Twitterにおけるアカウント乗っ取りによるスパムツイートの検出

和田 なぎさ† 奥谷 貴志‡ 山名 早人§¶

†早稲田大学基幹理工学部 〒169-8555 東京都新宿区大久保 3-4-1
‡早稲田大学大学院基幹理工学研究科 〒169-8555 東京都新宿区大久保 3-4-1
§早稲田大学理工学術院 〒169-8555 東京都新宿区大久保 3-4-1
¶国立情報学研究所 〒101-8430 東京都千代田区一ツ橋 2-1-2

E-mail: {wada-n, okutani, yamana}@yama.info.waseda.ac.jp

あらまし Twitterの利用者が増加する一方で，スパム被害が増加し，快適にTwitterを利用するうえでスパム検知の精度向上が重要となっている．Twitterのスパムの8割は，スパマーにアカウントを乗っ取られることによって発生するものであり，正規のアカウントが唐突にスパムツイートを投稿する．これまでに提案されてきた研究は，スパムアカウント，すなわち常時スパムツイートを投稿するアカウントを検知するものであり，これらの乗っ取られたアカウントによるスパムツイートを検知することは不可能であった．そこで本研究では，文字n-gramを用いた文体類似度，およびツイートのクライアント，投稿時間をもとに，スパムツイートを検出する手法を提案する．このことにより，アカウント乗っ取りによって投稿されるスパムツイートを検知することが可能となる．

キーワード Twitter，スパム，自然言語処理，文字n-gram

1. はじめに

近年，ソーシャルネットワークサービスの利用者数が増加している．ソーシャルネットワークサービスとは，日記や掲示板，メールなどの機能を用いて人とのつながりを電子化するサービスである．代表的なソーシャルネットワークであるTwitter[1]は，2012年6月の時点でユーザが5億人を超え[1]，1日のツイート数は4億件[2]にも達する．

Twitterのツイートは，情報拡散機能に長けており，自分が持つ情報を幅広く発信できるという利点がある一方，スパムに悪用されるという問題点もある．Twitterではアカウントを所有するために，Twitter IDとパスワードのみを設定する．つまり，Twitter IDとパスワードさえ手に入れば他人になりすましてツイートを投稿することが可能である．スパマーはこの性質を活用することで，他人のアカウントを乗っ取り，スパムツイートを投稿し，拡散する．スパムをツイートするアカウントは，16%がbotによって自動拡散しているスパムアカウントであり，84%がスパマーによって操作されている一般のアカウントである[3]．

スパムテキストを検知する研究として，電子メールのスパム検知の研究[4][5]があるが，ヘッダ等のメール固有の情報を用いているため，Twitterの乗っ取りには対応できない．Twitterのスパム検知に関するサービス[6]や研究[7][8]は存在するが，スパムを発信するアカウントを検知するものである．そのため，スパマーではなく乗っ取られた一般的なアカウントによって投稿されるスパムを検知することは不可能である．そこで本稿では，このようなアカウント乗っ取りによるスパムツイートを検知するために，ツイート単位でスパムかどうかを判断する手法を提案する．

提案手法では，あるアカウントのツイートの特徴を収集し，その特徴と異なるツイートが同じアカウントから発信された場合，それを検知する．具体的には，文字n-gramを用いて対象とするアカウントの複数のツイートから文章の特徴（口調，顔文字等）を得て，新たに投稿されたひとつのツイートについて文体類似度を算出し，スパムツイート判定に用いる特徴量とする．さらに，ツイートを投稿したクライアントと投稿時間の情報を用いて精度を高める．

本稿では以下の構成を取る．2節では関連研究について，3節では提案手法について，4節では評価方法について説明し，5節ではまとめを行う．

2. 関連研究

2.1. Twitterのスパム検知に関する研究

Twitterのスパム検知に関するサービスや研究は，数多く存在している．以下では，スパムアカウント検知するサービスと研究について紹介する．

2.1.1. スパムアカウント検知サービス

TWITBLOCK[6]では，Twitterアカウントによってログインすることで，ログインしたTwitterユーザ自身のフレンド，またはフォロワーをスパマーである可能性が高い順にランキングする．調べたいアカウントに対し，フォロー/アンフォローの勢い（量と間隔），ツイートの勢い等，スパマーと疑わしい行為を行っているアカウントに対してスコアを付与し，このスコアが高いほどスパマーである可能性が高いと判定する．

[1] Twitter, https://twitter.com/.

