いじめ表現辞書を用いたTwitter上のネットいじめの自動検出

大友　泰賀† 　張　　建偉†† 　中島　伸介††† 　李　　琳††††

† 岩手大学総合科学研究科理工学専攻　〒020–8551 岩手県盛岡市上田 4 丁目 3 番 5 号
†† 岩手大学理工学部　〒020–8551 岩手県盛岡市上田 4 丁目 3 番 5 号
††† 京都産業大学情報理工学部　〒603–8555 京都府京都市北区上賀茂本山
†††† 武漢理工大学計算機科学と技術学院　湖北省武漢市洪山区
E-mail: †{g0319030,zhang}@iwate-u.ac.jp, ††nakajima@cc.kyoto-su.ac.jp, †††cathylilin@whut.edu.cn

あらまし　パソコンやスマートフォンの普及に伴い，ネット上のいじめが深刻な問題となっている．本研究ではTwitter上のテキストを対象とし，いじめ表現辞書を構築する．いじめ表現辞書とは，単語とその単語がどれだけいじめと関連するかの程度を数値で表すものである．辞書に登録する単語は，特定の単語を使って収集したツイートに含まれている単語とし，その単語につける値は，SO-PMI というものを用いて算出する．構築したいじめ表現辞書を含む複数の特徴量を用いて複数の機械学習手法と組み合わせ、ネットいじめの自動検出に最適なモデルの構築を図った．構築したモデルを用いていじめ文、非いじめ文の分類の評価を行ったところ，多くの機械学習手法でいじめ表現辞書が正しい検出に貢献することが分かった．また，最も良かったモデルでは 90%を超える評価を得ることができた．

キーワード　ネットいじめ，いじめ表現辞書，Twitter，機械学習

1 はじめに

近年，パソコンやスマートフォンの普及に伴い，ネット上のいじめが深刻な問題となっている．平成 30 年 10 月の文部科学省の調査では，平成 29 年度のネットいじめの認知件数は 1 万 2632 件で，年々増え続けている．現状では，ネットいじめに関する研究の多くは教育分野で検討されている．情報分野では，ネットいじめの自動検出に係る研究はほとんどが英語データの分析であり，日本語データの分析はまだ少ない．さらに特徴量の分析が不十分である．よって，日本語のデータを対象とし，特徴量の分析を行い，ネットいじめを精度良く自動的に検出出来る技術が求められている．

本研究ではネットいじめの自動検出を目的とし，目的の実現のために機械学習を用いる．機械学習には特徴量の抽出と機械学習手法の選択が必要である．Twitter のテキスト (以後，ツイートと呼ぶ．) を対象とし，ネットいじめの検出に貢献度の高い特徴量の抽出を検討する．感情辞書を用いた感情値が有用であると考えているが，これまでの実験に用いた感情辞書 [1][2] は，リソースが新聞記事や国語辞書であり，ネット上特有の単語や最近生まれた単語は，既存の感情辞書には存在しないという問題があった．そこで，ネット上特有の単語や最近の単語に対応したいじめ表現辞書を作成し，特徴量として用いる．いじめ表現辞書の形式は，単語とその単語がどれだけいじめと関連するかの程度を数値で表すものとする．他にも N グラム，Word2vec，Doc2vec といった特徴量を使用する．また，複数の機械学習手法を用い，特徴量と組み合わせて，最適なモデルの構築を図る．収集したツイートをもとにモデルを構築し，いじめ文か非いじめ文かの自動検出の評価を行った結果，多くの機械学習手法でいじめ表現辞書が正しい検出に貢献することが分かった．また，最も良かったモデルでは 90%を超える評価を得ることができた．

本論文の構成を述べる．第 2 章では，関連研究を紹介する．第 3 章では提案手法の流れについて説明する．第 4 章ではデータの収集法といじめ表現辞書の作成方法について述べる．第 5 章ではいじめ文か非いじめ文かの分類に用いたデータのアノテーションについて述べる．第 6 章ではいじめ文か非いじめ文かの分類に用いた特徴量について述べる．第 7 章ではいじめ文か非いじめ文かの分類に用いた機械学習手法について述べる．第 8 章では収集したツイートをいじめ文か非いじめ文かに正しく自動分類できるかの実験とその結果についての考察を述べる．第 9 章では本論文のまとめと今後の課題について述べる．

