盛り上げる文であるとして、Wikipedia の各記事の文に対して、TF-IDF、単語の共起頻度、文長の 3 つの指標に基づきスコアを計算し、ルールを満たす文にはペナルティを与えることで意外性のある文の抽出を行っている。しかし、この手法では、対話を盛り上げると判断された文を十分に獲得できていないため、改善の余地が見られる。また、杉本ら [7] は、Wikipedia の記事中の文に対して、雑談対話システムに利用できる文であるかどうかを、記事内に置ける位置情報を用いて分類している。文字数や特定単語を用いたフィルタリングなどを行った後、位置情報に基づいた特徴量を用いて分類を行っている。

## 3. シズルワードから想起される料理名の獲得

本研究では、シズルワードから想起される料理名を、CookPad にあるレシピを用いて獲得する。レシピのタイトルには、「ク

表 1 シズルワード「ピリ辛」と材料との *TFIDF* の一例

材料名	<i>TFIDF</i> の値
XO 醤	4.84418708645859
あさつき	4.15103990589865
⋮	⋮

表 2 料理名「麻婆茄子」と材料との *TFIDF* の一例

材料名	<i>TFIDF</i> の値
XO 醤	4.57471097850338
おろしにんにく	6.26941298614615
⋮	⋮

リーミーでトロトロなカレーうどんの作り方」のように、シズルワードと料理名を含むものが存在する。よって、シズルワードと料理名の対応関係を獲得する最も単純な方法は、レシピタイトルにおける料理名とシズルワードの共起関係を獲得することである。しかし、この方法ではシズルワードと料理名の対応関係を十分に獲得することはできない。例えば、元々「ヘルシー」な料理 (「サラダ」など) に対して、「ヘルシーな〇〇」と明示的に書くことは少ない。また、味覚的な要素を示すシズルワード (「激辛」など) でも同様の傾向が考えられる。例えば、「キムチチゲ」のタイトルにわざわざ「激辛」と書くことは必ずしも多くない。したがって、レシピタイトルにおける共起関係以外から、シズルワードと料理名の対応関係を獲得する必要がある。ここで、先ほど挙げた事例から、食材の特性がその料理のシズルワードを推定するのに役立つということがわかる。例えば、キムチが使われている料理は少なくとも辛い料理であるため、「激辛」や「ピリ辛」といったシズルワードを推定することができる。よって、本研究では料理の食材から、シズルワードと料理名の対応関係を獲得する。

まず、料理名をクエリとして、CookPad に対して検索を行い、料理のレシピを獲得する。次に、獲得したレシピから材料名とシズルワードの共起関係を抽出する。この共起関係を利用して、あるシズルワード  $sw_i$  に対する材料  $m_j$  の重要度  $TFIDF(sw_i, m_j)$  を式 (1) で計算する。

$$TFIDF(sw_i, m_j) = tf(sw_i, m_j) \times idf(m_j) \quad (1)$$

式 (1) の  $tf(sw_i, m_j)$  はシズルワード  $sw_i$  に対する材料  $m_j$  の共起頻度、 $idf(m_j)$  はシズルワードを 1 文書とした時の逆文書頻度である。同様に料理名と材料名の共起関係も抽出を行い、ある料理名  $c_i$  に対する材料  $m_j$  の重要度  $TFIDF(c_i, m_j)$  を計算する。表 1 に、シズルワードと材料との *TFIDF* の一例を、表 2 に料理名と材料との *TFIDF* 値の一例を示す。次に、各シズルワード  $sw$  と料理名  $c$  に対して、材料名によるベクトルを作成する。ベクトルの値は重要度 *TFIDF* の値とする。次に、作成したベクトルを用いて、各シズルワードと料理名の  $\cos$  類似度を式 (2) で計算する。

$$COS(sw_x, c_y) = \frac{\sum_{i=1}^V x_i \cdot y_i}{\sqrt{\sum_{i=1}^V x_i^2 \times \sum_{i=1}^V y_i^2}} \quad (2)$$

式 (2) の  $x_i, y_i$  はそれぞれ  $sw_x, c_y$  のベクトル、 $V$  はベクトルの大きさである。最後に、各シズルワードに対して  $\cos$  類似度上位の料理名を獲得する。