2.1.2. スパムアカウント検知に関する研究

Benevenuto らの研究[7]では， 5,400 万件の Twitter アカウントから 19 億件のリンクと，18 億件のツイートを収集し，各アカウントをスパムアカウントか正規アカウントに手動で分類している．その際，スパムアカウントが持つ特徴を抽出し，SVM を用いて学習し分類を行う．特徴量としては，スパムアカウントと正規アカウント間で違いが大きく見られる以下に示すツイートの特徴3つとアカウントの行動3つを用いている．

① URL を含んでいるツイートの割合が大きい
② スパムワードを含むツイートの割合が大きい
③ ハッシュタグを含むツイートの割合が大きい
④ フレンドに対してフォローする割合が大きい
⑤ ひとつのアカウントの使用期間が正規アカウントに比べて短い
⑥ 受信するツイート数が少ない

このような特徴を学習させ，分類の結果，結果スパムアカウントの検知正答率は 69.7%であったと報告されている．

Wang の研究[8]では，Benevenuto らの研究と同様，スパムアカウントからスパムツイートの特徴を調査し，分類のために学習させている．大きく異なる点は，スパムアカウント発見に用いる特徴量と，分類手法である．スパムアカウントは，より効率よくスパムツイートを拡散させるために，より多くのフレンドとフォロワーを必要とする．そのため一般のアカウントには見られない，異常な行動が数値として表れる．それがフレンド数，フォロワー数，リプライ数である．500 件の Twitter アカウントから最近のツイート 20 件を収集し，フレンド数とフォロワー数，リプライ数を調査し，ベイジアンフィルタを用いて分類している．分類の結果，スパムアカウントの検知正答率は 89%となり，Benevenuto らの研究よりも精度向上に成功したと報告されている．

2.2. 著者推定に関する研究

本研究は，アカウント乗っ取りによるスパムツイートの検出を目的としており，2.1 項で述べた既存研究では，ツイート単位でスパムを検知することは難しい．そこで，調べたいアカウントのツイートの特徴（口調，顔文字等）を収集し，その特徴と異なるツイートが同じアカウントから発信された場合，それを検知する手法を取る．そのためにはツイートの文体を比較する必要があるため，文体比較に関する研究として著者推定の研究を参考にすることにした．

著者推定は以下の手順で構成される．手順 1 は，学習データとテストデータの収集．手順 2 は，学習データとテストデータとの文体類似度（2 つの文章において，これらの文体がどれほど似ているかを定量化したもの）の算出．次に手順 3 は，手順 2 で求めた文体類似度を用いた，テストデータ中の文章の著者推定を行う．

2.2.1 項と 2.2.2 項では，手順 2 で述べた文体類似度の判定方法が異なる著者推定の研究の手法を 2 つ紹介する．

2.2.1. 品詞 n-gram を用いた著者推定

中島ら[9]，井上ら[10]の研究では，2 つの文章 p, q において，それぞれ形態素解析を行い，品詞 n-gram の出現頻度分布に対するピアソンの積率相関係数の逆数を算出し，その値を文体類似度とする．提案手法では，文体類似度 $Dissim$ を式（1）のように定義している．C は文章 p, q の各々に存在する全ての品詞 n-gram の和集合である．f_{px}は文章 p における品詞 n-gram xの頻度を表しており，$\overline{f_{pq}}$は品詞 n-gram の頻度の平均を表している．

$$Dissim(p,q) = \frac{\sqrt{\sum_{i\in C}\left(f_{px}-\overline{f_{pq}}\right)^2}\sqrt{\sum_{i\in C}\left(f_{qx}-\overline{f_{qp}}\right)^2}}{\sum_{i\in C}\left(f_{px}-\overline{f_{pq}}\right)\left(f_{qx}-\overline{f_{qp}}\right)} \quad (1)$$