2 関連研究

英語の攻撃的なテキストの自動検出に関する研究は盛んに行われている，Burnap ら [3] は，Twitter 上からとある殺人事件についてのツイートをハッシュタグを用いて取得し，N グラム，単語の依存関係の特徴を用いて人種や宗教の点で攻撃的なツイートの自動検出を試みている．Rafiq ら [4] は Vine(現在はサービス終了) と呼ばれる動画共有サービス上から動画とその動画につけられたコメントをセットで取得し，動画の情報，投稿したユーザー，コメントの感情，N グラムの特徴を用いてネットいじめやネット攻撃の自動検出を試みている．Hosseinmardi ら [5] は Instagram と呼ばれる写真共有サービスから写真とその写真につけられたコメントをセットで取得し，写真の情報，投稿したユーザー，コメント間の時間，N グラムの特徴を用いてネットいじめやネット攻撃の自動検出を試みている．Nobata ら [6] は Yahoo!金融と Yahoo!ニュースのコメントから N グラム，言語，構文，分布意味論の特徴を使って有害なコメントの

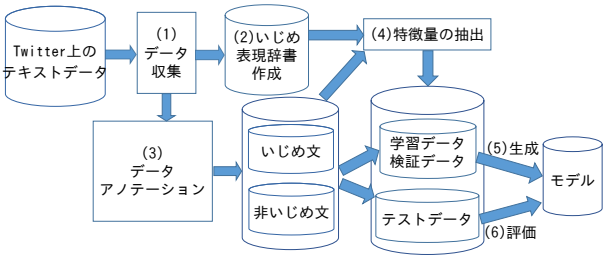

図 1　提案手法の流れ

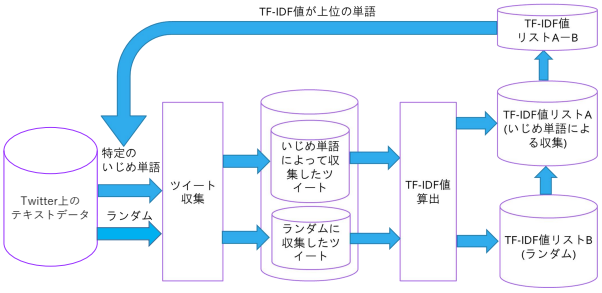

図 2　いじめ表現辞書に登録する単語の抽出の流れ

自動検出を試みている．Chatzakou ら [7] は，Twitter 上からハッシュタグを利用してツイートを取得し，それらのツイートをしたユーザーらをいじめユーザー，攻撃的ユーザー，スパムユーザー，普通のユーザーの 4 クラスに分類し，ユーザー，テキスト，ネットワークの特徴を用いて自動で正しく分類できるかを試みている．

日本では，三島ら [8] は，中高生向けソーシャルメディア上でソーシャルグラフを生成し，仲間集団の中で発生したネットいじめを検出している．中村ら [9] は，Twitter 上でネットいじめに関連する単語を組み合わせて検索し，ネットいじめに関連する投稿を発見できないかを試みている．さらにネットいじめ加害・被害ユーザ検出アルゴリズムを検討し，それらの流れについて述べている．

本研究では N グラムという基本的な特徴量だけではなく，いじめ表現辞書，Word2vec[10]，Doc2vec[11] などを特徴量として用いて Twitter 上の日本語のツイートに対して，ネットいじめの自動検出を試みるという点がこれらの研究とは異なっている．

3　提案手法の概要

提案手法の流れを図 1 に示す．

(1) データ収集 (4 章で説明)
いじめ表現辞書作成と分類に用いるツイートを Twitter 上から収集する．

(2) いじめ表現辞書作成 (4 章で説明)
いじめ表現辞書に登録する単語をツイートから抽出し，SO-PMI というものを用いていじめ度を算出する．

(3) データアノテーション (5 章で説明)
分類に用いるツイートを絞り込み，ツイートの中身に応じていじめ文，または非いじめ文のラベルを付ける．

(4) 特徴量の抽出 (6 章で説明)
ラベルを付けたツイートから各特徴量を抽出する．特徴量にはいじめ表現，N グラム，Word2vec，Doc2vec を用いる．

(5) モデルの生成 (7 章，8.1 節で説明)
ラベルを付けたツイートを学習・検証データとテストデータに分け，学習・検証データと各特徴量と各機械学習手法を用いてモデルを生成する．また，交差検証とグリッドサーチを用いてより良いモデルの生成を目指す．

(6) 評価 (8 章で説明)
生成したモデルがどれだけ正しくいじめ文か非いじめ文かを分類できるかの評価を行う．評価基準には F 値を用いる．

4　いじめ表現辞書作成

いじめ表現辞書の構築に必要なこととして，いじめ表現辞書に登録する単語をどのように自動抽出するかということと，登録した単語のいじめ度をどのように算出するかの 2 つが挙げられる．これらの方法について，説明していく．

