テキストの埋め込み表現に基づくデータ増強を用いた X（旧 Twitter）における日本語の皮肉検出

中井 紫音　宮本 友樹　内海 彰
電気通信大学大学院
n2330080@gl.cc.uec.ac.jp
{miyamoto,utsumi}@uec.ac.jp

概要

近年，数多くの SNS が存在する中で X（旧 Twitter）を対象とした皮肉判定の研究が活発に行われている．日本語の皮肉分類器の性能向上を図る手段の一つとしてデータ増強の適用が考えられるが，皮肉表現はその特性上，文章内の特定の単語を置き換えるとたとえわずかな変更であっても元の皮肉性が失われ，皮肉でなくなる（分類ラベルが反転する）可能性がある．本稿では，データ増強の際にノイズを加える箇所を Transformer の Attention を用いて選定するという，皮肉判定に特化したデータ増強を通じて皮肉分類器の性能向上を図る手法を提案する．本手法は既存手法や GPT を用いたデータ増強手法と比べ，正解率，F 値にともに高い性能を示した．

1　はじめに

SNS の普及に伴い，多くの人が個人の意見を日常的に発信している．このような大衆の意見について感情分析を行うことは，政治的イデオロギーの調査や企業がマーケティングを行う上で重要な役割を果たす．そしてこの感情分析における誤分類の原因の一つとして，皮肉による批判が挙げられる．ここでの皮肉とは，肯定表現を用いて否定的な意味を表すものを指す．文章中の極性表現は感情分析において重要な特徴となるが，皮肉文では発言者の感情や意図が，文章から読み取れる極性と一致しない．そのため皮肉を考慮せずに感情分析を行うと，真の感情と正反対の判定がなされてしまう危険がある．

近年，数多くの SNS が存在する中で X（旧 Twitter）を対象とした皮肉判定の研究が盛んである [1-8]．X とは 140 文字以内の文章を投稿し共有することができる Web サービスであり，その特性上，皮肉が含まれる投稿も多く見られる．このため，X 上の投稿内容を正確に理解するためには皮肉の判定が不可欠となっている．

投稿者は皮肉を示唆するハッシュタグを用いて自らのポストが皮肉であることを明示できるものの，そのようなハッシュタグを付けずに投稿される皮肉ポストが多く存在している [3]．よってこれらのポストを自動判定する必要があるが，皮肉判定を行うための分類器を学習する際には，正例となる皮肉ポストの集めやすさに言語による違いがある．これは英語圏では #sarcasm や #not といった皮肉を示唆するハッシュタグを用いて皮肉を投稿する習慣がある一方で，日本では こうしたハッシュタグ（#皮肉）の使用頻度が比較的低いことに起因する．そのため，日本語を対象する際には正例データの自動収集が難しいという問題がある．

このような背景から，日本語を対象とした皮肉判定に関する研究は少ない．魚住 [4] らは X の日本語投稿を対象とし，皮肉検出においてネガティブな感情を生起する要因を特徴量として用いることの有用性を検証している．また肥合 [5, 6] らは，皮肉に偏って現れる語を素性として用いる手法や，皮肉によって批判される対象とその対象に不満を持つ立場の関係を考慮した皮肉の検出手法を提案している．中東 [7] は，ランダムに収集したツイートを負例データとして用いると，データ内に正例データが含まれる可能性があることに着目し，この問題を解決するために半教師あり学習を適応した．そして中井 [8] はこの手法を拡張し，”皮肉”という単語を含むポストは皮肉文である可能性が高いという仮説のもと，この基準で収集したデータの中から正例データを抽出し，学習データに加えるという操作によって分類器の性能を向上させた．しかしこの手法は，増強するデータが特定のタイプに限定されていること，そして学習データの他に，先の基準を満たす大

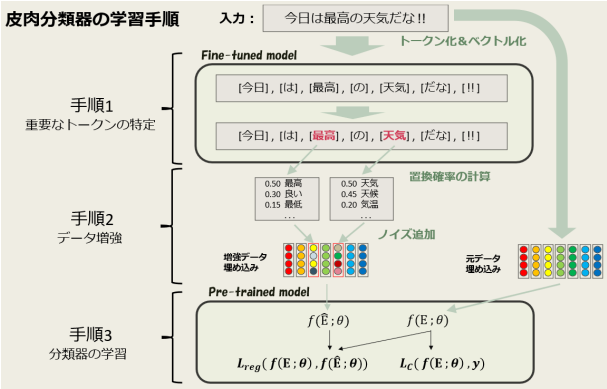

図 1　提案手法の概要

量のラベルなしデータが不可欠であるという問題がある．

このデータ増強という手法に関して近年，自然言語処理分野では TMix[9] や VDA[10] といった，テキストの埋め込み表現を活用する手法が提案されている．これにより元の文章の意味や文脈を保持しながら，単純な単語置換では得られない多様なデータを効率的に疑似生成することができる．しかしこれらの研究は，データに加えた変更の程度だけラベルも変化するという仮定に基づいている．

