電気学会論文誌 C（電子・情報・システム部門誌）
IEEJ Transactions on Electronics, Information and Systems
Vol.142 No.2 pp.177–189　DOI: 10.1541/ieejeiss.142.177

論　文

コーパス間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

非会員　羽田　拓朗*,**a)　非会員　清　雄一*
非会員　田原　康之*　正　員　大須賀昭彦*

Codewords Detection in Microblogs Focusing on Differences in Word Use Between Two Corpora

Takuro Hada*,**a), Non-member, Yuichi Sei*, Non-member,
Yasuyuki Tahara *, Non-member, Akihiko Ohsuga*, Member

（2020年11月25日受付，2021年8月15日再受付）

In recent years, the number of drug trafficking using microblogs has been increasing, which has become a social problem. While cyber patrols have been conducted to crack down on such crimes, those who post crime-inducing messages use terms that camouflage their criminal intentions so-called “codewords” to avoid keywords such as “enjo kosai,” “marijuana,” and “methamphetamine” that may be monitored and attract police attention. These codewords change once they become popular, so it is always necessary to keep track of the latest codewords. Therefore, we propose a new method for detecting the latest codewords. In this paper, we offer a new way of detecting code words from the differences in the words used in posts to detect codewords used in a crime. Specifically, we propose a new method in which we divide words into two corpora, depending on whether a post containing a word has a criminal intention and detect codewords from the differences between similar words of the same word between two corpora. To confirm the effectiveness of the proposed method, we conducted an experiment to detect codewords. The experimental results showed that the proposed method was able to detect codewords with an accuracy of 0.56 persentages points higher than that of the baseline method. The experiment shows that the proposed method can reduce the burden of continuously monitoring code words by rapidly and automatically detecting new codewords that change with time; thus, it provides the possibility of showing clues for crimes.

キーワード：隠語，類似語，単語分散表現，マイクロブログ
Keywords: codewords, similar words, word embedding, microblog

1. はじめに

世界中で，違法薬物売買と売春は問題となっており，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[1]。

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 の運営会社等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，犯罪に直接関係する単語（「援助交際」,「大麻」,「覚醒剤」等）を避け，Fig. 1 のように隠語を用いて，言葉の意味を知っている者同士だけで違法な取引を実施する傾向にある。

隠語は，たとえば，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。これらの隠語を定期的にキーワード検索により検知

a) Correspondence to: Takuro Hada. E-mail: hada.takuro@ohsuga.lab.uec.ac.jp
* 電気通信大学大学院情報理工学研究科
〒 182-8585　東京都調布市調布ヶ丘 1-5-1
Graduate School of Informatics and Engineering, The University of Electro-Communications
1-5-1, Chofugaoka, Chofu, Tokyo 182–8585, Japan
** 警察庁長官官房企画課兼情報通信局情報管理課
〒 100-8974　東京都千代田区霞が関 2 丁目 1-2
National Police Agency
2-1-2, Kasumigaseki, Chiyoda-ku, Tokyo 100–8974, Japan

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，一般的に認知されると監視を回避するために新しい隠語が作られたり，今まで使われていなかった一般的な言葉に隠語の意味が付与されるようになるからである。たとえば，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，それらを検知対象として追加していく必要があるため，負担は非常に大きい。このようなことから，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，隠語を含む犯罪を誘導する投稿の検出を目指す。

そこで本稿では，犯罪を誘導する隠語を検出するため，不正な取引に使用される単語の周りには，類似した関連する単語 (以下,「関連語」と定義する。) 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類似語の差異に着目した。今回 Twitter を対象とし，隠語に関するツイートにおいては，以下の 4 種類に分類できると考えた。

（1） 既知の隠語（及び犯罪に直接関係する単語）だけが利用されたツイート
（2） 未知の隠語だけが利用されたツイート
（3） 既知の隠語（及び犯罪に直接関係する単語）と未知の隠語が混在したツイート
（4） 既知の隠語（及び犯罪に直接関係する単語）も未知の隠語も利用されていないツイート

このうち，本研究では，(3) のツイートが存在することを前提として，既知の隠語（及び犯罪に直接関係する単語）を基に，未知の隠語を検出することを目指す。

具体的には，上記の (1), (3) を対象としたツイート群 (以下,「Bad コーパス」という。) と上記の (4) を対象としたツイート群 (以下,「Good コーパス」という。) の二つのツイート群に分類し，二つのコーパスのそれぞれで Word2vec [2] 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ーパスに間の類似語の差異から隠語を検出する方法について提案する。

なお，本稿における成果は以下の 3 点である。

（1） 二つのコーパス間の同じ単語の類似語の差異に着目し，単語分散表現を用いてモデルを構築し，コサイン類似度上位の単語について再帰的に検索することで，隠語及び未知の隠語（リスト外の隠語）検出の仕組みを考案。
（2） Twitter のような短文に取引ならではの特徴として，違法な取引に関連した隠語の周辺には隠語に関連した単語が出現しやすく，それらを「関連語」として定義。
（3） 実データを用いて隠語検出を実施し，事前に用意したリスト外の隠語（未知の隠語）を実際に 19 個検出。

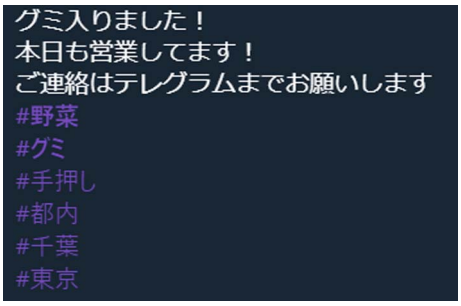


Fig. 1. Example sentences with codewords from Twitter.

なお，本論文の一部は，3rd IEEE International Conference on Computing, Electronics Communications Engineering(IEEE iCCECE '20) [3] 及び情報処理学会第 200 回知能システム研究会 [4] で発表したものである。

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，関連研究を紹介する。そして，第 4 章では本研究で提案する手法について述べ，第 5 章では実施した実験設定および結果を示し，第 6 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。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　現在，日本においても 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おり，社会的な問題となっている。たとえば，Fig. 2 は，年代別の大麻事犯における検挙人数の推移を表すものであり，特に 10 代と 20 代において年々増加傾向であることが見て取れる。

一方，援助交際についても，海外でも「enjo kosai」というローマ字で表記されるほど一般的であり，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 [6]。中でも 18 歳未満による援助交際が問題となっている。そして援助交際をきっかけに犯罪に巻き込まれるなど，SNS に起因した事犯の被害児童数は年々増加し，特に 2019 年に

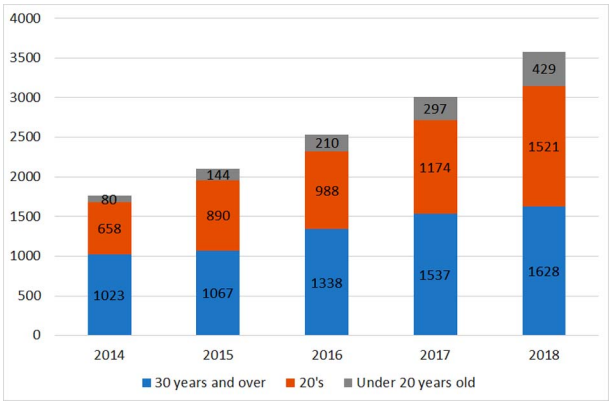


Fig. 2. Changes in the number of marijuana vending arrests by ages (according to data from the National Police Agency (Japan)) [5]