4.1　いじめ表現辞書に登録する単語の自動抽出

いじめ表現辞書に登録する単語をどのように自動抽出したかを説明する前に，どのような単語をいじめ表現辞書に登録したいかについて説明する．一つはその単語単独で悪口となる単語，もう一つは単語単独では悪口とはならないが、他の単語と共起することで，悪口表現となる単語である．例として,「サル以下の脳みその持ち主」という悪口表現があった場合,「サル」がこれに該当する．これは，いじめに関わる数多くの単語を選出する網羅性を重視している．いじめ表現辞書に登録する単語の抽出の流れを図 2 に示す．

4.1.1　ツイート収集

ツイートを収集するために GetOldTweets3 という Python のライブラリーを用いた．収集期間は 2019/06/16~2019/06/23 の 1 週間である．また，いじめに関するツイート (いじめ文) として特定のいじめ単語を用いて収集，いじめと関係のないツイート (非いじめ文) としてランダムに収集の 2 パターンで収集を行った．初期いじめ単語として，石坂ら [12] と新田ら [13] と畠山ら [14] の研究で用いられていたいじめ単語，合計 39 単語を使用した．

4.1.2　TF-IDF 値を算出

TF-IDF とは，文書中に含まれる単語の重要度を評価する手法の 1 つであり，主に情報検索やトピック分析などの分野で用いられている．TF-IDF は，単語の出現頻度：TF(Term Frequency) と逆文書頻度：IDF(Inverse Document Frequency) の二つの指標に基づいて計算される．IDF は一種の一般フィルタとして働き，多くの文書に出現する語は重要度が下がり，特定の文書にしか出現しない単語の重要度を上げる役割を果たす．それぞれを求める式は，以下のようになる．

$$TF\text{-}IDF_{ij} = TF_{ij} * IDF_i$$

$$TF_{ij} = \frac{n_{ij}}{\sum_k n_{kj}}$$

$$IDF_i = \log \frac{|D|}{|d : d \ni t_i| + 1} + 1$$

n_{ij}：文書 d_j における単語 t_i の出現回数
$\sum_k n_{kj}$：文書 d_j における全ての単語の出現回数の和
$|D|$：総文章数
$|d : d \ni t_i|$：単語 t_i を含む文書数

いじめ文と非いじめ文でそれぞれ別にツイートごとの単語の TF-IDF 値を求め，単語ごとに合計する．その前に収集したツイートの形態素解析を行う．なお，事前に形態素解析に不要と思われる特定の文字を排除する．排除する文字は Twitter 特有の単語である“RT”，“お気に入り”，“まとめ”や URL，半角記号，数字，英字，全角記号，改行文字，全角空白，半角空白である．形態素解析には JUMAN++[15] を使用した．JUMAN++ とは言語モデルを利用した高性能な形態素解析システムである．言語モデルとして Recurrent Neural Network Language Model(RNNLM) を用いることにより，単語の並びの意味的な自然さを考慮した解析を行う．

TF-IDF 値を算出する形態素は，JUMAN++の解析結果によって，品詞が「名詞」,「動詞」,「形容詞」,「未定義語」であるものとした．未定義語を含んだのは，新しい単語は未定義語に定義されている可能性が高いためである．いじめ文で TF-IDF 値を求めた場合，TF-IDF 値の合計値が大きいほど，いじめに関連する単語である可能性が高くなる．

4.1.3 ツイート再収集

いじめ文に対する TF-IDF に基づく抽出の結果から，TF-IDF 値が上位の単語を用いて再び Twitter 上からツイートを収集する．しかし，いじめ文の中には，日常的に使われる単語が含まれているため，TF-IDF 値が高い順にソートした際に，上位に現れるいじめ単語の割合が低くなる可能性がある．よっていじめ文に対する TF-IDF に基づく抽出の結果から，非いじめ文に対する TF-IDF に基づく抽出の結果を削除する．さらに，事前に筆者が目視で TF-IDF 値が上位の単語をチェックし，明らかにいじめに関係のなさそうな単語は除外している．初期いじめ単語数が 39 単語なので，それに近い上位 30 単語を用いて再び Twitter 上からツイートを収集した．

この流れを，複数回繰り返して行う．今回の実験では，3 回目の 4.1.1 節まで，すなわち 3 回目の特定のいじめ単語でツイートを収集するところまで行い，収集したツイートは，計 2,827,284 ツイートとなった．

収集した 3 回分のツイートの 10%を利用し，その中に含まれる形態素をいじめ表現辞書に登録した．なお，登録する品詞は「名詞」,「動詞」,「形容詞」,「未定義語」である．

