Vol.2020-ICS-200 No.2
2020/9/14

コーパス間での類似語の差異に着目したマイクロブログにおける隠語検出

羽田　拓朗[†1,a)]　清 雄一[†1]　田原 康之[†1]　大須賀 昭彦[†1]

概要：近年, マイクロブログにおける違法薬物取引等が増加の一途をたどっており, 社会的な問題となっている. こういった犯罪を取り締まるためのサイバーパトロールが行われてはいるものの, 犯罪を誘導する投稿を行う者たちは, 監視されそうな可能性のあるキーワード（「援助交際」,「大麻」等）を避け, 犯罪の意図をカモフラージュした用語, いわゆる「隠語」を駆使して, 監視の目をかいくぐりながら巧妙にやり取りを続けている. これらの隠語は, 一度把握したとしても, 一般的に広まれば陳腐化し, 新たな隠語が使われ始めてしまうと言われており, 既存の隠語を元に検索する方法では, 常に最新の隠語を把握しなくてはならず, 非常に労力がかかる. そこで, 本研究では犯罪を誘導する投稿に含まれる可能性が高い隠語を検出することを目的とし, 単語の用途の差異から隠語を検出する新たな手法を提案する. 具体的には, 二つのコーパス間における単語の用途の差異に着目し, コサイン類似度上位に出現する単語の差異から隠語を検出する方法を考案した. そして, 本提案手法の効果を確認するため, 隠語検出実験を行った. 実験の結果, 用意した隠語リスト以外の隠語を検出でき, また比較用に用意した手法の実験結果と比べて, 大きな差をつけることができた. 本手法を用いて, 時と共に変遷する新しい隠語や犯罪を誘導する投稿を自動的かつ迅速に発見することができるようになれば, 隠語の継続的把握の負担が軽減でき, 犯罪の端緒を迅速に掴むことが期待できる.

1. はじめに

現在, 日本では,Twitter に代表されるマイクロブログを悪用した援助交際や違法薬物に関連する事件が数多く発生している. ここで, 援助交際については, 海外でも「enjo kosai」という言葉で取り扱われており,Miller 氏は,「若い女性が見知らぬ男性との, お金や贈り物と引き換えに, 時にはセックスを含むデートをすること」と表現している [1]. 中でも 18 歳未満による援助交際が問題となっている.

このような援助交際や違法薬物のやり取りを目的とした投稿者は, 警察や SNS の運営会社によるサイバーパトロールによって, 自分たちの投稿を削除されたり, 自分たちのアカウントを凍結されたり, 警察に検挙されたりすることを警戒している. そのため, 図 1 のように隠語を用いて, 言葉の意味を知っている人たちだけで違法な取引を実施している.

隠語は, 例えば, 違法薬物売買においては, 大麻の場合,「マリファナ」,「ガンジャ」, 覚醒剤には「エス」,「シャブ」といった単語が用いられていることが一般に知られている.

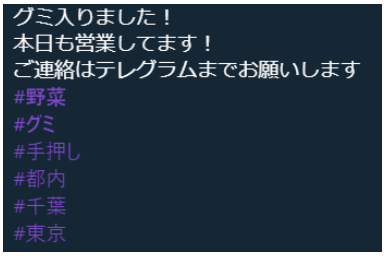

図 1: 隠語が用いられた文章例：違法薬物の入荷を隠語を使って表現している

これらの隠語をリスト化して定期的にキーワード検索により検知する対策をとったとしても, 効果は限定的と思われる. なぜなら, 隠語の特徴として, 一度広まると監視を回避するために新しい隠語に変化すると言われている [2]. 例えば, 大麻の場合,「草」,「雑草」「ジョイント」, 覚醒剤の場合,「アイス」「クリスタル」といった隠語が使われるようになっている.

その結果, 監視側は継続して新しい隠語を把握し続けなくてはならず, それらを検知する単語として追加していくことが考えられるが, 監視する側の負担は非常に大きい. このようなことから, マイクロブログのうち, 特に Twitter を対象として, 援助交際や違法薬物取引等の犯罪防止に向けたサイバーパトロールを支援するため, 隠語を含む犯罪を

[†1] 現在，電気通信大学 大学院情報理工学研究科
Presently with Graduate School of Informatics and Engineering, The University of Electro-Communications
[a)] hada.takuro@ohsuga.lab.uec.ac.jp

