社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
IN2018-81 (2019-01)

SNSトレンドを始点としたユーザの関心を煽る雑学探索

松田　純哉† 　荒澤　孔明† 　渡邊　稜平† 　服部　　峻††

†,†† 室蘭工業大学 ウェブ知能時空間研究室　〒 050–8585 北海道室蘭市水元町 27–1
E-mail: †{15024160,18096001,18043050}@mmm.muroran-it.ac.jp, ††hattori@csse.muroran-it.ac.jp

あらまし　雑学とは，くだらないことや些細なことの役に立つか立たないかを問わない知識のことである．そのような性質に反して人々の話の種として度々活用され，テレビ番組のネタとして使用されるなど，雑学は知識としてだけでなくコミュニケーションツールとしても活用することができる．しかし，日常生活上で雑学を知る機会というのは限られており，わざわざ雑学を調べる人もほとんどいない．また，Web 上には無数の雑学が存在しているが，Web の情報量の規模故に現時点では，面白い雑学を探し出す明確な方法が存在していない．そこで，世の中の流行を表すトレンドを始点として雑学を探索することで，より多くの人が関心を持つ面白い雑学を探し出すことができると考えた．本稿では，情報の流動性が高い SNS のトレンドを対象とし，ユーザの関心を煽る多種多様な雑学を導き出す手法について検討する．

キーワード　雑学探索，SNS，情報抽出，Web マイニング，機械学習

Searching the Web for Trivia Inducing Users' Interests by Starting with SNS Trends

Junya MATSUDA†, Komei ARASAWA†, Ryohei WATANABE†, and Shun HATTORI††

†,†† Web Intelligence Time-Space (WITS) Laboratory, Muroran Institute of Technology
27–1 Mizumoto-cho, Muroran, Hokkaido, 050–8585, Japan
E-mail: †{15024160,18096001,18043050}@mmm.muroran-it.ac.jp, ††hattori@csse.muroran-it.ac.jp

Abstract　Trivia are one kind of knowledge about trashy things and trivial things etc., and almost always we do not care whether or not they are useful for us. Contrary to such nature that trivia have, trivia can be utilized as knowledge as well as a communication tool. For example, they are often used as a topic of conversation by people, and moreover as a topic of TV programs etc. However, there is little opportunity for us to learn about trivia in our daily lives, and also there are few people who bother to search for trivia purposely. In addition, there is currently no clear way to search the Web for funny trivia due to the explosively-growing information volume of the Web, although there are countless trivia on the Web. Therefore, we propose a novel way to search the Web for funny trivia that more people are interested in, by using a trend word showing a trend of the world as a starting point of the searching. This paper examines a method to search the Web for various kinds of trivia inducing users' interests, by focusing on SNS trends with high mobility of information.

Key words　Trivia Search, SNS, Information Extraction, Web Mining, Machine Learning

1. まえがき

「江戸時代にはオナラをした人の身代わりになる役職があった」などといった知って得するかどうかを考えない面白い知識のことを雑学と言う．雑学は，一般的にはくだらないものとされるが，会話を盛り上げる題材として十分に活用可能である．2000 年代中期には，視聴者から投稿された雑学を紹介する「トリビアの泉 ～素晴らしきムダ知識～」というテレビ番組が放送されており，ゴールデンタイムでのレギュラー放送が平均視聴率 17.8%を獲得し，それが 2003 年以降に放送を開始したバラエティ番組では 1 位である [1] ことから，雑学の娯楽としてのポテンシャルには非常に大きな可能性が秘められていると考えられる．だだし，日常生活を送る上でそこまで重要となる知識ではない為，調べるほど価値のある情報とはされず，実際に調べる人はかなり少数である．また，「テレビ」や「雑誌」などの限られたメディアから知る程度で，日常

生活上で雑学を知る機会というのは限られている．ここで，その知っておいて損はない有用な雑学というものを，まだまだ未知の雑学が眠っているであろう Web という情報が膨大に蓄積されているデータの中から探し出し，提供できないかと考えた．しかしながら現時点では，雑学を探し出し面白さを測る明確な方法が存在していない．そこで，世の中の流行を表すトレンドを始点として雑学を探索することで，より多くの人が関心を持つ面白い雑学を探し出すことができると考えた．