4.2 登録した単語のいじめ度の算出

いじめ度の算出には SO-PMI(Semantic Orientation Using Pointwise Mutual Information)[16] を使用した．SO-PMI とは，事前に 2 つの基本単語を用意し、対象の単語がその 2 つのどちらと文書内共起しやすいかを計るものである．SO-PMI を求める式は，以下のようになる．

$$C(w) = \log \frac{hit(w, w_p) * hit(w_n)}{hit(w, w_n) * hit(w_p)}$$

$$f(\alpha) = \alpha * \log \frac{hit(w_p)}{hit(w_n)}$$

$$SO - PMI(w) = C(w) + f(\alpha)$$

w：SO-PMI 値を求める単語
w_p：いじめ基本単語
w_n：非いじめ基本単語
$hit(w)$：w を検索語とした時の Twitter のヒット件数
$hit(w, w_p)$：w と w_p を同時に検索語とした時の Twitter のヒット件数
$C(w)$：w_p と w_n のどちらと共起しやすいか
$f(\alpha)$：検索ヒット数の差による優位性を解消する関数
α：関数の重みを設定するための定数

関数の重みを設定するための定数 α についてだが，今回の実験では，0.9，0.6，0.3，0.0 の 4 種類で SO-PMI 値を算出し，筆者が単語につけられたいじめ値が最も人間と近い感覚と判断した 0.6 を採用とした．

SO-PMI において基本単語となる w_p と w_n をどのような単語にするかが重要である．元の定義では，基本単語は 1 つずつだが，本研究では，対象としているヒット件数が文量の少ない Twitter ということもあるので，複数設定することにした．

いじめ基本単語 w_p は，初期いじめ単語の中で検索ヒット数がある程度多く，かつその単語のみで悪口表現となり得る「死ね」,「カス」,「キモい」,「クズ」,「ブス」の 5 語とした．非いじめ基本単語 w_n は，石坂らの研究において，褒め言葉のような賞賛単語として用いられていた「可愛い」,「素敵」,「イケメン」,「素晴らしい」,「美しい」の 5 語とした．

4.3 結　　果

今回の手法では，86,906 語の単語を登録することが出来た．登録した単語の一部を図 3 に示す．本研究では，いじめ度が高いほど，その単語がいじめに関連する可能性が高いという意味を持ち，いじめ度が低いほど，その単語がいじめに関連する可能性が低いという意味を持つ．この辞書を，特徴量として後述する実験に用いた．

5 データアノテーション

分類に用いるデータは，4 章で収集した 3 回分のツイートである．これらのツイートを Yahoo!クラウドソーシングを利用してクラウドワーカーにいじめ・非いじめのラベル付けをしてもらう．しかし，この時点でツイートの数は膨大である．そこで，クラウドワーカーによる選別にかかるコストを低くするために，ツイート文にスコア付与しランキングを行った．ランキ

単語	いじめ度(SO-PMI値)
アホカス	5.247
消えろ	4.193
メスブタ	2.788
むかつく	1.833
陰湿な	0.836
悲しい	-1.007
感無量	-3.885
見とれて	-4.801

図 3　いじめ表現辞書の一部

ングは 3 回のツイート収集に使った特定のいじめ単語 97 語 (実際は 99 語あるが，そのうち 2 語は Twitter 上で 1 件もヒットしなかったため，97 語となっている．) がそれぞれ含まれているツイートごとに行い，スコアは，ツイートの中に，3 回のツイート収集に使った特定のいじめ単語が何種類含まれているかで付与を行った．スコアを付与した後，スコア上位ツイートと下位ツイートの 2 つに分け，スコア上位ツイートの上位 50 ツイートとスコア下位ツイートの 25 ツイートの計 75 ツイートを 97 単語分抽出した．これは，様々ないじめ単語の組み合わせや，種類数のツイートをラベル付けしてもらい，学習データとして用いたいためである．さらにここから重複しているツイートを削除した結果，5,399 ツイートを抽出した．この 5,399 ツイートを Yahoo!クラウドソーシングに提出した．

Yahoo!クラウドソーシングとは，Yahoo 社が提供するサービスの一種である．サービス内容として，企業や個人の大量タスクを多くのサービス利用者に依頼できる場を提供している．他のクラウドソーシングプラットフォームとの違いとして，依頼内容に関して Yahoo!と相談できる点，実施したタスクの結果を tsv ファイルとして受け取ることができる点がある．人間がある文に対していじめが関わっているかどうか判断する基準は様々であるため，[17] や [18] を参考に著者側で分類基準を定義した．分類基準として，

・特定の誰かに対して侮辱的な発言をしている
・特定の誰かをある集団から除外・無視しようとしている発言をしている
・特定の誰かの個人情報，または大衆に知られたくないような情報が含まれている発言をしている
・特定の誰かに対して性的嫌がらせ (セクハラ) な発言をしている
・上 4 つの表現のいずれかを現在進行形でされているような発言をしている
・上 4 つの表現を支持するような発言をしている