誘導する Tweet の検出を目指す．ここで, 隠語を検出する研究については, これまで, 掲示板などのウェブを対象とした隠語に関する研究は, いくつか発表されている．ただし, 一つの投稿につき文字数が短く限定される Twitter のような文章の意味の把握が難しい短文を対象としたものは, まだまだ数は少ない．そのため, 短文の中から未知の隠語を発見することは, 犯罪の未然防止及び早期検知による犯罪抑止が期待でき, 非常に意義深いと考える．

そこで本稿では, 犯罪を誘導する Tweet に含まれる可能性が高い隠語及び隠語と共に出現する傾向が高い単語 (以下,「関連語」という.) の検出するため, 悪意のあるやり取りに使用される単語の周りには, 類似した関連する単語が出現するとの仮説のもと, 二つのコーパス間の同じ単語の用途の差異に着目する．具体的には, 用意した Twitter データを悪い用途で用いられる Tweet 群 (以下,「Bad コーパス」という.) とそれ以外の Tweet 群 (以下,「Good コーパス」という.) の 2 つの Tweet 群に分類し, 二つのコーパスのそれぞれで Word2Vec [3] を用いて単語分散表現モデルを構築し, 同じ単語におけるコサイン類似度上位に出現する単語の差異から隠語等を検出する方法について提案する．

なお, 本稿の構成は以下のとおりである. 第 2 章では, 本研究の背景について記載する. 第 3 章では, 本研究で提案する手法について述べる．そして, 第 4 章では実施した実験設定および結果を示す. 第 5 章では, 実験を通じて, 提案手法に関する考察について述べる．第 6 章では, 関連研究を紹介する. 最後に, 本研究の結論を第 7 章で述べる.

2. 背景

2.1 薬物と援助交際等の犯罪の増加について

日本だけでなく世界でも, ドラッグと人身売買は問題となっている．国連のレポートを元にしたニュース記事においても, facebook,Twitter,Instagram を介したオンライン麻薬取引の増加について言及されている [4].

一方, 援助交際について, 警察庁の資料によると,SNS に起因した事犯の被害児童数は年々増加しており,2019 年には過去最高の被害数を記録しており（図 2）, また被害児童が多く利用していたサイトは Twitter とのことであった（図 3）.

このようなことから, マイクロブログの中でも特に Twitter に着目することとした.

2.2 隠語について

隠語は, 様々な業界で用いられており, 様々な言葉で表現される. 例えば,Yuan らは「Dark Jargon」と表現しているが [2], 本研究における隠語とは, 警察等の目をかいくぐり, 犯罪等に用いられる単語, 特に, 違法薬物売買や援助交際に関係する隠語について, 対象とする．具体的には, 以下のような隠語を対象とすることとした.

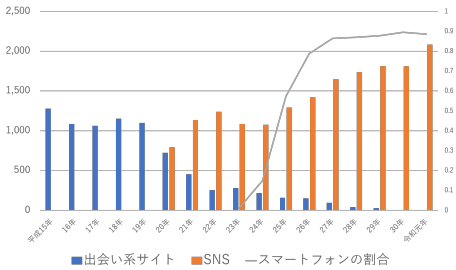

図 2: SNS 等に起因する被害児童数の推移（警察庁のデータ [5], [6] を元に作成． ただし, 出会いサイトについては,2018 年及び 2019 年のデータは無し)

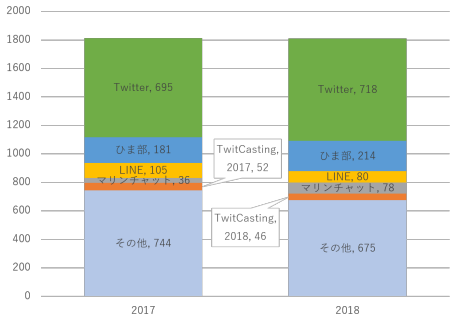

図 3: 被害児童が利用していたサイト (警察庁のデータ [7] を元に作成)

(1) カモフラージュされた隠語
一般的には別の意味をもっている無害な単語にカモフラージュさせて使われる単語がこれに該当する.
例えば, 違法薬物売買関連のジャンルでは, 大麻を表す隠語として,「野菜」や「草」(これは形が似ていることからこのように呼ばれるようになったと思われる) や覚醒剤を表す単語として, 覚醒剤には「アイス」「クリスタル」（これも形から連想される単語が選ばれるようになったと思われる）といった隠語がある．援助交際のジャンルでは,「諭吉」（これは 1 万円札に描かれた肖像画の人物名である「福沢諭吉」から一万円を指す隠語として用いられる.「1 ゆきち, 2 ゆきち」金額の単位として用いられる）や「苺」（これは援助交際の交渉する金額 (15,000 円) を表す）

