

日本語 Wikipedia エンティティベクトルを外部知識を利用した皮肉検出

徐 続非[†] 肥合 智史[†] 嶋田 和孝^{††}

[†]九州工業大学大学院 情報工学府 〒820-8502 福岡県飯塚市川津 680-4

^{††}九州工業大学大学院 情報工学研究院 〒820-8502 福岡県飯塚市川津 680-4

E-mail: †{x_xu,s_hiai,shimada}@pluto.ai.kyutech.ac.jp

あらまし 近年、皮肉を検出するタスクが多くの研究で取り組まれている。皮肉を検出することは、人々の実際の感情や考えを理解するために重要である。しかし、極性語を重要な特徴としている一般的な感情分析のモデルでは、文の表面上の意味と一致しない意図を伝える皮肉を理解できない場合が多い。すなわち、皮肉の理解において、表層以上の様々な外部知識や情報が必要である。本研究では、皮肉の検出の精度向上のために、外部情報を導入する。具体的には、Twitter から学習された単語分散表現をもとに Bi-LSTM+Attention によって皮肉を検出するモデルに外部知識として日本語 Wikipedia エンティティベクトルを加える。そして、ベースライン手法 (Bi-LSTM+Attention のみのモデル) と比較して、提案手法の有用性を検証する。また、先行研究の関係ベクトルとも統合し、その有効性を検証する。

キーワード 皮肉, 感情分析, 意見マイニング, 外部知識, マイクロブログ

Sarcasm Detection with Japanese Wikipedia Entity Vector

XU XUFEI[†], Hiai SATOSHI[†], and Shimada KAZUTAKA^{††}

[†] Graduate School of Computer Science and Systems Engineering, Kyushu Institute of Technology
680-4 Kawazu, Iizuka, Fukuoka 820-8502 Japan

^{††} Department of Artificial Intelligence, Kyushu Institute of Technology
680-4 Kawazu, Iizuka, Fukuoka 820-8502 Japan

E-mail: †{x_xu,s_hiai,shimada}@pluto.ai.kyutech.ac.jp

Abstract In recent years, the task of detecting sarcasm has been tackled in many studies. Sarcasm detection is important for understanding people's actual sentiments and thoughts. However, general sentiment analysis models, which take polar words as an important feature, often do not understand the sarcasm that conveys intent different from literal meaning. Sufficient relevant information about the sentence is needed for sarcasm detection. In this paper, we work on improving the accuracy of sarcasm detection by introducing external information. This research proposes a method for introducing external information and examines the structure of the model that introduces external information. We compare the methods with baseline methods. The results demonstrate the effectiveness of our method.

Key words Sarcasm, Sentiment analysis, Opinion mining, External knowledge, Microblogging

1. はじめに

皮肉は、相手の欠点や弱点を意地悪く遠まわしに非難する表現である^(注1)。

[例 1] またテストに落第? 見事な仕事だなあ。

例 1 においては、「見事な仕事」という肯定表現が用いられるが、実は相手への批判が示されている。近年、より高度な文

章理解のための挑戦的課題として、皮肉を検出する研究が進められている。皮肉とは、日常生活だけではなく、ツイートなどのソーシャルコミュニティにもよく現れる複雑な発話行為である [1]。皮肉の検出はレビューの要約、対話システム、レビューのランク付けなど、多くの感情分析に関する自然言語処理タスクに役立つものである [2]。文章中の極性表現は、主に文章の内容が肯定的か否定的か分類するタスクである感情分析における重要な特徴である。しかし、皮肉では、表層と意図が一致しないために本質的に曖昧な性質があり、感情分析の

(注1) : <https://www.weblib.jp/content/皮肉>

誤分類の原因となる。また、人間さえも発話が皮肉であるかどうかを判断するのが難しい場合がある。従って、皮肉を検出することは、人々の実際の感情や考えを理解するためにも重要である [3]。González-Ibáñez ら [4] は、皮肉には極性語が頻出する傾向がある特徴を踏まえ、皮肉の検出モデルを作っている。たとえば、例 1 でいえば、「落第」や「見事」といった極性表現が皮肉検出の手掛かりとなる。しかし、皮肉の理解の鍵となるのは「落第」や「見事」といった極性表現だけではない。その文が発せられた前提や背景など様々な情報を踏まえなければ、皮肉であることが分からない場合も多くある。

