情報処理学会論文誌　データベース　Vol.8　No.4　16–26 (Dec. 2015)

ツイート投稿位置推定のための単語フィルタリング手法

森國 泰平[1,a)]　吉田 光男[1,b)]　岡部 正幸[1,c)]　梅村 恭司[1,d)]

受付日 2015年6月20日, 採録日 2015年10月8日

概要：ツイートに含まれる特徴と位置情報を対応させることで，実世界を観測するセンサとして Twitter を活用することができる．しかし位置情報が付加されたツイートは少なく，Twitter をセンサとして活用するときの問題の 1 つとなる．そこで本研究では，ツイートの投稿位置を推定し，より多くのツイートに正確な位置情報を付与することを目的とする．この目的を達成するために，ツイート中のノイズとなる単語を除去するためのフィルタリング手法を提案する．また，単語の地理的分布を平滑化するためのスムージング手法も提案する．これらの提案手法が従来手法よりも有効に機能することを示し，その考察を行う．

キーワード：Twitter，位置情報推定，ノイズ除去

Geo-location Estimation of Tweets with Stop Words Detection

Taihei Morikuni[1,a)]　Mitsuo Yoshida[1,b)]　Masayuki Okabe[1,c)]　Kyouji Umemura[1,d)]

Received: June 20, 2015, Accepted: October 8, 2015

Abstract: Twitter can be considered as a real-time sensor that responds to real-world events by combining the content and location information of tweets. However, a problem persists: tweets containing location information are too small. To overcome this problem, we estimate the location where a tweet was posted. Our main method involves using word filters called *AF filter* and *TF–IAF filter* that detect stop words. In addition, we propose a smoothing method called *Distance smoothing* for overcoming sparsity of words. We show that both our methods improve location estimation accuracy and discuss the features of the results.

Keywords: Twitter, geo-location estimation, stop word detection

1. はじめに

Twitter[*1]は，ツイートと呼ばれる短文を投稿することができるソーシャル・ネットワーキング・サービス（SNS）である．Twitter ユーザは，ツイートに自身の近況などをリアルタイムに記述し，同時に記述時の位置情報（ジオタグ）を付加して投稿する場合もある．ツイートから特徴を抽出し，付加された位置情報と組み合わせることで，その瞬間における地域の特徴を知ることができる．このことから，Twitter は実世界でのイベントを観測するリアルタイムセンサ（ソーシャルセンサ）として活用することが可能である．Twitter をセンサとして活用する研究は，これまでに数多く行われている [1]．

Twitter をリアルタイムセンサとして活用する利点の 1 つとして，一般的なセンサでは観測できない事象を観測できることがあげられる．たとえば，一般的なセンサでも地震や台風などの災害自体を観測することはできるが，人的被害の状況や支援物資の不足を観測することは困難である．2011 年 3 月 11 日に発生した東日本大震災では，このような情報の伝達手段として Twitter が利用されており [2]，Twitter をリアルタイムセンサとして活用することには十分な意義があるといえる．

Twitter をリアルタイムセンサとして活用する場合に問題となるのは，位置情報が付加されたツイートが少なく [3]，日本語のツイートにおいても約 0.18%しか存在しないこ

[1] 豊橋技術科学大学
Toyohashi University of Technology, Toyohashi, Aichi 441–8580, Japan
[a)] morikuni14@ss.cs.tut.ac.jp
[b)] yoshida@cs.tut.ac.jp
[c)] okabe@imc.tut.ac.jp
[d)] umemura@tut.jp

[*1] https://twitter.com (accessed 2015–10–23)

とである [4]．そのため観測できる情報が少なく，実際にTwitter をリアルタイムセンサとして活用するときの問題となる．投稿位置の推定によって，できるだけ多くのツイートに位置情報を付与することができれば，この問題を克服できると考えられる．

本研究では，市区町村レベルの各エリアにおける単語の出現頻度を学習し，位置情報が付加されていないツイートの投稿位置を推定する．従来研究では推定の適合率を重視するものが多く見られるが，本研究ではできるだけ多くのツイートに正確な位置情報を付与することを目的とする．つまり，推定対象とするツイート集合にどの程度，正確な位置情報を付与できたのかを示す正解度を重視する．従来重視されていた適合率ではなく正解度を重視するのは，少量の推定結果の正確さよりも，多くの情報を集めることが必要とされる状況が考えられるためである．たとえば，人的被害を含む災害などが発生した場合に，適合率重視の手法では得られなかった位置情報を得ることで，人命救助に役立てられる可能性がある．また，投稿位置推定は位置情報が付加されているツイート数が少ないことを問題として行われるものが多いため，正解度を向上させることが本質的な問題解決につながる．

本研究のポイントは以下の 2 つの観点から正解度を向上させることである．

- ツイートに含まれる単語から地理的分布に偏りのない単語（ノイズ）を除去する．
- 単語の地理的分布がまばらであることによる観測誤差の影響を小さくする．

ノイズを除くことによって特定の場所と関連のある単語を抽出し，また，スムージングによって観測された単語の分布を真の分布に近づけることができれば，正解度が向上すると考えた．正解度が向上するということは，推定対象とするすべてのツイートに対して多くの正解が得られるということであり，できるだけ多くのツイートに正確な位置情報を付与することを意味する．

2. 関連研究

Twitter をリアルタイムセンサとして活用するための研究は，これまでにも多く行われている [1]．Aramaki ら [5] は，Twitter からインフルエンザの流行を検出する方法を提案し，都道府県レベルでインフルエンザの流行を可視化している[*2]．Sakaki ら [6] は，Twitter を用いて日本で発生した震度 3 以上の地震の震源地を高精度で特定している．

SNS を用いて位置情報推定を行う研究は多く存在する [7], [8]．推定手法は，投稿内容（テキスト）を用いて推定を行うコンテンツベースの手法とユーザの友人関係などを用いて推定を行うグラフベースの手法，さらにそれらを組み合わせる手法という 3 つの手法に大きく分けられる．

コンテンツベースの手法は，地理的な分布に偏りのある単語を抽出するフィルタリング手法を提案するもの，単語の地理的分布を表すモデル式を提案するもの，単語の地理的分布を平滑化するスムージング手法を提案するものという 3 つの提案に大きく分けられる．Cheng ら [3] がユーザの居住地を推定するために提案したフィルタリング手法は，地理的に狭い範囲にのみ出現する単語を抽出する．三木ら [9] は，ツイートの投稿位置を推定するために TF–IDF [10] の概念を用いたフィルタリング手法を提案している．三木らの提案したフィルタリング手法では，地理的分布に偏りのある単語の時間的局所性を考慮しており，一定期間ごとに単語データベースを更新することで，適合率が向上したと報告している．Han ら [11] は，IDF [12] の概念を用いたフィルタリング手法の提案に加え，生成モデルと識別モデルという 2 つのモデルを比較している．Kinsella ら [13] は，ベイズの定理と Kullback-Leibler divergence を用いた 2 つのモデルを提案している．Roller ら [14] は，k-Nearest Neighbors（kNN）を用いたモデル式を提案している．Yamaguchi ら [15] は，ツイートの時間的特徴を考慮し，モデル式を更新しながらユーザの位置を推定している．Cheng ら [3] は，ラプラススムージングといった一般的なスムージング手法に加え地理的な関係を考慮したスムージング手法を提案している．Cheng らはラプラススムージングによって推定性能が低下したと報告しており，スムージング手法の提案においては地理的な関係を考慮する必要性が示唆されている．単語の出現頻度の学習については，ツイートのジオタグに含まれる位置情報を用いて学習するものが多いが，伊川ら [16] は，位置情報サービス（Foursquare[*3] など）から投稿されたツイートを用いて学習している．