(2) 音から置き換えられた単語
読む際の音が同じことから, 別の漢字などで置き換えられ作られた単語であり, 一般的には使われない単語となっているものの, 音から連想できる単語が該当する.
例えば, 援助交際を指す言葉として「円光」援助交際を略した「援交」のうち, 同じ音のする感じを用いるものである

(3) 一般的に広まっていない単語
その言葉自体が広く知れ渡っておらず, 単語をそのまま用いたとしても特定の人々にしか分からず, 隠語と

同等の効果が認めあっれる単語が該当する. 例えば, 大麻の種類である「ホワイトクッシュ」,「ホワイトウィドー」などがこれに当たる

隠語の研究は, これまでウェブサイトでは検索する手法が研究されてきたが [8], Twitter などのマイクロブログではそのまま適用できないと思われる. その理由は次の通りである. マイクロブログの特徴として,Dela Rosa and Ellen の述べられた以下の 3 点の特徴がある [9].

(1) 短い文字数
Twitter では, 投稿できる一度の文字数は 140 字までの制限がある.

(2) スラングが多く, ミススペルも多い

(3) 文脈がないことが多い

そのため, 事前に用意した辞書とのマッチング手法では, スラングやミススペルを考慮し, 刻々と変遷す隠語に対して, 常に最新の辞書を更新していくことは運用に多大な労力がかかることから難しいと思われる. 一方で機械学習を用いての隠語の検出する手法については, 文章が短く, 単語が並んだりするだけのものもあるため, 文脈性がないことも多く, 文脈の理解が難しいため, 係り受けを考慮することが難しく, 正しい学習を妨げることが考えられるからである. 一方, 隠語の出現する Tweet 文を確認した結果, 同一 Tweet 内の隠語の周辺に似た用途の単語が記載されていることが多かった.

そのため, 単語分散表現を用いて, 単語をベクトル化し, その周辺に出現する類似語を確認する方法であれば, 隠語が検出できることが期待できると考えた.

このようなことから, 既知の隠語を手掛かりに, その類似する単語に着目し, 未知の隠語の検出を目指す.

3. アプローチ

3.1 アプローチの中心アイデア

犯罪を計画する者が「隠語」を用いて, いかに巧妙に犯罪の意図をカモフラージュしたとしても, 前後のやり取りの文脈性の変化は少ないと考えられる.

また悪意のあるやり取りに使用される単語は, その単語と類似度が高い単語（以下,「類似語」という.）も同じような意味で使われていると仮説を立てた. そのため, 隠語で構成されたコーパスには, 同様の使われ方をする単語が類似語として出現するのではないかと考えた.

そこで, 用意した Twitter データを 1 章でも説明した Bad コーパスと Good コーパスの 2 つの Tweet 群に分類した.

そして, それぞれのコーパスを単語分散表現 [3] を用いて, ベクトル化した. その後,Python のライブラリである gensim [10] を用いてコサイン類似度を求めた. その結果, 例えば, 覚醒剤の一種である「LSD」の隠語である「紙」という単語の類似語を調べたところ, 表 1 のとおりとなった.

表 1 から分かることは, 同じ単語であっても, 両コーパスで全く異なる単語が検出されること, さらには Bad コーパスで構成されたモデルの類似語からは隠語や関連語が多く検出された (表中の太字は隠語と判断した単語) ということである.

表 1: 「紙」における各コーパスの類似単語 (上位 8 位)

Good コーパス		Bad コーパス	
1	字詰め	1	業販
2	試筆	2	市内
3	便箋	3	営業中
4	裏紙	4	メニュー
5	ハードカバー	5	**スカンク**
6	アルシュ	6	**リキッド**
7	用紙	7	**ノーザン**
8	断裁	8	**グミ**
9	模造紙	9	**ハイレギュラー**
10	方眼	10	**ヘイズ**

これより, 二つのコーパス間での同じ単語にも関わらず, 検索される類似語が大きく異なるという点と,Bad コーパスで隠語の類似語を検索した場合, 似たような隠語や関連する悪意のあるやり取りに使われる単語が出現するのではないかという 2 つの点に着目し, 未知の隠語の発見を目指す. なお, 今回隠語リストに用いた単語は, 援助交際や違法薬物取引に関連するものを選定した.

3.2 アプローチの流れ

アプローチの実現方法として, コーパス内にある単語を検索する単語リストとして,2 つのコーパスのそれぞれで類似語を求め, その類似語を既知の隠語リストと照合させ, 結果を比較する方法を提案する（図 4）.