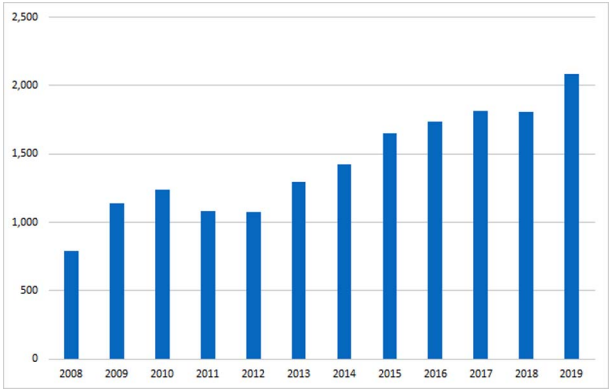


Fig. 3. Number of children victimized by SNSs according to data from the National Police Agency (JAPAN) [8][9]

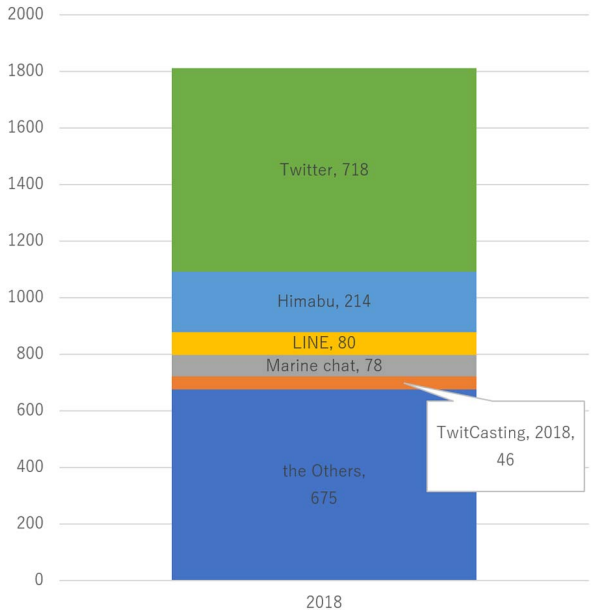


Fig. 4. Sites with multiple child victims according to data from the National Police Agency (Japan) [10]

は過去最高の被害数を記録していた（Fig. 3）。その中でも，被害児童が最も多く利用していたサイトは Twitter であり，約 4 割を占めていた（Fig. 4）。

このような状況を踏まえると共に，SNS の中では Facebook や Twitter が主に利用されていることと [7]，その中でも Twitter では不特定多数が閲覧できるため，違法な取引が行われやすい環境にあること，その際，隠語でやり取りされる傾向にあることなどから，本研究では特に Twitter に着目することとした。

〈2･2〉 隠語について 隠語は，特定の社会・集団内でだけ通用する特殊な語と定義されており †，たとえば，警察では犯人のことを「ホシ」といったり，すし屋ではすし飯を「シャリ」，お茶を「上がり」といったりするなど，我々の身の回りでも様々な場面で使用されている。本研究における隠語の対象は，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とした。具体的には，以下の分類の元，対象とする隠語を選定した。

（1） 犯罪行為となる対象名そのもの

（a） 認知度高

たとえば，「大麻」，「LSD」，「援助交際」などが該当する。これらは，一般的に認知されており，取引を隠れて行わせる効果のない単語であるため，隠語とは分類せず，後の〈4･3･1〉節で述べる関連語として分類した。

（b） 認知度低

対象の単語自体が一般的に認知されていないため，そのまま用いたとしても特定の人々にしか分からないことから，特段別の単語に置き換えなくとも隠語と同等の効果が認められる単語が該当する。たとえば，大麻の種類である「ホワイトクッシュ」，「ホワイトウィドー」，「ゴリラグルー」などがこれに当たる。

（2） 犯罪行為となる対象名そのものではない

（a） 転用 (カモフラージュ)

一般的に使用されている単語に隠語の意味を付与し，カモフラージュさせて使われる単語がこれに該当する。たとえば，違法薬物売買関連の隠語としては，大麻を表す隠語として，「野菜」や「草」，一方，覚醒剤を表す単語としては，「アイス」や「クリスタル」といった隠語がある。一方，援助交際関連の隠語としては，一万円の肖像画の人物名である「福沢諭吉」から一万円の単位を指す隠語として「諭吉」などが用いられている。

（b） 造語

違法な取引を行うため，意図的に造られた単語が該当する。たとえば，違法薬物売買関連の隠語としては，大麻の場合は「ハシシ」，「ポット」，覚醒剤には「シャブ」，「ガンコロ」といった隠語がある。一方，援助交際関連の隠語としては，「神待ち」といった隠語がある。中には，知らない言葉であったとしても，音が一緒であったり，漢字などから連想できるようなものもある。たとえば，援助交際関連の隠語として用いられる，「円光」や「えん」などがこれに当たる。

〈2･3〉 隠語の変遷について 言葉は時間と共に意味するものが少しずつ変化していく [11][12] が，中でも犯罪の用途で用いられる隠語については隠語の意味が一般的に認知されると変化する [13][14]。

また近年は SNS の発展に伴い，新しい隠語が次々と生まれるようになったと思われたことから，実際に 2016 年と 2020 年のツイートデータを用意し，隠語の出現の割合について調査を行った。具体的には，2016 年 11 月から 2017 年 2 月と 2019 年 11 月から 2020 年 2 月までのそれぞれ同じ期間に TwitterAPI により収集したツイートデータのうち，以

† デジタル大辞泉 (小学館) より引用

Table 1. The difference in the number of tweets in which each word appeared as codewords between the two years (2016, 2020)

Word	2016	2020
	Quantity	Quantity
神待ち	204	845
パパ活	0	10,100
クッシュ	1	157
ウィドー	0	70
野菜	0	894
手押し	0	1,163
Total number of tweets	111,408,818	69,301,877

Table 2. The difference in the percentage of tweets in which each word was used as codewords between the two years (2016, 2020)

	野菜		手押し	
	2016	2020	2016	2020
Number of detections	37,931	35,490	290	1,472
Number used as codewords	0	894	0	1,163
Percentage of codewords	0%	2.5%	0%	81.8%

下の単語の出現割合について調査を行った。その結果，以下のとおりであった (Table 1)。

Table 1 より，大麻を表す「ウィドー」,「クッシュ」，そして「パパ活」といった単語について，収集したツイートの範囲ではあるものの，2016 年のツイートデータからは隠語としてほぼ検出されなかった単語が 2020 年のツイートデータからは出現していることが見て取れる。

また野菜，手押しなど一般的に使われる単語について，2016 年と 2020 年のそれぞれにおいてどの程度隠語として用いられているのか調査を行った (Table 2)。

Table 2 より，収集したツイートの範囲において，手押しも野菜も 2016 年時点では隠語としての意味を付与されて用いられた割合は 0%であったのに対し，2020 年には，手押しの場合は 81.8%，野菜については一般的な意味で出現する数が多いため，2.5%ではあるが，それまで全く隠語として用いられていなかった単語が，2020 年には隠語としての意味が付与されて使われるようになっていることが分かる。

このように，Table 1，2 から，時間の経過とともに新しい隠語が使われるようになったり，一般的な単語の中に隠語の意味を付与させるような動きがあることが示唆される。

〈2･4〉 隠語の研究について 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[14]~[16]，Twitter などのマイクロブログではそのまま適用することは難しいと思われる。