たとえば、次の例を考えてみよう。

[例 2] John Doe 選手はチームウォリアーズの勝利の鍵だ。この文は普通に読めば、John Doe 選手への肯定的な言及であるように感じられる。しかし、実際には John Doe 選手はウォリアーズというチームの選手ではなく、ウォリアーズの対戦チームの選手である。また、この John Doe 選手がよくミスをし、それが勝敗に大きく影響していることが多い、という前提が成り立っているとすれば解釈は大きく変わる。加えて、実際に過去に John Doe のミスによってウォリアーズが勝ったことがあるという事実もあると仮定しよう。このような背景や前提を踏まえて読めば、例 2 は John Doe 選手に対する皮肉であると理解できる。つまり、この文の意図（皮肉）を正しく理解するには様々な背景知識が必要である。少なくとも John Doe 選手がウォリアーズの選手でないこと（固有名詞間の関係）は皮肉であると理解するためには必須となる。

次の例を考える。

[例 3] 君はしんちゃんのような静かな人ですね。この文をそのまま読めば、「君=静かな人」となる。しかし、「しんちゃん」が漫画のキャラクターで、その子供がいたずら好きで、決して静かな人ではないことを知っていれば、例 3 は「君はいたずらな子供のような静かな人ですね」ということを意味しており、遠まわしに「君≠静かな人」ということを意図していることを理解できる。これは「しんちゃん」という表現が持っている属性的な情報であり、文中には陽には表れない。

以上の例は、皮肉検出において、皮肉の対象の属性や対象間の関係などが不可欠であることを意味している。しかしながら、たとえば機械学習でモデルを作る際に、このような属性や関係が訓練データ中の文に存在する可能性は低く、単純な訓練データの拡充では対応できないだろう。このような問題に対応するには、皮肉に関する訓練データとは他に、前提や背景に関する情報を外部知識として導入する必要がある。本論文では、外部知識を用いた皮肉検出の手法について提案する。外部知識として利用するのは「日本語 Wikipedia エンティティベクトル」[5]（以降、Wikipedia ベクトル）である。Wikipedia ベクトルは、日本語のウィキペディア中に含まれる記事の本文情報だけでなく、エンティティ間の関係情報も同時に用いて学習されている。本論文では、機械学習のモデルに Bi-LSTM と ATTENTION [6] を適用し、この機械学習のモデルに Wikipedia ベクトルを追加したモデルの皮肉検出での有効性を検証する。さらに、外部知識として関係ベクトルを利用

した先行研究 [7] についても提案手法に組み込み、Wikipedia ベクトルとの統合の有効性について検証をする。

2. 関連研究

近年、英語の皮肉の検出が盛んに研究されている。皮肉の検出は、主に、文章が皮肉を含むかどうか分類するタスクとして取り組まれている。多くの研究で、機械学習を用いた分類手法が提案されている。機械学習のための素性としては、文章中の単語 N グラム [8] などが用いられることが多い。Riloff ら [9] と Joshi ら [10] は皮肉では否定的な状況への肯定的な言及があることに着目してモデルを構築している。しかしながら、このようなアプローチでは、例 3 に示すような否定表現が用いられない場合に対処できない。

近年、BERT を用いた皮肉検出の手法も多く提案されている。Lee ら [11] は、皮肉検出のワークショップ [3] において、BERT を用いた手法で最高精度を達成した。しかしながら、このようなアプローチでも例 2 や例 3 のような事例が正しく判別できるかは、機械学習の訓練データ中に関連する事例があるかに依存しており、適切に対応できない可能性が残る。

前述のように、皮肉では、文中には陽に表れない背景や前提、常識的な知識が不可欠な場合も多い。たとえば、

[例 4] 彼は部下に仕事を押し付ける素晴らしい上司だ。皮肉を意図する文である。肥合ら [7] は皮肉によって批判される対象と対象に不満を持つ立場の関係を考慮した手法を提案した。立場表現とは、例 4 中の「上司」と「部下」のような立場を表す表現である。肥合らはこの立場表現を関係ベクトルと呼ばれるもので表現し、その関係ベクトルが有効であることを確認している。Annervaz ら [12] は、知識グラフによって拡張されたニューラルネットワークが、様々なタスクにおいて有効であることを示した。本論文では、これらの研究のように、外部知識の補完によって皮肉検出の精度を向上することを目的としている。

