Vol.2013-DBS-157 No.7
Vol.2013-IFAT-111 No.7
2013/7/22

マイクロブログを対象とした 1,000 人レベルでの著者推定手法構築に向けて

奥野峻弥[†1]　浅井洋樹[†2]　山名早人[†3]

従来，著者推定研究は小説に対する著者推定を中心に研究が行われており，限定された人数の著者候補者群を取り扱ってきた．またこれまでに，インターネットに投稿された文章を対象に 1 万人レベルでの著者推定手法を提案し，8 割程度の精度を得ている．しかし，多数のユーザが存在する，マイクロブログに投稿されるメッセージは，投稿数は多いが一度に投稿される文章量が短く，未知語や誤字脱字が多いという特徴が存在するため，これまでの手法では精度が低下してしまう．そこで，本研究ではメッセージから辞書を作成し，その辞書を用いた形態素解析器を利用することで少数のメッセージを利用した大規模人数に対する著者推定を行う手法を提案する．900 人の候補者から著者を推定する評価実験を行った結果，既存の著者推定手法よりも精度が上昇することが確認できた．

1. はじめに

既存の著者推定手法[1][2][3]は小説などの文学的文章における著者推定を実現し，近年インターネットに投稿された日本語の文章に対して著者推定が応用[4][5][6]されている．このような文章の著者を推定する際には，大規模人数の著者候補者群に対して著者を推定する必要がある．なぜならば，インターネットに文章を投稿する著者は不特定多数であり，少人数に限定できないためである．そこで，われわれはこれまでに 10,000 人レベルでの著者推定手法を提案してきた．

しかし，これまでに提案してきた大規模候補者群に対する著者推定手法[7][8]をそのまま用いて，インターネット上の文章の一種であるマイクロブログのデータに対する著者推定を行うと，推定精度が低下する．原因としては，以下の 2 つの問題が考えられる．

1. 推定対象文章の短文化
2. マイクロブログ特有の単語・表現の頻出

1 つ目は，著者推定で用いる候補者毎の文章が，これまでの研究で用いてきた文章と異なり非常に短いという問題である．これは，マイクロブログに投稿される一回あたりの文章が短文であることに起因する．実際，twitter[a]に文章を投稿する際は，140 文字の文字数制限が存在する．長文が投稿できないため，マイクロブログに投稿される文章は，文法の整った文章だけでなく，マイクロブログ特有の文法を持った文章も多く存在する．それゆえに，これまでの研究において，著者推定の対象としてきた文章とは，文体や品詞分布が大きく異なる．そのため，これまでのわれわれの研究と同じ文体定量化方法を用いても，文章間の類似度を正確に計算できなくなるため，著者推定精度が低下する．

2 つ目は，マイクロブログに投稿される文章中に，既存の辞書には存在しない単語（未知語）が多く存在している[8]ことに加え，叫喚フレーズ[9]などの一般的な文章にはみられない表現が多数存在しているという問題である．この問題により，マイクロブログに投稿される文章に対する，既存の辞書を用いた形態素解析器の精度が低下する．形態素解析器の精度が低下することで，文章中の未知語の比率が高まり，文章間の類似度を正確に計算できなくなるため，著者推定精度が低下する．

本稿では，上記 2 つの問題に対応した著者推定手法を提案する．具体的には，文章間の類似度を計算する際に用いる素性の変更及び，推定の際に用いる文章に対して，形態素解析器で用いる辞書として，WEB サービスの単語リストやマイクロブログ特有の表現を収録したキーワードリストを利用することで，既存手法よりも高精度の著者推定を実現する．

本稿では以下の構成をとる．まず 2 節では，著者推定研究で取り扱われてきた著者推定タスクについて述べる．次の 3 節では，既存の著者推定手法について述べる．続く 4 節では，本稿で提案する著者推定手法について述べる．そして，5 節にて既存手法と提案手法とに対する評価実験の方法と結果について述べる．最後に 6 節で本稿をまとめる．

2. 著者推定タスク

著者推定とは，推定対象文章における文体の特徴から，その文章の著者を推定することである．推定対象文章とは，著者を推定する対象となる，著者不明の文章のことである．なお，本稿では日本語の推定対象文章を対象とした著者推定を取り扱う．また，本節で説明する著者推定タスクとは，数多く存在する著者推定手法を抽象化したものである．