皮肉表現はその特性上，文章内の特定の単語を置き換えると，たとえわずかな変更であっても元の皮肉性が失われ，皮肉でなくなる（分類ラベルが反転する）可能性がある．よって，文脈に強く依存する皮肉表現を対象とする場合，既存のデータ増強手法ではこの問題に対処できず，そのまま皮肉検出に適用することはできない．そこで本研究では，皮肉に特化した新たなデータ増強を通じて皮肉分類器の性能向上を図る手法を提案する．

2　データ増強を用いた皮肉判定手法

VDA(Virtual Data Augmentation)[10] とは，BERT などの MLM (Masked Language Modeling) を前提とした，埋め込み表現に基づくデータ増強手法の一つであり，入力されたすべてのトークン埋め込みに対してノイズを加えることで，意味的に関連した多様な埋め込みを生成する手法である．本研究ではこの手法をベースに，皮肉分類器の構築を目指す．提案手法の概要については図 1 に示す．分類器の学習手順は以下の通りである．

1. 重要でないトークンの特定：学習データで事前学習した皮肉分類器を用いて，学習データの各テキストの中から皮肉の判定において重要でないトークンを特定する．
2. データの増強：上記で選択したトークンに対してガウスノイズを加え，新たな文章の埋め込み表現を得る．
3. 正則化学習：元の学習データと増強したデータを用いて最終的な皮肉分類器を構築する．

2.1　重要でないトークンの特定

皮肉を構成する上で核となる，いくつかの重要な単語は，わずかなニュアンスの違いや言い換えによってその文章が持つ皮肉性を反転させてしまう．そのため皮肉文を拡張する際には，ノイズを加える処理を，その他の重要でない単語にのみ適応する必要があると考えられる．

Transformer には，文中の単語間の依存関係を捉えることを目的とした自己注意機構 (Self-Attention Mechanism) が組み込まれている．本研究ではこれを活用して皮肉検出において重要でないトークンを特定するために，Transformer をベースとしている BERT を利用する．BERT では入力の先頭に CLS トークンという特別なトークンを追加するが，これは文章全体の意味を要約したものとされている．この CLS トークンから文中の各単語へと向けられるアテンションスコアを利用し，その値が閾値を下回るトークンを，皮肉判定において重要でないトークンとして選定する．

2.2　データの増強

ガウスノイズを用いてデータを増強し，新たな埋め込み表現を生成する．ノイズを加える対象として選択された特定のトークン w_{target} に対して，以下の操作を行う．

1. MLM を使用して，テキスト S を入力したときの文脈に基づいて，ターゲットトークン w_{target} が語彙 W 内の単語 w_i に置き換わる確率 $p(w_i|S)$ を計算する．この処理を語彙内のすべての単語 $w_1, w_2, ..., w_V$ に対して行い，それぞれの置換確率 $\{p(w_1|S), ..., p(w_V|S)\}$ を得る．
2. 上記で得た確率分布に対してガウス分布から抽出したランダムノイズ e を加える．その後ソフトマックス関数で正規化し，式 (1) に示す新しい置換確率 $\{p'(w_1|S), ..., p'(w_V|S)\}$ を得る．

$$p'(w_i|S) = softmax(p(w_i|S) + e) \qquad (1)$$

3. 上記で得た，ノイズが加わった置換確率 $\boldsymbol{p}_w = \mathrm{vec}\{p'(w_i \mid S) \mid i = 1,2,\ldots,V\}$ と行列 $M_E \in \mathbb{R}^{V\times d}$ の積をとり，式 (2) に示す w_{target} に対する新しい埋め込みベクトル e_w を得る．

$$e_w = \boldsymbol{p}_w \cdot M_E \tag{2}$$

この M_E とは，MLM の語彙 W 内のすべての単語を表す埋め込みベクトルを行方向に並べたものである．これは BERT で単語をエンコードする際に用いられる行列であり，V は MLM の語彙数，d は埋め込みベクトルの次元数，となっている．

2.3 正則化学習

最終的な皮肉分類器は，事前学習済みのモデルを，皮肉判定タスク用の学習データを用いてファインチューニングすることで構築する．この際元の埋め込み表現は，正解ラベルとペアにして直接モデルの学習に使用する．一方で，新しい埋め込み表現 e_w は学習データとしては直接利用せず，正則化の目的で用いる．具体的には，元データのモデル出力と，ノイズを加えた埋め込みから得られるモデル出力が一致するように学習を行う．そのため本研究では図 1 の手順 3 で示すように，