3. 手法

本節では、外部知識を追加した皮肉検出モデルについて説明する。まず、外部知識として扱う日本語 Wikipedia エンティティベクトルについて説明する。その後、基本となる Bi-LSTM モデルを説明し、その基本となるモデルに Wikipedia ベクトルの組み込む方法の違いによる 4 つの提案手法について説明をする。

3.1 外部知識

機械学習のモデルに補完する知識として、本論文では日本語ウィキペディアの情報を利用する。具体的には、鈴木らが構築した「日本語 Wikipedia エンティティベクトル」[5]（以降、Wikipedia ベクトル）を用いる。

一般に、単純に、文から分散表現を学習すると、その近辺にある情報しか利用できない。その結果、多義語（たとえば、楽器メーカーの「ヤマハ」と輸送用機器メーカーの「ヤマハ」や自動車メーカーの「スズキ」と魚類の「スズキ」など）が正しく判別できない可能性がある。この Wikipedia ベクトルは、日

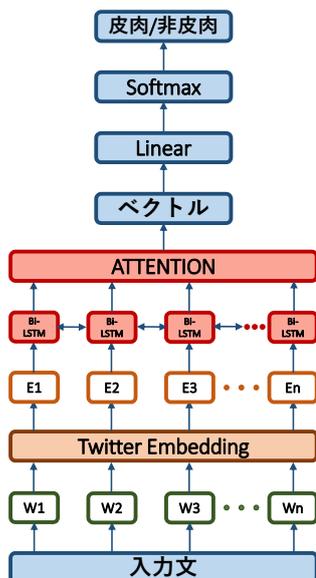


図1 ベースラインモデル (Bi-LSTM+Attention).

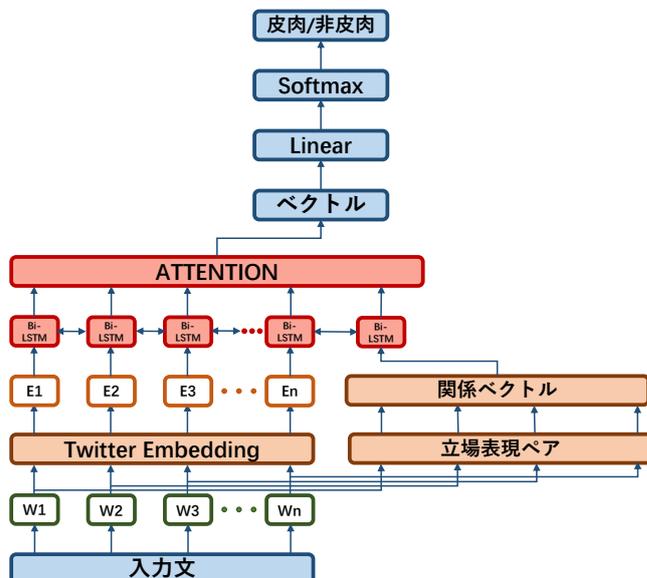


図2 関係ベクトルを用いたモデル (先行研究 [7]).

本語のウィキペディア中に含まれる記事の本文情報だけでなく、エンティティ間の関係情報も同時に用いて学習されている。具体的には、ある固有名詞の分散表現を獲得するために、その単語をリンク先の Wikipedia 記事タイトルに置き換える前処理を行った上で周辺文脈を学習させている。その結果、この Wikipedia ベクトルは固有表現と一般的な単語の識別や語彙曖昧性の問題に対して頑健という特徴がある。

たとえば、Wikipedia の「ロードレース世界選手権」という記事に「現在出場しているのは [ヤマハ], [ホンダ], [スズキ], [ドゥカティ], [アプリリア] の 5 メーカーと...」という文があった場合を考える。この例で [] で囲まれた単語は、Wikipedia 中に対応するエンティティが存在することを意味する。Wikipedia ベクトルでは、各単語をそのリンク先のウィキペディア記事タイトルに置き換える前処理を行った上で周辺文脈を学習させている。すなわち、先ほどの文を「現在出場しているのは [ヤマハ発動機], [本田技研工業], [スズキ_ (企業)], [ドゥカティ], [アプリリア] の 5 メーカーと...」と置き換えて分散表現を学習する。