その理由は次のとおりである。まず，マイクロブログの特徴として，以下の特徴が述べられている[17]。

（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） 文法が非公式で構造化されていない

会話調で編集もされていないため，スラングや略語や誤字も多い。

Twitter における隠語に関連するツイートの分析を行ったところ，短文が多く出現するだけでなく，中でも犯罪取引に用いられるツイートは，犯罪の意図を隠そうとするため，さらに文章の体をなしてないことが多く見受けられた。そのため，文の係り受けなどを用いた分析や機械学習などについては難しいと考えられる。一方で，取引をできる限り短いやり取りで成立させるため，一つの投稿の中に取引対象や場所・金額・品質等の必要な情報について書き込む必要があるため，本研究では関連語と定義した隠語周辺に犯罪に関連した単語が出現しやすい傾向を発見した。そこで，取引に関連したツイートについては，隠語の周辺に犯罪に関連した単語が出現する傾向を利用し，単語分散表現を用いることで，効果的に犯罪取引に用いられる隠語や関連語の類似語を見つけることができると考えた。

なお，単語分散表現として本研究では Word2vec を用いたが，Word2vec とコサイン類似度を用いた研究については，近年でもいくつか報告されているが，未知の隠語の検出に使った事例はない[18][19]。

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. 関連研究

これまでに掲示板などのウェブサイトを対象とした隠語等の検出等に関する研究は，いくつか報告されている[15][16]。たとえば，専門用語の検出として北村らの研究があげられる[20]。北村らの研究は，文対応の付いた対訳コーパスから共起する単語列を対応付けることにより，対訳表現を自動的に抽出する方法を提案するというものである。そして，特定分野特有の専門用語等の翻訳について，高精度で適切な対訳表現を抽出したと報告されている。しかしながら，北村らの手法は 2 言語間の専門用語の対応付けを行うものであるが，これを隠語に応用すると，隠語を使わずに表現された文と，まったく同じ意味を持つが隠語を使って表現された文のペアを多数用意する必要がある。たとえば,「都内 野菜　手押し」という文が「都内で大麻を手渡しで販売する」という文と同じ意味を持つというラベル付けを事前に行っておく必要がある。このようなペアが多数存在するとき,「野菜」という単語が「大麻」の隠語であると判断することが可能となる。北村らの元々のターゲットである翻訳のドメインでは，同じ意味を持つ日本語の文と英語の文のペアを多数取得することが可能である（たとえば，日本語と英語で書かれた特許文章）。しかし，隠語を含む文と含まない文で同じ意味の文のペアのデータベースは著者らの知る限り存在しない。したがって北村らの手法を未知の隠語発見に利用することはできない。さらに，文章単位で対応付け（対訳と単語組）が必要なことについて，実運用を考えると， データベースの規模の拡大に対応できず，運用担当者の負荷が非常に高いため，継続的な運用は難しいと思われる。一方で我々の手法は，既知の隠語（リスト内）か

ら未知の隠語（リスト外）を検出することがポイントであり，さらに再帰的手法を用いることで精度を上げて検出している。また，対応付けの文章も不要であるため，実運用を想定した場合でも有効であると言える。

隠語の検出として，橋本らの研究があげられが，本研究は，文章中の単語の語彙の決定及び係り受け関係にある2文節間の深層格の決定を行う意味解析システムの開発の中で，特に隠語の有害語意と文脈に登場する他の語（周辺語）の語意との共起頻度を辞書化しこれを元に隠語の語意を決定することで，有害語意を検出するというものである[21]。橋本らの研究は，隠語を事前に把握している前提で，ある単語が隠語として使われているか普通の意味で使われているかを判定するものであり，未知の隠語を発見するものではない。一方で我々の手法は未知の隠語を検出することを目的としている。また橋本らの研究は係り受けができる前提であり，Twitterなどの短文では係り受けがない場合が多く，そのままでの対応は難しいと考えられる。そのほかとして，大西らは，アンダーグラウンド系掲示板において，投稿された単語及びその周辺語に着目し，隠語検出を試みている[14]。ただし，これらの研究は係り受けができる前提であり，一つの投稿につき文字数が短く限定されるTwitterなどの短文では係り受けがない場合が多く，そのままでの対応は難しいと考えられる。また，Webサイトや掲示板以外を対象とした隠語検出の研究についても，いくつか報告されている。たとえば，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[13]。その際，Word2vec[2]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，北村らの研究[20]と同様に，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。ただし，前述の研究は，ダークウェブ上の隠語が対象であるが，一般の若年層が多く使用している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[22]。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[13]。

一方，日本語を対象としたものとしては，安西らのID掲示板を対象としたものが報告されている[23]。

安西らは，短文で文脈性のないID掲示板を対象に，テキスト分類（教師あり学習）を用いて有害性を分類している。ただし，本報告の中では，F値などで有意差が確認できなかったことと,「野菜」,「アイス」などダブルミーニングなものへの対応が難しいことと，そしてID掲示板については，違法な行為を目的とした投稿が多い一方，Twitterについては，Table 2でも言及したとおり，一般的な投稿の方が，たとえば，野菜については97.5%と圧倒的に多いため，そういった点でID掲示板とは大きく異なる。

本研究で対象としたTwitterについては，犯罪の軽減を目的とした研究がなされている[24][25]。またその中でも，攻撃的な単語や不正な単語を検出する研究についても，行われている[26]~[28]。

また，住田らは，不正なツイートを対象とし，機械学習を用いて有害性を分類している[29]。この研究では，ツイート全体が有害かどうかを判断しているが，隠語自体を別途理解する必要があると考える。なぜなら，機械学習させるにあたり，学習データをアノテーションする必要があり，その際，隠語の知識がなくては正しくアノテーションできないおそれがあるからである。そこで新しい隠語の情報が増えることは，アノテーションの精度が上がり，その結果，有害性の判断の精度も上がることが期待できる。

また青木らの研究についても，同様に相乗効果が期待できると考えられる。

青木らの研究は，橋本ら[21]と同様，周辺語に着目しており，ある単語が一般的ではない使われ方がされていた場合，その周辺単語は一般的な用法として使われた場合のものと異なるという仮説に基づいて，着目単語とその周辺単語の単語ベクトルを利用し，注目している単語の周辺単語が均衡コーパスにおける一般的な用法の場合の周辺単語とどの程度異なっているかを評価することにより，一般的ではない用法の検出を行う手法である[30]。青木らの手法では，前提として，隠語を事前に認知している必要があり，その隠語が出現する文章を用意する必要がある。しかしながら，隠語は監視の目をかいくぐり，特定の人しか分からないようにする特徴があることから，新しい隠語を把握し続けることは非常に労力を要する。つまり，我々の手法との最大の違いは，青木らの研究が未知の隠語を新たに見つけるものではない一方で，我々の手法は，ある単語の類似語から似たような使われ方をする単語を検出することであり，それは我々ですら認識していない未知の隠語を発見できるところにある。また青木らの手法については，我々の手法で新たな隠語を発見し，それが隠語であると認知でき，その隠語を用いた文章を集めることができたあとで，一般的な使われ方をしているか，そうでないか効果を発揮できる補完的な関係である。つまり,「野菜」が「大麻」の隠語と検出できるのが，我々の手法であり,「野菜」という単語が文章内に出現した際に，隠語としての使われ方をしているのか，一般的な意味で使われているのか判断するのが青木らの手法との認識であり，青木らの手法を効果的に使うためには，我々の手法が必須と考えている。

このようなことから，Twitterを対象とした短文の中から未知の隠語を検出することは，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。

4. アプローチ

〈4·1〉 アプローチの中心アイデア 隠語の周りには特徴的な単語が多く現れることから[21]，不正な取引に使用される単語は，その類似語も同じような意味で使われていると考えた。しかしながら，類似語を元に検出しようとしても，隠語には，前述した隠語例からもわかるとおり,「野菜」,「アイス」など日常的に使われる単語でカモフラージュされているものがあり，こういった単語は，類似語を探そうとしても，一般的な野菜に関連したツイートにまぎれてしまい，隠語を検出することができない。