これは小さければ小さいほど，文章 p, q の文体が類似していることを表している．文章中の文字統計を用いているため，話題が異なっていたとしても同一著者と判定することが可能である．また言語的，構造的，内容的な文章の解析をする必要がない．しかし，未知語が出現した場合，形態素を正確に抽出できないため，精度は低下する．

2.2.2. 文字 n-gram を用いた著者推定

文字 n-gram とは，文章中に現れる n 文字の文字列のことを指している．松浦ら[11]の研究では，長さ n の文字列を xで表し，文章pにおけるxの出現確率分布$P_p(x)$を用いる．文体類似度 $Dissim$ を式（2）のように定義している．C は文章 p, q 双方に現れる n-gram の集合を表している．

$$Dissim(p,q) = \frac{1}{|C|}\sum_{x\in C}\left|log_{10}\frac{P_p(x)}{P_q(x)}\right| \quad (2)$$

品詞 n-gram と大きく異なる点は，形態素情報や構文情報を必要とせず，著者を特徴づける文字列を抽出することができるため，どのような場所に書き手の癖が現れるかを求めることも可能である．また，対象とする文章が複数の話題を持つ場合，すなわち文字の出現頻度分布が変化すると，著者推定の精度が低くなる．

3. 提案手法

本研究では，アカウントを持っている本人ではない他者によって投稿が行われた場合に，直ちに検知できることを目標とする．従来のスパムアカウントを検知する手法は，当該アカウントからのツイートの多くがスパムであるという前提に立っているため，本稿が対象とする手法として用いることができない．そこで，提案手法では，あらかじめアカウント本人のツイート

から文章の特徴を抽出し，新たにツイートされた文章との類似度を測定することで，他人のツイートを検出する手法を提案する．具体的には，過去のツイートと大きく異なるツイートであると判断された場合にスパムツイートと判定する．これを実現するために本研究では，著者推定の研究手順を応用する．

しかし，著者推定の研究で用いる文章とは異なり，ツイートは 140 字という短文である．また，口語に近い文章であるため，辞書に記載されていないような独特の単語（未知語）が多く使われる．そのため，多量のデータを必要とし，文節が整った文章を得意とする品詞 n-gram を用いて文体類似度の判定を行うことは困難である．一方，文字 n-gram は，単語や文章の構成などを考慮する必要がなく，書き手の癖を抽出しやすい．また，単語や品詞を用いて文体比較するより，文字そのものを用いて比較する方が得られるデータが多くなるため，短文であるツイートには適している．そのため， 2.2.2 項で紹介した，文字 n-gram を用いて文体比較を行うことにする．ツイートは文字数が少ないため，文体類似度を単体で用いると，精度が出しにくい．そこで，ツイートのクライアントと投稿時間の情報を，スパムツイートの判定に用いることで，精度を向上させる．

3.1. 概要

本項では，提案手法の流れを示す．

手順１　データ収集

まず手順 1 として，Twitter API を用いてデータの収集を行う．ひとつのアカウントから本人が投稿したと分かっている，過去のツイートを 1,000 件収集し，同時にツイートを投稿したクライアントと，その投稿時間も収集する．次に，収集した 1,000 件のうち 900 件をツイート$\mathrm{A}=\{a_1,a_2,\ldots,a_{900}\}$とする．ここで，ツイートAが持つ情報を当該アカウントの特徴を表す基本データとして用いる．残りの 100 件はツイート$\mathrm{A}'=\{a'_1,a'_2,\ldots,a'_{100}\}$として収集し，ツイートAから抽出した特徴に対し，同アカウントからのツイートが持つ特徴の分散を求めるために用いる．また，同じアカウントから新しく投稿された新着ツイートをツイート b とし，このツイート b について本人ツイートかスパムツイートかどうかの判定を行う．ここでは，ツイートA, ツイートA′，ツイート b において本人の特徴を得やすくするため，本人以外のユーザが投稿した文章が含まれるツイートを除去するために，公式 RT と非公式 RT（または QT）を除く．また，ノイズを除去するために，リプライに含まれる「@ユーザ名」とハッシュタグ「#文字列」，URL をツイートから取り除く．

