常識的知識グラフ及び単語埋め込みを用いた重畳型駄洒落ユーモア認識

志方脩　谷津元樹　森田武史
青山学院大学理工学部
a5819059@aoyama.jp　{yatsu,morita}@it.aoyama.ac.jp

概要

機械による言語的なユーモアの認識能力を向上させるため, 駄洒落ユーモアのうち文内に潜在的な繋がりを持つ単語のペアを含む重畳型駄洒落の検出手法を提案する. 井上ら [1] が提案した常識的知識グラフを用いる手法に加えて, 単語埋め込みを活用することにより, 重畳型駄洒落の検出性能の精度の向上が確認された. 本稿では, 定性的評価実験において確認された, 重畳型駄洒落検出の成功事例, および失敗事例における主な失敗要因について説明する.

1　はじめに

近年機械がユーモアを表出・理解することに対する関心が高まっており, それに接するユーザの生活の質を向上させることが報告されている．人間のコミュニケーションは，画一的な形式によらず自由であり駄洒落やなぞかけに代表される言語的なユーモアを含んでいる．ユーモアな表現を機械が理解することによって人間同士の会話に近い自然な対応をすることができる.

1.1　研究課題と目的

ユーモアはジェスチャーなどを用いる身体的なユーモアと，駄洒落やなぞかけに代表される言語的なユーモアに大きく分類される．中でも駄洒落は比較的に認識が容易であり年齢層を問わず親しみやすいものと考えられる．知識グラフや自然言語処理技術を用いて駄洒落ユーモアの検出や理解の有効性を高めるためには下記の課題が存在する．後述する滝澤の研究 [2] によれば，駄洒落には併置型駄洒落と重畳型駄洒落の 2 つの分類が存在する．以下にそれぞれの例文と構造を示す.

(1) **猫が寝ころんだ (併置型駄洒落)**

この例は「猫」と「寝こ」のように，音韻的に類似するペアが文内に存在しているため, 駄洒落であると認識することができる.

(2) **鮭の卵は、いくら？ (重畳型駄洒落)**

しかしこの例は併置型のように音の類似するペアがないことから従来の手法では検出が困難である. 例文 (2) では,「鮭」と「いくら」の間に何らかの潜在表現を媒介する関係があると予測される. その場合, 外部の知識や単語埋め込みを用いることでその潜在的な表現及び関係を見出すことができる. すなわち,「『鮭』は『イクラ』という性質を持つ」という関係性である. このとき, 潜在表現である「イクラ」は文内の「いくら」に音韻的に一致するので, 上記の文を駄洒落文として認識できる. 本研究の目的は重畳型駄洒落に存在するこのような潜在的な表現及び文内の語との関係性を, 常識的知識グラフや単語埋め込みを用いて発見することである.

2　関連研究

はじめに，大規模知識グラフについて述べる．主要な大規模知識グラフとして著名なのが DBpedia[3] や Wikidata[4] であり，セマンティック Web 技術によるフリーの知識ベースであり，コミュニティベースで作成され，他のオープンデータとのリンクを持つリソースから構成されている.

ConceptNet[5] とは DBpedia，Wikidata と同様に集合知を知識源とした大規模な知識グラフである. ConceptNet は，DBpedia や Wikidata のような RDF による表現に類似した構造を持つ.

次に, 単語埋め込みについて述べる. 単語埋め込みとは自然言語処理で単語を表現する手法であり, 単語の意味をベクトル空間に埋め込むことで語彙を表現している.