- 元データのモデル出力と正解ラベルとの間で計算されるクロスエントロピー
- 元データのモデル出力と増強された埋め込み表現に対するモデル出力との間で計算される KL 情報量

という 2 つの損失の和を全体の損失として定義する．

3 実験

3.1 実験データ

学習に用いる日本語データとして，先行研究 [8] で皮肉分類器の学習を行う際に用いたポストデータ（正例 152 個，負例 152 個）を利用する．ただし，後述する比較手法 4 に関しては，分類器の学習時に用いる正例と負例の個数を均等にするため，GPT で作成した皮肉文と同数の負例データを，先行研究 [8] で使用された負例データからランダムにサンプリングし，追加した．

評価データには，先行研究 [7] で評価データとして用いたものを利用した．この評価データは，#皮肉 を含むものを正例，ランダムに収集したポストから人手でスパムではなく，かつ皮肉ではないと判断したものを負例し，前処理によって改行文字，URL，ユーザー名，皮肉ハッシュタグが削除されている．この正例評価データに対して人手による選別を行い，最終的に皮肉文であると判断された 172 個の正例ポストと，同数の負例ポストを最終的な評価データとして用いた．

3.2 実験の詳細

重要でないトークンの特定　BERT は 12 層の Transformer で構成されているため，手順 1 で皮肉判定に重要でないトークンを特定する際に，利用する層をこれらの 12 層から選択することができる．先行研究 [11] により，BERT では下層，中層，上層でそれぞれ表層的情報（文の長さ，単語の存在），構文的情報，意味的情報をとらえるとされている．そこで本研究では，1 から 12 までの各層をそれぞれ単体で用いるパターンに加え，中層（4-8），上層（8-12），中層＋上層（4-12），全層（1-12），7，9，12 層を用いた際の性能を比較した．複数の層を利用する際には，各層の Attention スコアの平均をそのトークンのスコアとした．また，$\mu - \sigma$（平均値-標準偏差）以下のアテンションスコアを持つトークンを重要でないトークンと判定した．

データの増強数　本実験ではすべての学習データに対して 1 度だけデータ増強を行い，疑似的にデータ数を 2 倍にして分類器を作成した．

モデル　皮肉分類器学習には，すべて BERT を使用した．事前学習モデルとしては、東北大学が提供している「cl-tohoku/bert-base-japanese-whole-word-masking」を使用し，バッチサイズを 16，エポック数を 5 とした．

また，手順 1 で重要でないトークンを特定する際には，事前学習モデルを学習データ（正例 152 個，負例 152 個）でファインチューニングしたものを用いた．

3.3 評価方法

提案手法による分類器の性能を，以下に示す手法で学習した分類器と比較評価した．

比較手法 1: データ増強なし　データ増強を行わず，学習データをそのまま用いて学習した分類器．

（手順 1 で重要でないトークンを特定する際に用いた分類器と同一のもの）

比較手法 2:VDA[10]　手順 1 を行わず，すべてのトークンに対してノイズ処理を適応した分類器．

比較手法 3:RandamNoise　手順 1 の代わりに，ノイズを加える対象を全トークン数の 30 %に当たる個数だけランダムに選択した分類器．

比較手法 4:GPT　GPT3.5Turbo, GPT4o をそれぞれ用いて皮肉文 152 個（提案手法と同数）生成し，正例学習データとして直接用いた分類器．

評価尺度としては，正解率，適合率，再現率，F 値を採用した．

3.4　実験結果と考察

表 1 に提案手法と比較手法で学習した分類器の性能を示す．正解率に関しては提案手法で 2,9,10 層を用いた際の分類器が，F 値に関しては提案手法で 12 層を用いた際の分類器がそれぞれ最も高い性能を示した．さらに比較手法の中では，正解率は RandomNoise，F 値は GPT4o が最も高い性能を示しているが，提案手法はすべてのパターンでこれらの値を上回る結果となった．

ノイズ処理に関して，すべてのトークンにノイズを加える VDA の手法がベースラインの値を正解率，F 値ともに下回っていることから，皮肉検出タスクにおいては既存のデータ拡張手法は悪影響であることが確認できる．また，この結果からはノイズを加える量に分類器の性能が左右される可能性が考えられるが，提案手法が RandomNoise（ノイズを加えるトークン数を揃えたもの）のスコアを上回っていることから，ノイズを加えるトークンを選択することの有効性が確認できる．これらの結果から提案手法は有効であると言える．

次に用いる層ごとの結果に着目すると，9,10 層が正解率，12 層が F 値に関して最高値を記録したことは，先行研究 [11] に基づく予想と概ね一致している．一方で，2 層も正解率において最高値を記録し，さらに F 値においても低層を結果が高層に匹敵していることから，X に投稿される皮肉文を判定する際には，用いられる単語の種類といった表層的情報も重要な役割を果たしていると考えられる．