グラフベースの手法は，主に友人の居住地の位置情報を用いてユーザの居住地を推定する [17]．Backstrom ら [18] は，Facebook[*4] において居住地が記述されたユーザを用いて学習し，友人は近くに住んでいるという仮説をもとに居住地を推定している．Rout ら [19] は，SVM を使ってユーザ間の社会的な関係と位置の関係を学習し推定している．Sadilek ら [20] は，社会的関係を推定するモデルを提案し，既知でない友人関係からも推定している．

これまでに述べたコンテンツベースの手法とグラフベースの手法を組み合わせた手法も提案されている．Li ら [21] は，Unified discriminative invluence（UDI）というモデルを提案し，ツイートとユーザのフォロー関係の両方を用いて推定している．彼らは Multiple location profiling（MLP）というユーザに複数の場所を割り当てるモデルも提案している [22]．

[*2] http://mednlp.jp/influ/ (accessed 2015–10–23)

[*3] https://ja.foursquare.com (accessed 2015–10–23)

[*4] https://www.facebook.com (accessed 2015–10–23)

位置情報に関連した分析を行う研究も存在する．Cheng ら [23] は，位置情報サービスの情報からユーザの行動パターンを定量的に評価している．Kamath ら [24] は，ジオタグ付きツイートを用いてハッシュタグの地理的な広がりについて調査している．Flatow ら [25] は，デバイスやプラットフォーム別の位置情報推定について比較している．Watanabe ら [26] は，ローカルイベントの位置を建物レベルで特定し，関連するツイートに位置を付与することで投稿位置も推定できている．

ツイートの投稿位置を推定する場合，グラフベースの手法では周辺ノードに対する重みの伝搬などが必要であり，リアルタイムに処理することが困難であると考えられる．そのため本研究では，コンテンツベースの手法によってツイートの投稿位置を推定する．従来の投稿位置推定は，推定できたツイートを母数とする正解の比率（適合率）向上と地理的分布に偏りのある単語の抽出精度向上を目的としていたが，本研究では，推定対象とするすべてのツイートを母数とする正解の比率（正解度）向上を目的とする．多くのツイートに位置情報を付与することができれば，Twitter をリアルタイムセンサとして活用することが可能になる．

3. 提案手法

3.1 概要

本研究では，位置情報付きツイートに含まれる単語から，各エリアでの単語の出現頻度を学習し，位置情報が付加されていないツイートの投稿位置を推定する．ツイート（投稿内容）には，方言やランドマーク名など，地理的局所性のある単語が含まれていると考えられる．これらの単語の分布を用いることで，位置情報が付加されていないツイートの投稿位置を推定できる可能性がある．そのため，本研究ではツイート内容のみを使用して投稿位置を推定する．また，Twitter をリアルタイムセンサとして活用するためにはリアルタイム性が求められることや，ツイート数の少ないユーザのツイートに対しても投稿位置を推定できることが望ましいため，推定に使用するのは推定対象とする 1 件のツイート内容のみとする．

投稿位置の推定は，式 (1) を用いた最尤推定によって行う．A を推定対象エリアの集合（4.1 節で詳述），a を推定対象エリア，W_t をツイート t に含まれる単語の集合，$p(w)$ をデータ中で単語 w が出現する確率，$p(a|w)$ を単語 w がエリア a で出現する確率として，ツイート t が投稿されたエリア a^* を推定する．

$$\begin{aligned} a^* &= \arg\max_{a \in A} p(a;t) \\ p(a;t) &= \sum_{w \in W_t} p(a|w)p(w) \end{aligned} \tag{1}$$

これは単純にツイートに含まれる単語を用いた推定であり，以下の 2 つの問題が存在する．

- ツイートに含まれる単語の多くは，地域を特定しえないノイズである．
- 単語の地理的分布がまばらであるため，観測誤差による影響が大きい．

これらの問題を克服するために，Cheng ら [3] はノイズとなる単語のフィルタリング手法と頻度情報のスムージング手法を提案した．ノイズとなる単語のフィルタリング手法とは，ツイートに含まれる単語を地理的分布に偏りのある単語とノイズとに二分する手法である．頻度情報のスムージングとは，エリアごとの単語の出現頻度を平滑化することによって，観測誤差を小さくするための手法である．Cheng らは，単語のフィルタリングが正解度の向上に大きな影響を与え，頻度情報のスムージングによる影響は小さかったと報告している．

本研究では，Cheng らと同様にフィルタリング手法とスムージング手法を提案し正解度の向上を図る．そして，スムージングについてはより地理的な関係を考慮した手法を提案する．

3.2 単語の出現頻度の学習

提案手法はエリアごとの単語の出現頻度を用いてツイートが投稿されたエリアを推定する最尤推定であり，すべてのエリアについて各単語の出現頻度を学習する必要がある．そのため，位置情報が付加されたツイートに対して形態素解析[*5]を行い，単語の集合に分割する．さらに，エリアデータの境界情報から，ツイートが投稿されたエリアを特定し，各エリアにおける各単語の出現確率を求める．$C(a,w)$ をエリア a における単語 w の出現回数，$C(w)$ を単語 w の総出現回数，n を単語数（タイプ数）として，式 (1) 中の確率値は次の式で計算される．

$$\begin{aligned} p(a|w) &= \frac{C(a,w)}{C(w)} \\ p(w) &= \frac{C(w)}{\sum_i^n C(w_i)} \end{aligned} \tag{2}$$

3.3 ノイズとなる単語の除去

本研究では，ツイートからノイズとなる単語を除去し，正解度を向上させるフィルタリング手法として 2 つのフィルタを提案する．一方は **AF フィルタ**，他方は **TF–IAF フィルタ**で，どちらのフィルタも情報検索などで用いられる TF–IDF [10], [12] の概念を用いている．$C(w)$ を単語 w の総出現回数，A を推定対象エリアの集合，A_w を単語 w の出現したエリアの集合として式 (3)，式 (4) で表される．各単語について AF および TF–IAF の値を求め，AF フィルタでは AF 値が閾値 α より大きいもの，TF–IAF フィルタでは TF–IAF 値が閾値 β よりも小さいものをノイズとして分類する．

[*5] Kuromoji 0.7.7 (http://www.atilika.org)

$$TF(w) = \log(C(w))$$
$$AF(w) = |A_w| \quad (3)$$
$$IAF(w) = \log \frac{|A|}{AF(w)}$$
$$TF\text{–}IAF(w) = TF(w) * IAF(w) \quad (4)$$