従来，著者推定は文学研究[10][11][12]で行われてきたが，近年ではテキストマイニング技術を用いた著者推定の手法[1][3][6]が提案されている．これらの手法は計算機上で容易に実装可能であることから，インターネットに投稿された文章の著者推定に応用[4][5][7][8]されている．計算機によるテキストマイニングによる著者推定手法の研究では，著者推定手法をもって著者推定タスクを行い，この結果によって当手法の評価を行う．本節では，著者推定タスクに

†1 早稲田大学大学院 基幹理工学研究科
†2 早稲田大学大学院 基幹理工学研究科
早稲田大学メディアネットワークセンター
†3 早稲田大学理工学術院，国立情報学研究所

a Twitter, https://twitter.com/

ついて，その内容と結果からの評価方法について述べていく．

2.1 著者推定タスクの分類

Stamatatos[13] は著者推定タスクを Profiled-Based Approach（PBA）と Instance-Based Approach（IBA）の 2 種類に分類した．本稿では，大規模候補者群に対する著者推定を行うため，PBA による著者推定タスクを取り扱う．これは，大規模候補者群に対する著者推定では，IBA による著者推定タスクに 2.1.2 項で示す問題があるためである．

2.1.1 PBA 及び IBA による著者推定タスク

PBA による著者推定タスクでは，事前に用意されている候補者の文章群と，推定対象文章を順に比較する．比較された候補者群の中から，推定対象文章の著者と文体が最も類似する候補者を得ることで，各著者推定手法は著者推定を行う．PBA に分類される著者推定タスクは，松浦ら[1], 安形ら[2], 中島ら[6], 及び井上ら[7][8]が取り扱っている．

一方で，IBA による著者推定タスクでは，機械学習により各候補者の文章群を学習し，推定対象文章を各候補者のいずれかに分類する．推定対象文章の分類先となる候補者を得ることで，各著者推定手法は著者推定を行う．IBA に分類される著者推定タスクは，金ら[3]，坪井ら[14]が取り扱っている．

2.1.2 IBA による著者推定タスクの問題点

大規模候補者群に対する著者推定における IBA の著者推定タスクでは，当該著者推定タスクにおける機械学習が上手く機能しない．これは，IBA の著者推定タスクで用いる学習データが不均衡データであるために起こる[15]．不均衡データとは，正例と負例の数に極端な差がある学習データを指す．IBA における著者推定タスクでは，学習データ中の文章群を，特定の 1 人の候補者の文章である正例，それ以外の複数候補者の文章である負例の 2 つに分類する．しかし，一般に負例を集めることは容易であるが，正例を多く集めることは困難である．このため， IBA における著者推定タスクでは，正例と負例の数に差が生まれ，学習データは不均衡データとなる．

不均衡データに対処するため，正例の数に合わせて負例の数を減らす，負例の数に合わせて正例の数を多くするといった対策が考えられる．しかし，前者の方法では学習が十分にできない問題が生じる．一方，後者の方法を講じることも難しい．これは，候補者ごとに集められる文章は数万文字の大量文章でなくてはならないが，マイクロブログを対象とした大規模候補者群においてこのような文章を 1 人の候補者に対し多く集めることは困難であるためである．

2.2 PBA による著者推定タスクの流れ

手順 1）学習データとテストデータの収集

学習データとは，著者が既知である文章群のことを指す．テストデータとは複数の推定対象文章を指す．ただし，著者推定タスクでは，推定したテストデータ中の文章の著者と実際の著者が同じであることを確かめるため，テストデータ中の文章の著者が既知であるものを用いる．また，テストデータ中の文章の著者は，学習データにおけるいずれかの文章の著者と同一であるとする．このような条件の下，著者推定の候補者群となる著者を決定した後，候補者ごとに学習データとテストデータの 2 種類の文章を取集する．

手順 2) 各文章の文体定量化

手順 1 で収集された学習データ及びテストデータ中のすべての文章に対して文体定量化を行う．文章の文体定量化とは，その文章の著者が持つ文体を，当該文章を用いて数値ベクトルに定量化することである．文体の定量化方法は，各著者推定手法によって異なる．