の 6 つを定義した．また，1 人に判断してもらうツイートは 10 ツイートとし，1 つのツイートに対し，3 人に評価してもらった．

その結果，660 ツイートがクラウドワーカーが 3 人一致でいじめであると判断したツイート，1,106 ツイートが 2 人がいじめであると判断したツイート，1,524 ツイートが 1 人だけいじめであると判断したツイート，2,109 ツイートが 3 人一致でいじめでないと判断したツイートとなった．この内，3 人一致でいじめと判断された 660 ツイートと 2 人がいじめであると判断した 1,106 ツイートの計 1,766 ツイートを後述する実験でいじめ文として用いた．

6　特徴量の抽出

機械学習を用いるいじめ文の検出には，分類に有効な特徴量の選択が重要である．本研究ではラベル付けしたツイートからいじめ表現，N グラム，Word2vec，Doc2vec を抽出し，これらを特徴量として使用した．

6.1　いじめ表現

単語とその単語がどれだけいじめと関連するかの程度を数値で表している．辞書に登録されている全ての単語を列ベクトル，ラベル付けされたツイートを行ベクトルとした行列を生成し，ツイート内の (各単語の出現頻度*各単語のいじめ度) を各要素の値とした．

しかし，このまま列ベクトル一つ一つを全て特徴量として使ってしまうと膨大な数の特徴量となってしまい，明らかに全く重要ではない列ベクトルも多数存在している．この問題を解決するために主成分分析という方法を用いた．

主成分分析とは，データセットの特徴量を相互に統計的に関連しないように回転する手法である．回転したあとの特徴量から，データを説明するのに重要な一部の特徴量だけを抜き出す．アルゴリズムとしてはまず最も分散が大きい方向を見つけ，それに「第 1 成分」というラベルを付ける．データはこの方向に対して最も情報を持つ．つまりこの方向は特徴量が最も相互に関係する方向である．次に「第 1 成分」と直交する方向の中から，最も情報を持っている方向を探す．それに「第 2 成分」というラベルを付ける．このようにして見つけていく方向がデータの分散が存在する主要な方向であり,「主成分」と呼ぶ．主成分はもとの特徴量と同じ数だけ存在するが，主成分はデータを説明するのに重要な順にソートされており，パラメータで主成分の残す数を決めることで，重要度の大きい主成分のみを残しながら次元削減をすることができる．本研究では 100 次元まで次元削減を行った．

6.2　N グラム

任意の文字列や文書を連続した N 個の文字列または単語で分割し，それがどの程度出現するかを調査する言語モデルである．文字列や単語の発生確率が直前の文字列や単語に依存すると仮定して扱われる．

本研究では文字単位での分割 (以下文字 N グラム) と単語単位での分割 (以下単語 N グラム) をともに使用し，文字 N グラムでは N=2~5，単語 N グラムでは N=1~5 とした．文字 N グラム，または単語 N グラムを生成した後，各文字，単語の連なりを列ベクトル，ラベル付けされたツイートを行ベクトルとした行列を生成し，ツイート内の各文字、単語の連なりの出現頻度を各要素の値とした．

しかし，このまま列ベクトル一つ一つを全て特徴量として使ってしまうと膨大な数の特徴量となってしまい，明らかに全く重要ではない列ベクトルも多数存在している．さらに文字 N グラムに関しては，それらのほとんどが「文章的には意味のない文字の組み合わせ」であり，ノイズの発生にもなってしまう．この問題を解決するためにこちらも主成分分析を用い、各 N グラムごとに 100 次元まで次元削減を行った．

6.3 Word2vec

Word2vec とは，大量のテキストデータを解析・学習し，各単語の意味をベクトル表現化する手法である．各単語のベクトル同士の内積を計算することで意味の近い単語を知ることができたり，単語のベクトル同士の加算減算などをすることによって単語間の関係性を理解することができる．

本研究では各単語のベクトルを利用することでいじめ文の分類に役立てようとした．学習には 4 章で述べた特定のいじめ単語を用いて収集した 2,827,284 ツイートを用いた．各単語の単語ベクトルは 100 次元に設定した．1 ツイート単位で分類をするため，1 ツイートに含まれる各単語の単語ベクトルの平均をそのツイートの Word2vec の特徴量とした．

6.4 Doc2vec

Doc2vec とは，大量のテキストデータを解析・学習し，ベクトル表現化する手法である．Word2vec と異なる点は，ベクトル表現化の際に文中の単語の語順を考慮する点と，ベクトル表現化するのが単語単位だけではなく，文書単位でもあるという点である．これらの点により文書間の関係性も理解することが出来る．