2. 提案システム

本章では，最終的な目標である Web 上から自動的に雑学を集め，面白さ順にランキング表示することを達成する為の雑学探索システムについて提案する．まず，提案システムの対象である雑学の特徴に関して整理した後，システム全体の概要，及び，各処理の詳細について述べていく．

2.1 雑学のパターン化

雑学とは，多岐のジャンルにわたる系統立っていない様々な事柄についての知識であり，その大きな特徴として，知識としての面白さを重視しているものである．その雑学は大きく分けて 2 つのパターンに分けられる．まず 1 つ目のパターンとして，「和菓子のコンペイトウは，角が 24 個あるのが良品とされている.」などといった 1 つの文で成立する雑学（以下，単文成立型雑学と呼ぶ）が挙げられる．

一方で，知識の文には大抵主語が存在しているが，指示語などによって「どこで」「何が」といった情報が文中に存在しない場合がある．具体例で言うと「また，防水にするには，コストがかかるとともに完全密封になる為，熱がこもりやすくなり壊れやすくなることがある.」といった文では「何が」に当たる情報が欠損しており，この文だけではどういった知識なのか意味が伝わらない．この文は前文の「日本産のケータイには防水が多いが，海外では防水ケータイ，スマートフォンはほとんど存在しない.」という文があって初めて意味が伝わる．これが 2 つ目のパターンの複数の文が集まることで初めて意味が伝わる雑学（以下，複文成立型雑学と呼ぶ）である．ただし，複文成立型雑学は基本的に情報量が多く，読むことだけでユーザに負荷がかかる．また，文と文の関係が複雑化しているものであれば，意味を理解できるまで時間を要することになる．そういった意味で複文成立型雑学は単文成立型雑学に比べストレスがかかりやすく，雑学としての面白さ，分かりやすさという点で単文成立型雑学の方が優れていると考える．

以上より本稿では，雑学探索の対象として単文成立型雑学を中心に Web 上から雑学を抽出し，面白さ順にランキングして提示することを目指す．

2.2 雑学探索の始点の決定

Web 上で雑学を見つけようとする時，Web 検索におけるキーワードが必要となる．トレンドキーワードは対象の Web リソースにおいて多くのユーザの検索対象となっているワードであり，現時点でのユーザの興味関心を反映しているワードとも言える．また，Web リソースの種類に依って情報の伝播する速度は変化し，特にツイートと呼ばれるメッセージや画像等を投稿できる SNS の 1 つである Twitter においては，リツイートという投稿に対する拡散機能により，トレンドキーワードが伝播する速度が他の Web リソースに比べて突出している．実際に吉田ら [2] の研究でも，Twitter のトレンドキーワードの言及量とトレンドの即時性の高さについて述べられている．その為，Twitter は Web 上で比較的早くトレンドキーワードを掴むことが可能であり，早期のトレンドに関する雑学は今後話題性を持つ可能性が高いと考えられる為，Twitter のトレンドキーワードを雑学探索の始点とすることが最適であると考える．しかし，トレンドキーワード中にはハッシュタグと呼ばれるツイートの共有目的に利用されるものも含まれる場合があり，これは純粋なトレンドキーワードとは言えないので，本稿では除くこととする．

2.3 雑学探索システム全体の概要

本稿における雑学探索システムの流れを図 1 に示す．始めに，Twitter からその時点におけるハッシュタグを除いたトレンドキーワード上位 10 件を抽出し，それぞれそのまま検索エンジンに入力する．そのトレンドキーワード 1 つから関連性を持つ幅広い分野に属する雑学の探索を可能とする為に，トレンドキーワードで Web 検索した結果表示された Web ページの要約テキスト各々から人物，場所などといったトレンドキーワードに関連する名詞を雑学検索単語として抽出する．これらを用いて再度 Web 検索を行い，検索結果（URL）からリンクされている各 Web ページのコンテンツより雑学となるテキストを抽出する．このように抽出された雑学の面白さを評価し，その後ランキングしてユーザに雑学を提示する．以上の流れでユーザの関心を煽る雑学提供を目指す．