本研究はこの Wikipedia ベクトルを外部知識として、基本となる Bi-LSTM モデルに追加する。

3.2 ベースラインモデル

本研究では Bi-LSTM に Attention 機構を加えたものを基盤モデル (以降ベースライン) とする。図 1 に概略図を示す。まずは、入力文を MeCab [13] を用いて単語に分割する (W_1, \dots, W_n)。次に Twitter のデータから skip-gram モデル [14] によって学習された単語分散表現を適用する (E_1, \dots, E_n)。この Twitter Embedding は肥合ら [7] が作成したものを用いる。それを Bi-LSTM の入力とし、Attention 層を介して得られたベクトルを Linear 層へ渡し、Softmax 関数を経て、皮肉か非皮肉かの二値に分類する。

3.3 関係ベクトルモデル

前述のように、外部的な要因を考慮したモデルとして肥合

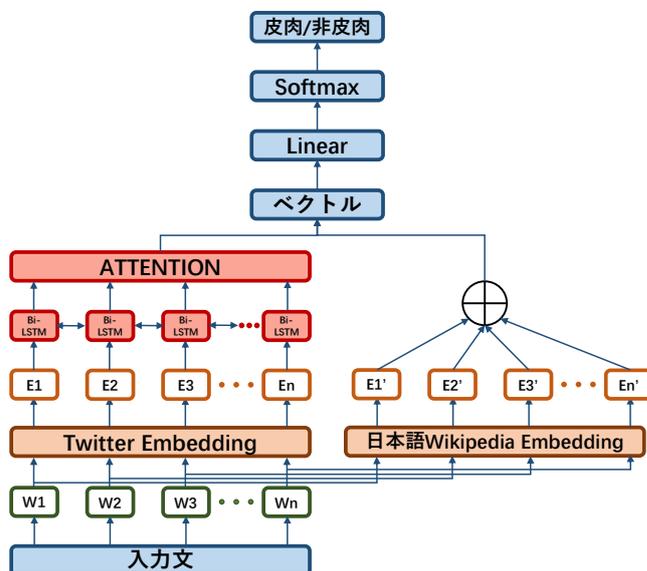


図3 AllLinear : 文中のすべての単語に対する Wikipedia ベクトルを Linear 層に入力するモデル。

らによる関係ベクトルを用いたモデルがある [7]。本節ではそのモデルについて説明する。

図 2 に関係ベクトルを用いたモデルの概要を示す。図の左側は前節のベースラインと同じである。肥合らは、事前に獲得した立場表現に基づき、立場表現ペアの関係ベクトルを構築している。そして、皮肉検出の対象となる文に出現する立場ペアリストをこの関係ベクトルに適用し、その結果を利用して皮肉・非皮肉の分類問題を解いている。関係ベクトルの求め方などについての詳細は先行研究の論文を参照して欲しい。

3.4 提案手法

提案手法では、3.2 節で説明したベースラインモデルに、3.1 節で説明した Wikipedia ベクトルを導入する。本論文では、この Wikipedia ベクトルの導入方法について次の 4 つのパターンを考える。

