SNS 上の攻撃的表現の検出と位置特定

牧元大悟 [1]　徳永健伸 [1]
[1] 東京工業大学 情報理工学院
makimoto.d.aa@m.titech.ac.jp　take@c.titech.ac.jp

概要

本研究では日本語における攻撃的表現の検出及び位置特定のタスクのためのデータセット構築し，このデータセットを用いて，最先端の機械学習モデルを含む複数のモデルで実験を行いその結果を比較した。英語を対象とした関連研究の学習モデルが本研究で作成したデータセットにおいても良い結果を達成することを確認した。また，攻撃的表現の検出と位置特定をまとめたタスクにおける評価手法について考察し，サブタスクに分割して取り組む方針の有効性について検証した。

1　序論

本研究の目的は SNS 上の日本語の投稿に対する攻撃的表現の検出とその位置の特定である。具体的には、SNS 上の投稿を入力とし，投稿が攻撃的表現を含むかどうかを判定し，攻撃的表現を含むものに対してその位置を出力するモデルを構築する。本研究で扱う攻撃的表現とは，第三者が投稿を見た際に攻撃的な表現であると判断できるものである。

攻撃的表現の検出と位置特定を同時に扱う先行研究はなく，特に日本語の投稿を対象とした研究では攻撃的表現の位置特定を扱った研究はない。本研究では日本語のデータセットの作成し、攻撃的表現の検出と位置特定を同時に行なう手法の提案と英語における先行研究を参考にした複数の機械学習モデルの実装と比較，評価手法の検討を行なった。また，攻撃的表現の検出と位置特定を包括的に行う上での評価方法やタスクの取り組み方，データセットの利用方法を複数の設定で実験し，比較を行なった。

本研究の貢献を以下にまとめる。

- 攻撃的表現がアノテーションされた日本語データセットの作成方法の提示
- 最先端の機械学習を用いた攻撃的表現の検出と位置特定タスクの実装と評価
- 本タスクに対する評価手法の考察

2　関連研究

2.1　英語における攻撃的表現の検出

Zampieri らは，Offensive Language Identification を含むタスク群を扱った共有タスクを提案している [1, 2]。このタスクで用いられたデータセット OLID [3] は，キーワードを用いて Offensive Language を含みやすいツイートを収集し，アノテーションしたものである。Zampieri らの Offensive Language の検出では，ALBERT[4] のアンサンブルモデルが最も良い結果を達成した [1]。

2.2　英語における攻撃的表現の位置特定

Pavlopoulos らはは Toxic Span Detecion を扱った共有タスクを提案している [5]。データセットは，既存のデータセットから “Toxic” とラベルされたコメントを取り出し，それらに対しスパンのアノテーションしている。BERT ベースの 3 つのモデルのアンサンブルが最も良い結果を達成した。

2.3　日本語における関連研究

尼崎らは Twitter から人が嫌がる語を含む投稿を収集し，人手により不適切な投稿であるかどうかのラベルを付与した [6]。作成したデータに対して Bag of Words(BoW) で特徴量を抽出し線形 SVM を用いて不適切投稿の検知を行なっている。他の類似研究として，学校非公式サイトでの有害な書き込み検出 [7, 8] や掲示板の書き込みが児童買春を示唆しているかの有害性評価 [9] や SNS 上で非難が殺到してしまう投稿を検知する炎上検出 [10, 11] などがある。

2.4　本研究の位置づけ

本研究は，Twitter により収集したデータに対し，人手でアノテーションを行いデータセットを作成す

る。作成したデータセットに対して最先端の機械学習モデルを使用して攻撃的表現の検出及び位置特定を行う。英語を対象とした研究で用いられているような最先端の機械学習モデルが用いられた日本語における先行研究はなく，攻撃的表現の位置特定については先行研究が存在しない。また，検出と位置特定を同時に扱うような研究は英語においても行われていない。

3 データ

3.1 データ収集

本研究では，SNS 上の投稿としてツイートを扱う。ツイートの収集には TwitterAPI[1] を利用する。リプライ及び引用リツイート（以下，まとめてリプライと呼称）を収集対象とし、既定の攻撃的単語を含むツイートを複数回しているアカウントからリプライを収集した。既定の攻撃的単語には，畠山ら [12] が用いた人手のアンケートで半数以上の人が有害及び少し有害と判断した種単語群のうち卑猥語を除いた以下の 17 単語を用いた。

死ね, 消えろ, 蛆虫, カス, 殺す, きもい, うざい, 不細工, ビッチ, クズ, マスゴミ, 脱糞, 糞虫, ダセー, ゴキヲタ, マジキモ, 死ね

上記で得られた最終的な収集データに対し，メンションやハッシュタグ除去等の前処理を行い，最終的に 306 のアカウントから合計 34,123 リプライが集まった。また，そのうち既定の攻撃的単語を含むリプライ数は 1,292 で全体の 3.7%である。