学習には Word2vec と同じく 4 章で述べた特定のいじめ単語を用いて収集した 2,827,284 ツイートを用いた．文書 (ツイート) ベクトルは 100 次元に設定した．1 ツイート単位で分類をするため，ベクトル表現化も 1 ツイート単位で行い，そのベクトルをそのツイートの Doc2vec の特徴量とした．

7 機械学習モデルの選択

機械学習を用いるいじめ文の検出には，機械学習手法の選定も重要である．本研究では単純なモデルである線形モデルの (1) 線形サポートベクトルマシンと (2) ロジスティック回帰，木構造のモデルである (3) 決定木と (4) ランダムフォレストと (5) 勾配ブースティング回帰木，複雑なモデルであるニューラルネットワークの一種である (6) パーセプトロンを使用した．

8 評価実験

8.1 実験の設定

ラベル付けされたツイートを用いて，各特徴量，各機械学習手法によって学習させ，どれだけ正しく検出できるか，そしてどの特徴量，どの機械学習手法が分類に役立つかの実験を行った．

分類に用いるツイートは 5 章で述べた，クラウドワーカーによって 3 人一致でいじめと判断された 660 ツイートと 2 人がいじめであると判断した 1,106 ツイートの計 1,766 ツイート (いじめ文) とランダムに収集した 1,766 ツイート (非いじめ文) の計 3,532 ツイートである．なお，ランダムに収集したツイートに対しては，あらかじめ，いじめ表現辞書構築に用いた特定のいじめ単語 97 語がツイート内に含まれないように収集した．

特徴量に関してはまず，1 種類ずつ使用した．いじめ表現，文字 N グラム (N=2∼5)，単語 N グラム (N=1∼5)，Word2vec，Doc2vec の 5 種類である．次に良い評価をした特徴量をベースとして，いじめ表現と組み合わせて使用した．最後は全ての特徴量を用いたものと，全ての特徴量からいじめ表現辞書を除いたものを使用した．

機械学習に関しては全ての機械学習手法共通で，ネストされた層化 5 分割交差検証を用いた．まず，データを各分割内でのラベルの比率が全体の比率と同じになるように 5 個に分割し，5 個のうち 4 個を学習・検証データ，1 個をテストデータとする．さらにこの学習・検証データを各分割内でのラベルの比率が全体の比率と同じになるように 5 個に分割し，4 個を学習データ，1 個を検証データとし，グリッドサーチを用いてあらかじめ指定していたパラメータの全ての組み合わせに対して交差検証を行う．もっとも良い評価を示したパラメータのモデルを生成し，テストデータを使ってモデルの評価を行う．一度モデルの評価が終わったら 4 個のうち 1 個の学習・検証データとテストデータを入れ替え，再びモデルの生成と評価を行う．これを 5 回行い，5 回の平均分類評価を出し，その値を分類評価として使うことで，データの分割によらない頑健な評価を可能にしている．

テストデータの評価に関しては F 値 (F-measure) を利用した．F 値 (F-measure) は，適合率 (Precision) と再現率 (Recall) の調和平均をとった値である．なお，適合率 (Precision) は，モデルがいじめ文と判断したデータのうち，実際にいじめ文であるものの割合であり，再現率 (Recall) は，全てのいじめ文のうち，モデルがいじめ文と判断したものの割合である．

それぞれの値を求める式は，以下のようになる．

$$F - measure = \frac{2 * Recall * Precision}{Recall + Precision} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

なお，

TP:モデルがいじめ文と判断したいじめ文
FP:モデルがいじめ文と判断した非いじめ文
FN:モデルが非いじめ文と判断したいじめ文

である．

8.2 実験結果

結果を表 1，表 2，表 3 に示す．使用した特徴量と機械学習

表 1　分類評価 (1 種類ずつと全ての特徴量)

	いじめ表現	文字 N グラム	単語 N グラム	Word2vec	Doc2vec	全ての特徴量
線形サポートベクトルマシン	0.822	0.870	0.812	**0.902**	0.881	0.819
ロジスティック回帰	0.838	0.870	0.813	0.882	**0.883**	**0.921**
決定木	**0.805**	0.759	0.669	0.774	0.706	**0.827**
ランダムフォレスト	0.801	0.802	0.695	**0.839**	0.767	**0.840**
勾配ブースティング回帰木	0.828	0.830	0.760	**0.868**	0.810	**0.892**
パーセプトロン	0.845	0.869	0.816	**0.910**	0.882	**0.914**

表 2　分類評価 (評価の良い特徴量と「評価の良い特徴量+いじめ表現」)