そこで，用意したツイートデータを 1 章で定義した Bad コーパスと Good コーパスについて，以下の方法で二つのツイート群に分類した。

（1） Bad コーパス

不正な取引目的で使用されていた単語が含まれるツイート群と定義し分類した。今回は，不正な取引目的で使用された単語群（以下,「コーパス分けリスト」という。）のうち，いずれかの単語が一つ以上出現するツイート群とした。なお，コーパス分け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。

（2） Good コーパス

一般的な単語のツイート群と定義し，分類した。Good コーパスについては，一章において既知の隠語も未知の隠語も利用されていないツイートと説明したところ，その作成方法として，次の二つの方法を考えた。

（a） 良いツイートのみを抽出する方法

i. 作成方法

全体のツイートから Bad コーパスへ分類したもの以外の全ツイートを Good コーパスに分類する。

ii. 利点

単語分散表現モデルやパラメータを自分で自由に設定可能であり，簡単に，そして柔軟に作成できる。

iii. 欠点

Bad コーパスで悪い意図のツイートを網羅できていない場合，Good コーパス内に悪い意図のツイートが紛れ込むおそれがあり，影響が少ないかもしれないが，Good コーパスの純度が下がるおそれがある。コーパス規模を広げた場合，単語分散表現による処理に膨大な時間を要する。実際に 3 か月規模のコーパス（約 10 Gbyte）の作成を試み，一般的なデスクトップ端末の性能で 10 日以上の処理を要した。

（b） 大規模なツイートコーパスを利用する方法

i. 作成方法

Twitter 上では不正な取引に関するツイートの数は全体から見ると無視できる数であると考え，既に作成されている SNS における一般的なコーパスモデルを使用する[31]。なお，当該コーパスを用いての類似語検出実験を行ったところ，検出結果は Table 3 のとおりであり，これまでの Good コーパスと大きな差はなく，一般的な単語のみ出現していることを確認した。

ii. 利点

大規模コーパスを簡単に利用することができる。

iii. 欠点

作成しているモデルに依存する。単語分散表現が用意されていなかったり，パラメータが設定が自由に変更できないおそれがある。また，提供元が少ない。

Table 3. Similar words by Hotlink's large-scale SNS corpus.

順位	手押し		アイス		氷		野菜		冷たい	
1	手押	0.8652	アイスバー	0.8193	氷雪	0.7952	キャベツ	0.8972	冷たく	0.8406
2	手押し車	0.8407	ロックアイス	0.8155	氷ら	0.7810	果物	0.8969	冷たき	0.8079
3	手押しポンプ	0.7695	アイスター	0.7914	凍っ	0.7754	豆類	0.8669	温かい	0.8052
4	軌道自転車	0.7355	アイスタ	0.7854	氷塊	0.7676	トマト	0.8666	暖かい	0.7928
5	トロッコ	0.7268	アイスティー	0.7723	凍ら	0.7594	キュウリ	0.8659	冷た	0.7820
6	起重機	0.7263	アイススター	0.7690	海氷	0.7582	ニンジン	0.8628	熱い	0.7677
7	車輪	0.7166	アイスノン	0.7591	氷筍	0.7564	ジャガイモ	0.8618	生暖かい	0.7675
8	荷車	0.7046	アイスバイン	0.7574	浮氷	0.7550	軟弱野菜	0.8537	冷やさ	0.7464
9	人力	0.7001	レジアイス	0.7572	氷山	0.7537	根菜	0.8492	冷や	0.7371
10	馬車	0.6972	アイススケートリンク	0.7560	融け	0.7455	タマネギ	0.8480	吹き出す	0.7301

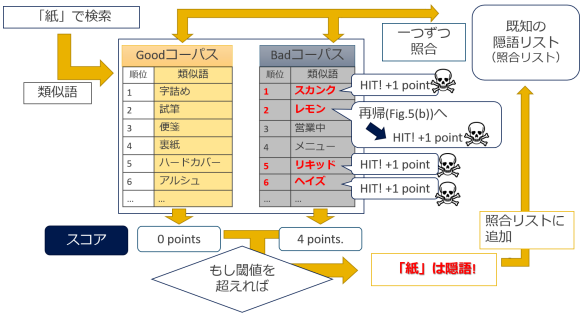
Table 4. Top 10 similar words to "paper" in each corpus. (Bolded words are those defined as codewords.)

	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**

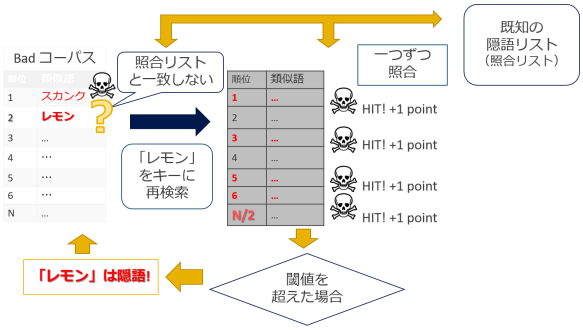
そして，その後は，コーパスに対し，Word2vec[2] を用いて，単語分散表現を獲得し，Python のライブラリである gensim[32] を用いてコサイン類似度を求めた。たとえば，覚醒剤の一種である「LSD」の隠語である「紙」という単語のそれぞれのコーパスにおける類似語を調べたところ，Table 4 のとおりとなった。ここにおける Good コーパスの作成方法は,「良いツイートのみを抽出する方法」とした。

Table 4 から，同じ単語であっても二つの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が多く検出された (Table 4 中の太字は隠語と判断した単語) ということがわかった。

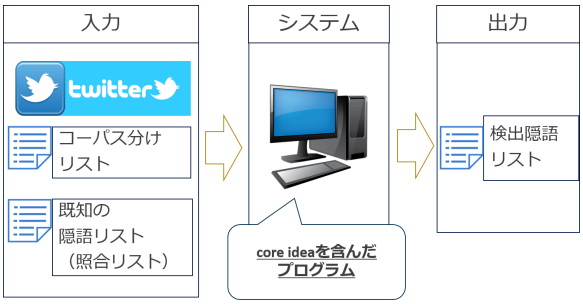
これより，二つのコーパス間で同じ単語にも関わらず，検索される類似語が大きく異なるという点と，Bad コーパスで隠語の類似語を検索した場合，似たような隠語や関連する不正な取引に使われる単語が出現するのではないかという二つの点に着目し，未知の隠語の発見を目指す。



(a) The Core idea of the system



(b) The Core idea of the system (if the word does not match any in the codeword list.)



(c) Input/output of the system

Fig. 5. Schematic of the system.