(注1) : <https://ja.wikipedia.org/>

(注2) : <https://twitter.com/>

(注3) : <https://cookpad.com/>

## 4. 雑学文の抽出

本研究では、Wikipedia から雑学文の抽出を行い、独自に設定した指標に基づき順位づけを行う。4.1 節で Wikipedia からの雑学文抽出方法について説明し、4.2 節で独自の指標に基づく順位づけについて説明する。

### 4.1 Wikipedia からの雑学文抽出

Wikipedia から雑学文を抽出する手法について説明する。本研究では、前述の太田らをベースとした手法を用いる。太田らは、Wikipedia から記事を 1 つ取り出し、記事内の文に対して TF-IDF、単語の共起頻度、文の長さの 3 つの指標に基づくスコアを計算し、ルールによってペナルティを与えた後スコア上位の文を抽出している。本研究では、太田らが用いた指標に変更を加えることで、雑学文の抽出を行う。

#### 4.1.1 IDF

太田らは、Wikipedia の記事中にある重要な文の評価を高くするために、文中の名詞に対して TF-IDF を求め、その平均値をスコアとして用いている。しかし、本研究で対象とする雑学文が、記事中の重要な文であるとは限らない。また、専門性の高い文は雑学文である可能性が高いと考えられる。よって本研究では、TF-IDF ではなく IDF のみを用いる。文  $S_i$  に対する IDF に基づくスコア  $IDF(S_i)$  は式 (3) を用いて計算する。

$$IDF(S_i) = \frac{\sum_{w \in S_i} idf(w)}{n(S_i)} \quad (3)$$

式 (3) の  $w$  は文  $S_i$  中の名詞、 $idf(w)$  は  $w$  の Web 日本語 N グラム [10] の 1-gram における IDF 値、 $n(S_i)$  は文  $S_i$  中の名詞の総異なり数である。

#### 4.1.2 単語の共起頻度

太田らは、共起頻度の低い単語で構成される文は、意外性のある内容を含んでいる可能性が高いとして、単語の共起頻度に基づくスコアを計算している。まず、文  $S_i$  から名詞を抽出し、各名詞に対して 7-gram の範囲にある名詞との名詞ペアを作成する。次に、作成した名詞ペアの共起頻度を、Web 日本語 N グラムの 7-gram を用いて算出する。最後に、式 (4) で単語の共起頻度に基づくスコア  $coFreq(S_i)$  を計算する。

$$coFreq(S_i) = \begin{cases} \frac{cf_{max} + 1 - cf_i}{cf_{max}} & (cf_i \neq 0) \\ 0 & (cf_i = 0) \end{cases} \quad (4)$$

式 (4) の  $cf_i$  は文  $S_i$  中の全ての名詞ペアの共起頻度の平均、 $cf_{max}$  は  $cf_i$  の最大値である。しかし、式 (4) は、 $cf_{max}$  の値が非常に大きく、 $cf_i$  と  $cf_{max}$  の間に大きな差があると、ほとんどの文のスコアがほぼ同じ値になるという問題点がある。よって、本研究では式 (4) に対数を適用した式 (5) を用いて、単語の共起頻度に基づくスコア  $coFreq_{log}(S_i)$  を計算する。

$$coFreq_{log}(S_i) = \begin{cases} \frac{\log cf_{max} + 1 - \log cf_i}{\log cf_{max}} & (cf_i \neq 0) \\ 0 & (cf_i = 0) \end{cases} \quad (5)$$

#### 4.1.3 文の長さ

太田らは、長すぎる文や短すぎる文はシステムの発話文として不適切であると考え、収集した雑談対話コーパスにおける各発話の形態素数の割合をスコアに用いている。しかし、本研究の対象である雑学文は、一般的な雑談対話における発話文よりも長いと考えられる。よって、本研究では利用しない。

#### 4.1.4 ルール

太田らは、以下のルールを満たす文は、1 文のみで意味を理解することができない不適切な文であるとして排除している。

- 末尾が読点やピリオドではない
- 助詞の「が」もしくは「は」を含まない
- 体言止めで終わっている
- 接続詞から始まる
- 指示語や特定の単語を含む

一方で、これらの一部を満たす文は、文脈情報の追加や文末表現の補完などを行うことで、発話文として利用可能になると考えられるため、排除するのは必ずしも適切ではない。そこで本研究では以下のルールを設け、排除するのではなくルールを満たす文にスコアに 0.1 を乗算することで、4.1.1 節や 4.1.2 節で求めたスコアを重視しつつ、一定のスコアを割り当てる。