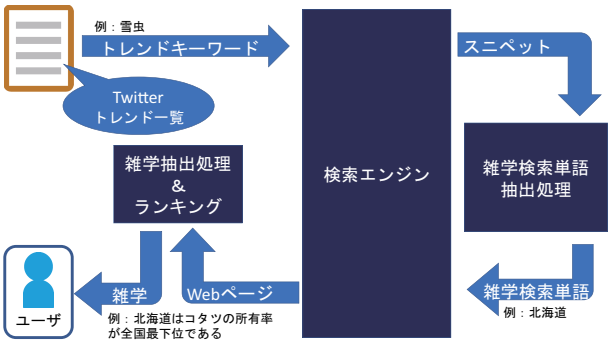

図 1 システムの流れ

2.4 雑学検索単語抽出処理

Twitter から雑学探索の始点となるトレンドキーワードを取得後，Web 検索を行う．そして検索結果として表示された Web ページから検索結果（URL）各々のスニペットを取得する．スニペットとはそれぞれの検索結果（URL）の Web ページの要約文であり，その Web ページ内でトレンドキーワードに関連した部分がテキストとして抽出されたものである．取得したスニペットから雑学検索単語を取得する為に Google Cloud Natural Language API [3] を用いる．この API によって，入力されたテキストに含まれている既知の名詞を「人物，場所，組織，イベント，メディア，商品」の 6 種類にラベル付

けした名詞をエンティティとして抽出することが可能となり，本稿では雑学検索単語として使用する．また，ラベル付けによって，より少ない雑学検索単語で幅広いカテゴリの雑学を探索することも可能となる．

ただし，雑学と言ってもユーザが知らない主題の雑学は関心を煽ることができないと考えられる．その為，比較的認知度が高い雑学検索単語に絞る為に，各ラベルの雑学検索単語毎に Google での検索数を比較し，検索数がトップの雑学検索単語をそのラベルにおける高認知度の雑学検索単語として抽出を行う．今後，パーソナライズも必要であると考える．

2.5 雑学抽出処理の概要

雑学が載っている Web ページを見つける為に，2.4 節で抽出された雑学検索単語と「雑学」という単語を組み合わせて AND 検索を行う．この時，検索結果として表示された上位約 10 件で各 URL からリンクされた先の Web ページ内のテキスト部分全体を抽出する．そして抽出したテキストを「。」「！」「？」「!」「?」を区切り文字として文単位に分割し，雑学検索単語が含まれている文を雑学候補のテキストとして抽出する．この雑学候補となるテキスト群に対し，雑学かどうかの判定，及び，雑学としての面白さを評価し，最終的に，これまで抽出された雑学検索単語それぞれの雑学全てを 1 つの面白さのランキングにしてユーザに提示する．

3. 提 案 手 法

本章では，雑学探索システムによって得られた Web ページ内のテキスト部分全体から雑学だけを抽出する手法，その取得した雑学の面白さをギャップの大きさという観点から評価する手法について述べる．

3.1 テキスト分類による雑学のピックアップ

雑学探索システムによって得られた Web ページ内のテキスト部分全体にはアフィリエイト目的の広告による文や，雑学とはなんら関わりもない文が多分に含まれている．このままではユーザに雑学を上手く提供することができないので，これらのノイズとなる文を除去する必要がある．そこで，以下の 2 つの手法について提案する．

- 機械学習による分類

雑学の文とノイズの文とを分けることができれば良いので，文のある単語から，その前後に出現する単語を推測するモデルを生成するアルゴリズムである Skip-gram を実装した機械学習の手法の 1 つである fastText [4] という手法を利用する．この fastText で大量の学習用テキストに対し，「雑学」，「ノイズ」とカテゴリ別ラベルを付与し，Skip-gram にて教師付き学習を行うことでモデルを構築する．このモデルを活用することで新たなテキストに対する「雑学」，「ノイズ」のカテゴリ分類を自動化可能にする．

- ルールベースによる分類

雑学，ノイズ双方にはそれぞれ特徴的な単語が存在しており，特にノイズに分類される文には特徴的な単語（以下，ノイズキーワードと呼ぶ）が多く含まれている．具体例としては，「いたします」，「スポンサーリンク」，「PR」などが挙げられる．このノイズキーワードをより多く見つけ，ノイズキーワードが含まれている文をノイズ文として除去する．