3.2 アノテーション

3.1 で収集したリプライに対し人手で以下の 2 つのアノテーションを行なった。1 リプライにつき 3 人のアノテーターが担当した。

(1) 攻撃的表現を含むかどうかを，「0: 含まない」「1: 含む」「2: ツイートの内容が理解できない」に分類する
(2)「1: 含む」の攻撃的表現を括弧で囲む（複数可）

3.3 データセット構築

攻撃的表現を含むかのラベルについては，2 人以上が「2: ツイートの内容が理解できない」としたリプライを除外し，それ以外のリプライについては 1 人以上が「1: 含む」としたリプライのラベルを「1: 含む」とし，残りを「0: 含まない」とした。

そして「1: 含む」のラベルがついたリプライについて以下の手順で攻撃的表現の位置を決定する。

(1) 括弧で囲まれた攻撃的表現を文字インデックスの集合に変換し，3 人の文字のインデックスの集合を統合する
(2) リプライを MeCab[13] を用いてトークン単位に分割し，(1) で作成した集合の要素のインデックスが含まれるトークンを攻撃的表現トークンとし，IOB タグに変換する

攻撃的表現の位置決定における手順 (1) の例を表 1 に示す。以上の操作で得られた BI タグで表される攻撃的表現のトークン列を 1 つのスパンと呼ぶ。

表 1 攻撃的表現の位置の統合例

アノテーション	攻撃的表現インデックス
{バカ}な{クソ}ガキどもめ	[1, 2, 4, 5]
{バカ}な{クソガキども}め	[1, 2, 4, 5, 6, 7, 8, 9]
{バカなクソガキ}どもめ	[1, 2, 3, 4, 5, 6, 7]
統合後のインデックスの集合	[1, 2, 3, 4, 5, 6, 7, 8, 9]

作成したデータセット内のラベルの分布を表 2 に示す。また，攻撃的表現を「1: 含む」とされた 4,828 ツイートの約 20%に収集に用いた既定の攻撃的単語が含まれていた。得られたデータセットを 6:2:2 で訓練・開発・テストに分割した。

表 2 ラベルの分布

ラベル	例数
「0: 含まない」	28,828
「1: 含む」	4,828
合計	33,656

4 タスク設定と評価

4.1 タスク設定

本研究では，SNS 上の攻撃的表現の検出と位置特定という目的を 2 つのサブタスク，攻撃的表現の検出と攻撃的表現の位置特定に分けて考える。攻撃的表現の検出は入力されたリプライが攻撃的表現を含むかどうかの二値分類を行う。出力は「0: 含まない」「1: 含む」のタグとなる。攻撃的表現の位置特定は攻撃的表現が存在すればそのスパンを出力する。出力はスパンを表現する BIO タグとなる。

それぞれのサブタスク単体の実験と 2 つのサブタスクをまとめた実験を行う。サブタスクのまとめ方

1） https://developer.twitter.com/en

として，2 つのサブタスクをパイプライン式に行う方法と両方のサブタスクを一括式に行う方法の 2 通りで実験を行う。パイプライン式では攻撃的表現の検出を行った後，検出結果が「1: 含む」の例についてその位置特定を行う。一括式では攻撃的表現の位置特定を行い，出力にスパンが含まれないような例を攻撃的表現を含まないとすることで攻撃的表現の検出と位置特定の両方の結果を得る。

4.2 評価

攻撃的表現の検出の評価には，「1: 含む」を対象とした Accuracy，Precision，Recall，F1 値を用いる。

攻撃的表現の位置特定の評価には，Partial Match F1，Exact Match F1，Char-offsets F1 を用いる。Partial Match と Exact Match は固有名認識 (Named Entity Recognition) における評価であり，スパンをベースに評価を行う。Partial Match は正解のスパンの一部にモデル出力のスパンが被っていればそのスパンについては TruePositive としてカウントし，Exact Match は正解とモデル出力のスパンが完全に一致してる場合を TruePositive としてカウントする。

Char-offsets F1 は Pavlopoulos ら [5] が用いた評価尺度である。リプライ内のスパンを文字オフセットに変換し，正解とモデル出力の文字オフセットリストから F1 を計算する。データセット中の各リプライについて上記の F1 を計算し，平均したものが Char-offsets F1 である。ただし，正解とモデル出力の文字オフセットリストが共に空の場合は F1 を 1，どちらか一方が空の場合は F1=0 として計算する。後述の表中では Partial Match F1，Exact Match F1，Char-offsets F1 をそれぞれ PM F1，EM F1，Co F1 と略記する。