- 接続詞や指示語を含む
- 文中の名詞の数が 2 つ以下

逆に、Wikipedia の記事の最初に書いてある定義文や箇条書きで書かれている文には、雑学文でない文が多く存在するため、このような文のスコアは 0 とする。

#### 4.1.5 文の順位づけ

4.1.1 節と 4.1.2 節で計算したスコアに基づき、文  $S_i$  に対するスコア  $Score(S_i)$  を式 (6) を用いて計算する。

$$Score(S_i) = IDF(S_i) \times coFreq_{log}(S_i) \times \alpha \quad (6)$$

$$\alpha = \begin{cases} 0.1 & (4.1.4 \text{ 節のルールを満たす}) \\ 0 & (\text{定義文もしくは箇条書きの文}) \\ 1 & (\text{上記のどれにも当てはまらない}) \end{cases}$$

この  $Score(S_i)$  を 1 つの記事の全文に対して計算し、4.1.4 節のルールでペナルティを与えた後、スコア上位の文を抽出する。

## 4.2 対応可能度による順位づけ

4.1 節で抽出した文を、スコア  $Score(S_i)$  順で対話システムに用いる場合の問題点について説明する。例えば、「インド料理」の記事から「ハイデラバードのハリームが有名」という文が、スコア  $Score(S_i)$  が最も高い文として抽出できたとする。ここで、スコア  $Score(S_i)$  順で抽出した雑学文を用いる場合、この「ハイデラバードのハリームが有名です」という文がはじめて出力される。この時、この文に対するユーザの応答文として、「ハイデラバードいいところですよ」や「ハリームって何ですか」というような文が考えられる。しかし、「ハイデラバード」の記事の文数が少ない場合、ハイデラバードについての雑談を続けていくことが難しくなる。また、「ハリーム」の記事が存在しない場合、わからないという旨の応答文を出力することになり、ユーザの対話欲求を削ぐことになる。このように、抽出した文の内容に対して、Wikipedia の記事が持つ知識量が貧弱な

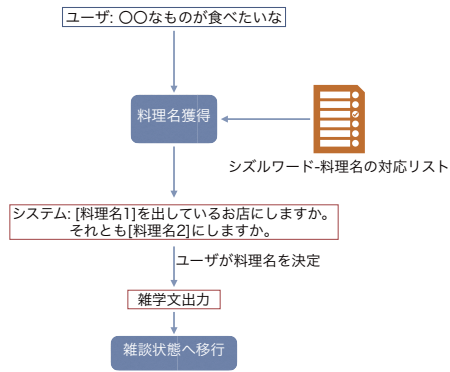


図1 シズルワードを含むユーザ発話文から雑談状態移行までの概要図

場合、ユーザからの質問や話題の移行などに上手く対応できないという危険性がある。本研究では、この問題点を解決するために、ユーザの質問や話題の移行に対応できるかどうかを表す指標（対応可能度: Response Score）を計算し、4.1節で抽出した文に対して再度順位づけを行う。

対応可能度 ( $RS$ ) の計算について説明する。対応可能度については、4.1.5節で計算したスコア  $Score(S_i)$  上位10文のみを対象に計算を行う。まず、Wikipedia から1つの記事を取り出し、4.1節で抽出した文  $S_i$  から固有名詞  $pn_j$  を抽出する。次に、固有名詞  $pn_j$  とタイトルが一致する記事  $A_{pn_j}$  を検索し、スコア  $RS(pn_j)$  を式(7)を用いて計算する。

$$RS(pn_j) = \log sn(A_{pn_j}) + \frac{\sum_{i=1}^N Score(T_i)}{N} \quad (7)$$

$$N = \begin{cases} 10 & (sn(A_{pn_j}) \geq 10) \\ sn(A_{pn_j}) & (sn(A_{pn_j}) < 10) \end{cases}$$