3.2 雑学の面白さの判定

面白い雑学を見た場合，「へぇ～そうなのか！」という感想が思い浮かぶ傾向にある．この感想が浮かぶ要因として，文章に感じる意外性が面白さを生んでいると考えられる．意外性を感じさせる雑学のパターンとしては「A が実は B だった！」，「A は過去に B をしていた！」というようなものが多く，A と B のギャップが雑学としての面白さの肝であると分かる．このギャップが生まれる要因として，雑学中の A と B に当たるワードの類似度が低いのではという仮説を立てた．この A と B の類似度を評価するには，単語をベクトル化しそのコサイン類似度の算出をもって評価する方法が挙げられる．そこで，本稿ではコサイン類似度を算出するのに fastText と同じく Skip-gram を実装した Word2Vec [5] という手法を利用する．この手法により各単語をベクトル表現化でき，単語間のコサイン類似度を算出することが可能となる．

まず，事前実験として A と B のコサイン類似度を算出することで雑学の面白さを評価することができるか検証する．文中の A と B に当たる単語の抽出は手作業で行い，Word2Vec の学習モデルには白ヤギコーポレーション [6] が提供している日本語版 Wikipedia を学習したものを利用する．実験に使用するデータは，以前放送されていた人気バラエティ番組「トリビアの泉」にて紹介された雑学全 1034 件 [7] を使用する．この番組において雑学は「へぇ」という 0 から 100 までの値で雑学の面白さが評価されるので，各獲得「へぇ」数ごとにデータ数の上限値を 5 個に定め，各々の雑学から抽出した A と B に当たる単語間のコサイン類似度を算出し，各雑学の面白さ（へぇ数）とコサイン類似度との関係性を表す散布図を作成すると，図 2 のようになる．

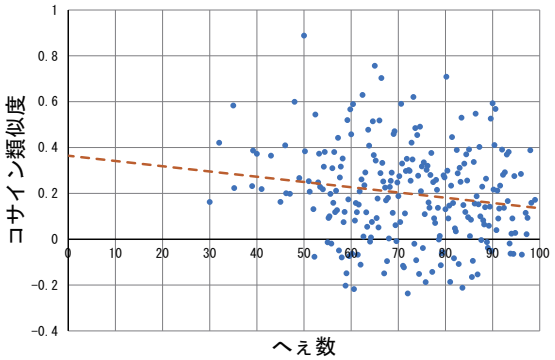

図 2 各雑学におけるへぇ数と AB 間のコサイン類似度の相関関係

図 2 においてコサイン類似度が低いほどギャップが大きくなり，面白い雑学と評価される．また，線形近似（回帰直線）をオレンジの点線で示している．実験結果から，相関係数 $r = -0.165690091$ となり，若干ではあるが「へぇ」数とコサイン類似度に負の相関の傾向が見られた．しかし，「へぇ」はあくまで番組基準の面白さ評価値であり，その 1 文に関わる補足トリビアの紹介によっても「へぇ」数は伸びる傾向にあったので，「へぇ」数が単純にその雑学の評価となっていない場

合が多く含まれていた為，仮説通りではなかった．

また，この事前実験では A と B に当たる単語を手作業で選出していたが，システム化に向けて Support Vector Machines に基づく日本語係り受け解析器である CaboCha [8] を利用する．この CaboCha を利用することで，ある単語に係る単語が自動的に抽出でき，大規模な雑学データに対する評価を容易に行うことが可能となる．

しかしながら，事前実験では雑学中の A と B に当たる単語が Word2Vec の学習モデルに含まれていない未知語であった場合，単語のベクトル表現が得られず，意図的にその雑学の評価を避けることしかできないことが判明した．そこで，3.1 節で提案した手法で利用されている fastText を利用して，コサイン類似度を算出する．この fastText には単語をさらに分割した部分語を利用したベクトル表現の獲得が可能となる subword [9] という仕組みが取り入れられている．この仕組みにより Word2Vec ではベクトル表現化できなかった未知語に対しても対応可能となる．例えば，Word2Vec では未知語である「ゴジラ座」は，fastText では部分語「ゴジラ」と「座」に分割され，その各部分語のベクトルの和として「ゴジラ座」のベクトル表現が可能となる．

4. 評価実験

本章では，雑学検索単語で Web 検索された結果の各 Web ページ中の雑学候補の文からノイズを取り除き，雑学のみを抽出する手法，及び，雑学の面白さを定量的に評価する手法に関する評価を行う．