AF フィルタは，少ないエリアで出現する単語を高く評価する．これにより地理的分布に偏りのある単語が高く評価され，正解度の向上が期待できる．

TF–IAF フィルタは，単語の出現頻度を TF 値で評価するため，ツイートに含まれる可能性の低い単語がノイズとして判定されやすいという特徴を持っている．そのため，日常的に使用されることの少ない単語をノイズとして除外することができる．また，AF フィルタと同様に，IAF の値によって地理的な分布に偏りのない単語が除かれるため，正解度の向上も期待ができる．TF–IAF フィルタのもう 1 つの特徴として，三木ら [9] とは異なり，TF–IDF の概念をそのままフィルタリングに適用したものではない点があげられる．TF–IDF の概念をそのままフィルタリングに適用すると TF の計算式が，単語 w の総出現回数ではなく，あるエリアにおける単語 w の出現回数となる．本研究では，地理的分布に偏りのある単語の抽出精度向上を目的とするのではなく，多くのツイートに位置情報を付与することを目的としているため，単語の出現頻度を全体で評価し，ツイートに多く含まれるであろう単語を高く評価することが重要となる．ただし，本質的には地理的分布に偏りのある単語を評価する必要があるため，TF 値の差による影響を小さくする目的で $C(w)$ に対数をとっている．

3.4　頻度情報のスムージング

本研究では，エリアごとの単語の出現頻度を平滑化し，観測誤差を小さくするために **Distance スムージング**を提案する．a を推定対象エリア，A_w を単語 w が出現したエリアの集合，λ を距離による影響の強さ，$distance(a,i)$ をエリア a とエリア i 間の距離，$p(i|w)$ を単語 w がエリア i で出現する確率として，式 (5) で表される．なお，式 (2) 中の $p(a|w)$ が確率であるのに対し，$p'(a|w;\lambda)$ は確率ではない．

$$p'(a|w;\lambda) = \sum_{i \in A_w} \exp(-\lambda * distance(a,i)) * p(i|w) \quad (5)$$

Distance スムージングでは，距離の近いエリアにおける単語の出現頻度が大きな影響を与える．つまり，単語の出現頻度が高いエリアがあった場合に，その周囲のエリアでも単語の出現頻度が高くなるという特徴がある．Cheng らが提案したスムージングでは近いエリアではなく，同じ格子内のエリアの影響を強く受ける．Distance スムージングでは，距離が近いエリアほど影響が強くなるため，より地理的な関係を考慮したスムージングとなる．

4.　実験設定

4.1　エリアデータ

エリアデータとは，学習と推定における領域を区別するための境界データと重心データである．山や川で隔てられたエリアでは，生活の様子や文化，方言などが異なる可能性がある．地理的な意味を考慮せずに緯度経度でエリアを分割すると，エリアの中で複数の方言が混在してしまうおそれがある．そこで，本研究では地理的に意味のある境界で分けられていると考えられる，都道府県や市区町村の領域データの利用を検討する．エリアとして扱う領域は，いくつかの分割の粒度を考えることができ，関東や関西などの地方単位や，都道府県単位，市区町村単位，町丁（丁目や字など）単位などがあげられる．これらのうち，本研究では市区町村単位の領域をエリアとして使用する．

エリアデータは総務省統計局の「平成 22 年国勢調査（小地域）2010/10/01」[*6]から，境界データ（世界測地系緯度経度・GXML 形式）を取得した．このデータは，町丁単位で分けられたデータである．町丁は県番号 (KEN) と県内通し番号 (SEQ_NO2) の 2 つのメタデータで一意に求まる．それぞれのデータについて，Boundary タグで囲まれた境界全体を含む矩形座標データ，Geometry タグで囲まれた境界を示す詳細な座標データ，Property タグで囲まれた都道府県名や中心点座標などを示すメタデータの 3 つの情報が含まれている．

本研究では Geometry タグに含まれる詳細な境界データをエリアの境界として使用する．また，取得したデータは町丁単位のデータであるため，本研究でエリアとして扱う市区町村単位の境界データに変換する必要がある．市区町村は県番号 (KEN) と県内都市番号 (CITY) の 2 つのメタデータで一意に求まることから，次の手順に従って，市区町村単位の境界データに変換した．

(1) (KEN, CITY) の 2 つのメタデータをキーとして，町丁単位のデータをグループ化する．
(2) 各グループについて，境界データを結合する（境界データ完成）．
(3) 各グループについて，結合された境界データの重心を求める（重心データ完成）．

境界データはツイートが投稿されたエリアを調べるために使用し，重心データは推定結果のエリアとツイートの投稿位置との誤差，およびスムージングにおけるエリア間の距離を計算するために使用する．なお，境界データの結合と重心の計算には，MySQL 5.6 の MultiPolygon 型と Centroid 関数を使用した．

[*6] http://e-stat.go.jp/SG2/eStatGIS/page/download.html (accessed 2015–10–23)

4.2　ツイートデータ

2013年1月1日から2013年12月31日および2014年1月8日から2014年1月21日に投稿された位置情報付きのツイートデータをTwitter Streaming API[*7]を用いて収集した．通常，Streaming APIから得られるツイートはサンプリングされたものであり，すべてのツイートを取得することはできないが，ジオタグが付加されているツイートについてはそのすべてが含まれると報告されている[27]．収集したツイートを日本国内からの投稿に限定するため，ツイートの投稿位置を示すメタデータcoordinates[*8]に記述された座標が，エリアデータのいずれかのエリアに含まれるツイートのみを抽出した．また，Botによる自動投稿の影響を少なくするために，次の条件にあてはまるツイートをBotによる投稿として除いた．

(1) Twitterクライアント名に「NightFoxDuo」を含む．
(2) ツイート内容に「きつねかわいい！！！」を含む．
(3) 緯度経度が（34.967096, 135.772691）である．
(4) Twitterクライアント名，アカウント名，表示名，プロフィールのうち，1つ以上に「BOT」「Bot」「bot」「人工無能」のいずれかを含む．

先の4つのBot条件の中で(1)(2)(3)は「夜狐八重奏＋[*9]」の影響を除くことを目的としている．「夜狐八重奏＋」はiOS向けのTwitterクライアントであり，ツイートのジオタグに京都府の伏見稲荷大社の座標を自動で追加する機能や，「きつねかわいい！！！(X回目)」（Xには投稿回数が入る）というツイートを自動投稿する機能がある．これらの機能によって多くのツイートが本来の位置情報と関係なく京都府伏見区で観測されるため，本研究ではこのクライアントのユーザをBotとして扱う．収集した80,652,258件のツイートからBotによる投稿を除き，75,389,276件のツイートを実験で利用する．

Twitterをリアルタイムセンサとして活用するためには，ツイートが投稿されたときに位置を推定することが望ましい．そのため，学習に利用するツイートデータは評価に利用するデータよりも過去に投稿されたものである必要がある．よって，2013年1月1日から2013年12月31日までの71,187,807件のツイートを学習データとし，2014年1月8日から2014年1月14日までの2,140,350件のツイートから10%サンプリングを行った214,035件のツイートをパラメータ獲得のためのデベロップメントデータとする．また，2014年1月15日から2014年1月21日の2,061,119件のツイートを性能評価のためのテストデータとする．

[*7] https://dev.twitter.com/streaming/overview (accessed 2015–10–23)
[*8] https://dev.twitter.com/overview/api/tweets (accessed 2015–10–23)
[*9] http://www58.atwiki.jp/nightfox (accessed 2015–10–23)