手順 3）各文章間の文体相違度計算

テストデータ中の文章ごとに，学習データ中の各文章との間の文体相違度をすべて計算する．2 つの文章間の文体相違度とは，各文章の著者の文体がどれほど異なるかを定量化したものである．2 つの文章間の文体相違度は，手順 2 で得られる定量化された文体を用いて算出される．文体相違度をどのように算出するかは，各著者推定手法によって異なる．

手順 4）文体類似度順位の算出

テストデータ中の文章ごとに文体類似度順位を算出する．文体類似度順位とは，文体相違度の低い順に候補者群を並び替えたとき，推定対象文章の著者が何位に順位付けされたかを表す．

手順 5）著者推定手法の評価

手順 4 で得られたテストデータ中の各文章に対する文体類似度順位に基づいて，手順 2 及び手順 3 で用いた著者推定手法の評価を行う．得られた文体類似度順位からどのように著者推定手法を評価するかは，著者推定手法評価方法によって異なる．

2.3 評価方法

既存の著者推定研究[1][2][3][5][6][8]で行われる著者推定手法の評価は，2.2 項で述べた著者推定タスクの手順 5 において，テストデータ中の文章群の中で文体類似度順位が 1 位となる文章の割合である，PRECISION@1 を指標として評価を行ってきた．これは，テストデータ中の各文章に対して著者推定を行うとき，著者推定タスクの手順 4 で並び替えられる候補者群において 1 位となる候補者を推定対象文章の著者であると推定するためである．

井上ら[7]は大規模候補者群に対する著者推定手法評価方法として，文体類似度順位の累積相対度数分布を定量的に評価する MRR 及び，正解が上位 k 件以内に入っていれば 1 と, そうでなければ 0 としてその平均をとる mean top-k call を評価方法として用いた．具体的には，MRR については式(1)によって算出される．ここで，Q はテストデータ中の文章の著者の集合，N_qは出力される候補者群順列中にお

ける候補者の順位である．

$$MRR = \frac{1}{|Q|}\sum_{q \in Q}\frac{1}{N_q} \quad (1)$$

井上らが MRR 及び mean top-k call による評価方法を用いたのは，著者推定タスクにおける候補者群の並び替えにおいて，実際の著者が 1 位に順位付けされているかだけでなく，上位に順位付けされているかを評価するためである．これは，誤った推定をしない著者推定手法が存在しない以上，推定結果を実用するためには複数の候補から人手によって選択することが要求されるためである．特に，推定精度低下が顕著となる大規模候補者群に対する著者推定では，人手による確認が要求される．人手による推定を行う際は，複数の推定結果から著者を精査することで，正しい著者推定を行うことができる．しかし，そのためには 2 位以降の上位に正解が含まれていなければならない．よって，大規模候補者群に対する著者推定の評価には，MRR による評価方法が適しているといえる．

3. 従来の著者推定手法

3.1 大規模候補者群に対する著者推定手法

井上ら[7]はインターネット上の文章を用いた大規模候補者群に対する著者推定手法を提案した．井上らは，文体定量化の際に品詞タグ・文字混合 n-gram 頻度分布を用いた．ここで，品詞タグ・文字混合 n-gram とは，文章を文字または品詞タグの羅列に変換したときに，当該羅列中に存在する n 個の連続した要素順列を指す．

井上らが提案する文章中の文体定量化は，文章p中における品詞タグ・文字混合 n-gram xの生起回数d_{px}の集合D_pを得ることで行う．文章を文字または品詞タグの羅列に変換するために以下の手順をとる．まず，形態素解析器を用いて文章を形態素に分割する．なお，形態素解析器は Sen[16]を用いている．次に，「動詞」「接続詞」「記号」「副詞」「形容詞」「感動詞」の形態素については，文字列をそのまま採用し，これら 6 種類の品詞以外について品詞タグを用いる．