手順２　文体非類似度・重み付き文体非類似度算出

手順 1 で収集したツイート$a_i \in \mathrm{A}$とツイート$a'_j \in \mathrm{A}'$に対し，文字 n-gram の出現パターンを用いて文体非類似度を算出する．また，その文体非類似度に対し，ツイートAのクライアント・投稿時間の情報から重み付けを行い，重み付き文体非類似度を算出する．なお，本研究では文体非類似度にクライアント情報のみで算出する重み付き文体非類似度①と，クライアントと投稿時間の情報を用いる重み付き文体非類似度②を算出する．

手順３　閾値の算出

手順 2 で算出した，文体類似度と重み付き文体非類似度①，②各々に対し，平均と標準偏差を求め，スパムツイートを判定するための閾値をそれぞれ算出する．

手順４　スパムツイート算出

手順 1 で収集したツイートa_iと新着ツイート b に対し，手順 2 と同様に文体非類似度，重み付き文体非類似度①，②のいずれかを算出する．そしてそれらの値が手順 3 で算出したそれぞれの閾値より，高い値を示すツイートをスパムツイートとして判定する．

3.2. 文字 n-gram を用いたツイートの文体類似度測定

本節では，ツイート$a_i \in \mathrm{A}$に対する各ツイート$a'_j \in \mathrm{A}'$の文体非類似度$Dissim(a_i,a'_j)$の算出方法について説明する．

ツイートa_iに対するツイートa'_1の文体非類似度$dissim(a_i,a'_1)$を式（3）から計算する．$P_{a_i}(x)$はツイートa_iにおける文字 n-gram x の出現確率を示し，C はツイートa_iとツイートa'_1の双方に現れる n-gram の集合を表している．1 件ずつのツイートa_iに対し，ツイートa'_1との文体非類似度を算出する．900 件の文体非類似度のうち，中央値にあたる文体非類似度を$Dissim(a_i,a'_1)$とする．このようにして，各ツイート$a'_j(1 \leqq j \leqq 100)$の文体非類似度$Dissim(a_i,a'_j)$を算出する．文字 n-gram には，比較する文章間の文字数の差が大きいと精度が下がってしまうため，本研究ではツイート 1 件ごとに文体類似度を算出する方法を提案した．

$$dissim(a_i,a'_j)=\frac{1}{|C|}\sum\left|\log_{10}\frac{P_{a_i}(x)}{P_{a'_j}(x)}\right| \tag{3}$$

$$Dissim(a_i,a'_j)=\widetilde{dissim}(a_i,a'_j) \quad 1 \leqq i \leqq 900 \tag{4}$$

また，1 件ずつのツイート$a_i \in \mathrm{A}$に対し，ツイート b の文体非類似度$Dissim(a_i,b)$も同様にして算出する．$Dissim(a_i,b)$の値が小さければ小さいほど，ツイートa_iとツイート b の特徴が似ていることを示しており，ツイート b が本人によるツイートである可能性が高くなる．逆に大きいほど，ツイート b がスパムツイートである可能性が高くなる．

3.3. ツイートのクライアントによる重み付け

ツイートを投稿する際に利用するクライアントは人によって異なる．本研究では，その違いを重み付けとして用いる．クライアント q から投稿されたツイート a_j' に対し，ツイートA内でクライアント q から投稿されたツイート数が占める割合 $P(q)$ を用いて，式（5）をもとに重み付け $Weight(q)$ を算出する．また，式（5）の重み付け係数は，20 件のアカウントに対し予備実験を行い最も良い結果を示した 0.9 と 1.0 を選択した．そして，$Weight(q)$ は頻繁に使われているクライアントであれば値が小さくなり，全く使われていないクライアントであれば値が大きくなる．そして，3.2 項で求めた $Dissim(a_i, a_j')$ の値に $Weight(q)$ を掛け合わせることによって，重み付き文体非類似度①として $Score(a_i, a_j')$ を算出する．

$$Weight(q) = \begin{cases} 0.9(1 - P(q)) & (P(q) > 0) \\ 1.0 & (P(q) = 0) \end{cases} \quad (5)$$

$$Score(a_i, a_j') = Dissim(a_i, a_j') * Weight(q) \quad (6)$$

ただし，q はツイート a_j' を投稿したクライアント．

また，ツイート b に対しても，3.2 項で求めた $Dissim(a_i, b)$ の値に $Weight(q)$ を掛け合わせることによって重み付き文体非類似度① $Score(a_i, b)$ を算出する．