4.3　評価方法

推定結果の評価には，正解度（Accuracy），適合率（Precision），それらの調和平均（以降，H値）（HScore）を用いる．正解度はどれだけ多くのツイートに正確な位置情報を付与できたか，適合率は推定結果の正確さ，H値はその両方を評価する指標である．各評価指標は以下の式で表される．

$$ErrDist(t) = d(l_{act}, l_{est}(t))$$
$$Accuracy(T,x) = \frac{|\{t|t \in T \wedge ErrDist(t) \leq x\}|}{|T|}$$
$$Precision(T,x) = \frac{|\{t|t \in T \wedge ErrDist(t) \leq x\}|}{|T_{est}|}$$
$$HScore(T,x) = \frac{2 * Accuracy(T,x) * Precision(T,x)}{Accuracy(T,x) + Precision(T,x)}$$

$ErrDist(t)$はツイートtの実際の位置l_{act}と推定エリアの重心l_{est}との距離（エラー距離），Tは推定対象ツイートの集合，T_{est}は位置を推定できたツイートの集合，xは推定成功とする最大の距離（正解距離）である．なお，T_{est}に関し，位置を推定できなかったツイートとは，含まれる単語のすべてが学習データで出現しなかったツイート，あるいはフィルタリングによってすべての単語が除かれたツイートである．

すでに述べたように，本研究はできるだけ多くのツイートの投稿位置を推定することを目的としているため，3つの評価指標のうち特に正解度を重視する．また，位置情報の推定結果を利用するアプリケーションごとにエラー距離の許容範囲が異なることが想定されるため，評価の際には正解距離を変化させた場合の性能値も確認する．

4.4　関連研究との比較

提案手法の有効性を確認するために，フィルタリング手法とスムージング手法について，関連研究と比較する．

フィルタリング手法については，三木ら[9]が提案したフィルタとChengら[3]が提案したフィルタ，および名詞のみを使用した場合を比較対象とする．三木らが提案したフィルタでは，本研究と同様にTF–IDFの概念を用いており，出現したエリア数が少なく，1つ以上のエリアで多く出現する単語を高く評価する．三木らは単語の時間的局所性を考慮したフィルタリング手法も提案しているが，位置を推定できるツイートが少なくなることから，正解度を重視した実験においては適していないと考えられるため，今回の実験では時間的局所性は考慮しないものとする．Chengらが提案したフィルタでは，単語の分布モデルを構築し，すべての単語についてモデル値を最大化するパラメータを求める．パラメータには，単語の出現の中心となる位置，および距離に比例したペナルティの強さの2つがある．このうち，距離に比例したペナルティの強さが閾値αより小

表 1　獲得したパラメータ — フィルタリング手法
Table 1　Parameters using in test — Filtering methods.

手法	パラメータ	ノイズの割合
AF フィルタ（提案手法）	α=32	1.2%
TF–IAF フィルタ（提案手法）	β=9.56	93.0%
三木らのフィルタ	θ=17	99.0%
Cheng らのフィルタ	α=2.36	3.0%

さいものをノイズとする．この手法には地理的に狭い範囲内に出現した単語を高く評価するという特徴がある．名詞のみを利用するフィルタは，名詞以外の単語のすべてを禁止語とする簡易的な禁止語リストとして機能する．なお，実験の際には本節に記載したフィルタリング手法以外のフィルタリング処理は行わない．

スムージング手法については，Cheng らが提案した 4 つのスムージングのうち，正解度の向上に最も効果があったとされる Lattice-based Neighborhood Smoothing（以降，Lattice スムージング）を比較対象とする．Lattice スムージングでは，まず緯度経度 1 度単位で日本を格子状に区切り，各格子に含まれる市区町村レベルのエリアを求める．次に，それぞれの格子について，単語の出現確率を求め，周囲 8 近傍の格子における単語の出現確率を重み付けして足し合わせる．最後に，格子内のすべてのエリアについて，単語の出現確率を求め，前手順により求めた格子における単語の出現確率を重み付けして足し合わせる．

4.5　パラメータの獲得

すべてのフィルタリング手法およびスムージング手法について，正解距離を 30 km としたときに正解度が最大化するパラメータをデベロップメントデータを用いてそれぞれ獲得する．なお，式中に距離を表すパラメータが導入されているものについては，メートルを単位として距離を計算した．

フィルタリング手法について獲得したパラメータを**表 1** に示す．ノイズの割合とは，学習データに出現した全単語のうち，ノイズとしてフィルタされた単語種類数（タイプ数）の割合である．AF フィルタでは，α が 0 から 400 の範囲でパラメータを探索し，α が 32（約 1.2%の単語をノイズとした）のときに正解度が最大化した．TF–IAF フィルタでは，ノイズの割合が 0%から 100%の範囲で 1%刻みに変化するよう β を探索し，β が約 9.56（約 93.0%の単語をノイズとした）のときに正解度が最大化した．三木らが提案したフィルタでは，出現エリア数（AF 値）を 1 としたときの出現回数の下限値 θ を 0 から 100 の範囲で探索し，θ が 17（約 99.0%の単語をノイズとした）のときに正解度が最大化した．Cheng が提案したフィルタでは，出現エリア数（AF 値）が 1 のときにモデル値 $f(C,\alpha)$ が無限大に発散し，距離に比例したペナルティの強さを表すパラメータ α を獲得することができない．よって，AF が 1 の単語をすべてノイズとする，あるいは AF が 1 の単語をすべてノイズとしないという選択が考えられる．この 2 つの条件についてパラメータを獲得した結果，後者の AF が 1 の単語はすべてノイズとしない場合に正解度が最大化した．パラメータの獲得は，ノイズの割合が 0%から 100%の範囲で 1%刻みに変化するよう α を探索し，α が約 2.36（約 3.0%の単語をノイズとした）のときに正解度が最大化した．

表 2　獲得したパラメータ — スムージング手法
Table 2　Parameters using in test — Smoothing methods.

手法	パラメータ
Distance スムージング（提案手法）	$\lambda=4\times10^{-5}$
Lattice スムージング	μ=0.7, λ=0.6

スムージング手法について獲得したパラメータを**表 2** に示す．Distance スムージングでは，距離による影響の強さ λ を $\{1, 0.1, 0.01, \ldots, 1\times10^{-10}\}$ と変化させ，正解度が最大となった 1×10^{-5} 周辺の $\{1\times10^{-6}, 2\times10^{-6}, \ldots, 1\times10^{-5}, 2\times10^{-5}, \ldots, 9\times10^{-5}\}$ の範囲でパラメータを探索した．結果として，λ が 4×10^{-5} のときに正解度が最大化した．Lattice スムージングでは，スムージングの強さ μ と λ をそれぞれ $\{0.1, 0.2, \ldots, 0.9, 1.0\}$ の範囲で変化させ，μ が 0.7，λ が 0.6 のときに正解度が最大化した．

5.　実験結果・考察

5.1　フィルタリング手法の比較

正解距離を 10 km から 100 km まで変化させた場合の正解度，適合率，H 値を比較した結果を**図 1**，**図 2**，**図 3** に示す．すべての評価で AF フィルタが最高性能を示した．正解距離 30 km においては正解度約 0.43，適合率約 0.79，H 値約 0.56 であった．正解度をみると，従来手法である Cheng らのフィルタリング手法よりも約 0.04 ポイント向上している．正解距離 30 km において符号検定を行ったところ，有意水準 1%で AF フィルタが他のフィルタよりも正解度が高いことを確認した．