5 学習モデル

5.1 攻撃的表現の検出

攻撃的表現の検出には BiLSTM モデルと ALBERT fine-tuning モデルを用いる。

BiLSTM モデルは 1 層の BiLSTM 層と二値分類器から成るモデルを使用した。埋め込み表現の獲得には ELMo[14] を用いた。

ALBERT Fine-tuning モデルは [1] の Offensive language の検出で最も良い結果のモデルを参考にした。学習済みの ALBERT[2] を本研究のデータセットで fine-tuning したモデルである。ALBERT の入力トークン列の先頭に挿入される特殊トークン [CLS] の最終隠れ状態を二値分類器の入力とする。

5.2 攻撃的表現の位置特定

攻撃的表現の位置特定には BiLSTM-CRF モデルと BERT-CRF fine-tuning モデルを用いる。

BiLSTM-CRF モデルは 1 層の BiLSTM 層と CRF 層から成るモデルを使用した。CRF[15] 層は，各トークンに対する各タグのスコアを入力としてタグの依存関係を考慮した全体の最適なタグ列を出力する。埋め込み表現の獲得には ELMo を用いる。

BERT-CRF fine-tuning モデルは [5] の Toxi Span Detection において最も良い結果のモデルを参考にした。学習済みの BERT[3] を本研究のデータセットで fine-tuning したモデルである。BERT-CRF は NER において最先端のモデルの 1 つである。各トークンに対する BERT の最終 Transformer 層の隠れ状態を CRF に入力する。

5.3 一括式のタスク設定

一括式のタスク設定では，5.2 の BERT-CRF fine-tuning モデルと同様のものを用いる。ただし，5.2 のモデルについては攻撃的表現を含む例のみで学習を行うが，一括式のタスク設定におけるモデルでは全ての例を用いて学習を行う。

6 評価実験

6.1 サブタスクの実験結果

表 3 に攻撃的表現の検出の結果，表 4 に攻撃的表現の位置特定の結果を示す。攻撃的表現の位置特定については入力が必ず攻撃的表現を含むという前提であり，単体のタスクとして考えた場合の結果である。これらの結果は異なるシード値における 5 回の実験の結果の平均値である。

表 3 攻撃的表現の検出モデルの単タスク性能評価

モデル	F1	Precision	Recall	Accuracy
BiLSTM	.540	.742	.426	.894
ALBERT fine-tuning	.592	.695	.517	.896

2） https://github.com/cl-tohoku/bert-japanese

3） https://github.com/alinear-corp/albert-japanese

表 4　攻撃的表現の位置特定モデルの単タスク性能評価

モデル	Co F1	PM F1	EM F1
BiLSTM-CRF (ELMo emb.)	.464	.716	.247
BERT-CRF fine-tuning	.584	.777	.329

6.2　タスク全体の結果

6.1 でそれぞれのサブタスク単体としての結果を示した。それらのサブタスクで最も結果が良かったモデルである ALBERT fine-tuning と BERT-CRF fine-tuning を接続したパイプライン式のモデルの結果と BERT-CRF fine-tuning を用いた一括式のモデルの結果を表 5 と表 6 に示す。

表 5　攻撃的表現の検出：パイプライン式・一括式

モデル	F1	Precision	Recall	Accuracy
パイプライン式	.592	.695	.517	.896
一括式	.538	.729	.427	.901

表 6　攻撃的表現の位置特定：パイプライン式・一括式

モデル	Co F1	PM F1	EM F1
パイプライン式	.874	.505	.209
一括式	.883	.492	.221

6.3　パイプライン式と一括式の比較

パイプライン式と一括式の「1: 含む」の検出成功数を表 7 に示す。また，パイプライン式と一括式の Partial Match 基準のスパン特定成功数を表 8 に示す。それぞれモデルは 6.2 と同様である。攻撃的表現のスパンについてはパイプライン式と一括式のそれぞれで位置特定に成功しているスパンのうち 5 割以上が各モデルでのみ特定に成功している。そのため，位置特定についてはアンサンブルすることで結果が向上する可能性が高いといえる。

6.4　評価尺度について

攻撃的表現の位置特定の評価に用いた Char-offsets F1 は，攻撃的表現のスパンがない例がデータセット中に多い場合にスパンがある例の影響が小さくなり，スパンを特定できているかどうかの差が値に出にくくなってしまう。Char-offsets F1 を用いていた先行研究 [5] では，既存のデータセットで “Toxic” と判定されたコメントのみを対象としてデータセットを作成しているため上記の問題は発生しない。本研究においては，攻撃的表現の位置特定のみの実験では攻撃的表現を含む例のみを対象としているため上記の問題は発生しないが，タスク全体での攻撃的表現の位置特定の評価ではデータセットの多くが攻撃的表現のスパンを含まない例のため評価指標に適していないと考えられる。