単語埋め込みが取得できる手法として Word2vec [6] が著名である．Word2vec の他に, 近年では BERT [7] のような Transformer モデルが注目されている. 従来の Transformer モデルと比べて, BERT では双方向の文脈情報を学習しているため高い精度を示している. 本研究では Transformer モデルとして,

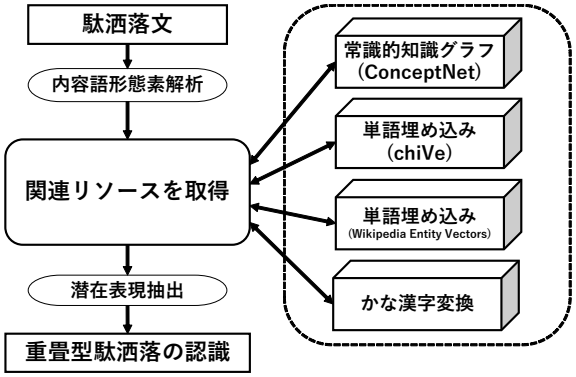

図 1　各手法を統合したシステムの構成図

BERT に基づく ja-ginza-electra[1)] を利用している．このため ja-ginza-electra は文脈に応じた意味を表す単語ベクトルを取得することができる．

駄洒落の検出・理解を目的とする手法の基礎に関わる提案は以前から行われている．滝澤 [2] は駄洒落を理解するシステムの構築の一環として併置型駄洒落と重畳型駄洒落を音素列や長さの一致などを基に分類し明確な基準を初めて提示している．また，谷津ら [8] は本研究の目的と同様に，ユーザによる発話をユーモアとして検出することを目的に韻文ユーモアである駄洒落の教師あり学習に基づく検出手法を提案している．そして先行研究である井上ら [1] は重畳型駄洒落の構造に着目し，常識的知識グラフを用いて潜在表現を抽出する手法を提案している．

3　提案

3.1　提案システム

手法は大きく分けて 4 つあり，先行研究で提案された知識グラフを用いた手法に加えて，単語埋め込み，Transformer モデル，かな漢字変換を用いたそれぞれの手法である．最終的には，知識グラフを用いた手法，単語埋め込みとして chiVe[9] と Wikipedia Entity Vectors[10] を適用したそれぞれの手法，かな漢字変換を用いた手法の計 4 つを統合し，いずれかの手法から検出成功が 2 件以上確認された文を重畳型駄洒落として認識する．統合したシステムの構成を図 1 に示す．

3.1.1　知識グラフを用いた手法

駄洒落文が入力されると形態素解析が行われ，内容語形態素が得られる．次に内容語形態素ごとに知識グラフから関連リソースを探索し，リソースの探索に成功した場合に音韻類似度関数を用いて潜在表現の抽出を試みる．本研究では大規模知識グラフとして ConceptNet 5 [5] を使用し，音韻類似性を検出するためにはレーベンシュタイン距離 [11] に基づく類似度関数を用いる．内容語と関連リソースの音韻類似度がしきい値を超えた場合，潜在表現の抽出及び重畳型駄洒落の検出に成功となり，すべての内容語及び関連リソースについて潜在表現が抽出できなかった場合，重畳型駄洒落の検出は失敗となる．

3.1.2　単語埋め込みを用いた手法

駄洒落文が入力されると形態素解析が行われ，内容語形態素が得られる．次に内容語形態素ごとに単語埋め込みを利用して関連リソースを探索する．具体的には Python の gensim ライブラリ[2)] を用いて単語埋め込みが取得できるモデルを利用し，内容語に対してコサイン類似度がしきい値 (0.40) 以上かつ上位 500 件以内の単語を関連リソースとして取得する．この手法では，単語埋め込みが得られるモデルとして chiVe [9], fastText [12], Wikipedia Entity Vectors [10] を利用している．そして取得した各単語に対して音韻類似度関数を用いて潜在表現の抽出を試みる．内容語と関連リソースの音韻類似度がしきい値を超えた場合，潜在表現の抽出及び重畳型駄洒落の検出に成功となり，すべての内容語及び関連リソースについて潜在表現が抽出できなかった場合，重畳型駄洒落の検出は失敗となる．