本研究で重要となる正解度において，AF フィルタが最高性能を示したことや Cheng らが提案した手法で AF が 1 の単語を含めた方が正解度が高くなったことから，単純に AF 値が低い単語が正解度の向上に大きな影響を与えていると考えられる．三木らが提案したフィルタの正解度がフィルタなしの場合と比べてほとんど変化していない理由は，TF 値を大きく評価しすぎてしまっているためだと考えられる．つまり，ツイートに多く含まれる単語はノイズとして分類されずにそのまま残るため，フィルタなしの場合とほとんど結果が変わらなかったと考えられる．同様に TF–IAF フィルタでも TF を高く評価しすぎている可能性があるため，TF 値と IAF 値の重みを調整するパラメータを導入することで正解度が向上する可能性がある．名詞の

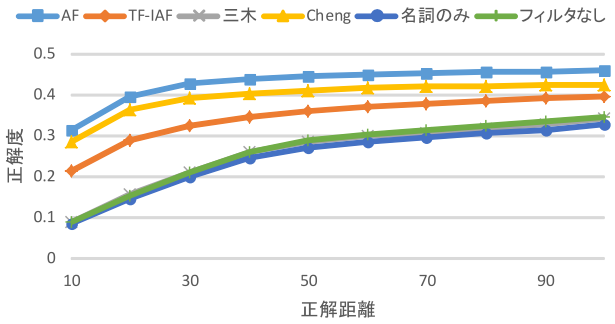


図 1　フィルタリング手法の比較 — 正解度
Fig. 1　Comparison of filtering methods — Accuracy.

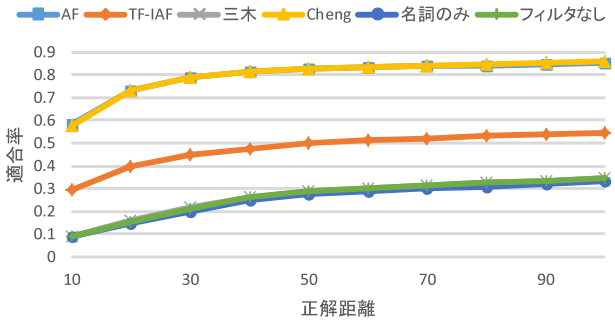


図 2　フィルタリング手法の比較 — 適合率
Fig. 2　Comparison of filtering methods — Precision.

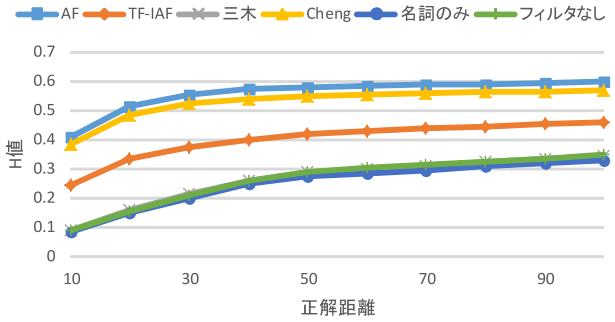


図 3　フィルタリング手法の比較 — H 値
Fig. 3　Comparison of filtering methods — HScore.

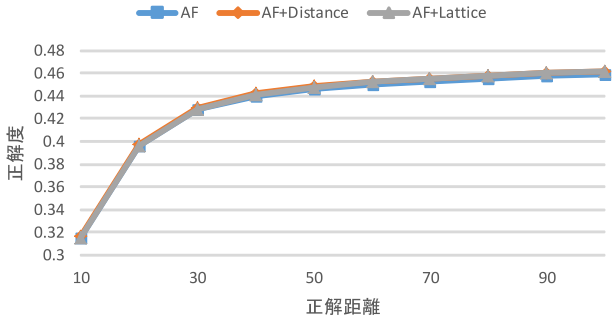


図 4　スムージング手法の比較 — 正解度
Fig. 4　Comparison of smoothing methods — Accuracy.

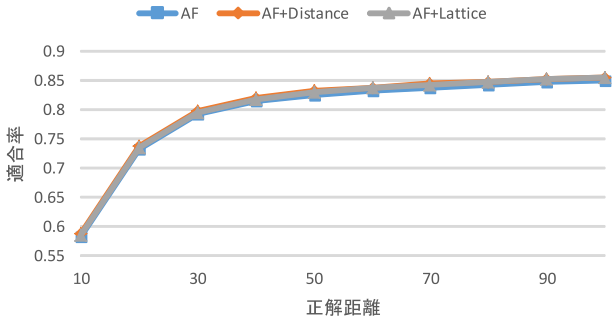


図 5　スムージング手法の比較 — 適合率
Fig. 5　Comparison of smoothing methods — Precision.

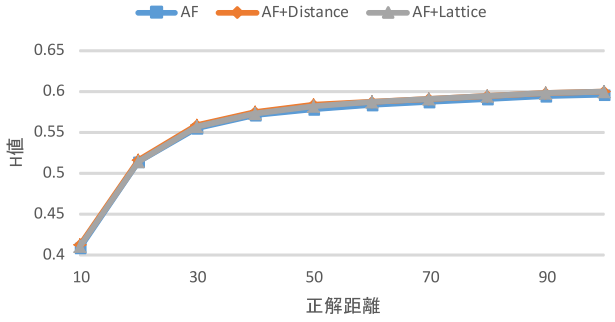


図 6　スムージング手法の比較 — H 値
Fig. 6　Comparison of smoothing methods — HScore.

みを用いた場合には，正解度が悪化しており，単純な禁止語リストでは正解度が向上しないことが明らかになった．

本研究では正解度を向上させる手法を提案することを目的としているため，以降の実験では正解度において最も高性能を達成した AF フィルタを用いる．

5.2　スムージング手法の比較

AF フィルタによってノイズとなる単語を除き，正解距離を 10 km から 100 km まで変化させた場合の正解度，適合率，H 値を比較した結果を**図 4**，**図 5**，**図 6** に示す．正解度，適合率，H 値の 3 つすべてにおいて，正解距離 100 km でのみ Lattice スムージングが最高性能を示し，それ以外の距離では Distance スムージングが最高性能を示している．正解距離 30 km における Distance スムージングの性能は，正解度約 0.43，適合率約 0.80，H 値約 0.56 であった．正解距離 30 km において符号検定を行ったところ，有意水準 1%で Distance スムージングが Lattice スムージングよりも正解度が高いことを確認した．

本節では，AF フィルタによってノイズを除いた後にスムージングを行ったが，予備実験により，フィルタリング手法を適用しない場合でも正解度の向上を確認した．

5.3　都道府県別の正解度

都道府県ごとにツイート数やエリアの面積が異なるため，都道府県別の正解度を比較する．ここにおける正解度は，各都道府県別のツイート数を分母とし，その中で推定に成功したツイートの割合である．また，地理的に分割した結果はスムージングの特徴も見ることができるため，同時にスムージング手法も比較する．正解距離を 10 km としたときの正解度を**図 7** に示す．このときのマクロ平均は，