表 7　パイプライン式と一括式の「1: 含む」の検出成功数

		一括式 検出成功	一括式 検出失敗
パイプライン式	検出成功	288	184
	検出失敗	82	349

表 8　パイプライン式と一括式の特定成功スパン数

		一括式 特定成功	一括式 特定失敗
パイプライン式	特定成功	160	212
	特定失敗	239	505

同じく攻撃的表現の位置特定の評価に用いた Partial Match F1 及び Exact Match F1 については，例ごとにではなくスパンごとに評価しているため，Char-offsets F1 と同様の問題は発生しない。Partial Match F1 の問題点としてはモデル出力のスパンの長さを評価において考慮しなため，スパンをより長く出力するモデルにおいて評価が高くなってしまうというものがある。一方 Exact Match F1 では，過不足なく正解スパンと一致させるのは難しいという問題がある。実際に 3 人のアノテーターが全員「1: 含む」とした例について，そのスパンが完全一致しているものは 1032 例中 283 例であり，人手においてもスパンが完全一致する例は少ない。

7　結論

本研究では日本語における攻撃的表現の検出及び位置特定のタスクを提案した。本タスク用のデータセットの作成方法について述べ，作成したデータセットを用いて実験を行った。実験には最先端の機械学習モデルを用い，従来モデルでの結果との比較やいくつかの異なる条件での性能の比較を行なった。また，英語における既存の先行研究と関連研究を参考に評価手法について考察した。

今後の取り組みとしては，攻撃的であると判断したアノテーターの人数を利用した重み付きのロス関数の導入や本研究におけるパイプライン式のモデルと一括式のモデルのアンサンブルモデルを用いることが挙げられる。また，fine-tuning に用いた機械学習モデルの事前学習に Twitter のデータを用いることでも結果の改善が期待できる。

参考文献

[1] Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. SemEval-2020 task 12: Multilingual offensive language identification in social media (OffensEval 2020). In **Proceedings of the Fourteenth Workshop on Semantic Evaluation**. International Committee for Computational Linguistics, 2020.

[2] Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In **Proceedings of the 13th International Workshop on Semantic Evaluation**. Association for Computational Linguistics, 2019.

[3] Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. Predicting the Type and Target of Offensive Posts in Social Media. In **Proceedings of NAACL**, 2019.

[4] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. In **International Conference on Learning Representations**, 2020.

[5] John Pavlopoulos, Jeffrey Sorensen, Léo Laugier, and Ion Androutsopoulos. SemEval-2021 task 5: Toxic spans detection. In **Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)**. Association for Computational Linguistics, 2021.

[6] 尼崎航成, 向井宏明, 松井くにお. SNS における不適切投稿の検知. 第 82 回全国大会公演論文集, 2020.

[7] 松葉達明, 桝井文人, 井須尚紀. 学校非公式サイトにおける有害情報検出を目的とした極性判定モデルに関する研究. 言語処理学会第 17 回年次大会発表論文集, 2011.

[8] 新田大征, 桝井文人, プタシンスキ・ミハウ, 木村泰知, ジェプカ・ラファウ, 荒木健治. カテゴリ別関連度最大化手法に基づく学校非公式サイトの有害書込み検出. 第 27 回人工知能学会全国大会発表論文集, 2013.

[9] 安彦智史, 長谷川大, プタシンスキ・ミハウ, 中村健二, 佐久田博司. Id 交換掲示板における書きこみの隠語表記揺れを考慮した有害性評価. 情報システム学会誌, pp. 41–58, 2018.

[10] 三宅剛史, 松本和幸, 吉田稔, 北研二. 分散表現を用いた有害表現判別に基づく炎上予測. 人工知能学会第二種研究会資料, 2017.

[11] 大西真輝, 澤井裕一郎, 駒井雅之, 酒井一樹, 進藤裕之. ツイート炎上抑制のための包括的システムの構築. 人工知能学会全国大会論文集, 2015.

[12] 畠山鈴生, 桝井文人, プタシンスキ・ミハウ, 山本和英. 有害表現抽出に対する種単語の影響に関する一考察. 人工知能学会全国大会論文集, 2016.

[13] 工藤拓, 山本薫, 松本裕治. Conditional random fields を用いた日本語形態素解析. 情報処理学会研究報告. NL, 自然言語処理研究会報告, 2004.

[14] Wanxiang Che, Yijia Liu, Yuxuan Wang, Bo Zheng, and Ting Liu. Towards better UD parsing: Deep contextualized word embeddings, ensemble, and treebank concatenation. In **Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies**. Association for Computational Linguistics, 2018.

[15] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In **Proceedings of the Eighteenth International Conference on Machine Learning**. Morgan Kaufmann Publishers Inc., 2001.