3.1.3　Transformer モデルを用いた手法

まず，形態素解析から得られた各内容語の全ペアにおいてコサイン類似度を算出する．次に内容語のペアを「と」で連結した文を生成し，その文における各ペアのコサイン類似度を再度算出する．「と」で連結した後に，コサイン類似度がしきい値 (0.30) 以上上昇し，さらに値自体もしきい値 (0.40) 以上の場合そのペアを潜在表現として検出する．潜在表現の抽出に成功した場合は重畳型駄洒落の検出に成功となる．すべての内容語において潜在表現が抽出できなかった場合，重畳型駄洒落の検出は失敗となる．

3.1.4　かな漢字変換を用いた手法

駄洒落文が入力されると形態素解析が行われ，内容語形態素が得られる．各内容語形態素のかな漢字変換から得られた単語が潜在表現の候補となる．具体的には，GoogleTransliterate[3)] を用いて，各内容語の

1）　https://github.com/megagonlabs/ginza

2）　https://radimrehurek.com/gensim/

3）　https://www.google.co.jp/ime/cgiapi.html

表 1　各手法での検出性能

手法名	適合率	再現率	F 値
知識グラフ	0.31	0.60	0.41
単語埋め込み (chiVe)	0.46	0.56	0.50
Transformer モデル	0.28	0.56	0.37
かな漢字変換	0.46	0.53	0.49
統合したシステム	0.53	0.56	0.55

新たな漢字変換を取得する. そして各候補と内容語のコサイン類似度を算出し, 値がしきい値 (0.40) 以上であった場合潜在表現として抽出する. 潜在表現の抽出に成功した場合は重畳型駄洒落の検出に成功となり，すべての内容語及び潜在表現候補について潜在表現が抽出できなかった場合, 重畳型駄洒落の検出は失敗となる.

4　評価と考察

提案手法を用い, 駄洒落データベース [13] に収録された 1103 文の重畳型駄洒落のうち, 68 文を研究の対象に指定した. 基準として, 重畳型駄洒落の特徴が明確になっている駄洒落文を要件とした. 特徴とは, 駄洒落文内の語に関連した潜在表現を介して, 音韻的に類似するペアが存在していることである. また, 正例・負例各 68 文からなるテストデータを作成し, 検出性能の評価を行った．負例は駄洒落データベースに収録されている併置型駄洒落文の中から無作為に 68 文を選択したものである.

比較実験に使用した手法は 3.1 節に基づき, 3.1.1 から 3.1.4 節の各手法及びこれらを統合した手法とした. ただし 3.1.2 節の fastText を適用した手法と 3.1.3 節の Transformer モデルを用いた手法は予備実験において検出精度が要求される水準を下回ったため, 統合したシステムより除外した.

評価の結果は表 1 に示す. 新たに単語埋め込みを用いた手法を加え統合したシステムでは適合率が 0.22 ポイント, F 値が 0.14 ポイントの上昇が確認された.

4.1　事例別の考察

本節では，認識の成功及び失敗の事例を対象に個別的に定性的評価を行い，今後の手法改良に繋がると考えられる発見について述べる.

4.1.1　成功例文

知識グラフを用いた手法

・煙とともに灰さようなら

内容語抽出を行うと「煙，灰」になる．探索した関連リソースと各内容語形態素との音韻類似の比較を行うと，「灰」に対し RelatedTo（灰– RelatedTo– 煙）という関係性で存在する「煙」が完全一致する. したがって，「煙」を潜在表現とした検出が成功する.

単語埋め込みを用いた手法

・育児休暇くれるの？サンキュー

内容語抽出を行うと「育児，休暇，サンキュー」になり, それぞれの単語における関連リソースを取得すると，「育児」とコサイン類似度が高い単語の上位 9 位に「産休」が存在し音韻類似の比較を行った結果, 入力文中の「サンキュー」と一致する. したがって,「産休 (サンキュー)」という言葉が潜在的な語として抽出することが可能となり検出成功となる.