式(7)の  $sn(A_{pn_j})$  は記事  $A_{pn_j}$  の文数、 $T_i$  は記事  $A_{pn_j}$  における4.1.5節のスコア  $Score$  上位  $i$  番目の文である。そして、 $RS(pn_j)$  を文  $S_i$  の全ての固有名詞に対して計算を行い、その平均値を文  $S_i$  の対応可能度として算出する。最後に、各文に対して対応可能度を計算し、順位づけを行う。

## 5. 雑学文を利用する対話システム

本研究では、3.節で獲得したシズルワードと料理名の組と、4.節で抽出した雑学文を用いた対話システムを構築する。本システムの処理の流れは、シズルワードを含むユーザ発話文から料理名を決定し、雑談状態へ移行するまでの処理と、雑談状態での処理の2つに分かれる。

まず、シズルワードを含むユーザ発話文から雑談状態移行までの処理の流れについて説明する。概要図を図1に示す。ユーザからシズルワードを含んだ入力文を受け取った時、3.節で獲得したシズルワードと料理の組の中から2つの料理名を選択し、ユーザに提示を行う。料理名を提示した後、ユーザに料理名を決定してもらう。料理名が決定したら、その料理名に関する雑学文を出力し、雑談状態へ移行する。

提示する料理名を選択方法について説明する。選択する料理名は、シズルワードに対して3.節で計算した  $\cos$  類似度上位5個の料理名から選択する。料理名選択の最も単純な方法は、 $\cos$  類似度の高い順に料理名を選択する方法である。しかし、この

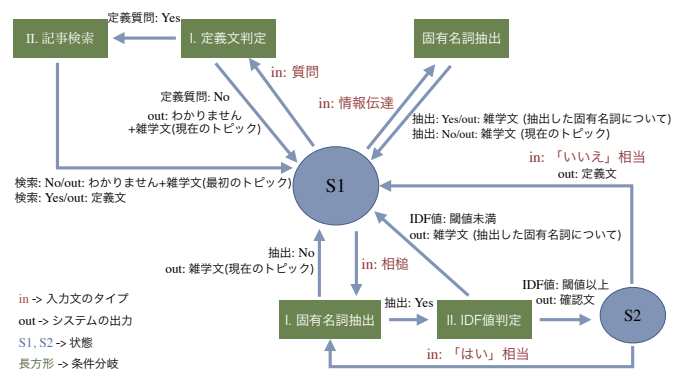


図2 雑談状態時の処理の概要図

方法では似たような料理を提示する危険性がある。例えば、シズルワード「スパイシー」について、 $\cos$  類似度が高い順に「チキンカレー」、「ドライカレー」、「タコライス」、「クリームカレー」、「エスニックライス」の5個の料理名が獲得された場合、 $\cos$  類似度が高い順に提示する料理名を選択すると、「チキンカレー」と「ドライカレー」が選択される。しかし、この2つは似た料理であるため、両方をユーザに提示する意味はない。したがって、本研究では単語の意味的類似性を用いて、互いに似ていない2つの料理名を選択する。選択する2つの料理名のうち、1つは  $\cos$  類似度が最も高い料理名  $c$  である。もう1つの料理名の選択について、まず  $c$  に含まれる名詞を抽出し、その名詞を含まないものを他の4つの料理名から抽出する。そして、抽出した料理名と  $c$  との類似度を、単語の分散表現 [11] に基づき計算し、最も類似度が低かった料理名を選択する。

次に、雑談状態でのシステムの処理の流れについて説明する。概要図を図2に示す。雑談状態でのシステムは  $S1$  と  $S2$  の2つの状態を持つ。状態  $S1$  での処理の流れについて説明する。状態  $S1$  では、ユーザから入力文を受け取った時、入力文を以下の3つのタイプにルールを用いて分類する。

- 質問

構文解析器 KNP<sup>(注4)</sup> で「疑問」タグが付与される、もしくは疑問詞が存在する。

- 情報伝達

固有名詞を含み、かつ動詞もしくは形容詞を含む。

- 相槌

固有名詞を含まない、もしくは固有名詞は含むが動詞及び形容詞を含まない。

次に、以下のようにタイプに沿った応答文を生成する。

- 質問に分類された場合 (図2の「in:質問」)