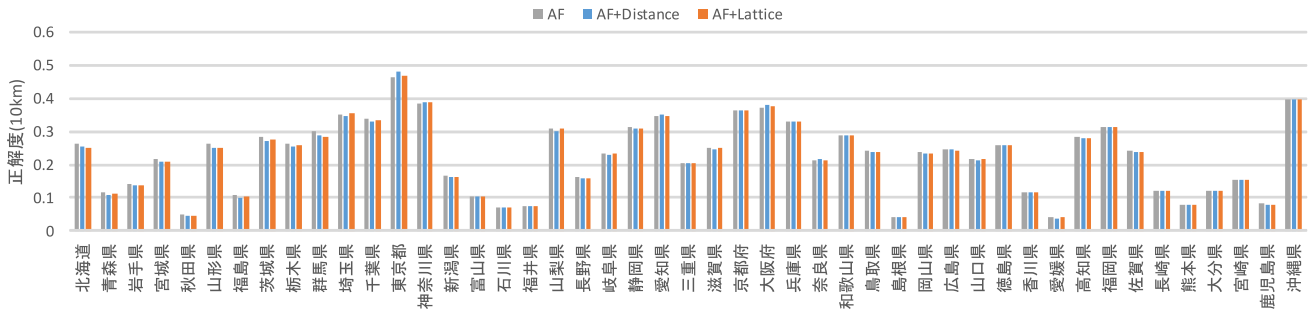


図 7　都道府県別の正解度（正解距離 10 km）

Fig. 7　Accuracy for each prefecture (ErrDist within 10 km).

すべての手法において約 0.22 であった．全体的な傾向としては，AF フィルタのみの場合に正解度が高くなり，関東圏では Lattice スムージングの正解度が高くなった．ただし，東京都においては Distance スムージングの正解度が最大となった．

東京都の正解度が特に高いことについては，東京都で投稿されたツイート数が多いことが原因と考えられる．ツイート数が多ければ，東京都で出現する単語の種類も多くなり，東京都に推定されやすくなっている可能性がある．

埼玉県，千葉県，東京都，神奈川県で Lattice スムージングが高い性能を示したことについては，この 4 都県が Lattice スムージングにおける同じ格子（東経 139 度–140 度，北緯 35 度–36 度）に属していることが関係していると考えられる．この格子は東京都の大部分を含んでいるため，東京都における単語の出現頻度の影響を大きく受ける．前段落で述べた理由で東京都に推定されやすくなっているのであれば，Lattice スムージングによって埼玉県，千葉県，神奈川県にも推定されやすくなっている可能性がある．

東京都で Distance スムージングが最大となっていることについては，東京都の面積が小さいことが原因であると考えられる．Distance スムージングでは距離が近いエリアの影響を大きく受けるため，面積が小さい都道府県ではエリア間の距離が小さくなり，相対的にスムージングの影響を大きく受ける．東京都における単語の出現頻度が高い状況で，Distance スムージングによって東京都内のエリアの出現頻度がさらに高くなり，より東京に推定されやすくなっていると考えられる．

地方の正解度が下がっているのは，これまでに述べた理由によって東京都周辺に推定されやすくなり，地方へ推定される可能性が下がっているためだと考えられる．つまり，地方においてスムージングの効果がないわけではなく，東京都に対する影響が特に強いために，相対的に正解度が低下したと考えられる．

5.4　時刻別の正解度

AF フィルタによってノイズとなる単語を除き，正解距離を 30 km としたときの時刻別の正解度を図 **8** に示す．こ

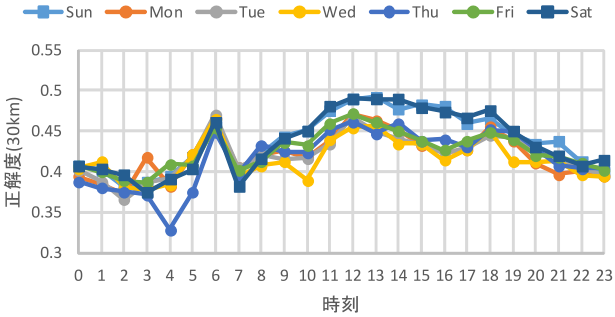


図 8　時刻別の正解度（正解距離 30 km）

Fig. 8　Accuracy for each time (ErrDist within 30 km).

こにおける正解度は，各時間別のツイート数を分母とし，その中で推定に成功したツイートの割合である．曜日によって人の移動パターンが異なる可能性があるため，曜日ごとに正解度を比較する．土曜と日曜においては，平日と比較すると，日中の正解度が高くなるという傾向が見られる．マクロ平均は土曜と日曜が約 0.44，金曜が約 0.43，それ以外が約 0.42 となった．夜間は自宅でツイートが投稿されると予想されるため，実際にツイートの投稿位置を推定する必要があるのは日中であると考えることもできる．6 時から 19 時の 13 時間についてのマクロ平均は土曜と日曜が約 0.46，月曜と木曜と金曜が約 0.44，火曜と水曜が約 0.43 となり，日中についてはより正確に推定できている．

18 時付近で正解度が高くなっていることから，通勤の時間帯において正解度が高くなる可能性が考えられる．これは，通勤の移動中に行われるツイートに駅名などが含まれる可能性が高くなっているためだと考えられる．

5.5　単語数と正解度の関係

フィルタリングによってノイズとなった単語の割合と正解度の関係を図 **9** に示す．なお，この結果はデベロップメントデータを用いてパラメータを獲得した際のものである．AF フィルタと Cheng らが提案したフィルタでは，ほとんどの単語をノイズとしない場合に正解度が最大化し，TF–IAF フィルタでは多くの単語をノイズとした場合に正解度が最大化している．三木らのフィルタでは，単語数による変化は小さい．

表 3　ロケーションサービスツイートの有無による性能比較（正解距離 30 km）
Table 3　Results in having location service tweets or not (ErrDist within 30 km).

	正解度		適合率		H 値	
	なし	あり	なし	あり	なし	あり
AF フィルタ（提案手法）	0.39	0.43	0.77	0.79	0.51	0.56
TF–IAF フィルタ（提案手法）	0.28	0.32	0.40	0.45	0.33	0.38
三木らのフィルタ	0.19	0.21	0.20	0.21	0.19	0.21
Cheng らのフィルタ	0.35	0.39	0.77	0.79	0.48	0.52
名詞のみ	0.18	0.20	0.18	0.20	0.18	0.20
フィルタなし	0.19	0.21	0.19	0.21	0.19	0.21

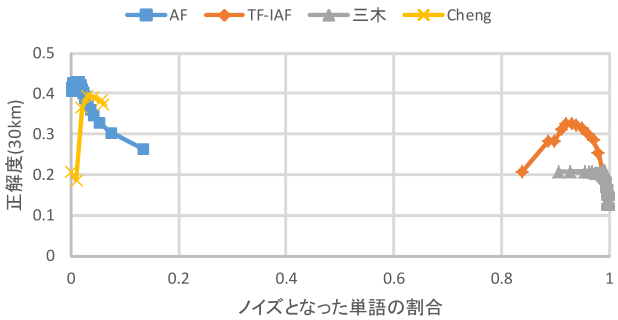


図 9　ノイズとなった単語の割合と正解度の関係
Fig. 9　Relationship of stop word ratio and estimation accuracy.