井上らが提案する著者推定タスクにおける文体相違度計算では，文章p,qについてのD_pだけではなく，C_{pq}, a_pを用いる．C_{pq}は，文章pと文章qの各々に存在するすべての品詞タグ・文字混合 n-gram の和集合である．a_pは，文章pを構成する記事の数である．記事とは，電子掲示板における 1 つの記事や，1 件の電子メールのように，一度に投稿する文のまとまりを指す．井上らは，C_{pq}, D_p, a_pを用いることで 2 つの文章p,qにおける文体相違度$Dissim_{pos}$を以下のように定義している．

$$Dissim_{pos}(p,q) = \frac{\sqrt{\sum_{i \in C_{pq}}(f_{pi} - \overline{f_{pq}})^2}\sqrt{\sum_{i \in C_{pq}}(f_{qi} - \overline{f_{qp}})^2}}{\sum_{i \in C_{pq}}(f_{pi} - \overline{f_{pq}})(f_{qi} - \overline{f_{qp}})} \quad (2)$$

$$\overline{f_{pq}} = \frac{\sum_{i \in C_{pq}} f_{pi}}{|C_{pq}|} \quad (3)$$

$$f_{pi} = \begin{cases} 0.4 & (f'_{pi} > 0.4) \\ f'_{pi} & (f'_{pi} \leq 0.4) \end{cases} \quad (4)$$

$$f'_{pi} = \frac{d_{pi}}{a_p} \quad (5)$$

文体相違度$Dissim_{pos}$は，その値が小さいほど 2 つの文章p,qの文体が似ていることを表す．

3.2 既存手法の問題点

エラー! 参照元が見つかりません。項で説明した，井上ら[7]の大規模候補者群に対する著者推定手法を用いてマイクロブログのデータを用いた大規模候補者群に対する著者推定を行った場合，著者推定精度が大きく低下する．それは，井上らの手法で対象としているデータセットと，本稿で対象としているデータセットの違いに起因するものである．

井上らの手法で対象としているデータはインターネット上の文章を用いたデータであるが，最低でも 500 文字以上の文章量がある記事のみを，テストデータ及び学習データの作成に用いている．しかし，Twitter などの多くのマイクロブログの文章は，一度に投稿できる文書量に制限があり，また着想から投稿までのタイムラグが短いことから，短文かつ整合性のない文章が多い．そのため，他のデータセットと比べても，使用される品詞の分布が大きく異なる．それにより，文字列として採用する品詞を変更しなくてはならないという問題が存在する．

また，井上らの手法で対象としている掲示板のデータにくらべ，本稿で対象としているマイクロブログのデータは未知語の存在する割合が高い．そのため，形態素解析器の精度が低下するという問題が存在する．

4. 提案手法

4.1 概要

3.2 項では，これまでのわれわれの著者推定手法をマイクロブログのデータに対して適用する際の，2 つの問題について説明した．本節では，井上らの手法を改変した著者推定手法を提案する．具体的には，形態素解析器に用いる辞書を追加し，文章を品詞タグ・文字列に変換する際に文字列としてそのまま採用する品詞群である文字列採用品詞群を変更する．

辞書に単語を追加することで，3.2 項で述べた未知語に

ついての問題に対応する．これは，既存の辞書には登録されていない単語を追加することで，形態素解析の際に未知語として出力される形態素を減らすためである．同様に，マイクロブログ特有の表現である叫喚フレーズを浅井ら[9]の手法を用いて正規化し，形態素として扱うことで，マイクロブログ特有の表現の存在による形態素解析精度の低下についての問題に対応する．

また，文字列採用品詞群を貪欲法により新たに決定することで，マイクロブログ内の文章の品詞の分布や文体が通常の文章と異なるという問題に対応する．

4.2 マイクロブログ特有の表現の除外方法

本項では，メッセージからマイクロブログ特有の表現である叫喚フレーズを取り除く方法について述べる．

浅井ら[9]は，「〇〇きたあああああ」のような語尾の母音を 3 回以上繰り返す表現に注目することで，マイクロブログ上で投稿される突発的な感情を表わす「叫喚フレーズ」と呼ばれる表現を抽出する手法を提案した．

浅井らは叫喚フレーズを以下のように定義した．

- 語尾の母音が 3 回以上繰り返して付加されている
- 母音は大文字，小文字を区別しない
- 母音はひらがな，カタカナの大小文字すべて

この定義から，浅井らは以下の正規表現に基づいて，叫喚フレーズの含まれるメッセージを抽出した．

あ{3,}|い{3,}|う{3,}|え{3,}|お{3,}|ぁ{3,}|ぃ{3,}|ぅ{3,}|ぇ{3,}|ぉ{3,}|ア{3,}|イ{3,}|ウ{3,}|エ{3,}|オ{3,}|ァ{3,}|ィ{3,}|ゥ{3,}|ェ{3,}|ォ{3,}