(I) まず定義を問う質問かどうかの判定を行う。この時、判定は「何」、「どんなもの」などのキーワードを含んでいるかどうかで行う。定義を問う質問でない場合は、「わかりません」という旨の文と最初のトピックに関する雑学文を出力する。

(II) 定義を問う質問であった場合には、質問対象となっている単語とタイトルが一致する Wikipedia の記事を検索を行う。そして、記事があった場合には定義文を出力する。記事

(注4) : <http://nlp.ist.i.kyoto-u.ac.jp/?KNP>

がなかった場合には「わかりません」という旨の文と最初のトピックに関する新たな雑学文を出力する。

- 情報伝達に分類された場合 (図2の「in:情報伝達」)

ユーザの入力文から固有名詞を抽出する。この時、今までの対話の中でトピックになっていない固有名詞のみを対象とする。抽出できた場合には、その固有名詞に関する雑学文を出力し、抽出できなかった場合には最初のトピックに関する雑学文を出力する。なお、抽出できる固有名詞が2つ以上ある場合は、4.2節で計算したスコア  $RS$  が最も高い固有名詞を抽出する。

- 相槌に分類された場合 (図2の「in:相槌」)

(I) 直前にシステムが出力した雑学文から固有名詞を抽出する。この時、情報伝達の時と同じく今までの対話の中でトピックになっていない固有名詞のみを対象とする。抽出できなかった場合は、その時点でのトピックに関する雑学文を出力する。なお、抽出できる固有名詞が2つ以上ある場合は、4.2節で計算したスコア  $RS$  が最も高い固有名詞を抽出する。

(II) 抽出できた場合は、その固有名詞の IDF 値が閾値より大きいかの判定を行う。IDF 値が閾値より大きいなら「○○ってご存知ですか」という確認文を出力して状態 S2 に遷移し、閾値以下なら固有名詞に関する雑学文を出力する。

状態 S2 での処理の流れについて説明する。状態 S2 では、ユーザからの入力文を「はい」相当と「いいえ」相当の2つにキーワードを用いて分類する。

- 入力文が「はい」相当の場合 (図2の「in:はい相当」)

(I) 直前に出力した雑学文から別の固有名詞の抽出を行う。抽出できなかった場合は、その時点でのトピックに関する雑学文を出力し、状態 S1 に遷移する。

(II) 抽出できた場合は、その固有名詞の IDF 値が閾値より大きいなら再度確認文を出力し、閾値以下なら固有名詞に関する雑学文を出力して状態 S1 に遷移する。

- 入力文が「いいえ」相当の場合 (図2の「in:いいえ相当」)

確認文に含まれている固有名詞の定義文を出力し、状態 S1 に遷移する。

## 6. 実験

シズルワードの獲得手法、雑学文抽出手法、及び雑学文を用いた対話システムの有効性について検証を行った。

### 6.1 実験設定

シズルワードと料理名については、人手で収集した 615 個の料理名と 341 個のシズルワードを用いた。また、レシピについては CookPad から 1 つの料理名につき最大 30 件収集した。その結果、14,704 件のレシピを獲得することができた。なお、341 個のシズルワードの中で、獲得したレシピ中に出現していたシズルワードの数は 127 個であった。Wikipedia の記事については、2017 年 5 月 21 日付でウィキメディア財団より提供されているデータベース・データ<sup>(注5)</sup>から入手したものを用いた。単語の分散表現については、Word2Vec<sup>(注6)</sup>を用いて獲得した。

(注5) : <https://dumps.wikimedia.org/jawiki/>

(注6) : <https://radimrehurek.com/gensim/models/word2vec.html>

### 6.2 実験方法

シズルワードから想起される料理名は、各シズルワードに対して 3. 節で計算した  $\cos$  類似度が最も高い上位 5 個を獲得した。評価については、被験者 5 人に獲得されたシズルワード-料理名の組を見てもらい、「料理名に対してシズルワードが適切であるかどうか」を、適切な場合は 1、そうでない場合は 0 で評価した。その後、3 人以上が 1 と評価したシズルワード-料理名の組数の割合を算出した。