4.1 Web ページ上の雑学の抽出手法の評価

本節では，3.1 節で提案した手法によって Web ページ中から抽出された雑学とノイズがどの程度正しく分類できたかを明らかにしていく．

- fastText を用いた機械学習による分類

ここでは機械学習の fastText による分類精度の検証を行う．使用するデータは，雑学収集システムから取得できた Web ページのテキストより，単体で雑学として意味が伝わる文のみを雑学カテゴリの正解データとし，それ以外の文はノイズカテゴリの正解データとして手作業で分類する．また，2018 年 11 月 28 日 16 時頃に実施した雑学収集システムにより収集し，テキスト全 4383 件を対象に分類した雑学カテゴリ 561 件，ノイズカテゴリ 3822 件から各 500 件を学習データとする．それとは別に雑学カテゴリ 61 件，ノイズカテゴリ 477 件を判定データとして取り扱う．分類精度に大きく直結するパラメータとして主に学習データ数，単語ベクトルの次元数，学習を繰り返す回数であるエポック数の 3 要素がある為，それぞれ数値を変更した場合ごとの精度を比較する．

図 3 に雑学カテゴリ，ノイズカテゴリ各 50 件を合わせた学習データ数 100 件，各 100 件を合わせた 200 件，300 件，...，1000 件ごとにそれぞれ学習を行ったモデルを利用した際の分類精度について示す．図 3 から見ても分かる通り学習データ数の増加に依る分類精度の向上は見られなかった．唯一，学習データ数 700 から 800 にかけて一時的に分類精度が向上しているが，これは学習データの内容に依るところが大きく，学習データ数の増量には関わっていないと考えられる．

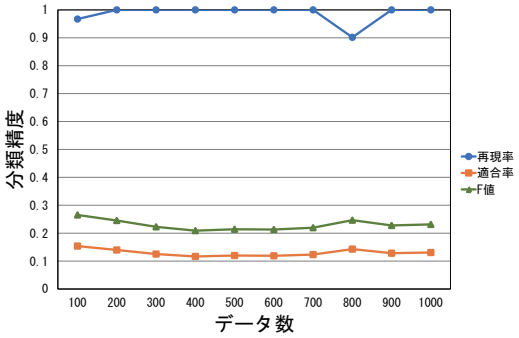

図 3　学習データ数に依る機械分類精度の変化

次に図 4 に，前述の図 3 で F 値が最も高かった学習データ数 100 件で次元数 50，100，150，...，300 ごとにそれぞれ学習を行ったモデルを利用した際の分類精度について示す．図 4 から見ても分かる通り次元数が低くなると再現率は急激に悪化している．一方，適合率に関しては，安定的な推移となっている．F 値で見ると，次元数 100，150 辺りが最も高い値を取っており，これは本稿で取り扱っているデータセットの大きさが小さい為，低い次元が適当であるからであると考えられる．

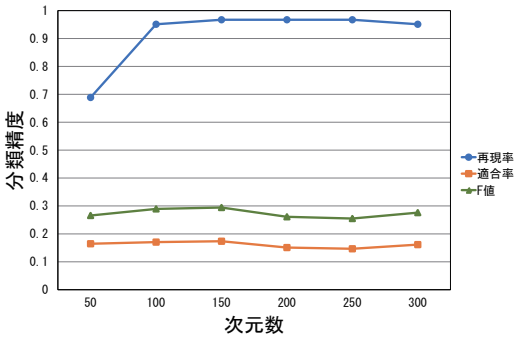

図 4　単語ベクトルの次元数に依る機械分類精度の変化

最後に図 5 に，前述の図 3 で F 値が最も高かった学習データ数 100 件でエポック数 5，10，15，...，30 ごとにそれぞれ学習を行ったモデルを利用した際の分類精度について示す．図 5 から見ても分かる通りエポック数の増加に依って，再現率は好転しているが，適合率，F 値はともに悪化している．