〈4·2〉 アプローチの流れ コアアイデアを内包したシステムを実装し，入力はコーパス内にある単語を入力する単語群（以下,「入力単語リスト」という。），ツイートデータ，事前に用意した照合用の既知の隠語の単語群（以下,「照合リスト」という。）とし，二つのコーパスのそれぞれで類似語を求め，その類似語を照合リストと照合させ，結果を比較し，隠語の一覧（以下,「検出隠語リスト」という。）を出力する方法を提案する（Fig. 5）。

詳細な手法の流れは以下のとおりである (Algorithms 1, 2)。

（1） 入力単語リストの各単語について，二つのコーパスのそれぞれで照合リストと Hit した類似語数（スコア）を計算する (Function SIMILAR)（Fig. 5 (a)）。

（a） 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。

（b） 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。

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)

（c） もし照合リストの単語と合致した場合，スコアを加点する (X=X+1)。つまり今回の実験では最大で 20 点となる。

（d） 照合リストのどの単語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（Fig. 5 (b)）。

（e） その類似語のうち，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。

（2） 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 コーパス単体で一定以上のスコアであれば，当該単語を隠語と判定する。なお，閾値については，フレキシブルに変更可能となっているところ，本実験では検出精度が一番良い値を閾値として決定した。

〈4·3〉 精度向上方策及び汎用性についての検討 コアアイデアに加え，さらに隠語検出の精度を向上させるため，いくつかの機能を検討及び検証し，追加した。

〈4·3·1〉 関連語の検出 〈2·4〉でも述べたところ，隠語を確認する中で，たとえば，一万円の単位を表す「諭吉」や郵送ではなく手渡しを意味する「手押し」などが確認さ

れたところ，一般的に取引には,「取引対象」,「取引対象を形容するもの（高品質等）」,「時間」,「場所」,「取引方法」,「取引量」,「金額」などの情報が必要となる。これらの情報を表すための語として，取引を行う者たちの間で共通認識が生じれば隠語が発生し得るが，まだ隠語として確立している単語がない状況では，お互いに誤解が生じないよう一般的な言葉が利用される。そのため，隠語を用いて巧妙に意図を悟られずに取引しようとしても,「取引対象」,「場所（都内等）」,「金額」という少なくとも三つの情報が含まれている必要があると考えられる。さらには監視の目をかいくぐりながら迅速なやり取りを実現させるためには,「取引方法（手押し，郵送等）」や「取引対象を形容するもの（高品質等）」なども文章内に含まれている必要があると思われる。このような，その単語自体だけでは隠語として成立しないが，隠語と一緒に出現する傾向が高い単語を関連語と定義した。

隠語について，たとえば，覚せい剤の隠語である「アイス」について，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，Table 5 のとおりとなった。ここで，単語を以下の区分で分類した。

（1）「隠語」
〈2･2〉節で定義した単語

（2）「関連語」
上でも述べた，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い単語

（3）「無関係」
上の二つ以外の一般的な意味で使用される単語

Table 5 より,「アイス」の類隠語の中には,「隠語」は上位 10 位までのうち，2 個と少ないものの，関連語である「郵送」や「営業中」といった単語が上位 10 位までのうち，7 個確認された。

そのため，関連語を照合する仕組みを導入することで，Recall（再現率）を向上させることが期待できる。

〈4･3･2〉 品詞分類によるフィルタ　コアアイデアでは，隠語と判定した単語は，スノーボール方式で自動的に照合リストに追記していく仕組みとなっているところ，関連語と隠語と区別して自動的に追加するためには，検出した単語を隠語か関連語に分類する必要がある。

そこで，隠語と関連語を分類するための特徴を把握するため，形態素解析器である Sudachi [33] を用いて，ツイートデータのうち 191,079 語を対象に，品詞分類を行い，分析した。

単語の属性について分析した結果，隠語として識別した単語が主に名詞句であったことから，名詞に絞ることとし，さらに分類を行った。分類方法として，網羅的に単語を確認した結果，名詞の中で原則として全て隠語として照合リストに追記することとすることとした。ただし，項目によっては隠語対象からの除外もしくは隠語ではなく関連語への分類とした。各品詞の分類及び考察については Table 6 の

Table 5.　Top 10 similar words to “ice” in Bad corpus.

Rank	Result	Annotation
1	市内	関連語
2	郵送	関連語
3	営業中	関連語
4	野菜	隠語
5	極上	関連語
6	業販	関連語
7	ブラック	隠語
8	おはようございます	無関係
9	メニュー	関連語
10	テレ	関連語

Table 6.　Classification by part-of-speech classification

Categories	Subcategories	Groups	Example	Classification	Reason
名詞	固有名詞	地名	カルフォルニア，名古屋	関連語	犯罪行為に関わる取引きでは，お互いを信用することが難しいため，対面での取引が行われることが多い。また居住地から遠い場所での取引はコストと時間がかかるため，事前に場所の指定がなされることが多い。地名は場所の指定に用いられることになるため，もしこれが隠語として別の意味を持たせる場合，その別の意味のものを指すのか，場所の指定を指すのか混乱が生じる。したがって隠語にはなりづらい一方，隠語とともに出現することは多いと考えられる。
		一般	バイアグラ	関連語	
		人名	ホフマン，ジャック	隠語	
	普通名詞	一般	氷，クリスタル，葉っぱ	隠語	
		サ変可能	絶賛，宅配，味見	関連語	本分類の単語は取引対象自体を指すものではなく，取引対象の質や取引に関する情報を指すものが多い傾向にあった。そして，取引対象を形容するために使われる単語が取引対象を指す意味を持ってしまうと，取引対象を形容するために使われているのか，取引対象自体のどちらを指すのか混乱するため，取引対象を指す隠語にはなりづらい。
		サ変形状詞可能	安心，満足，直接	関連語	
		形状詞可能	安全，好評	関連語	
		助数詞可能	キロ，ドル，袋	無関係	これらの単語は取引においても使われるので，幅広にやり取りを行う Twitter では一意に特定の取引対象を比喩することは難しことから，隠語にも関連語にもなりづらい。
		副詞可能	ただいま，朝方，来週	無関係	
	数詞		二，十	無関係	

とおりである。

そして，Table 6 を元に，隠語と関連語を分類し，それぞれを自動的に追加登録可能にした。これにより，Precision を向上させることが期待できる。と考えた。

5. 実　　験

〈5·1〉 実験の概要　　入力単語リストとして，出現頻度が 20 回以上の事前にアノテーション済の単語群 1,892 語を用いて，隠語検出の実験を行った。なお，本単語群は，羽田ら [3] における実験で使用した入力単語リストを引用した。

具体的には，1,892 語に含まれる 45 語の既知の隠語のうち，10 語を照合リストとする。そして，入力単語リストの類似語と照合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。実験の工程については，次節で記載する。なお，前処理の内容は〈5·2·2〉節に記載している。

〈5·2〉 実験のプロセス　　実験において Fig. 6 の流れで処理を実施した。

〈5·2·1〉 データ収集　　TwitterAPI を利用し，ツイートデータを 47 日間分収集した（5.4 GByte）し，本文データのみを使用した。

〈5·2·2〉 前処理の実施　　隠語検出に無関係な単語については，事前に削除した。本実験において削除した項目は，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」

〈5·2·3〉 コーパス作成　　用意したコーパスについては，以下の二つである。

（1） Bad コーパス（8 MByte）
本実験では，前処理が完了したツイートデータ群を一文ずつ二つのジャンル（違法薬物売買，援助交際）における不正な取引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，Bad コーパスを作成した。

（2） Good コーパス
〈4·1〉節で説明した定義について，大規模なツイートコーパスを利用する方法を選択し，株式会社ホットリンクの作成した日本語大規模 SNS+Web コーパスを利用した [31]。

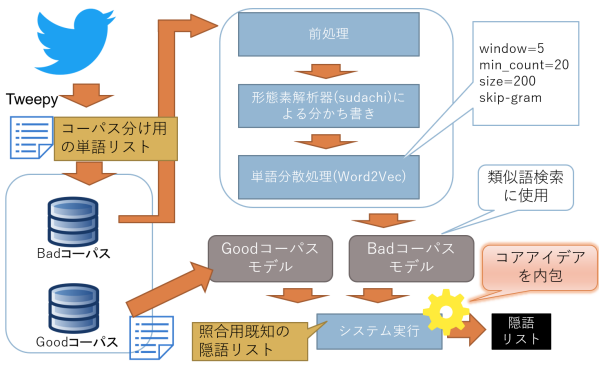


Fig. 6.　Experimental Process

〈5·2·4〉 形態素解析　　日本語は特有の文章構造を保有しており，英語等と異な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。分かち書きとは，事前に単語の辞書を内部で持ち，それに従い，文章を適切に単語単位に分割するものである。これによって，単語単位で文章を分けることができる。ここで問題となるのが，いかに適切に文章を分けることができるかということである。なぜなら，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からである。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，以下の二つの理由から形態素解析器として Sudachi [33] を採用した。