雑学文については、ランダムに獲得した 50 個の Wikipedia の記事に対して、本研究と太田らの手法を用いてそれぞれスコア  $Score(S_i)$  上位  $n$  文を抽出した。評価については、利用した手法を伏せた上で、被験者 3 人に抽出した各文を評価してもらった。評価方法は、各文の内容に対して「日常生活において自然と身につく(暗黙的に得られる)知識でないか」を、自然と身につく知識の場合は 0 を、そうでないなら 1 で評価した。なお、説明している対象がわからないもの、もしくは知識として成り立っていないもの(「本項では 1. の朝鮮半島由来の麺料理について解説する」など)については 0 とした。その後、2 人以上が 1 と評価した文の文数の割合を手法ごとに算出した。なお、本研究では  $n = 5, 10$  とした。

雑学文を利用する対話システムについては、対応可能度を用いたものとそうでないものの 2 つを用いて、被験者 5 人に対して実際に対話を行った。評価については、被験者に対して Q1: 対話の流れは自然であったか、Q2: 対話は面白かったか、Q3: また対話したいと思うか、それぞれの項目を 5 段階で評価した。

### 6.3 結果

シズルワードから想起される料理名の獲得手法について、被験者 3 人以上が 1 と評価したシズルワード-料理の組数の割合は 0.511 であった。実際に獲得されたシズルワード-料理の組を表 3 に示す。表 3 の「○」は被験者 3 人以上が 1 と評価したことを、「×」は 1 と評価した被験者が 2 人以下であることを表す。この結果を踏まえ調査を行った。本実験では表 3 のように、あるシズルワードに対してシズルワード-料理名の組を 5 組獲得している。そこで、各シズルワードごとに被験者 3 人以上が 1 と評価したシズルワード-料理名の組について、3 組以上の場合を集合  $Set_{high}$  とし、2 組以下の場合を集合  $Set_{low}$  としてシズルワードを分類した。例えば、表 3 の「ピリ辛」は○が 5 個であるため、 $Set_{high}$  に分類される。一方、「プリプリ」は○が 1 つしかないため、 $Set_{low}$  に分類される。そして、それぞれの集合に対して、獲得したレシピにおけるシズルワードの出現数の平均値を算出した。結果を表 4 に示す。表 4 より、 $Set_{low}$  に属するシズルワードのレシピ出現数が、 $Set_{high}$  に属するシズルワードのレシピ出現数より少ないことがわかった。よって、レシピ出現数が少なかったシズルワードについて、シズルワードの特性を表す材料名のベクトルをうまく作成できなかったために、適切な料理を獲得することができなかったと考えられる。

雑学文抽出手法の評価結果を表 5 に示す。表 5 より、提案手法は太田らの手法より雑学を含んでいると判断された文の文数の割合が高いことがわかる。また、t 検定による検定では、 $n = 5$  の場合の結果に関して、 $p < 0.05$  の範囲で有意差を確認

表 3 獲得したシズルワード-料理名の例

シズルワード	料理名 1	料理名 2	料理名 3	料理名 4	料理名 5
ピリ辛	マーボ丼 (○)	麻婆茄子 (○)	麻婆茄子丼 (○)	回鍋肉 (○)	麻婆豆腐 (○)
ホクホク	かぼちゃの煮物 (○)	かぼちゃのそばろあんかけ (○)	里芋のそばろあんかけ (○)	さといもご飯 (○)	かぼちゃがゆ (×)
プリプリ	しゃぶしゃぶ (×)	かきご飯 (○)	ターメリックライス (×)	土鍋ご飯 (×)	鯛めし (×)

表 4 集合別のレシピ中での出現数

集合	レシピ出現数の平均
$Set_{high}$	20.641
$Set_{low}$	8.465

表 5 雑学文抽出の評価結果

抽出手法	$n = 5$	$n = 10$
太田ら	0.612	0.656
提案手法	0.700	0.706

表 6 対話システムの評価結果

	Q1	Q2	Q3
対応可能度あり	3.0	2.8	3.2
対応可能度なし	3.0	3.4	3.2

表 7 対話例

発話者	発話文
ユーザ	ホクホクしたものが食べたい
システム	かぼちゃの煮物を出しているお店にしますか? それともさといもご飯にしますか?
ユーザ	さといもご飯が食べたい
システム	飯といえば、保温すれば「デンプンの劣化」は防げるますけど、質が劣化し臭くなるそうですよ
ユーザ	知らなかった
システム	デンプンといえば、ジャガイモについて、いわゆる、片栗粉は、カタクリの地下茎から採取したデンプンを指す言葉であった、近年市販されている片栗粉と呼ばれるもののほとんどは馬鈴薯澱粉となっているそうですよ
ユーザ	片栗粉はじゃがいもからできてるんだ・・・