3.4. ツイートの投稿時間比較

複数のクライアントから投稿するアカウントの場合，時間や場所によってクライアントを使い分けている．例えば，学生ならば家から投稿するときはパソコンのウェブブラウザから，通学中はスマートフォン向けクライアントから……というような形である．

時間 t にクライアント q から投稿されたツイート a_j' に対し，ツイート A_i 内で，時間 t から前後 1 時間に投稿されたツイートの中で，クライアント q から投稿されたツイートが占める割合 $P(q,t)$ を用いて，式（7）をもとに重み付け $Weight(q,t)$ を求める．また式（5）と同様に，式（6）の重み付け係数は 20 件のアカウントに対し，予備実験を行い，最も良い結果を示した 0.9 と 1.0 を選択した．そして $Weight(q,t)$ はツイート a_j' が投稿された時間帯 t の前後 1 時間において頻繁に使われるクライアントであれば値が小さくなり，滅多に投稿されないクライアントであれば値が大きくなる．そして，3.2 項で求めた $Dissim(a_i, a_j')$ の値に $Weight(q,t)$ を掛け合わせることによって，重み付き文体非類似度②として $Score(a_i, a_j')$ を算出する．

$$Weight(q,t) = \begin{cases} 0.9(1 - P(q,t)) & (P(q,t) > 0) \\ 1.0 & (P(q,t) = 0) \end{cases} \quad (7)$$

$$Score(a_i, a_j') = Dissim(a_i, a_j') * Weight(q,t) \quad (8)$$

ただし，q と t はツイート a_j を投稿したクライアントと投稿時間．

また，ツイート b に対しても，3.2 項で求めた $Dissim(a_i, b)$ の値に $Weight(q,t)$ を掛け合わせることによって重み付き文体非類似度② $Score(a_i, b)$ を算出する．

3.5. 閾値の算出

本研究で，乗っ取りによるスパムツイートを検知するために基準とするのは，文体非類似度であるが，これはアカウントによって値の広がりが異なるため，アカウントごとの閾値が必要とされる．そこであらかじめ，各アカウント内で本人による複数のツイートに対し，文体非類似度を求め，標準偏差を取ることで，各アカウントの閾値としている．つまり，式(4)，式(6)，式（8）で求めた 100 件ずつの文体非類似度と重み付き文体非類似度①，②のそれぞれに対し，標準偏差σを求め，式（9）または式（10）から閾値 $\alpha(a_i, a_j')$ を算出する．また，$\overline{Dissim(a_i, a_j')}$ は 100 件の文体非類似度の平均，$\overline{Score(a_i, a_j')}$ は 100 件の重み付き文体非類似度①または②の平均を示しており，係数の 0.08 は，実験をした 0.01～1 の中で最も良い結果を示したものを用いている．

$$\alpha(a_i, a_j') = \sigma + 0.08\overline{Dissim(a_i, a_j')} \quad (9)$$

$$\alpha(a_i, a_j') = \sigma + 0.08\overline{Score(a_i, a_j')} \quad (10)$$

この閾値 $\alpha(a_i, a_j')$ より，$Dissim(a_i, b)$ または $Score(a_i, b)$ の値が高いとき，ツイート b をスパムツイートとして判定する．

4. 実験・評価

4.1. データセット

Twitter Search API を用いて，2013 年 1 月 1 日に英語でツイートしているアカウント 1,000 件を収集し，2013 年 1 月 28 日から過去 1,030 件以上ツイートしているアカウント 100 件から，それぞれツイートと各クライアント情報，投稿時間を 1,030 件収集し，直近の 30 件をツイート b の本人ツイート，次の 100 件をツイートA′とし，残りの 900 件をツイートAとする．また，Twitter Search API を用いて，スパムツイートに頻出する単語をクエリとして 100 件のスパムツイートを収集したうちからランダムに選んだ 30 件をツイート b のスパムツイートとして利用する．