（1） 内部辞書が定期的に更新されており，新語にできる限り対応している保守の観点
（2） 新語を想定し，単語の分割単位を選択できるという観点

〈5·2·5〉 単語分散表現処理　　形態素解析処理の実施後，Word2vec [2] を用いて，単語分散表現処理を実施した。Word2vec のパラメータは Table 7 のとおり設定した。

〈5·2·6〉 入力単語リストの作成　　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（1,872 単語）と Bad コーパスのみに出現した単語（10 単語）を抽出した。

それらの単語を，隠語に関する知識を有していない 3 名により，対象の単語が出現するツイート本文を確認した上で，〈4·3·1〉節で説明した 3 種類へ分類した。

そして，作成した入力単語リストを元に実装したシステムを実行した。

〈5·3〉 評価指標　　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。なお，式中には，真陽性 (TP)，偽陽性 (FP)，偽陰性 (FN)，真陰性 (TN) で表す。

（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。

$$Precision = \frac{TP}{TP + FP} \cdots\cdots\cdots\cdots\cdots\cdots (1)$$

（2） Recall

Table 7.　Parameters of Word2vec

Parameters	Value
size	200
min-count	20
window	5
Skip-Gram or CBow	Skip-Gram [34]

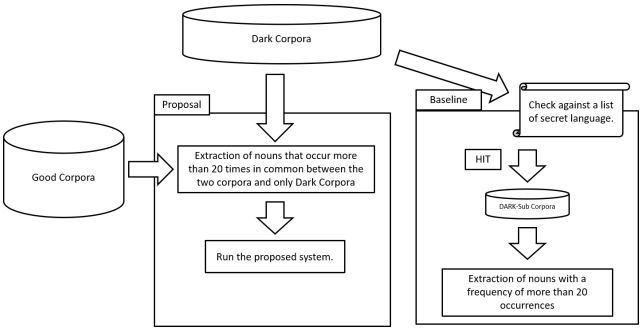


Fig. 7. Relationship between the proposed and baseline methods

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。

$$Recall = \frac{TP}{TP + FN} \cdots\cdots\cdots\cdots\cdots\cdots (2)$$

（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。

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \cdots\cdots (3)$$

（4） F-measure

F 値と呼ばれ，Precison（適合率）と Recall（再現率）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。

$$F - measure = 2\frac{Precision * Recall}{Precision + Recall} \cdots\cdots (4)$$

〈5·4〉 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，不正な取引に使用される単語の周りには似たような目的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，不正な取引に使用される単語が出現したツイートのうち，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 Fig. 7 のとおりである。

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照合リストを照合させる。

（2） 照合リストのうち，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。

〈5·5〉 実験結果 アノテーションの結果，隠語は 45 語あり，そのうち，10 単語を照合リストとして用意した。

その照合リストとして用いた 10 単語を除いた 1,882 語を入力単語リストとして，システムを実行した。その結果，隠語として，31 語検出され，そのうち 19 語の隠語が含まれていた。

Table 8. Evaluation results

	All Words		Proposal method		Baseline method	
Clasified	Quantity	Rate	Quantity	Rate	Quantity	Rate
Codewords	35	1.9%	19	61.3%	23	6.0%
Others	1,847	98.1%	12	38.7%	379	94.3%
Total	1,882		31		402	

Table 9. Details of the results

Evaluation Method	Proposal method	Baseline method
Precision	0.613	0.057
Recall	0.543	0.657
Accuracy	0.970	0.584
F-measure	0.576	0.105

提案手法とベースライン手法の結果は Table 8 のとおりであった。

Table 8 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，若干少なくなったものの，隠語の検出率としては，大きな差をつけることができた。さらに，Precision（適合率），Recall（再現率），Accuracy（正解率），F-measure（F 値）の 4 つの指標を求めた (Table 9)。

Table 9 より，Precision（適合率)，Accuracy（正解率)，F-measure（F 値）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，検出できた隠語の例としては「ディーゼル」,「ジョイント」などがあった。また，通常は一般的に用いられる「レモン」,「アイス」,「スカンク」,「グミ」等の隠語も検出できていた。

6. 考　察

〈6·1〉 精度向上に向けて 今回，本提案手法を用いることにより，隠語検出についてベースライン手法に比べ，適合率において 0.556 ポイント高い結果を出すことができた。一方で，出現頻度が 20 回以上を対象にアノテーションしていたため，20 回未満の隠語（たとえば,「バブルガム」等）を検出できていなかった。

この理由として，単語分散表現を獲得する際の Word2vec [2] のパラメータのうち，n 回未満登場する単語を破棄する「min_count」オプションの値を 20 回としていたため，これらの単語は出現頻度が設定値を下回っていたことから，分散表現モデル生成時に破棄されたことが原因と考えられる。

今回の実験では，ノイズを減らすため，設定値を 20 で分散表現モデルを生成したが，本研究では隠語と対象としており，隠語の出現割合は Table 2 でも言及したとおり，たとえば，野菜については 2.5%と低かったことから，元々の出現数は少ないことが考えられる。そのため，出現頻度については閾値を大幅に下げることなどにより元々出現頻度が低い隠語のうち，特に低い単語についてについても検出できる方法についても，今後検討していく。

〈6·2〉 閾値について 本手法では，閾値をどう設定するかによって精度（Precision 及び Recall）は変化する。

Table 10. Differences in results by function.

Evaluation Method	(1)	(2)	(3)
Precision	0.613	0.581	0.112
Recall	0.543	0.514	0.600
Accuracy	0.970	0.968	0.807
F-measure	0.576	0.545	0.188

Table 11. Details of the result including related words in True.

Evaluation Method	
Precision	0.871
Recall	0.771
Accuracy	0.987
F-measure	0.818

Precision と Recall はトレードオフの関係にあるため，どちらをより重視するかによっても設定すべき閾値は変化していく。そのため，たとえば，事前に用意した隠語のうち，5個隠し，その 5 個のうち，何個検出できるか見つけられるかを事前に実施し，目標となる Recall を 60%としたとき，5 個中 3 個見つけた時の設定値を閾値とすることができる。Precision についても同様の方法で閾値を設定することが可能である。このように，状況によって，Recall を重視したり，Precision を重視したり，フレキシブルに閾値を変更できる仕組みとなっている。本論文の実験においては，隠語及び関連語検出機能の検出における閾値を設定できるようにしたところ，それぞれの項目を変更し，一番良い値を閾値として決定した。

〈6·3〉 機能の効果の検証 今回の実験では，コアアイデアにいくつかの機能を加えて実装を行った。そこで追加した機能が効果的に働いていたのか検証を行った。今回は以下の条件で比較した。

（1） 今回の提案手法
主に品詞分類によるフィルタ機能，関連語検出機能の二つが含まれている。

（2） 品詞分類によるフィルタ機能のみ（関連語検出機能を含まない）

（3） 品詞分類によるフィルタ機能，関連語検出機能の両方の機能を含まない

機能の有無による結果は Table 10 のとおりである。

これより，コアアイデアに加えて追加した機能の中で，今回の実験では特に品詞分類が精度の差に大きく影響を与えていたことがわかった。そのため，品詞分類次第で見落としてしまう隠語があるおそれもあるので慎重に設定していく必要があると思われる。また別の事前実験では関連語検出機能も大きく効果を出していたことも確認していることから，共起性をさらに考慮することで，より効果的に機能することが期待できる。