Transformer モデルを用いた手法

・鮭の卵は、いくら？

内容語抽出を行うと「鮭，卵，いくら」になり, 各単語のペアのコサイン類似度を算出すると，「鮭」と「いくら」のコサイン類似度は 0.22 である. そして各ペアを「と」で連結させた「鮭といくら」という文章を生成し, 再度「鮭」と「いくら」のコサイン類似度を算出すると, 0.53 となる. したがって, 各ペアを「と」で連結した後でコサイン類似度の上昇値がしきい値を超えたため，「いくら」という言葉が潜在的な語として抽出することが可能となり検出成功となる.

かな漢字変換を用いた手法

・一番頭を使う作業は, 農作業!

内容語抽出を行うと「一番，頭，使う, 作業, 農, 作業」になり, それぞれの単語におけるかな漢字変換を取得し, 内容語形態素とコサイン類似度の算出を行う. すると，「農」のかな漢字変換に「脳」が存在し各内容語とのコサイン類似度の算出を行うと,「脳」と「頭」のコサイン類似度がしきい値を超える. したがって,「農 (脳)」という言葉が潜在的な語として抽出することが可能となり検出成功となる.

4.1.2　音韻類似の検出漏れや表記体系の違いによる失敗

・これを運ぶの？うん，そう.〈運送〉」

内容語抽出を行うと，「運ぶ，うん」となる.「運ぶ」の関連リソースに「貨車」,「貨車」の関連リソースに「運送」があるが，音韻類似度比較の対象が形

態素単位であるため,「うん，そう」のように途中に読点がある場合違う単語として処理されてしまい,音韻類似の比較を行うことができない.

4.1.3 知識グラフの関連リソース不足による失敗

・オリンピックなんて, 銅でもいい.

「オリンピック」という言葉の ConceptNet 上の関連リソースにおいて「銅」という言葉が存在しなかった. Wikidata を探索した場合は正解に相当する関連リソースとして発見されるため, 異なる知識グラフについての実装は今後の課題といえる.

4.1.4 単語埋め込みにおける関連リソース取得範囲外による失敗

・この坊主, いい読経してるぜ.

Wikipedia Entity Vectors での「坊主」と「読経」のコサイン類似度は 0.44 であり, しきい値は超えていたが類似度が高い単語の上位 500 件以内には入らなかったため, 潜在表現として取得できなかった.

5 おわりに

本研究の目的は，常識的知識グラフや単語埋め込みを用いて潜在的な音韻類似の対を含む駄洒落文の検出を行いユーモアな表現を機械が理解できるようにすることである. 先行研究から新たに単語埋め込みを利用することによって関連リソースの取得できる幅が広がり精度も向上した. 具体的には, 先行研究に相当する知識グラフを用いた手法に比べて, 新たに単語埋め込みを用いた手法を加え統合したシステムでは適合率が 0.22 ポイント, F 値が 0.14 ポイントの上昇が確認された. しかし探索の仕方や音韻類似度の比較においては, 依然として手法を見直さなければならない点がいくつか見受けられた.

5.1 今後の課題

知識グラフを用いた手法

先行研究から引き続き課題は 4 つである. 1 つ目は音韻類似度の比較を行う際に比較対象となる単語の文字数が異なる場合正確な比較ができないという点である．2 つ目の課題は英語表記及びローマ字表記の単語の比較ができないという点である．3 つ目は関連候補の出現数上限の制限により求める関連リソースの取得が困難，という点である．4 つ目は例文中に読点がある場合違う単語として処理され，音韻類似の比較を行うことができないという点である.

単語埋め込みを用いた手法

4.1 節の考察を経て得られた課題は 4 つである. 1 つ目は知識グラフを用いたシステムと同様, 音韻類似度の比較を行う際に比較対象となる単語の文字数が異なる場合正確な比較ができないという点である．2 つ目も知識グラフを用いたシステムと同様, 例文中に読点がある場合違う単語として処理され，音韻類似の比較を行うことができないという点である．3 つ目は内容語と潜在表現のコサイン類似度がしきい値を下回るという点である. 4 つ目は内容語と潜在表現のコサイン類似度がしきい値以上になっても, 上位 500 件以内に入らないという点である. これらの課題において, 関連リソースを取得する範囲の拡張や, コサイン類似度のしきい値の設定などにおいて, 最適な条件を考える必要がある.