4.2. 評価方法

事前に収集した 1,000 件のツイートとは別に，本人によるツイート 30 件とスパムツイート 30 件を用意した．そして，合計 60 件のツイートを正しく分類できるかを判断基準とする．実験結果を示すのに表 1 を用いる．表 1 において，TP（True Positive）はスパムツイートとして正しく分類されたもの，FP（False Positive）はスパムツイートであるにも関わらず，本人によるツイートと誤分類されたもの．FN（False Negative）は本人によるツイートであるにも関わらず，スパムツイートと誤分類されたもの．そして TN（True Negative）は本人によるツイートとして正しく分類されたもの．こ

れらのツイートの割合を表 1 にまとめた．本研究では，TP と TN の割合(TP+TN)/60 を正答率とする．

表 1　実験結果を示す表

		実験結果	
		スパムツイート	本人ツイート
実際の分類	スパムツイート	TP	FP
	本人ツイート	FN	TN
正答率		(TP+TN) / 60	

4.3.　事前実験

実験を行う前に，精度が最も高くなる環境設定を行うために，事前実験を行った．文字 n-gram の文体非類似度で最も良い精度を出す n の大きさを調査した結果を図 1 に示す．正規のアカウント 20 件に対し n= 2, 3, 4 それぞれの文体非類似度を算出し，スパムツイート検出の正答率を求め，比較を行った．図 1 からわかるように，n=2,3 のとき高い正答率を出していることがわかる．文字 n-gram というのは，文字数が n 未満であると算出不能である．ツイートは文章ではなく，1 単語でも成立するため，文字数が 4 文字以下のツイートは頻繁に発生する．そのため，n の値が大きくなればなるほど，文体非類似度が算出不能となる確率が高くなる．また n の値が大きくなると，2 つのツイート間に同じ文字列が存在する確率が低くなる．そこで，本研究では n=2 として実験を行うことにする．

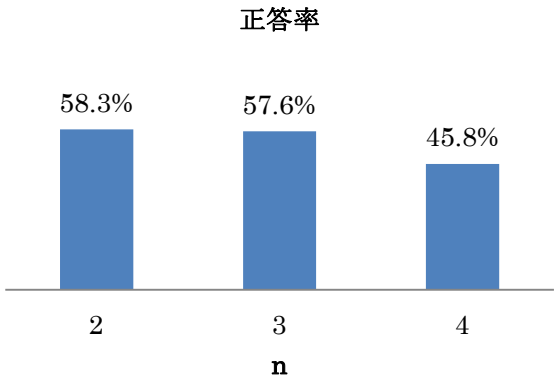

図 1　n-gram の最適な n の調査結果

4.4.　実験結果

100 件のアカウントに対し実験を行い，正答率を算出した．文体非類似度のみで実験した結果の平均を表 2，文体非類似度に対しクライアントで重み付けをし，実験した結果の平均を表 3，そして文体非類似度に対し，クライアントと投稿時間で重み付けをし，実験した結果の平均を表 4 に示す．また，比較をするためにクライアントと投稿時間の情報のみで実験した結果の平均を表 5 に示す．

表 2　文体非類似度のみ
実験結果のツイート数の平均

		実験結果	
		スパムツイート	本人ツイート
実際の分類	スパムツイート	15.1	14.9
	本人ツイート	9.9	20.1
正答率		58.7%	

表 3　文体非類似度×クライアント
実験結果のツイート数の平均

		スパムツイート	本人ツイート
実際の分類	スパムツイート	26.5	3.5
	本人ツイート	7.6	22.4
正答率		81.4%	

表 4 文体非類似度×（クライアントと投稿時間）
実験結果のツイート数の平均

		スパムツイート	本人ツイート
実際の分類	スパムツイート	27.0	3.0
	本人ツイート	7.3	22.7
正答率		82.8	

表 5 クライアントと投稿時間
実験結果のツイート数の平均

		スパムツイート	本人ツイート
実際の分類	スパムツイート	27.5	2.5
	本人ツイート	13.8	16.2
正答率		72.9%	

4.5.　実験評価

実験結果をもとに，F 値の算出結果を表 6 に示す．表 6 からわかるように，文体非類似度×（クライアントと投稿時間）の手法が，安定して最も良い結果を出していることがわかる．従って，本研究では文体非類似度にクライアントと投稿時間の情報を用いて重み付けを行う方法によって 82.8%という高い正答率を出すことに成功した．