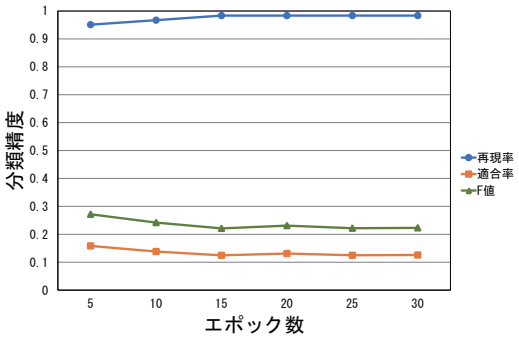

図 5　エポック数に依る機械分類精度の変化

上記よりまとめると，次元数は対象となる学習データ数に応じて適宜調整が必要となるが一般的に増やせば増やすほど分類精度が向上するとされている学習データ数とエポック数の増加に依る分類精度の向上は見られなかった．このような結果となってしまった大きな原因として，複文成立型雑学がノイズカテゴリの学習データとして含まれてしまっている為，学習データのノイズとしての専門性が低くなってしまい，分類精度の低下に繋がっているかと考えられる．また，雑学とノイズの双方に言えることであるが，そもそもこれらの取り扱っている文書は多岐のジャンルにわたる，系統立っていない様々な事柄について述べられているものであり，出現する単語の傾向が表れにくいカテゴリである為，Word2Vec と同じく Skip-gram を実装している fastText ではこれらのカテゴリ特徴を捉えにくいと考えられる．

- ルールベースによる分類

ここではノイズキーワードを用いたルールベースによる分類精度の検証を行う．本稿で，ノイズカテゴリに分類される文から手動で選出したノイズキーワード全 161 個を用いて分類を行う．選出したノイズキーワードの一例を図 6 に示す．

あの，あるある，あるよ，いかが，いきます，いたします，いただき，いるよう，うれしい，おすすめ，コツ，マジ，ヤバ，サイト，シェア，ハンパ，ベスト，ページ，リンク，オススメ，一方，一覧，出典，方法，以来，今回，比較，当時，追記，記事，ID，PR，TOP，NAVER，by，jp，com，etc，jpg，net，↑，→，↓，←，⇒，・・，…，※，>，≫

図 6　ノイズキーワードの一例

選出した単語群が 1 つでも含まれている文をノイズカテゴリと分類し，それ以外の文を雑学カテゴリとして分類することで，正しい分類を行うことができるのかを検証する．精度評価には，fastText による分類時に利用した判定データをこの手法でも利用する．

表 1 に前述の fastText における最も精度が高い，学習データ数 100 件，次元数 150 の場合の精度と，ルールベースによる分類精度を示す．表 1 の F 値に注目すると分かるように単純な精度を比較するとルールベースによる分類の方が優れているという結果となった．また，分類が上手く行われなかった文について分析すると，ノイズキーワードに含まれる「あの」などの指示語による分類ミスが多く発生していた．複文成立型雑学をノイズとして分類する上で指示語をノイズキーワードに選出しており，雑学とノイズの手動による分類は指示語が文中に含まれていても，その文単体で意味が伝わるものであれば単文成立型雑学として分類を行っている．具体例としては「その初テレビ放送の第一声は“JOAK-TV，こちらは NHK 東京テレビジョンであります”でした.」などがある．よって，ノイズキーワードに含まれている指示語の存在が，そのようなケースの雑学をノイズとして分類するということが再現率の低下の主な要因になっていると判明した．

以上，2 つの手法についてまとめると，本稿では単文成立型雑学に限定して分類を行った為，複文成立型雑学がノイズとして扱われることとなってしまい，このことが両手法いずれにも悪影響を与えていると考察できる．また，Web ページから雑学候補テキストを取得する際，文の分割処理が「。」「！」「？」「!」「?」の 5 つの区切り文字によって行われているが，必ずしも文末がこれらの区切り文字になっている訳ではなく，上手く分割できていない文も見受けられた．よって，その分割不備が同じように両手法に悪影響を与えていると考えられる．これらの問題の有効的な解決策として，形態素解析などにより前文との関係性を分析することで，複文成立型雑学をまとめて一つとして取り出す，または正しく文を分割することなどが挙げられる．更に精度が上昇する手法として，ルールベースによる分類は，ルールを増やし厳しくすればするほど適合率が上がるので，適合率 9 割程度までルールを増やし分類する．その後，ルールベースによる分類によって失われた再現率を機械学習による分類で補うという組み合わせも考えられる．

表 1　機械学習とルールベースの分類精度

手法	再現率	適合率	F 値
fastText	0.967213	0.173529	0.294264
ルールベース	0.524590	0.359551	0.426667