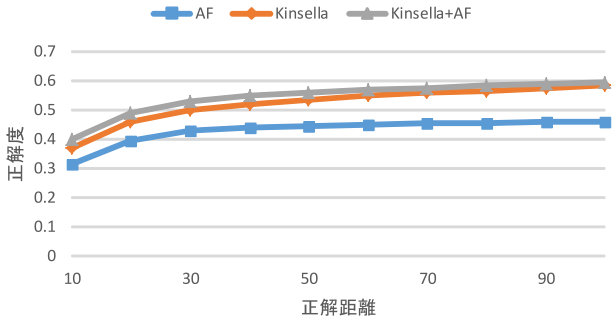


図 10　推定式の比較 — 正解度
Fig. 10　Comparison of estimated formulae — Accuracy.

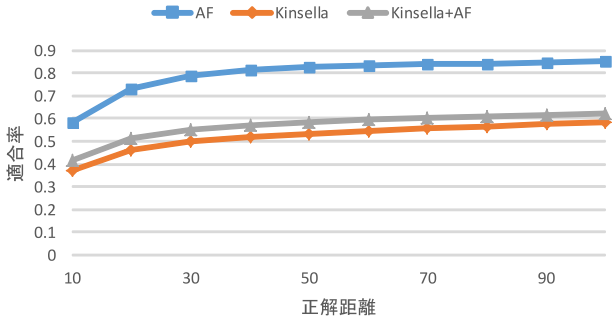


図 11　推定式の比較 — 適合率
Fig. 11　Comparison of estimated formulae — Precision.

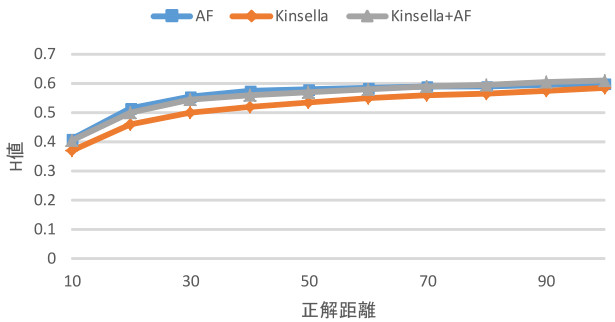


図 12　推定式の比較 — H 値
Fig. 12　Comparison of estimated formulae — HScore.

正解度を最大化するフィルタリング手法は AF フィルタとなったが，TF–IAF フィルタでは約 93%もの単語をフィルタリングした場合でも正解度を向上させている．このことから，正解度を最優先とする場合には AF フィルタ，多くの単語を除去しつつも正解度を維持したい場合には TF–IAF フィルタが適しているといえる．

5.6　ロケーションサービスを除外した場合の評価

4.2 節に示したテストデータには，Foursquare などのロケーションサービスからのツイートが含まれる．これらのツイートは「I'm at 豊橋駅 (Toyohashi Sta.) in 豊橋市，愛知県」のようにツイート本文から投稿位置が明らかであったり，ロケーションサービス内の地理データベースとの照合が可能であったりする．そのため，実運用を想定する場合には，これらのツイートを除外した評価も必要である．

Twitter クライアント名に「foursquare」「loctouch」を含むツイートをロケーションサービスからのツイートであると見なし[*10]，それらのツイート 148,443 件（7.2%）を除外した場合の性能を評価した．ロケーションサービスからのツイートの有無と性能（正解距離 30 km）との関係を**表 3** に示す．ロケーションサービスからのツイートを除外した場合（表 3 の「なし」），除外しない場合（表 3 の「あり」）よりも全体的に性能の低下が認められる．除外した場合であっても，提案した AF フィルタが正解度および H 値で最高性能を示し，適合率に関しても従来手法と同等の性能を示している．正解距離 30 km において符号検定を行ったところ，有意水準 1%で AF フィルタが他のフィルタよりも正解度が高いことを確認した．これらより，容易に推定可能であると考えられる一部のツイートを除外した状況下であっても，AF フィルタが有効に機能することが明らかになった．

[*10]「loctouch」はロケタッチ（http://tou.ch/）からのツイートであるが，2015 年 6 月 30 日 12 時にサービスが終了した．

5.7　異なる推定式での評価

5.1 節では式 (1) に対して単語のフィルタリングが有効に機能するかどうかを評価した．本節では，ほかの推定式においても提案手法（AF フィルタ）が有効に機能するかどうかを評価するために，Kinsella ら [13] が提案した query likelihood Language Model with Dirichlet smoothing と比較する．

Kinsella らの提案手法にはパラメータ μ が存在するため，4.5 節でのパラメータ獲得方法と同様に μ を $\{1, 5, 10, 50, 100, 500, \ldots, 5 \times 10^{10}\}$ と変化させたところ，$\mu = 1$ で正解度が最大化した．また，Kinsella らの提案手法に AF フィルタを組み合わせたもの（Kinsella+AF）については，$\mu = 1$ において，α を 0 から 800 と 1300 から 1900 とのそれぞれを 10 刻みで探索し，$\alpha = 1470$ で正解度が最大化した．本節では，これらのパラメータを利用して性能を評価した．

本研究で用いた式 (1) に AF フィルタを適用した手法（AF），Kinsella ら [13] らの提案手法（Kinsella），Kinsella らの提案手法に AF フィルタを適用した手法（Kinsella+AF）を比較した結果を**図 10**，**図 11**，**図 12** に示す．正解距離 30 km において，正解度は Kinsella+AF が最高性能を示し約 0.53，適合率と H 値は AF が最高性能を示し，それぞれ約 0.79 と約 0.54 であった．正解距離 30 km において符号検定を行ったところ，有意水準 1%で Kinsella+AF が他の手法よりも正解度が高いことを確認した．実験により，AF フィルタは推定式によらず正解度を向上させることが明らかになった．

6.　おわりに

本研究では，単語の出現頻度を学習し，推定対象とする 1 件のツイート内容のみから，より多くのツイートに正確な位置情報を付与する問題に取り組んだ．さらに，ツイート中からノイズとなる単語を取り除く AF フィルタと TF–IAF フィルタ，および単語の地理的分布を平滑化する Distance スムージングを提案した．フィルタリング手法に関し，正解度について AF フィルタが最高性能を示したことから，単純に AF 値が低い単語が正解度の向上に大きな影響を与えていることが示唆された．TF–IAF フィルタは，多くの単語をノイズとした場合にも正解度を大きく向上させていることから，単語数を減らしつつも位置情報を推定したい場合に有効に機能する．スムージング手法に関し，Distance スムージングは従来手法よりも有意に正解度を向上させる効果があることを確認した．

今後の課題として，1 件のツイート内容のみではなく，ユーザの情報やツイート間の時間的関係を考慮した推定を行うことがあげられる．使用できる情報が増えることで，本質的に推定を行うことが困難なツイートや単語数が少ないツイートに対しても，より正確な投稿位置推定を行えると考えられる．

参考文献

[1] 榊　剛史，松尾　豊：ソーシャルセンサとしての Twitter–ソーシャルセンサは物理センサを凌駕するか?–，人工知能学会誌，Vol.27, No.1, pp.67–74 (2012).

[2] 吉次由美：東日本大震災に見る大災害時のソーシャルメディアの役割–ツイッターを中心に–，放送研究と調査，Vol.61, No.7, pp.16–23 (2011).