〈6·4〉 精度と誤検出した単語について 現場の担当者へのヒアリングをした結果，0.6 という精度は実用に耐える精度であるとのコメントを得た。またヒアリングの中で，隠語に限らず，隠語と関連した関連語の検出も効果的であるとの意見もあった。関連語は隠語そのものではないが，隠語を用いた取引に利用されることが多く，犯罪捜査においては関連語を把握することも重要である。そこで，本研究において隠語であると検出した 31 語について確認したところ，〈5·5〉節のとおり，19 語は隠語であり，誤検出した 12 語について確認したところ，そのうち関連語が 8 語であった。関連語も検出対象として正解に含めた結果は Table 11 のとおりであった。Precision に注目すると，0.871 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。ただし，関連語については品詞分類を用いて自動的に追加登録するようにしているところ，隠語として検出されているものもあることからより隠語，関連語の検出精度を向上させる方法について検討する必要がある。

〈6·5〉 複合語型隠語の検出について 前処理として分かち書き作業を実施した中で，複合語が分割されるという課題があった。その分かち書きを元に単語分散表現を構築し，構築した単語分散表現に内包する単語を元に隠語を検出するシステムであることから，分かち書きの時点で複合語が分割されていると複合語が検出できないこととなる。複合語とは，本来独立した単語が二つ以上結合して，新たに一つの単語としての意味・機能をもつようになったものと定義されている[†]。たとえば，複合語の例として,「ほん・ばこ（本箱）」「やま・ざくら（山桜）」などがある。一方，大麻の隠語としては,「レモンスカンク」「ゴリラグルー」「ホワイトウィドー」などがこれまで確認したツイートの中から確認できている。これらの単語については，それぞれ「レモン・スカンク」「ゴリラ・グルー」「ホワイト・ウィドー」というように文節で切れてしまい,「ゴリラ」が検出されることはあっても「ゴリラグルー」として検出ができていなかった。

複合語の隠語（以下,「複合語型隠語」という。）が分割される対策として，分かち書きの文節単位を調節することが考えられるところ，たとえば Sudachi [33] では文節の長さを 3 種類設定できる機能がある。この機能のうち，一番文節を長くする設定を分かち書きを実施しても,「レモン・スカンク」のように文節が区切られてしまった。

本手法では，構築された単語分散表現モデル内に内包された単語をベースに隠語を判定するシステムであり，分かち書きの結果,「レモンスカンク」として区切られていなければ，単語分散表現の中に存在しないことになるため，隠語として判定されることがない。

一方で，Sudachi 内部で用いられる辞書に登録されていれば分かち書きは可能だが，本研究で検出対象とする隠語は当然ながら一般的な辞書には登録されておらず，さらに文節の中に「レモン」のような一般的な語が含まれている場合，文節が区切られやすいことが原因として考えられる。

[†] デジタル大辞泉（小学館）より引用。

そのために，事前に複合語を辞書登録し，その単位で文節を区切るようにすれば，課題が解決でき，複合語型隠語を検出できるようになると考えられる。しかしながら，実際の環境における複合語型隠語は，登録すべき単語が不明であると思われる。また辞書が更新されたとしても，複合語型隠語の場合，造語や認知度の低さの点から，辞書に追加される可能性が低いと思われる。そのため，複合語隠語発見の観点からも複合語を自動的に検出できることが望ましいため，将来課題とする。

〈6·6〉 共起する単語の特徴について 本手法により隠語を認識したうえで次の課題となるのは，隠語が用いられている不正なツイートを検出することである。「野菜」が隠語と認識した上で,「野菜」をキーワードにツイートを検索したとしても，大量及び不正な取引とは関係のない一般的な「野菜」に関するツイートが数多く検索されることは，〈2·3〉節のとおりである。隠語の出現するツイートを確認する中で，共起する隠語や関連語がジャンルによって異なることがわかった。大麻などの薬物取引関連の単語においては，たとえば「野菜」と一緒に「手押し」や「高純度」などが共起することが多いが，援助交際関連の単語として，たとえば「神待ち」では「手押し」や「高純度」といった単語とは共起せず,「諭吉」,「苺」,「パパ」,「JK」などといった単語が共起する頻度が高いことがわかった。さらに同じジャンルにおいても，隠語によって共起する単語が異なることもあった。一方で,「野菜」の場合,「生活」（飲料水の製品名と思慮）や「トマト」などと共起する場合，その野菜が出現するツイートは，一般的な用途でのみ使われている可能性が高い傾向がみられた。そのため，ジャンル（薬物取引や援助交際等）や単語単位で共起する単語を細かく調整可能とすることで，隠語・関連語との共起による不正なツイートの検出，一方で一般的な単語との共起による無関係なツイートの除外をより効果的に実現させることが期待できる。

7. 結　論

本稿では，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，コーパス間の単語の類似語の差異に着目する手法を提案した。提案手法では，犯罪に関係する単語の類似語もまた，犯罪を意図するものであると考え，同じ単語であっても一般的なコーパスと不正な目的のコーパスでは類似語が異なるという仮説の元，用意した二つのコーパス間の同じ単語の類似語を比較した。

そして，提案手法を用いて隠語検出実験を実施した結果，照合用に用いた単語以外の隠語を検出することができた。また比較用に用意したベースライン手法と比べても，Precision（適合率），Accuracy（正解率），F-measure（F 値）において高い精度を得ることができた。

これらのことから，本提案手法を拡張することで，変遷していく隠語を自動的に検出でき，サイバーパトロール等への貢献が期待できる。

今回の実験では，1 章で分類した隠語のツイートのうち，(1) の既知の隠語及び (3) の既知の隠語と未知の隠語が混在したツイートを対象として，コーパス分けリストに一致したツイートを元に隠語検出を目指したが，今後は隠語の対象を (2) も含めるため，犯罪の意図を含めたツイートを行うユーザーに着目し，ユーザー単位でツイートを収集し Bad コーパスを構築することで，より広い範囲から未知の隠語を検出することを目指す。また，それ以外にも複合語型隠語の検出や隠語を含む不正なツイートの検出などについても目指す。

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。

文　献

（1） “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.
（2） T. Mikolov, K. Chen, G. Corrado, and J. Dean: “Efficient estimation of word representations in vector space”, in 1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings, Y. Bengio, Y. LeCun, Eds. (2013)
（3） T. Hada, Y. Sei, Y. Tahara, and A. Ohsuga: “Codewords detection in microblogs focusing on differences in word use between two corpora”, in 2020 International Conference on Computing, Electronics Communications Engineering (iCCECE), pp.103–108 (2020)
（4） 羽田拓朗・清　雄一・田原康之・大須賀昭彦：「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」，情報処理学会研究報告 (2020)
（5） 警察庁：「平成 30 年における組織犯罪の情勢」，https://www.npa.go.jp/sosikihanzai/kikakubunseki/sotaikikaku04/h30.sotaijousei.pdf（2020/11/25 参照)
（6） L. Miller: “Those naughty teenage girls: Japanese kogals, slang, and media assessments” (2004)
（7） J. Mangnejo, A. Khuhawar, M. Kartio, and S. Soomro: “Inherent flaws in login systems of facebook and twitter with mobile numbers”, *Annals of Emerging Technologies in Computing*, Vol.2, pp.53–61 (2018)
（8） 警察庁：「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 (2020/11/25 参照)
（9） 内閣府：「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（2020/11/25 参照)
（10） 内閣府：「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 (2020/11/25 参照)
（11） R. Mihalcea and V. Nastase: “Word epoch disambiguation: Finding how words change over time”, in Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp.259–263, Jeju Island, Korea (2012)
（12） D. Wijaya and R. Yeniterzi: “Understanding semantic change of words over centuries” (2011)
（13） K. Yuan, H. Lu, X. Liao, and X. Wang: “Reading thieves’ cant: Automatically identifying and understanding dark jargons from cybercrime marketplaces”, in 27th USENIX Security Symposium (USENIX Security 18), pp.1027–1041, Baltimore, MD (2018)
（14） 大西　洋・田島敬史：「語の出現の偏りに基づく新たな隠語の発見」，DBSJ journal, pp.103–108（2013）
（15） W. Lee, S.S. Lee, S. Chung, and D. An: “Harmful contents classification us-