4.2　雑学の面白さ算出手法の評価

本節では，3.2 節で提案された手法によって Web ページ中から抽出された雑学候補の面白さがどの程度の精度で評価されるのかを検証する．使用するデータは，4.1 節で利用していた雑学カテゴリ 561 件を判定データとして取り扱う．この判定データ 1 件ごとに，雑学検索単語に CaboCha を適用し，雑学検索単語に係る単語を抽出する．そして，その 2 単語間のコサイン類似度を算出後，コサイン類似度が低い順にランキングすることで雑学の面白さを評価する．コサイン類似度の算出には Word2Vec と fastText を適用する．Word2Vec の学習モデルは 4.1 節で適用していたものを利用し，fastText の学習モデルは Facebook Open Source から公開されている日本語版 Wikipedia を学習したもの [10] を利用する．また，両手法によってランキングされた雑学を順位に基づいて等間隔にそれぞれ 30 件雑学をピックアップし，被験者 40 人に面白いと思った雑学を任意の数，選択してもらうアンケートを実施した．ランキングの一例として，2018 年 11 月 28 日 16 時頃に取得し，ハッシュタグの除いたトレンドキーワード上位 10 件を始点に探索した雑学ランキング結果を図 7 に示す．

そして，Word2Vec と fastText の両手法により単語間のコサイン類似度を算出し，各雑学の被験者に面白いと選択された数とコサイン類似度との関係性を表す散布図を作成すると，図 8 と図 9 のようになる．

図 8 と図 9 において被験者 40 人の内の選択数が大きいほど面白い雑学となる．また，線形近似（回帰直線）をオレンジの点線で示している．実験結果から，Word2Vec では相関係数 $r = -0.01544$ となり，fastText では相関係数 $r = -0.33708$ となった．理想では強い負の相関が得ることができれば良かったが，結果として両手法ともにはっきりとした相関は得られなかった．しかし，Word2Vec に比べて fastText は若干負の

【Word2Vecでのランキング】
1位.　「六甲おろし」と言えば間違いなく通じるので、
　　曲名だと思っている方は大半ですが、正しくは「阪神タイガースの歌」です。
2位.　1980年代、ノルウェーの水産業者が日本を訪れたことがきっかけで、
　　日本にサーモンが輸入されることになった。
　　⋮
29位.　42.チンパンジーと人間の体毛の数は変わらない。
30位.　えきねっとを利用した予約方法であれば、
　　従来の手順に比較して手間が減らせて手軽にチケットの購入が済ませられるのですが、
　　優れているのは購入に際する手順に限らず購入時に支払う料金も例外ではありません。

【fastTextでのランキング】
1位.　36.日本には「謎のフルーツ味」、「天才エネルギー」など、
　　奇妙なフレーバーのファンタが70種類以上ある。
2位.　25.日本にはウサギだらけの島があるウサギ島よばれるうさぎたちの楽園は、
　　広島県の大久野島、瀬戸内海にある周囲4. 3kmの小さな無人島にある。
　　⋮
29位.　しかし、チャーチルがヒトラーに対してVサインをした写真が世界に出回った際に、
　　現在の「平和」という意味でピースサインが広まりました。
30位.　■補足同様に、手に汗をかくという行為は、木の枝をつかみやすくして
　　すばやく逃げるためという、人が木の上で生活をしていた時のなごりだそうです。