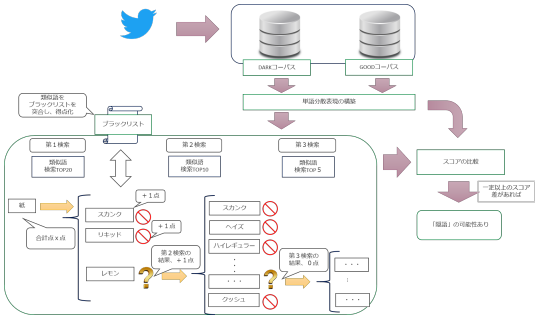

図 4: システム概要図

詳細な手法の流れは以下のとおりである (Algorithm 1 and Algorithm 2).

(1) 単語リストの各単語について, 2 つのコーパスのそれぞれで隠語点を計算する (Function SIMILAR).

(2) それぞれの単語は, 構築した単語分散表現モデル (Good_Corpus, Bad_Corpus) を使ってコサイン類似度上位 N 位までの類似語を検索する (*Get similar words*). なお, 今回の実験では N に 20 を使った.

Algorithm 1 ***Main***

Input: Word_List,N,Good_Corpus,Bad_Corpus
Output: Codewords
  for all $Word$ in $Word\_List$ **do**
    $Cnt\_Bad \Leftarrow SIMILAR(Word, N, Bad\_Copus, 1)$
    $Cnt\_Good \Leftarrow SIMILAR(Word, N, Good\_Copus, 1)$
    $Diff \leftarrow abs(Cnt\_Bad - Cnt\_Good)$
    if $(Cnt\_Bad/N >= 0.2)$ or $((Diff/N >= 0.15)$ and $(Cnt\_Bad/N >= 0.1))$ **then**
      $Codeword\_List.append(WORD)$
    end if
  end for
  $return(Codeword\_List)$

Algorithm 2 Function ***SIMILAR***

Input: Word,N,Corpus,Loop_count
Output: Number of matches with codewords
  $X \Leftarrow 0$
  $Sim\_words \Leftarrow Corpus.Get\_similar\_words(Word, N)$
  for all Sim_word in Sim_words **do**
    if Sim_word in $Codeword\_List$ **then**
      $X \Leftarrow X + 1$
    else if $Loop\_count <= 2$ **then**
      $Y \Leftarrow SIMILAR(Sim\_word, N/2, Corpus, Loop\_count+1)$
      if $Y/N >= 0.2$ **then**
        $X \Leftarrow X + 1$
      end if
    end if
  end for
  $return(X)$

(3) N 個の類似語について, 一つずつ照合リスト (Codeword_List) と照合させる.
(4) もし隠語リストの単語と合致した場合, 加点する (X=X+1). つまり最大で 20 点となる.
(5) 閾値以上であれば当該単語を隠語と判定する. また同じ単語の二つのコーパスのスコアを比較し, 一定以上のスコア差かつ Bad コーパスでの値が一定以上のスコア値であれば, 当該単語を隠語と判定する.
(6) 照合リストと合致しなかった場合, 照合リストにはまだ登録されていない未知の隠語である可能性を考慮し, その単語を元にコサイン類似度上位 $N/2$ 位までの単語を検索し, スコアを求め隠語かどうか判定する.
(7) その類似語についても, 照合リストに合致しなかった単語についても, さらに $N/4$ 個の類似語を検索する.

4. 実験

4.1 実験の概要

事前にアノテーションした単語群 950 語を用いて, 隠語の検出する実験を行った.

具体的には,950 語に含まれる 45 語の既知の隠語のうち,10 語を照合用の隠語リストとする. そして, 単語群の類似語と隠語リストを照合することで, そして, 残りの 35 語の検出を目指す.

以下の工程で, 実験を実施した.

4.2 実験のプロセス

4.2.1 データ収集

TwitterAPI を利用し,Twitter のデータを 47 日間分収集した (5.4GByte). 本文データのみを使用した.

4.2.2 前処理

隠語検出に無関係な単語については, 事前に削除した. 削除した項目は以下のとおりである.

(1) 半角英数字
(2) URL
(3) 全角記号
(4) 改行文字
(5) Twitter に定型でよく現れる単語
「RT」「まとめ」「お気に入り」

4.2.3 コーパス作成

前処理が完了した Twitter データ群を一文ずつ 2 つのジャンル（違法薬物売買, 援助交際）における確実に悪意のある目的で使用されていたと判断した 10 個の単語と照合させ, 以下の二つのコーパスに分類分けを行った.