	Word2vec	Word2vec+いじめ表現	Doc2vec	Doc2vec+いじめ表現
線形サポートベクトルマシン	**0.902**	0.894	**0.881**	0.872
ロジスティック回帰	0.882	**0.899**	0.883	**0.899**
決定木	0.774	**0.812**	0.706	**0.804**
ランダムフォレスト	0.839	**0.853**	0.767	**0.813**
勾配ブースティング回帰木	0.868	**0.890**	0.810	**0.844**
パーセプトロン	**0.910**	0.907	0.882	**0.892**

表 3　分類評価 (全ての特徴量と「全ての特徴量-いじめ表現」)

	全ての特徴量	全ての特徴量-いじめ表現
線形サポートベクトルマシン	0.819	**0.833**
ロジスティック回帰	**0.921**	0.920
決定木	**0.827**	0.783
ランダムフォレスト	**0.840**	0.824
勾配ブースティング回帰木	**0.892**	0.881
パーセプトロン	**0.914**	**0.914**

手法の組み合わせと，その F 値を示している．

まず，表 1 から見ていく．特徴量を 1 種類ずつ用いた中で，各機械学習手法ごとに最も良い評価の特徴量の部分のセルを太字にしている．さらに，全ての特徴量がその機械学習手法の時に最も良い評価の場合も，そのセルを太字にしている．特徴量を 1 種類ずつ用いた中で，最も良い評価を示したモデルは，特徴量は Word2vec，機械学習手法はパーセプトロンを用いたもので，91%を示した．各機械学習手法ごとに見てみると，Word2vec が線形サポートベクトルマシン，ランダムフォレスト，勾配ブースティング回帰木，パーセプトロンの 4 つの機械学習手法で最も良い評価を示しており．Doc2vec がロジスティック回帰で最も良い評価を示している．いじめ表現は決定木において最も良い評価を示した．したがって，基本的には Word2vec の特徴量が正しい検出に大きく貢献しているが，機械学習手法が決定木の場合，いじめ表現が特に正しい検出に大きく貢献してくれることが分かった．

次に表 2 を見ていく．特徴量を 1 種類ずつ用いた中で評価の良かった Word2vec と Doc2vec をいじめ表現と組み合わせて実験した．各機械学習手法ごとに Word2vec と Doc2vec それぞれで単体といじめ表現を組み合わせた場合を比較し，より評価が良かった方の部分のセルを太字にしている．Word2vec では，ロジスティック回帰，決定木，ランダムフォレスト，勾配ブースティング回帰木の 4 つの機械学習手法でいじめ表現と組み合わせた方が評価が良くなり，Doc2vec では，それらにパーセプトロンを加えた 5 つの機械学習手法でいじめ表現と組み合わせた方が評価が良くなった．良くならなかった残りの部分はどれも 1%未満の差であり，同程度と考えることが出来る．したがって評価の良かった特徴量にいじめ表現を組み合わせることによって，さらに正しい検出に貢献してくれることが分かった．

最後に表 3 を見ていく．各機械学習手法ごとに全ての特徴量と全ての特徴量からいじめ表現を除いた場合を比較し，より評価が良かった方の部分のセルを太字にしている．ロジスティック回帰，決定木，ランダムフォレスト，勾配ブースティング回帰木，パーセプトロンの 5 つの機械学習手法でいじめ表現を除いた場合，評価が悪くなる，または同等だった。したがって，多くの特徴量の中の 1 つとしていじめ表現を使っても，正しい検出に貢献してくれることが分かった．

8.3 考　　察

特徴量について考察する．Word2vec や Doc2vec がその特徴量単体で良い評価を示したのは，学習に用いた 2,827,284 ツイートという大量のテキストデータによって単語をうまくベクトル表現できていたからだと考えられる．一方，いじめ表現は，機械学習手法が決定木の時や，他の特徴量と組み合わせることによって，正しい検出に貢献してくれることが分かったが，機

械学習手法が決定木以外のときにいじめ表現単体で用いた場合，他の特徴量よりも良い評価を示したとは言えなかった．これは考えられる原因が 2 つある．1 つ目はいじめ表現辞書に登録する単語が少なかったということである．Word2vec や Doc2vec が 2,827,284 ツイートのテキストデータからモデルを構築したのに対し，いじめ表現辞書はそのうちのわずか 10%内の単語しか登録しなかった．時間の制約上，10%しか登録できなかったのだが，もっと多くのツイート内から単語登録すれば，結果が変わる可能性がある．2 つ目は，SO-PMI 値を算出する際の基本単語である．本研究では，基本単語の選定を，主に過去の研究を参考に決めてしまった．今回の目的にふさわしい基本単語を，あらかじめ予備実験を行うことで厳選した上で，SO-PMI 値を算出することによって，結果が変わる可能性がある．したがって，いじめ表現辞書に改善の余地があると考えられる．