[3] Cheng, Z., Caverlee, J. and Lee, K.: You Are Where You Tweet: A Content-Based Approach to Geo-locating Twitter Users, *Proc. 19th ACM International Conference on Information and Knowledge Management*, pp.759–768 (2010).

[4] 橋本康弘，岡　瑞起：都市におけるジオタグ付きツイートの統計，人工知能学会誌，Vol.27, No.4, pp.424–431 (2012).

[5] Aramaki, E., Maskawa, S. and Morita, M.: Twitter Catches The Flu: Detecting Influenza Epidemics using Twitter, *Proc. 2011 Conference on Emperical Methods in Natural Language Processing*, pp.1568–1576 (2011).

[6] Sakaki, T., Okazaki, M. and Matsuo, Y.: Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors, *Proc. 19th International Conference on World Wide Web*, pp.851–860 (2010).

[7] Jurgens, D., Finethy, T., Mccorriston, J., Xu, Y.T. and Ruths, D.: Geolocation Prediction in Twitter Using Social Networks: A Critical Analysis and Review of Current Practice, *Proc. 9th International AAAI Conference on Web and Social Media*, pp.188–197 (2015).

[8] Schulz, A., Hadjakos, A., Paulheim, H., Nachtwey, J. and Mühlhäuser, M.: A Multi-Indicator Approach for Geolocalization of Tweets, *Proc. 7th International AAAI Conference on Weblogs and Social Media*, pp.573–582 (2013).

[9] 三木翔平，新田直子，馬場口登：単語の地理的局所性の経時変化を考慮したツイートの発信位置推定，第 6 回データ工学と情報マネジメントに関するフォーラム (2014).

[10] Salton, G. and McGill, M.J.: *Introduction to Modern Information Retrieval* (1984).

[11] Han, B., Cook, P. and Baldwin, T.: Geolocation Prediction in Social Media Data by Finding Location Indicative Words, *Proc. International Conference on Computational Linguistics 2012*, pp.1045–1062 (2012).

[12] Jones, K.S.: Index Term Weighting, *Information Storage and Retrieval*, Vol.9, No.11, pp.619–633 (1973).

[13] Kinsella, S., Murdock, V. and O'Hare, N.: "I'm Eating a Sandwich in Glasgow": Modeling Locations with Tweets, *Proc. 3rd International Workshop on Search and Mining User-Generated Contents*, pp.61–68 (2011).

[14] Roller, S., Speriosu, M., Rallapalli, S., Wing, B. and Baldridge, J.: Supervised Text-based Geolocation Using Language Models on an Adaptive Grid, *Proc. 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp.1500–1510 (2012).

[15] Yamaguchi, Y., Amagasa, T., Kitagawa, H. and Ikawa, Y.: Online User Location Inference Exploiting Spatiotemporal Correlations in Social Streams, *Proc. 23rd ACM International Conference on Conference on Information and Knowledge Management*, pp.1139–1148 (2014).

[16] 伊川洋平，榎　美紀，立堀道昭：マイクロブログのメッセージを用いた発信場所推定，第 4 回データ工学と情報マネジメントに関するフォーラム (2012).

[17] Wang, D., Pedreschi, D., Song, C. and Giannotti, F.: Human Mobility, Social Ties, and Link Prediction,

Proc. 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp.100–1108 (2011).
[18] Backstrom, L., Sun, E. and Marlow, C.: Find Me If You Can: Improving Geographical Prediction with Social and Spatial Proximity, *Proc. 19th International Conference on World Wide Web*, pp.61–70 (2010).
[19] Rout, D., Bontcheva, K., Preoţiuc-Pietro, D. and Cohn, T.: Where's @wally? A Classification Approach to Geolocating Users Based on their Social Ties, *Proc. 24th ACM Conference on Hypertext and Social Media*, pp.11–20 (2013).
[20] Sadilek, A., Kautz, H. and Bigham, J.P.: Finding Your Friends and Following Them to Where You Are, *Proc. 5th ACM International Conference on Web Search and Data Mining*, pp.723–732 (2012).
[21] Li, R., Wang, S., Deng, H., Wang, R. and Chang, K.C.-C.: Towards Social User Profiling: Unified and Discriminative Influence Model for Inferring Home Locations, *Proc. 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.1023–1031 (2012).
[22] Li, R., Wang, S. and Chang, K.C.-C.: Multiple Location Profiling for Users and Relationships from Social Network and Content, *Proc. International Conference on Very Large Databases Endowment*, Vol.5, No.11, pp.1603–1614 (2012).
[23] Cheng, Z., Caverlee, J., Lee, K. and Sui, D.Z.: Exploring Millions of Footprints in Location Sharing Services, *Proc. 5th International AAAI Conference on Weblogs and Social Media*, pp.81–88 (2011).
[24] Kamath, K.Y., Caverlee, J., Lee, K. and Cheng, Z.: Spatio-Temporal Dynamics of Online Memes: A Study of Geo-Tagged Tweets, *Proc. 22nd International Conference on World Wide Web*, pp.667–678 (2013).
[25] Flatow, D., Naaman, M., Xie, K.E., Volkovich, Y. and Kanza, Y.: On the Accuracy of Hyper-local Geotagging of Social Media Content, *Proc. 8th ACM International Conference on Web Search and Data Mining*, pp.127–136 (2015).
[26] Watanabe, K., Ochi, M., Okabe, M. and Onai, R.: Jasmine: A Real-time Local-event Detection System Based on Geolocation Information Propagated to Microblogs, *Proc. 20th ACM International Conference on Information and Knowledge Management*, pp.2541–2544 (2011).
[27] Morstatter, F., Pfeffer, J., Liu, H. and Carley, K.M.: Is the Sample Good Enough? Comparing Data from Twitter's Streaming API with Twitter's Firehose, *Proc. 7th International AAAI Conference on Weblogs and Social Media*, pp.400–408 (2013).

森國 泰平

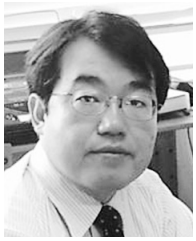
2015 年豊橋技術科学大学工学部情報・知能工学課程卒業．同年同大学院工学研究科情報・知能工学専攻博士前期課程進学．

吉田 光男　（正会員）

2011 年筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻博士前期課程修了，2014 年同博士後期課程修了．博士（工学）．同年豊橋技術科学大学大学院工学研究科（情報・知能工学系）助教．ウェブ工学，自然言語処理，情報検索に関する研究に従事．言語処理学会，人工知能学会，日本データベース学会各会員．

岡部 正幸

2001 年東京工業大学大学院総合理工学研究科知能システム科学専攻博士課程修了．博士（工学）．同年科学技術振興機構（CREST）研究員．2003 年豊橋技術科学大学情報メディア基盤センター助手，2007 年同助教．知的情報検索，インタラクティブデータマイニングに関する研究に従事．人工知能学会会員．



梅村 恭司　（正会員）

1983 年東京大学大学院工学系研究科情報工学専攻修士課程修了．博士（工学）．同年日本電信電話公社電気通信研究所入所．1995 年豊橋技術科学大学工学部情報工学系助教授，2003 年同教授．自然言語処理，システムプログラム，記号処理に関する研究に従事．情報電子通信学会，日本ソフトウェア科学会，言語処理学会，計量国語学会，ACM 各会員．

（担当編集委員　岡崎 直観）