(1) Bad コーパス (8MByte)
10 個の単語のうち, いずれかの単語が一つ以上出現する Tweet 群であり, 違法な取引に関連する Tweet が集まったと想定した.
(2) Good コーパス (4GByte)
Bad コーパスに該当した以外の全ての Tweet 群. そのほとんどが一般的なやり取りがされていると想定した.

4.2.4 形態素解析

日本語は特有の文章構造を保有しており, スペース等で区切られないため, 単語分散処理を行う前に, 形態素解析処理及び分かち書きが必須である.

今回, マイクロブログの中でも Twitter を対象としたが, その特徴として, 短文であり, 新語やスラングが多く, 意図的に文章を切っているものも多くみられる等の特徴のため, 正しく分かち書きされないおそれもある. また, 今回の検出対象が隠語であるため, 中には造語に近いものもあることが考えられ, 正しく分かち書きがされる必要がある.

これらのことから, 辞書が定期的に更新されており, 新語

にできるだけ対応しているという保守の観点と, 単語の分割単位を選択できるという 2 つの理由から, 形態素解析器として Sudachi [11] を採用した.

4.2.5 単語分散表現処理

形態素解析処理の実施後, Word2Vec を用いて, 単語分散表現処理を実施した.

パラメータは以下のとおり設定した (表 2).

表 2: Word2Vec のパラメータ

パラメータ項目	設定値
size	200
min-count	20
window	5
手法	Skip-Gram [12]

4.2.6 提案システムの実行

単語分散表現処理によって生成したモデルから, 両方のコーパスで共通して出現する単語及び Bad コーパスのみに出現する単語を調べ, それを, 提案システムにより類似語を検索するための単語をしたその結果, 両コーパスに共通して出現する単語は 940 語であり,Bad コーパスのみに出現する単語は 10 語であった.

4.3 Performance Metcric

評価について, 以下の 4 つの指標を用いて, 評価を実施した.

(1) Precision

適合率と呼ばれるもので, 正と予測したデータのうち, 実際に正であるものの割合で求める. 計算式は, 数式 (1) のとおりである.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

(2) Recall

再現率と呼ばれるもので, 実際に正であるもののうち, 正であると予測されたものの割合で求める. 計算式は, 数式 (2) のとおりである.

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

(3) Accuracy

正解率 (精度) と呼ばれ, 正や負と予測したデータのうち, 実際にそうであるものの割合計算式は, 数式 (3) のとおりである.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{3}$$

(4) F Score

Precison と Recall の加重調和平均として定義される. 計算式は, 数式 (4) のとおりである.

$$F_1 = 2\frac{Precision * Recall}{Precision + Recall} \tag{4}$$

4.4 比較手法

提案手法による効果を検証するため, 比較手法を用意した（以下,「ベースライン手法」という）. 本研究における提案手法は, 悪意のあるやり取りに使用される単語の周りには似たような悪意のあるやり取りに使用される単語が現れるとの仮説の元, 類似語に着目している. そこで, ベースライン手法では, 悪意のあるやり取りに使用される単語が出現した tweet のうち, 名詞を全て隠語とした. 提案手法とベースライン手法の関係は図 5 のとおりである.

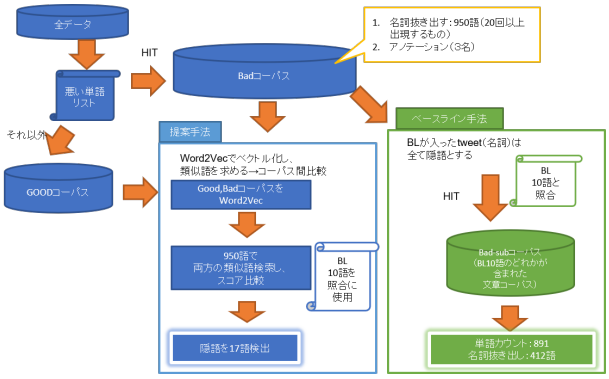

図 5: ベースライン手法と比較手法の関係

ベースライン手法による, 隠語の検出方法は以下のとおりである.

(1) Bad コーパスに対し, 提案手法で用いたものと同じ隠語リストを照合させる

(2) 隠語リストのうち, いずれかの単語が含まれた文章を全て抜き出し,Bad-Sub コーパスを作成する.

(3) Bad-Sub コーパスのうち, 名詞だけを抜き出し, 全て隠語とする.

4.5 評価方法

二つのコーパスから作成された単語のうち, 共通する単語 (1,894 単語) と Bad コーパスのみに出現した単語 (18 単語) を抽出した.

そのうち, 隠語の候補となりうるものとして, 形態素解析器の Sudachi を用いて, 名詞 950 単語を選定した.