Transformer モデルを用いた手法

4.1 節の考察を経て得られた課題は 2 つである. 1 つ目は各単語のペアを「と」で連結させた文において再度各ペアのコサイン類似度を算出すると, 値が下がってしまうという点である. 2 つ目は各単語のペアを「と」で連結させた文においてコサイン類似度が上昇した場合でも, 値自体がしきい値以上にならないという点である. 前提として 2 つの単語のコサイン類似度がしきい値の 0.40 を上回らない場合, その単語同士は似ているとは判断できないため潜在表現として取得できない. どちらの課題点においても,「と」以外にも「の」や「は」などで連結させてみたり, 他の Transformer モデルを利用してみることで解決できる可能性がある.

かな漢字変換を用いた手法

4.1 節の考察を経て得られた課題は 2 つである. 1 つ目は各単語を形態素解析する際に, 必要以上に細かく分割してしまったり, 読点を含んだり文字数が違ったりすると意図した漢字変換ができず, 検出に失敗するという点である. 2 つ目はコサイン類似度を算出する際に, 値がしきい値を下回るという点である. 今回はコサイン類似度の算出に chiVe を用いたが, fastText ではしきい値を上回ることができる例もあったので今後他のモデルも活用したシステムの実装が課題である.

謝辞

本研究は JSPS 科研費 JP21K12007 の助成を受けたものです.

参考文献

[1] 井上蒼一朗, 谷津元樹, 森田武史. 重畳型駄洒落ユーモアにおける常識的知識グラフを用いた潜在表現抽出. 言語処理学会第 28 回年次大会, F5-5, 2022.

[2] 滝澤修. 記述された「併置型駄洒落」の音素上の性質. 自然言語処理, Vol. 2, No. 2, pp. 3–22, 1995.

[3] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. Dbpedia-a crystallization point for the web of data. **Journal of web semantics**, Vol. 7, No. 3, pp. 154–165, 2009.

[4] Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. **Communications of the ACM**, Vol. 57, No. 10, pp. 78–85, 2014.

[5] Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In **Thirty-first AAAI conference on artificial intelligence**, 2017.

[6] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. **Advances in neural information processing systems**, Vol. 26, , 2013.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[8] 谷津元樹, 荒木健治. 子音の音韻類似性及び svm を用いた駄洒落検出手法. 知能と情報, Vol. 28, No. 5, pp. 875–886, 2016.

[9] 真鍋陽俊, 岡照晃, 海川祥毅, 髙岡一馬, 内田佳孝, 浅原正幸. 複数粒度の分割結果に基づく日本語単語分散表現. 言語処理学会第 25 回年次大会, pp. NLP2019–P8–5. 言語処理学会, 2019.

[10] Masatoshi Suzuki, Koji Matsuda, Satoshi Sekine, Naoaki Okazaki, and Kentaro Inui. A joint neural model for fine-grained named entity classification of wikipedia articles. **IEICE Transactions on Information and Systems**, Vol. 101, No. 1, pp. 73–81, 2018.

[11] Vladimir I Levenshtein, et al. Binary codes capable of correcting deletions, insertions, and reversals. In **Soviet physics doklady**, Vol. 10, pp. 707–710. Soviet Union, 1966.

[12] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. **Transactions of the association for computational linguistics**, Vol. 5, pp. 135–146, 2017.

[13] 荒木健治, 佐山公一, 内田ゆず, 谷津元樹. 駄洒落データベースの拡張及び分析 (ことば工学研究会 (第 58 回)(異次元) 空間と創作). ことば工学研究会: 人工知能学会第 2 種研究会ことば工学研究会資料, Vol. 58, pp. 1–15, 2018.