図 7　Word2Vec と fastText によるランキングの一例

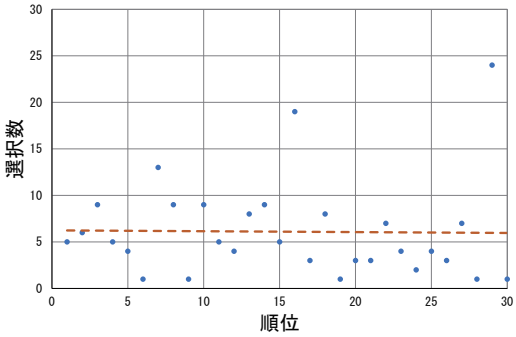

図 8　Word2Vec によるコサイン類似度に基づく雑学ランキングと主観的面白さ評価との相関関係

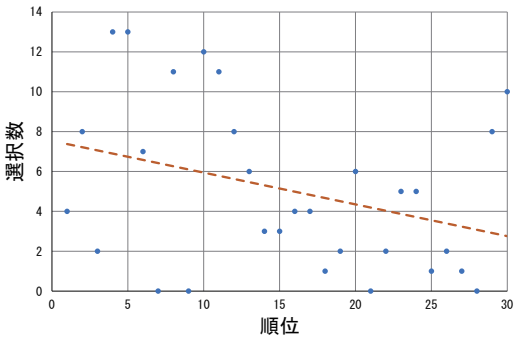

図 9　fastText によるコサイン類似度に基づく雑学ランキングと主観的面白さ評価との相関関係

相関が得られており，比較的 fastText による雑学の評価の方が優れていると言える．はっきりとした負の相関が得られなかった原因としては，雑学検索単語 A に対して係り受け関係にある単語はコサイン類似度を算出する対象 B として必ずしも適当ではないことが大きな要因であると考えられる．現時点では，手動以外で最も適当な単語を抽出することは不可能であるが，コサイン類似度を算出する単語を 1 つのペアに絞り込まずに，全係り受け関係にある単語の平均コサイン類似度をその雑学の評価とする手法，もしくは全係り受け関係にある単語中から最も低いコサイン類似度をその雑学の評価とする手法などが，評価の改善に繋がるのではと考えられる．ただし，これらの手法では雑学検索単語との関連性をほとんど失ってしまう為，単語間のギャップとは別の評価基準を見つけるべく，高く評価されている雑学の更なる分析が必要である．また，表 2 からも分かる通り fastText の評価可能雑学数が Word2Vec に比べ 2 倍近い差があることから，ランキング対象数が大幅に減少し評価精度が下がったのではとも考えられる．ちなみに，判定データの 561 件に対し fastText の評価可能雑学数が 510 件に減少している原因は，CaboCha によって雑学検索単語に係る単語が抽出できなかった文に因るものであると考えられる．

表 2　雑学の面白さ算出手法における Word2Vec と fastText のコサイン類似度算出不可回数と評価可能雑学数の比較

手法	コサイン類似度算出不可回数	評価可能雑学数
Word2Vec	548	229
fastText	0	510

5. ま と め

本稿では，単文成立型雑学を中心に Web 上から幅広いカテゴリの雑学を探索し，収集した雑学の面白さを評価してランキングする雑学探索システムを提案した．この提案によって，多少ではあるが Web 上の雑学コンテンツのマイニングが容易になった．また，雑学と雑学ではない文を分類する幾つかの手法，及び，雑学の面白さをギャップという観点から定量的に評価する手法を考案し，評価実験を行った．評価実験の結果から，今後の最優先に解決すべき問題として複文成立型雑学の認識，及び，取得が明らかとなった．

文　献

[1] Video Research Ltd.（2003）2003 年 年間高世帯視聴率番組 30（関東地区）, <https://www.videor.co.jp/tvrating/past_tvrating/top30/200330.html>.
[2] 吉田 光男, 荒瀬 由紀, “トレンドキーワードに関するウェブリソースの横断的分析,” 情報処理学会論文誌, データベース, Vol.9, No.1, pp.20–30 (2016).
[3] Google（2018）Google Cloud Natural Language API, <https://cloud.google.com/natural-language/?hl=ja>.
[4] Facebook Inc.（2018）fastText, <https://fasttext.cc/>.
[5] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean, “Distributed Representations of Words and Phrases and their Compositionality,” Advances in Neural Information Processing Systems 26, pp.3111–3119 (2013).
[6] Tanida Kazuaki（2017）word2vec の学習済み日本語モデルを公開します, <http://aial.shiroyagi.co.jp/2017/02/japanese-word2vec-model-builder/>.
[7] Noncky.（2011）トリビアの泉 パーフェクトデータベース, <http://www.noncky.net/trivia/>.
[8] 工藤 拓, 松本 裕治, “チャンキングの段階適用による日本語係り受け解析,” 情報処理学会論文誌, Vol.43, No.6, pp.1834–1842 (2002).
[9] Piotr Bojanowski, Edouard Grave, Armand Joulin, Tomas Mikolov, “Enriching Word Vectors with Subword Information,” Transactions of the Association for Computational Linguistics, Vol.5, 2307-387X, pp.135–146 (2017).
[10] Facebook Inc.(2018)Wiki word vectors･fastText, <https://fasttext.cc/docs/en/pretrained-vectors.html>.