それらの単語を, 隠語等の知識のない 3 名が, 対象の単語が出現する Tweet 本文を確認した上で, 以下の 3 種類へ分類した. なお, 評価が分かれたものについては, 多数決で分類した.

(1)「隠語」

隠語として, 本来とは別の意味を指す使い方をしていると判断した単語

(2)「関連語」

その単語自体は隠語とは言えないが, 隠語と一緒に出現する傾向が高く, 一般的な Tweet には表れる頻度は少ないと判断した単語（「在庫」,「値段」等）

(3) 無関係
上の 2 つ以外の単語

4.6 結果

アノテーションの結果, 隠語は 45 語あり, そのうち,10 単語を照合用の隠語リストとして用意した.

その認知済みの隠語リスト 10 単語を除いた 940 単語を検索語として, システムを実行した. その結果, 隠語として,39 語検出され, そのうち 17 語の隠語が含まれていた.

提案手法とベースラインの結果は表 3 のとおりであった.

表 3: 評価結果

	全単語		提案手法		ベースライン	
分類	個数	割合	個数	割合	個数	割合
隠語	35	3.7 %	17	43.6 %	23	6.0 %
その他	905	96.3 %	22	56.4 %	379	94.3 %
合計	940		39		402	

表 3 より, 提案手法は隠語の検出数はベースライン手法に比べ少ないものの, より高い割合で隠語を検出していることがわかった. さらに,Precison,Recall,Accuracy,F score の 4 つの指標を求めた (表 4).

表 4: 結果の詳細

評価方法	提案手法	ベースライン	精度の差
Precision	0.436	0.057	0.379
Recall	0.486	0.657	-0.171
Accuracy	0.957	0.584	0.373
F_score	0.459	0.105	0.354

表 4 より,Precision,Accracy,F Score において, 提案手法はベースライン手法と比べ, より優れた結果を得ることができたことがわかった.

なお, 提案手法において, 検出できた単語の例として,「ディーゼル」「スカンク」「グミ」「レモン」「ジョイント」などがあった.

5. 考察

5.1 検出の課題

今回,Twitter のような短文から隠語を検出することを目的としたところ, 既知の隠語の検出実験によって, 検出能力を確認できたことから, 本手法を用いることによって, 新しい隠語を発見できる可能性が明らかになったといえる. また幅広く隠語とするベースライン手法と比べても,Recall は低くなることは想定はしていたが, 大きく差が開いたわけではなく, それ以上にベースライン手法に比べ高い Precision,Accuracy,F_Score となったことからも, 本手法は隠語に絞って検出できたといえる.

一方で課題も見つかった. 具体的には, 例示した「アイス」, 野菜といった, 典型的な隠語と考えていたものが検出できなかったことである. そこで,「アイス」という単語で Bad コーパスから作成した単語分散表現モデルにおける類似語の確認を行ったところ, 表 5）のとおりとなった.

表 5: 「アイス」の類似語

	検出された隠語	アノテーション結果
1	市内	△
2	郵送	△
3	営業中	△
4	野菜	○
5	極上	△
6	業販	△
7	ブラック	○
8	おはようございます	×
9	メニュー	△
10	テレ	△

表 5 より,「アイス」の類似語として, 悪意のあるやり取りに使用されるた単語やその関連語は多く確認されていたことがわかった. しかしながら, 用意した照合リストの単語数が少なすぎたため, 一致した単語数が少なかったことが原因と考えられる. そのため, 照合リストとして, ある程度の単語数を用意するが Recall を向上させる一つの方法であると考えられる. 一方で,「郵送」や「営業中」といった, 関連語は多く出現していたことがわかった.

そのため, 今後は, 隠語の検出数を増やす方法として, 照合リストの単語数を増やすことを考えているが, それ以外にも, 関連語を照合する仕組みを導入することで,Recall の向上が期待できると考えられる.

5.2 関連語の検出

本研究では, 結果を隠語かそうでないかで評価を行ったが, アノテーションでは, さらに関連語も分類している.

関連語は,「隠語とは言えないが, 隠語との出現頻度が高く, 無関係な単語とは言えない単語」と定義した.

関連語も True に含めた結果は表のとおりであった. Precision に注目すると,0.718 と非常に高く, 検出した単語の中には, 隠語もしくは隠語の関連語を多く検出していることがわかる. そのため, 隠語と関連語を区別し, 検出できる仕組みを今後導入する予定である.