また，複数の層を利用することで分類器の性能が大きく向上することはなかった．これは，各層で算出した Attention スコアを単純に平均する操作によって，層ごとに異なる特徴が均されてしまったことが原因として考えられる．

表 1　提案手法と比較手法で学習した分類器の性能

提案手法	正解率	適合率	再現率	F 値
1 layer	0.7820	0.7487	0.8488	0.7956
2 layer	0.7907	0.7688	0.8314	0.7989
3 layer	0.7878	0.7565	0.8488	0.8000
4 layer	0.7878	0.7368	0.8953	0.8084
5 layer	0.7733	0.7217	0.8895	0.7969
6 layer	0.7878	0.7565	0.8488	0.8000
7 layer	0.7762	0.7363	0.8605	0.7936
8 layer	0.7791	0.7264	0.8953	0.8021
9 layer	0.7907	0.7475	0.8779	0.8075
10 layer	0.7907	0.7475	0.8779	0.8075
11 layer	0.7820	0.7321	0.8895	0.8031
12 layer	0.7878	0.7346	0.9012	0.8094
4-8 layers	0.7878	0.7619	0.8372	0.7978
8-12 layers	0.7791	0.7286	0.8895	0.8010
4-12 layers	0.7762	0.7387	0.8547	0.7925
1-12 layers	0.7849	0.7450	0.8663	0.8011
7, 9, 12 layers	0.7762	0.7340	0.8663	0.7947
比較手法	**正解率**	**適合率**	**再現率**	**F 値**
データ増強なし	0.7645	0.7156	0.8779	0.7885
VDA	0.7558	0.7245	0.8256	0.7717
RandomNoise	0.7733	0.7350	0.8547	0.7903
GPT3.5Turbo	0.7471	0.6906	0.8953	0.7797
GPT4o	0.7703	0.7246	0.8721	0.7916

4　おわりに

本稿では，特定の埋め込み表現に対してのみノイズを加えるという皮肉に特化したデータ増強を通じて，皮肉分類器の性能向上を図る提案手法を提案した．そして比較手法との評価実験を通じて，皮肉検出という特定のタスクにおいては既存のデータ増強手法は不適切であること，また本稿における提案手法が有効であることを示せた．

今後の課題として，各層が捉えている皮肉の特徴を可視化することが挙げられる．本研究では各層で実際に選出されたトークンや，提案手法の操作に伴って正負の判定が変わった正解データの実例を用いた分析を行っていない．このような分析を通じて得た知見を基にトークン選択の最適化を図ることで，皮肉分類器のさらなる性能向上が期待される．

参考文献

[1] Arifur Rahaman et al. Sarcasm Detection in Tweets: A Feature-based Approach using Supervised Machine Learning Models. *International Journal of Advanced Computer Science and Applications*, Vol.12, No.6 (2021).

[2] Anita Saroj et al. Ensemble-based domain adaptation on social media posts for irony detection. *Multimedia Tools and Applications*, Vol.83, pp.23249–23268 (2024).

[3] Silviu Oprea and Walid Magdy. iSarcasm: A dataset of intended sarcasm. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp.1279–1289 (2020).

[4] 魚住ゆい, 内田ゆず，荒木健治. 皮肉検出における感情生起要因の有効性. 第 17 回情報技術フォーラム (FIT2018)，pp.163-166,2018.

[5] 肥合智史，嶋田和孝. 偏りのある特徴語を考慮した皮肉の検出. 言語処理学会第 23 回年次大会 , pp.990-993, 2017.

[6] 肥合智史，嶋田和孝. 関係ベクトルを利用した皮肉の検出. 言語処理学会第 24 回年次大会，pp.829–832, 2018.

[7] 中東 三四郎, 半教師あり学習を用いた Twitter における日本語の皮肉の自動判定, 平成 27 年度総合情報学科卒業論文 (2016).

[8] 中井紫音, 宮本友樹, 内海彰. PU 学習と NU 学習を用いた Twitter における日本語の皮肉検出. 日本知能情報ファジィ学会 ファジィ システム シンポジウム 講演論文集. 日本知能情報ファジィ学会 (2023).

[9] Jiaao Chen et al. MixText: Linguistically-Informed Interpolation of Hidden Space for Semi-Supervised Text Classification. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp.2147–2157 (2020).

[10] Kun Zhou et al. Virtual Data Augmentation: A Robust and General Framework for Fine-tuning Pre-trained Models. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp.3875–3887 (2021).

[11] Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does BERT learn about the structure of language?. *In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp.3651–3657 (2019).