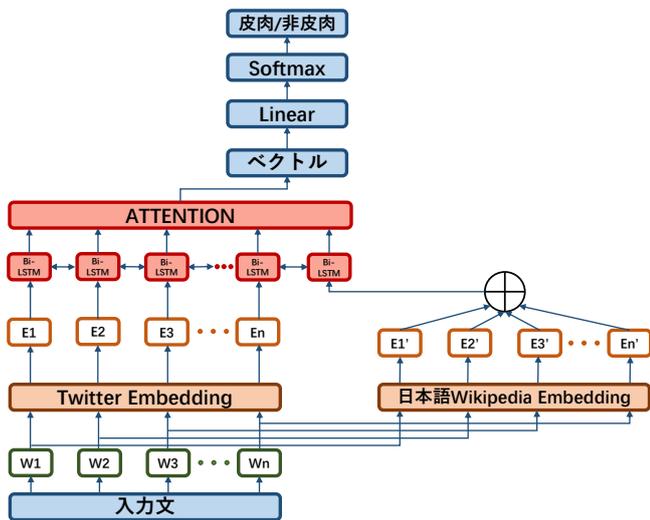


図4 AllBiLSTM : 文中のすべての単語の Wikipedia ベクトルを Bi-LSTM に入力するモデル。

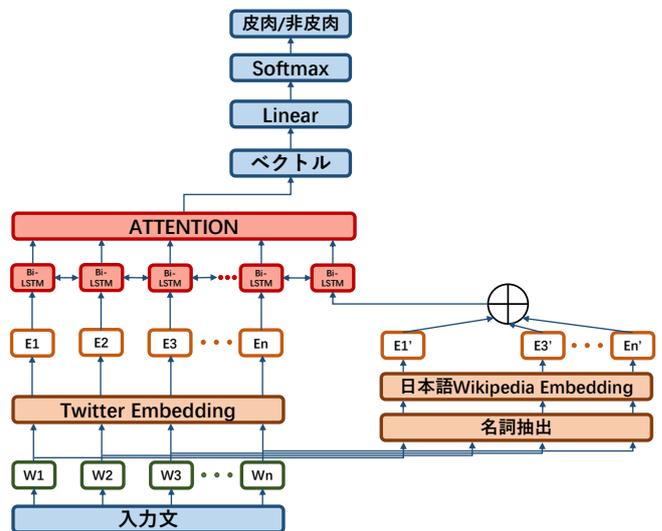


図6 NounBiLSTM : 文中の名詞のみの Wikipedia ベクトルを Bi-LSTM に入力するモデル。

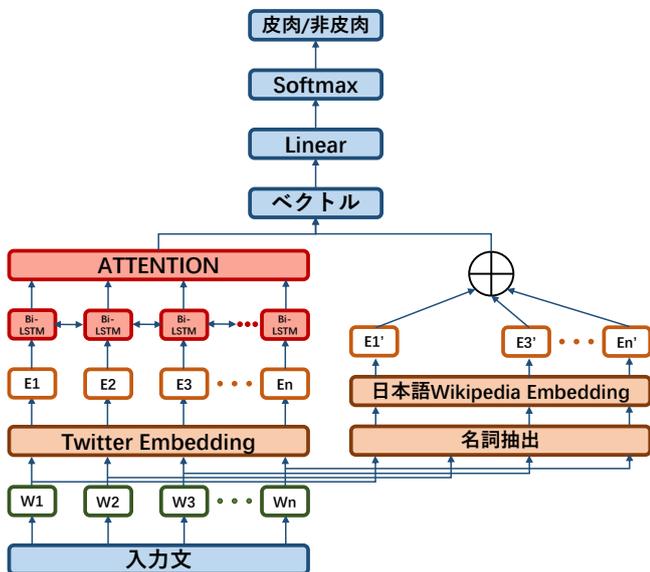


図5 NounLinear : 文中の名詞のみの Wikipedia ベクトルを Linear 層に入力するモデル。

- AllLinear

図3にその概要を示す。図の左側はベースライン(図1)と等価である。このモデルでは、各単語を Wikipedia ベクトルにも適用する。Wikipedia ベクトルを介して得られた単語ベクトルを加算し、それを外部知識としてのベクトルとして扱う。このベクトルをベースラインモデルの出力と結合し、Linear 層に渡す。

- AllBiLSTM

図4にその概要を示す。図の左側はベースライン(図1)と等価である。このモデルも AllLinear と同様にすべての単語について Wikipedia ベクトルに適用し、その和をベースラインのモデルの Bi-LSTM の末尾に追加し、学習をする。

- NounLinear

図5にその概要を示す。この図の基本構成は AllLinear (図3) と似ている。単語に対して適用した Wikipedia ベクトルの加

算結果をベースラインモデルの出力と結合し、Linear 層に渡すものである。ただしすべての単語を使うのではなく、名詞のみを Wikipedia ベクトルに適用する点が AllLinear との違いである。

- NounBiLSTM

図6にその概要を示す。この図の基本構成は AllBiLSTM (図4) と似ている。単語に対して適用した Wikipedia ベクトルの加算結果を Bi-LSTM の末尾に追加するものである。ただし、NounLinear と同様、すべての単語を使うのではなく、名詞のみを Wikipedia ベクトルに適用する点が AllBiLSTM との違いである。