表 6: 関連語も分けた評価結果

	全単語		検出結果		
分類	個数	割合	個数	割合	割合の差
隠語	35	3.7 %	17	43.6 %	+39.9 %
関連語	119	12.7 %	11	28.2 %	+15.5 %
無関係	786	83.6 %	11	28.2 %	-55.4 %
合計	940		39		

表 7: 関連語も True に含めた結果の詳細

評価方法	割合
Precision	0.718
Recall	0.182
Accuracy	0.854
F Score	0.290

5.3 精度向上に向けて

また, 今回の仕組みでは,Bad コーパス内の類似語検出の仕組みにより主に隠語が検出されていた. 現時点では, まだコーパス間の比較を効果的に活用できているとは言えないため,Good コーパスとの比較を効果的に活用する仕組みの構築を検討する. 例えば, 確実に隠語でない単語を選定したホワイトリストを用意することで, より確実に隠語を検出できることが期待できる.

また,Bad コーパスに分類された Twitter の文章を確認していたところ, 明らかに隠語として使用されていた (「バブルガム」「タンジェリンドリーム」等) 単語を目視では発見できた. これらの単語について, 類似語を検索できなかった. なぜなら, 構築したコーパスモデルに単語がなく,「not vocabrary」と表示されたからである. この理由として, 次のことが考えられる. それは, 単語分散表現モデルを構築する際の設定として, 出現頻度が 20 回以下の単語を切り捨てるようにしたため, 出現頻度が少ない単語は, 分散表現モデル生成時に, 無視されたことが原因と考えられる.

そのため, 出現頻度の低い単語について検出できる方法についても, 今後検討していく.

6. Related Work

Twitter を端緒にした犯罪が増加していることはこれまでにも述べてきたが, 一方で,Twitter を対象とした犯罪の軽減を目的とした研究もいろいろなされている [13], [14]. その中でも, 攻撃的な単語や不正な単語を検出する研究についても, いろいろと行われている. [15], [16], [17].

犯罪のやり取りの中でも, 隠語が用いられて取引が行われることもあり, それらの単語は取り締まりを回避するため, 一般的な単語の中に巧妙に隠されているものがある. このような隠語の検出を目指した研究もいくつかなされている. 例えば,Yuan らは, ダークウェブ上では, ポップコーンやブルーベリーの名で大麻がやり取りされていたり, チーズピザという名でチャイルドポルノがやり取りされていることから, ダークウェブから自動的に「隠語」を識別する手法について提唱している. その際, Word2Vec による単一のコーパスでは, 隠語が発見できないとのことから, 複数のコーパスを用意し, そのうちの二つの異なるコーパスに現れる用語の意味的な矛盾から隠語を検出している [2]. ただし, 本研究は, 短文で文脈性のないマイクロブログを対象としていない.

また, 中国語について,Zhao らは, 中国におけるアンダーグラウンドマーケットにおけるサイバー犯罪に使われる隠語に着目し, 教師なし学習を用いて, 隠語の検出を実施している [18]. その際,「CBOW + NS」の組み合わせが Word2Vec で最適な設定であり,LDA アプローチよりも約 20 %高いという結論に至っている. ただし, 前述の筆者らによると, まだファーストステージの研究と評されている [2].

また Aoki らは, 隠語に限定せず単語の中の本来の意味とは異なる意味でやり取りされる単語（例えば, サーバの意味で用いられる「鯖」等）の検出を目指している [19]. ただし, 犯罪に限定しているわけではなく, 犯罪の場合, より巧妙に意図が隠されていることが考えられる.

7. 結論

サイバーパトロールを支援するため, 隠語を検出することを目的として, 今回の提案手法を用いて, 隠語検出実験を実施した. 実験の結果, 用意した照合用の隠語リストにはない隠語を検出することができた. また, 比較用に用意した Baseline モデルと比べても,Precision,Accuracy,FScore で良好な結果を得たことが確認できた.

これらのことから, 本提案手法を用いることで, 刻々と変わっていく隠語を自動的に検出することが期待できるといえる. 一方でまだ Precision,Recall,Accuracy を向上させる余地はあるため, 精度向上に向けて検討を進める. そして, さらなる精度向上及び犯罪誘導 Tweet 検出へ寄与することを目指す.

謝辞 本研究は JSPS 科研費 JP17H04705, JP18H03229, JP18H03340, JP18K19835, JP19H04113, JP19K12107 の助成を受けたものです. 本研究を遂行するにあたり, 研究の機会と議論・研鑽の場を提供して頂き, 御指導頂いた早稲田大学 本位田真一教授, 鄭顕志准教授をはじめ, 活発な議論と貴重な御意見を頂いた研究グループの皆様に感謝致します.