することができた。本研究では、 $IDF$  と  $coFreq_{log}$  に対しての重みを考慮していないため、重みをつけることで精度の向上を図ることができるのではないかと考えられる。

雑学文を利用する対話システムの評価結果を表 6 に、実際の対話例を表 7 に示す。表 6 より、対応可能度を用いたシステムの Q2 の評価値は、対応可能度を用いない場合に比べ低いことがわかる。これは、対応可能度を用いたシステムが、1 つのトピックについての雑学を多く出力していたことが原因であると考えられる。対応可能度を用いたシステムは、固有名詞抽出において、現在のトピックとなっている単語のスコア  $RS$  よりも高いスコアを持つ固有名詞のみを抽出し、その固有名詞の雑学文を出力する。よって、あるトピックのスコア  $RS$  が高い場合、他のトピックに移行することが難しくなる。よって、あるトピックに関する雑学文の出力回数を制限することで、ユーザが対話に飽きにくくなると考えられる。

## 7. むすび

本研究では、シズルワードから想起される料理名を、材料名に着目したベクトルを用いて獲得する手法と、太田らをベースとした手法に対応可能度を組み合わせた雑学文抽出手法を提案した。また、提案手法を組み合わせ、シズルワードから想起される料理名の雑学を話す対話システムの構築を行った。実験の結果、51.1%の精度でシズルワードに対して適切な料理名を獲得することができた。また、雑学文抽出手法については太田らの手法に対して有意差を確認することができた。対話システムについては、対応可能度の有効性を確認することができなかった。今後の課題として、雑学文抽出の各指標の重み調整や、同一トピックにおける雑学文出力回数の制限などがあげられる。

**謝辞** 本研究の一部は平成 29 年度 FAIS 新成長戦略推進研究開発事業（実用化研究開発事業）の助成を受けた（株）コン

ピュータサイエンス研究所からの受託研究によるものです。

## 文 献

- [1] Steve Young, Milica Gašić, Blaise Thomson, and Jason D Williams. Pomdp-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, Vol. 101, No. 5, pp. 1160–1179, 2013.
- [2] Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gasic, Lina M. Rojas Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pp. 438–449, Valencia, Spain, April 2017.
- [3] Oriol Vinyals and Quoc Le. A neural conversational model. *arXiv preprint arXiv:1506.05869*, 2015.
- [4] Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. Deep reinforcement learning for dialogue generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, D16-112*, pp. 1192–1202, Valencia, Spain, November 2016.
- [5] 加藤大介, 宮部真衣, 荒牧英治, 灘本明代. インターネット上のメディア毎の「おいしさ」表現比較分析. In *DEIM Forum2015*, F5-5, 2015.
- [6] 太田知宏, 鳥海不二夫, 石井健一郎. 発話生成を目的とした wikipedia からの文抽出. 人工知能学会第 23 回全国大会, 2G1-NFC5-11, 2009.
- [7] 杉本俊, 植木拓, 林宏幸, Nichols Eric, 中野幹生. Wikipedia からの特定ドメインの雑談対話システムのための発話候補文集合の獲得. 人工知能学会第 31 回全国大会, 3A1-3, 2017.
- [8] 稲葉通特, 吉野友香, 高橋健一. 対話システムが話し手役になるためのオープンドメイン独話生成. 人工知能学会論文誌, Vol. 31, No. 1, pp. DSF-F\_1-9, 2016.
- [9] 江頭勇佑, 柴田知秀, 黒橋禎夫. 雑談対話システムにおける強化学習を用いた応答生成モジュールの選択. 言語処理学会第 18 回年次大会発表論文集, D3-7, Vol. 18, pp. 654–657, 2012.
- [10] 工藤拓, 賀沢秀人. Web 日本語 N グラム第 1 版. 言語資源協会発行, 2007.
- [11] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, Vol. abs/1301.3781, pp. 1–12, 2013.