表 6　F 値算出結果

手法	F 値
文体非類似度	0.520
文体非類似度×クライアント	0.821
文体非類似度×(クライアントと投稿時間)	0.835
クライアントと投稿時間	0.774

5.　おわりに

本研究では Twitter において，スパマーによるアカウント乗っ取りによって発生する，正規アカウントから投稿されたスパムツイートの検知することを目標とした．文字 n-gram を用いてアカウントの複数のツイートから文章の特徴（口調，顔文字等）を得て，新たに投稿されたひとつのツイートについて文体非類似度を算

出し，ツイートを投稿したクライアントと投稿時間を用いて文体非類似度に対して重み付けを行い，違った特徴を持つツイートをスパムツイートとして検出する手法を提案した．実験の結果，n-gram において n=2 のとき，文体非類似度のみで行う正答率より，クライアントと投稿時間の情報を取り入れる手法の方が，正答率が高くなることが判明した．本研究では，クライアントと投稿時間の関係を用いて，文体非類似度に重み付けを行う手法で，82.8%の正答率を得ることができた．

将来的には，スパム発信元のアカウント情報を利用しない本手法を適用することで，近年被害が多発している，電子メールにおけるアカウント乗っ取りによるスパムメール[12]の検知や Twitter のみならず，SNS 全般における成りすまし・乗っ取り問題[13]に貢献できると期待される．

参　考　文　献

[1] Twitter reaches half of a accounts more than 140 millions in the U.S., http://semiocast.com/publications/2012_07_30_Twitter_reaches_half_a_billion_accounts_140m_in_the_US.（2012 年 12 月 4 日アクセス）

[2] Twitter hits 400 million tweets per day, mostly mobile, http://news.cnet.com/8301-1023_3-57448388-93/twitter-hits-400-million-tweets-per-day-mostly-mobile/.（2012 年 12 月 4 日アクセス）

[3] C. Grier, K. Thomas, V. Paxspm, and M. Zhang :“@spam: The Underground on 140 Characters or Less”, Proc of the 17th ACM conference on Computer and communications security, pp.27-37, 2010.

[4] 伊藤朋哉，寺田真敏，土居範久:“発信元情報を適用したベイジアンスパムフィルタ方式の提案”，情報処理学会報告,Vol.2008, No.21, pp.285-290, 2008.

[5] 王卉歓，中谷直司，小池竜一，厚井裕司，朴美娘:“ベイズ学習アルゴリズムのスパムフィルタとウイルスフィルタへの適用の最適化”，情報処理学科論文誌, Vol.48, No.9, pp.3125-3135, 2007.

[6] TWITBLOCK, http://twitblock.org/.（2012 年 12 月 4 日アクセス）

[7] F.Benevenuto, G. Magno, T. Rodrigues, and V.Almeida :“Detecting Spammers on Twitter”, Proc of the Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS), 2010.

[8] A. Wang:“Don’t follow me: Spam detection in twitter”, Proc of the 2010 International Conference on Security and Cryptography (SECRYPT), pp.1-10, 2010.

[9] 中島泰，山名早人:“品詞と助詞の出現パターンを用いた類似著者の推定とコミュニティ抽出”，DEIM2011, B6-5, 2011.

[10] 井上雅翔，山名早人:“品詞　n-gram を用いた著者推定手法－話題に対する頑健性の評価”，DBSJ Journal, Vol.10, No.3, pp.7-12, 2012.

[11] 松浦司，金田康正:“近代日本文学者 8 人による文章における文字 n-gram の分布を利用した近代日本語分の著者推定”，計量国語学，Vol.22，No.6, pp.1-9, 2000.

[12] 独立行政法人情報処理推進機構，http://www.ipa.go.jp/security/txt/2012/10outline.html.（2013 年 1 月 8 日アクセス）

[13]「なりすまし」や「乗っ取り」SNS で被害急増，http://www.nikkei.com/article/DGXNZO51272100R00C13A2TJ2000/?dg=1.（2013 年 2 月 2 日アクセス）