9 まとめと今後の課題

本研究では Twitter 上のテキストを対象とし，いじめ表現辞書を作成した．この辞書を含む複数の特徴量を用いて複数の機械学習手法と組み合わせ，モデルを生成することで，ネットいじめの自動検出を試みた．本研究では用いた特徴量や機械学習手法によっては高い分類評価が出ており，いじめ表現辞書が正しい検出に貢献するということが分かったが，考察でも述べた通り，まだ改善の余地が残っている．

今後はより多くの単語を登録し，SO-PMI 値を算出する際の基本単語の厳選を行い，いじめ表現辞書を改善していきたい．

謝　　辞

本研究は JSPS 科研費 19K12230 の助成を受けたものである．

文　　献

[1] 熊本忠彦，河合由起子，張建偉，“ユーザ印象評価データの分析に基づく印象マイニング手法の設計と評価，” 情報処理学会論文誌データベース，vol.6，no.2，pp.1-15，2013 年.
[2] 高村大也，乾孝司，奥村学，“スピンモデルによる単語の感情極性抽出，” 情報処理学会論文誌，vol.47，no.2，pp.627-637，2006 年.
[3] Pete Burnap, Matthew L. Williams, “Cyber Hate Speech on Twitter: An Application of Machine Classification and Statistical Modeling for Policy and Decision Making,” Policy & Internet, vol.7, no.2, pp.223-242, 2015.
[4] Rahat Ibn Rafiq, Homa Hosseinmardi, Richard Han, Qin Lv, Shivakant Mishra, Sabrina Arredondo Mattson, “Careful what you share in six seconds: Detecting cyberbullying instances in Vine,” ASONAM 2015, pp.617-622, 2015.
[5] Homa Hosseinmardi, Sabrina Arredondo Mattson, Rahat Ibn Rafiq, Richard Han, Qin Lv, Shivakant Mishra, “Detection of Cyberbullying Incidents on the Instagram Social Network,” AAAI 2015, 2015.
[6] Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, Yi Chang, “Abusive Language Detection in Online User Content,” WWW 2016, pp.145-153, 2016.
[7] Despoina Chatzakou, Nicolas Kourtellis, Jeremy Blackburn, Emiliano De Cristofaro, Gianluca Stringhini, Athena Vakali, “Mean Birds: Detecting Aggression and Bullying on Twitter,” WebSci 2017, pp.13-22, 2017.
[8] 三島 浩路，本庄 勝，“技術的観点からのネットいじめ対策，” 通信ソサイエティマガジン，No.34 秋号，2015 年.
[9] 中村 健二，寺口 敏生，“CGM 解析に基づくネットいじめ被害の検出手法の検討，” 大阪経大論集，第 66 巻第 5 号，2016 年.
[10] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean, “Distributed Representations of Words and Phrases and their Compositionality,” NIPS 2013, pp.3111-3119, 2013.
[11] Quoc Le, Tomas Mikolov, “Distributed Representations of Sentences and Documents,” ICML 2014, pp.1188-1196, 2014.
[12] 石坂達也，山本和英，“Web 上の誹謗中傷を表す文の自動検出，” 言語処理学会第 17 回年次大会発表論文集，2010 年.
[13] 新田大征，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，” 第 30 回人工知能学会全国大会，2016 年.
[14] Suzuha Hatakeyama, Fumito Masui, Michal Ptaszynski, Kazuhide Yamamoto, “Statistical Analysis of Automatic Seed Word Acquisition to Improve Harmful Expression Extraction in Cyberbullying Detection,” IJETI 2016, vol.6, no.2, pp.165-172, 2016.
[15] Hajime Morita, Daisuke Kawahara, Sadao Kurohashi, “Morphological Analysis for Unsegmented Languages using Recurrent Neural Network Language Model,” EMNLP 2015, pp.2292-2297, 2015.
[16] Guangwei Wang, Kenji Araki, “Modifying SO-PMI for Japanese weblog opinion mining by usinga balancing factor and detecting neutral expressions,” ACL 2007, pp.189-192, 2007.
[17] Cynthia Van Hee, Gilles Jacobs, Chris Emmery, Bart Desmet, Els Lefever, Ben Verhoeven, Guy De Pauw, Walter Daelemans, Vronique Hoste, “Automatic detection of cyberbullying in social media text,” PLOS ONE, 2018.
[18] Michal Ptaszynski, Juuso Kalevi Kristian Eronen, Fumito Masui, “Learning Deep on Cyberbullying is Always Better Than Brute Force,” LaCATODA 2017, vol.1926, pp.3-10, 2017.