名詞のみの導入パターンを考えるのは、今回想定しているような皮肉検出の問題(例2や例3など)を考えた場合、文中のすべての単語よりは名詞のほうが大きな意味を持っていると考えられるためである。

3.3節で示した関係ベクトルと本節で説明した Wikipedia ベクトルはそれぞれ独立にモデルの中に組み込むことができる。すなわち、上記の4パターンに関係ベクトルを導入することも可能である。本論文では、それぞれのパターンにさらに関係ベクトル(RV)を導入した4つのモデル(AllLinearRV, AllBiLSTMRV, NounLinearRV, NounBiLSTMRV)も比較・検証する。関係ベクトルを用いた場合のベースラインモデルとの結合方法は、それぞれの Wikipedia ベクトルの結合方法に準拠する。たとえば、AllLinearRV の場合は、関係ベクトルは AllLinear と様にベースラインモデルの出力と統合して、Linear 層に渡され、AllBiLSTMRV の場合は、Wikipedia ベクトルと同様に Bi-LSTM の末尾に関係ベクトルも追加される。

4. 実験設定

本節では、これまでに説明した2つの先行モデル(ベースラインと関係ベクトル)と Wikipedia ベクトルを考慮した4つのモデル、さらにその4つに関係ベクトルも統合したモデル

表 1 実験結果.

	皮肉			非皮肉		
	P	R	F1	P	R	F1
ベースライン	0.766	0.764	0.763	0.766	0.764	0.764
関係ベクトル	0.813*	0.787	0.800*	0.794*	0.819*	0.806*
AllLinear	0.749	0.753	0.757	0.751	0.747	0.750
AllLinearRV	0.751	0.752	0.751	0.753	0.751	0.710
AllBiLSTM	0.761	0.794	0.777	0.785	0.750	0.767
AllBiLSTM RV	0.744	0.810*	0.775	0.792	0.719	0.753
NounLinear	0.762	0.759	0.759	0.761	0.760	0.759
NounLinearRV	0.772	0.759	0.765	0.764	0.775	0.769
NounBiLSTM	0.748	0.802	0.774	0.787	0.730	0.757
NounBiLSTM RV	0.758	0.783	0.770	0.776	0.748	0.761

の合計 10 個のモデルを比較する。また、実行結果を分析し、提案手法の有効性などについて考察する。

4.1 データセットと評価尺度

評価実験に利用するデータについて説明する。本論文では先行研究の肥合らの実験で用いられた日本語のデータセットを用いる。肥合らは Twitter への投稿から「(皮肉)」という表現を含んだ tweet を収集し、それを皮肉の正例として利用している^(注2)。このデータセットは、正例と負例^(注3)をそれぞれ 21000 件ずつの合計 42000 件の tweet で構成されている。先行研究に倣い、30000 件を訓練用データ、10000 件を開発用データ、2000 件を評価用データとした。

評価尺度は適合率 (P)、再現率 (R)、F 値を利用する。この 3 つの尺度について、皮肉、非皮肉の両方について評価する。分類実験は 5 回行い、その平均値を各手法の値とした。

4.2 実験結果

実験結果を表 1 に示す。表で、太字の部分がベースライン(外部知識を用いない場合)と比較して高い値となった部分である。*を伴う数字は各評価項目で最も高い値を意味する。

ベースラインと比較すると、特に皮肉に対する F 値は 5 つのモデル (AllBiLSTM, AllBiLSTM RV, NounLinearRV, NounBiLSTM, NounBiLSTM RV) で高い値を収めた。AllBiLSTM RV については皮肉の再現率 (R) においては最も良い精度となっている。実験結果は提案手法についての一定の有効性を表している。一方で、F 値で見れば提案した 4 つのモデルはいずれも先行研究である関係ベクトルのみの手法に大きく差を付けられている。また、提案モデルに対する関係ベクトルの導入は有効な場合とそうでない場合が混在しており、提案手法と関係ベクトルの統合には議論の余地が残る。