参考文献

[1] Miller, L.: Those Naughty Teenage Girls: Japanese Kogals, Slang, and Media Assessments (2004).
[2] Yuan, K., Lu, H., Liao, X. and Wang, X.: Reading Thieves' Cant: Automatically Identifying and Understanding Dark Jargons from Cybercrime Marketplaces, *27th USENIX Security Symposium (USENIX Security 18)*, Baltimore, MD, USENIX Association, pp. 1027–1041 (2018).
[3] Mikolov, T., Chen, K., Corrado, G. and Dean, J.: Efficient Estimation of Word Representations in Vector Space, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings* (Bengio, Y. and LeCun, Y., eds.) (2013).
[4] : Asia-Pacific drug trade thrives amid the COVID-19 pandemic, `https://www.reuters.com/article/us-asia-crime-drugs/asia-pacific-drug-`

trade-thrives-amid-the-covid-19-pandemic-idUSKBN22R0E0.
[5] : 令和元年における少年非行, 児童虐待及び子供の性被害の状況, https://www.npa.go.jp/safetylife/syonen/hikou_gyakutai_sakusyu/R1.pdf.
[6] : SNS 等に起因する被害児童の現状と対策, https://www8.cao.go.jp/youth/kankyou/internet_torikumi/kentokai/40/pdf/s4.pdf.
[7] : 平成 30 年における SNS に起因する被害児童の現状, https://www8.cao.go.jp/youth/kankyou/internet_torikumi/kentokai/41/pdf/s4-b.pdf.
[8] Lee, W., Lee, S. S., Chung, S. and An, D.: Harmful Contents Classification Using the Harmful Word Filtering and SVM, *Computational Science – ICCS 2007* (Shi, Y., van Albada, G. D., Dongarra, J. and Sloot, P. M. A., eds.), Berlin, Heidelberg, Springer Berlin Heidelberg, pp. 18–25 (2007).
[9] Dela Rosa, K. and Ellen, J.: Text Classification Methodologies Applied to Micro-Text in Military Chat, pp. 710–714 (online), DOI: 10.1109/ICMLA.2009.49 (2009).
[10] Řehůřek, R. and Sojka, P.: Software Framework for Topic Modelling with Large Corpora, *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, Valletta, Malta, ELRA, pp. 45–50 (2010).
[11] Takaoka, K., Hisamoto, S., Kawahara, N., Sakamoto, M., Uchida, Y. and Matsumoto, Y.: Sudachi: a Japanese Tokenizer for Business, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, European Language Resources Association (ELRA) (2018).
[12] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. and Dean, J.: Distributed Representations of Words and Phrases and their Compositionality, *CoRR*, Vol. abs/1310.4546 (2013).
[13] O'Day, D. and Calix, R.: Text message corpus: Applying natural language processing to mobile device forensics, pp. 1–6 (online), DOI: 10.1109/ICMEW.2013.6618380 (2013).
[14] Kansara, C., Gupta, R., Joshi, S. D. and Patil, S.: Crime mitigation at Twitter using Big Data analytics and risk modelling, *2016 International Conference on Recent Advances and Innovations in Engineering (ICRAIE)*, pp. 1–5 (online), DOI: 10.1109/ICRAIE.2016.7939484 (2016).
[15] Xiang, G., Fan, B., Wang, L., Hong, J. and Rose, C.: Detecting offensive tweets via topical feature discovery over a large scale twitter corpus, pp. 1980–1984 (online), DOI: 10.1145/2396761.2398556 (2012).
[16] Wiedemann, G., Ruppert, E., Jindal, R. and Biemann, C.: Transfer Learning from LDA to BiLSTM-CNN for Offensive Language Detection in Twitter, *CoRR*, Vol. abs/1811.02906 (2018).
[17] Hakimi Parizi, A., King, M. and Cook, P.: UNBNLP at SemEval-2019 Task 5 and 6: Using Language Models to Detect Hate Speech and Offensive Language, *Proceedings of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, Association for Computational Linguistics, pp. 514–518 (online), DOI: 10.18653/v1/S19-2092 (2019).
[18] Zhao, K., Zhang, Y., Xing, C., Li, W. and Chen, H.: Chinese underground market jargon analysis based on unsupervised learning, *2016 IEEE Conference on Intelligence and Security Informatics (ISI)*, pp. 97–102 (2016).
[19] Aoki, T., Sasano, R., Takamura, H. and Okumura, M.: Distinguishing Japanese Non-standard Usages from Standard Ones, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark, Association for Computational Linguistics, pp. 2323–2328 (online), DOI: 10.18653/v1/D17-1246 (2017).