ing the harmful word filtering and svm”, in Computational Science – ICCS 2007, Y. Shi, G.D. van Albada, J. Dongarra, P.M.A. Sloot, Eds., pp.18–25, Berlin, Heidelberg (2007)
（16） 三谷亮介・小野　守・松本裕治・隅田飛鳥・服部　元・小野智弘：「有害性スコアリングによる web テキストにおける隠語の発見」，言語処理学会第 19 回年次大会，pp.461–464 (2013)
（17） K. Dela Rosa and J. Ellen: “Text classification methodologies applied to micro-text in military chat”, pp.710–714 (2009)
（18） K. Yao, H. Wang, Y. Li, J.J.P.C. Rodrigues, and V.H.C. de Albuquerque: “A group discovery method based on collaborative filtering and knowledge graph for iot scenarios”, *IEEE Transactions on Computational Social Systems*, pp.1–12 (2021)
（19） L. Huang, F. Liu, and Y. Zhang: “Overlapping community discovery for identifying key research themes”, *IEEE Transactions on Engineering Management*, Vol.68, No.5, pp.1321–1333 (2021)
（20） 北村美穂子・松本裕治：「対訳コーパスを利用した対訳表現の自動抽出」，情報処理学会論文誌，Vol.38, No.4, pp.727–736 (1997)
（21） 橋本広美・木下嵩基・原田　実：「フィルタリングのための隠語の有害語意検出機能の意味解析システム sage への組み込み」，情報処理学会研究報告。自然言語処理研究会報告，Vol.196, pp.N1–N6 (2010)
（22） K. Zhao, Y. Zhang, C. Xing, W. Li, and H. Chen: “Chinese underground market jargon analysis based on unsupervised learning”, in 2016 IEEE Conference on Intelligence and Security Informatics (ISI), pp.97–102 (2016)
（23） 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，Vol.13, No.2, pp.41–58 (2018)
（24） D. O’Day and R. Calix: “Text message corpus: Applying natural language processing to mobile device forensics”, pp.1–6 (2013)
（25） C. Kansara, R. Gupta, S.D. Joshi, and S. Patil: “Crime mitigation at twitter using big data analytics and risk modelling”, in 2016 International Conference on Recent Advances and Innovations in Engineering (ICRAIE), pp.1–5 (2016)
（26） G. Xiang, B. Fan, L. Wang, J. Hong, and C. Rose: “Detecting offensive tweets via topical feature discovery over a large scale twitter corpus”, pp.1980–1984 (2012)
（27） G. Wiedemann, E. Ruppert, R. Jindal, and C. Biemann: “Transfer learning from LDA to bilstm-cnn for offensive language detection in twitter”, *CoRR*, Vol.abs/1811.02906 (2018)
（28） A. Hakimi Parizi, M. King, and P. Cook: “UNBNLP at SemEval-2019 task 5 and 6: Using language models to detect hate speech and offensive language”, in Proceedings of the 13th International Workshop on Semantic Evaluation, pp.514–518, Minneapolis, Minnesota, USA (2019)
（29） 住田　淳・亮　隆弘・菱田隆彰：「児童被害を抑止するための sns 上の不正コメント抽出方法」，第 80 回全国大会講演論文集，Vol.2018, No.1, pp.117–118 (2018)
（30） 青木竜哉・笹野遼平・高村大也・奥村　学：「ソーシャルメディアにおける単語の一般的ではない用法の検出」，自然言語処理，Vol.26, No.2, pp.381–406 (2019)
（31） 松野省吾・水木　栄・榊　剛史：「日本語大規模 sns+web コーパスによる単語分散表現のモデル構築」，人工知能学会全国大会論文集，Vol.JSAI2019, pp.4Rin113–4Rin113 (2019)
（32） R. Řehůřek, and P. Sojka: “Software framework for topic modelling with large corpora”, in Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, pp.45–50, Valletta, Malta (2010)
（33） K. Takaoka, S. Hisamoto, N. Kawahara, M. Sakamoto, Y. Uchida, and Y. Matsumoto: “Sudachi: a Japanese tokenizer for business”, in Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan (2018)
（34） T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean: “Distributed representations of words and phrases and their compositionality”, *CoRR*, Vol.abs/1310.4546 (2013)

羽　田　拓　朗

（非会員） 2004 年静岡大学大学院情報学研究科情報学専攻博士前期課程修了。2008 年より警察庁中部管区警察局。現在，警察庁長官官房企画課兼情報通信局情報管理課。2021 年電気通信大学大学院情報理工学研究科情報学専攻博士前期課程修了。同年より同大学大学院博士後期課程に在籍中。主として自然言語処理に関連した機械学習の研究など犯罪軽減に向けた研究に従事。

清　　雄　一

（非会員） 2009 年東京大学大学院情報理工学系研究科博士後期課程修了。同年（株）三菱総合研究所入社。2013 年より電気通信大学。現在，同大学大学院情報理工学研究科准教授。博士（情報理工学）。エージェント，プライバシ保護技術等の研究に従事。2016 年度土木学会水工学論文賞，情報処理学会論文賞受賞。情報処理学会，電子情報通信学会，日本ソフトウェア科学会，IEEE Computer Society 各会員。

田　原　康　之

（非会員） 1991 年東京大学大学院理学系研究科数学専攻修士課程修了。同年（株）東芝入社。1993-1996 年情報処理振興事業協会に出向。1996-1997 年英国 City 大学客員研究員。1997-1998 年英国 Imperial College 客員研究員。2003 年国立情報学研究所着任。2008 年より電気通信大学 准教授。博士（情報科学）（早稲田大学）。エージェント技術，およびソフトウェア工学などの研究に従事。情報処理学会，日本ソフトウェア科学会会員。

大須賀　昭　彦

（正員） 1981 年上智大学理工学部数学科卒。同年 (株) 東芝入社。同社研究開発センター，ソフトウェア技術センター等に所属。1985-1989 年 (財) 新世代コンピュータ技術開発機構 (ICOT) 出向。2007 年より電気通信大学。現在，同大学大学院情報理工学研究科教授。2012 年より国立情報学研究所客員教授兼任。工学博士 (早稲田大学)。情報処理学会フェロー。ソフトウェア工学，エージェント，人工知能の研究に従事。1986 年度及び 2016 年度 情報処理学会論文賞，2013 年度人工知能学会研究会優秀賞，2014 年度同学会功労賞，2018 年度電子情報通信学会 ISS 活動功労賞 受賞。IEEE Computer Society Japan Chapter Chair，人工知能学会理事，日本ソフトウェア科学会理事，同学会監事等を歴任。情報処理学会，電子情報通信学会，人工知能学会，日本ソフトウェア科学会，IEEE Computer Society 各会員。