4.3 事例分析・考察

本節では、ベースラインと比較して上手くいった事例や、関係ベクトルでは上手くいかなかったが、提案手法では正しく検出できた例などを挙げ、分析・考察する。

次の例は、ベースラインでは正しく検出できなかったが、提

案手法により正しく検出できた皮肉の事例である。

[例 5] 綺麗なタイプのメンヘラ。

Wikipedia の「メンヘラ」という記事には、「何らかの [精神疾患] を抱えている人や、抱えていると思われる人を指す [ネットスラング]」という記述がある。この「メンヘラ」の記事には、「精神疾患」にハイパーリンクが付けられている。3.1 節で説明したように、Wikipedia ベクトルの構築の際に、このハイパーリンクの情報が組み込まれて学習される。このような知識が、皮肉文の認識に有効に機能したのだと考えられる。これは、提案手法が名詞の属性の知識を補完できている可能性を示唆しており、例 3 のような皮肉文を検出に貢献すると考えられる。

次に関係ベクトルのみによる手法では正しく検出されなかったが、提案手法で正しく検出された例を示す。

[例 6] EU 残りたい連中は他の EU 諸国へ「移民」すればいいんじゃない。

肥合らの関係ベクトルは、事前に用意されたパターンにマッチするもののみを自動的に収集しており、網羅性の点では課題がある。もちろん、本手法で利用している Wikipedia ベクトルでも網羅性の問題は生じるが、知識的な要素を多く含んでいる Wikipedia が情報源であるため、知識的な単語には有効に機能しやすいと考えられる。たとえば、Wikipedia には「イギリスの欧州連合離脱」の記事があり、「イギリスの欧州連合離脱 (イギリスのおうしゅうれんごうりだつ) とは、[イギリス] が [欧州連合] (EU) から離脱することを指す」と記述されている。そして、同様にリンク先の「欧州連合」という記事にも、「欧州連合 (おうしゅうれんごう, 英: European Union, 略称: EU) は...」のような記述がある。このような知識の連鎖が Wikipedia ベクトルを経由して皮肉検出に影響を与えたのではないかと考えられる。このように、精度の面では提案手法の追加要素は関係ベクトルに基づくものに及ばないが、先行研究の関係ベクトルでは得られない情報を考慮できているため、今後より有効な使い方を検討したい。

最後に、提案手法では正しく皮肉だと検出されず、現状の Wikipedia ベクトルだけでは十分に対応できないと考えられる例を示す。

(注2): 当然だが、実験データとして利用する場合は、その「(皮肉)」という表現は削除されている。

(注3): (皮肉) を含んでいない tweet をランダムに収集。

[例 7] 福岡の空気が綺麗すぎて喉が全然痛くない！
これが皮肉になる解釈はいくつかある。たとえば、投稿者の認識として、「福岡の空気」はいつも綺麗ではない、というような前提がある場合に成り立つ皮肉である。このような感覚的な情報は Wikipedia から得ることは難しい。また別の解釈として投稿者の居住地が空気が綺麗ではなく、立ち寄った福岡の空気が綺麗だと感じた場合は、自身の居住地に対する皮肉である。このような投稿者の属性や皮肉に対する対象に関する情報も皮肉であるかどうかの判別に大きく関わっており、難しい問題である。

また、次のような事例も提案手法では分類が上手くいかない。

[例 8] やったー！今日は7時間も寝れるぞお。
この例には、Wikipedia のエンティティとなるような語句は出現していない。しかし、「7時間」という睡眠時間として一般的な数字を強調することで、普段の睡眠が不足している状況に関する皮肉を表現している。このような常識的知識を Wikipedia のような百科事典的な知識源から抽出することは困難であると考えられる。よって、常識的知識の獲得やその利用に関する手法を検討し、本手法との適切な統合方法などを検討することが必要である。

5. おわりに

本研究では、皮肉検出の精度向上のために、外部知識の導入による皮肉検出モデルを提案した。本論文では、外部知識として日本語 Wikipedia エンティティベクトルを導入し、基盤となる Bi-LSTM+Attention モデルに組み込むことで、精度向上を目指した。Wikipedia ベクトルの導入方法として4つのパターンを提案した。さらに先行研究である関係ベクトルをそれぞれに導入し、先行研究に相当するベースラインモデルと関係ベクトルモデルに対して、8つの提案手法を比較した。

実験結果では、Wikipedia ベクトルを用いた手法はベースラインモデルと比較して良好な結果を得られる傾向があることが確認された。皮肉検出の再現率についてはすべての手法の中で最も良い値を得た。一方で、F 値の面で見ると皮肉の検出においても非皮肉の検出においても先行研究である関係ベクトルのみを利用した手法には劣る結果となった。実験結果を踏まえ、事例を分析したところ、Wikipedia ベクトルが有効に機能する事例や Wikipedia ベクトルでは対応できない問題など、多くの知見が得られた。

今回の手法は、先行研究である関係ベクトルも組み合わせることが簡単にできる。加えて、分析から、Wikipedia ベクトルだけでもしくは先行研究の関係ベクトルだけですべての皮肉をカバーできるわけではないことも分かった。今後は、Wikipedia ベクトル以外の知識源や常識的知識 [15], [16] の導入などを経て、さらなる精度向上を目指す。

文 献

[1] Y. Tay, A.T. Luu, S.C. Hui, and J. Su, “Reasoning with sarcasm by reading in-between,” Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp.1010–1020, Association for Computational Linguistics, Melbourne, Australia, July 2018.

[2] D. Davidov, O. Tsur, and A. Rappoport, “Semi-supervised recognition of sarcasm in twitter and amazon,” Proceedings of the fourteenth conference on computational natural language learning, pp.107–116, 2010.

[3] D. Ghosh, A. Vajpayee, and S. Muresan, “A report on the 2020 sarcasm detection shared task,” Proceedings of the Second Workshop on Figurative Language Processing, pp.1–11, Association for Computational Linguistics, Online, July 2020.

[4] R. González-Ibáñez, S. Muresan, and N. Wacholder, “Identifying sarcasm in Twitter: A closer look,” Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp.581–586, Association for Computational Linguistics, Portland, Oregon, USA, June 2011.

[5] 鈴木正敏, 松田耕史, 関根聡, 岡崎直観, 乾健太郎, “Wikipedia 記事に対する拡張固有表現ラベルの多重付与,” 言語処理学会第 22 回年次大会発表論文集, pp.797–800, 2016.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” Advances in neural information processing systems, pp.5998–6008, 2017.

[7] S. Hiai and K. Shimada, “Sarcasm detection using RNN with relation vector,” The International Journal of Data Warehousing and Mining (IJDWM), vol.15, no.4, pp.66–78, 2019.

[8] A. Reyes, P. Rosso, and T. Veale, “A multidimensional approach for detecting irony in twitter,” Language resources and evaluation, vol.47, no.1, pp.239–268, 2013.

[9] E. Riloff, A. Qadir, P. Surve, L. De Silva, N. Gilbert, and R. Huang, “Sarcasm as contrast between a positive sentiment and negative situation,” Proceedings of the 2013 conference on empirical methods in natural language processing, pp.704–714, 2013.

[10] A. Joshi, V. Sharma, and P. Bhattacharyya, “Harnessing context incongruity for sarcasm detection,” Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pp.757–762, Association for Computational Linguistics, Beijing, China, July 2015.

[11] H. Lee, Y. Yu, and G. Kim, “Augmenting data for sarcasm detection with unlabeled conversation context,” Proceedings of the Second Workshop on Figurative Language Processing, pp.12–17, Association for Computational Linguistics, Online, July 2020.

[12] K. Annervaz, S.B.R. Chowdhury, and A. Dukkipati, “Learning beyond datasets: Knowledge graph augmented neural networks for natural language processing,” 2018.

[13] T. Kudo, K. Yamamoto, and Y. Matsumoto, “Applying conditional random fields to Japanese morphological analysis,” Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, pp.230–237, Association for Computational Linguistics, Barcelona, Spain, July 2004.

[14] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” The International Conference on Learning Representations: Workshops Track, 2013.

[15] R. Speer, J. Chin, and C. Havasi, “Conceptnet 5.5: An open multilingual graph of general knowledge,” AAAI Conference on Artificial Intelligence, pp.4444–4451, 2017.

[16] K. Yamamoto and K. Shimada, “Acquisition of periodic events with person attributes,” Proceedings of the International Conference on Asian Language Processing (IALP), pp.229–234,, 2020.