浅井らは，以下の手順でメッセージ内の叫喚フレーズを正規化した．

1. 前処理としてデータセット内の tweet に含まれるメンション（@username），ハッシュタグを除去する
2. 叫喚フレーズの含まれる文章を，本項で説明した正規表現を用いて抽出する．
   例) うわあぁあどうしようぅぅぅう
3. 繰り返される母音を大文字化する．
   例) うわあああどうしようううううう
4. すべての繰り返される母音部分に対して，母音一文字とそれ以前の文字列を削除する．
   例) うわあどうしよう

本稿では，浅井らが抽出した，正規化済みの叫喚フレーズのリストを形態素解析器の辞書に追加する．そのため，データセット内の叫喚フレーズを浅井らと同じ方法で正規化することで，形態素解析を行う際に，叫喚フレーズを正しく形態素として扱うことができる．具体的には，上記手順の 2 から 4 までをデータセット内の文章に対して施すことで，叫喚フレーズの正規化を行う．

4.3 学習データとテストデータの作成

本稿では，評価実験に用いる学習データ及びテストデータを作成する手順を説明する．前提として，実験に使用できる全データセットから，ランダムに n 名のユーザを選択し，選択した各ユーザについて m 件のメッセージを抽出し，実験用データセットD_{in}とする．文字列として採用する品詞群を P とし，以降 P を文字列採用品詞群と呼ぶ．

step 1. D_{in}に含まれる全てのメッセージに対し叫喚フレーズの正規化を行う

step 2. D_{in}に含まれる全ユーザ ID 集合 N からユーザ ID を 1 つ選択し，選択したユーザ ID を N から取り除く．

step 3. 選択したユーザ ID が投稿したすべてのメッセージを抽出する

step 4. step 3 で抽出した全メッセージについて，ランダムにメッセージ集合T_{test}もしくはT_{train}のどちらかに分類する．この際，T_{test}及びT_{train}に含まれるメッセージが同数になるようにする．

step 5. T_{test}の中のメッセージに対し形態素解析を行い，各形態素を品詞もしくは文字列の混合列に変換する．ここで，P に含まれる品詞に変換される形態素については文字列を採用する．T_{train}についても同様の操作を行う．具体例を図 1 に示す．

step 6. T_{test}に含まれる混合列をすべて結合し，テストデータとする．T_{train}も同様に，含まれる全ての混合列を結合し，学習データとする．

step 7. step 2 から step 6 の操作を，N が空集合になるまで行う

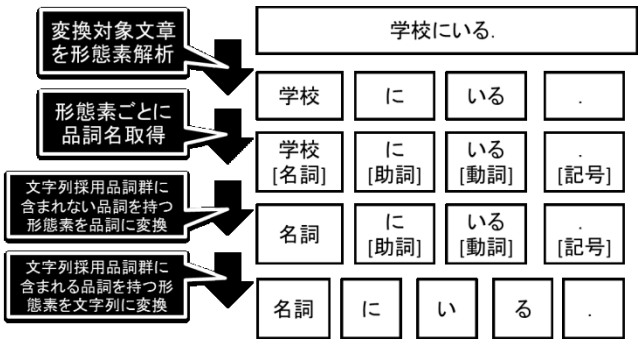

図 1　P={動詞，助詞，記号}のときの変換例

4.4 辞書に追加する単語の品詞の設定

本項では，提案手法で使用する文字列採用品詞群 P の決定手法について説明する．

1. $P = \emptyset$とおく
2. 品詞の全体集合Qに対し，P の補集合$\bar{P}$を得る
3. $|\bar{P}|$個の集合$P_i = P \cup \{p_i\}(p_i \in \bar{P}, 1 \leq i \leq |\bar{P}|)$を得る
4. 各P_iを文字列採用品詞群として，学習データおよびテストデータを改めて作成する．
5. $i_{max} = maxarg_{1 \leq i \leq |\bar{P}|} MRR(P_i)$を得る
6. $MRR(P) > MRR(P_i)$のとき，処理を終了する
7. $P = P \cup \{p_{i_{max}}\}(p_{i_{max}} \in \bar{P})$として，3 に戻る．

ここで，$MRR(P)$は文字列採用品詞群を P として，提案手法を行ったときの MRR の値である．なお，手順 4 では，データに依存した文字列採用品詞群の決定を避けるために，MRR の計算のたびにデータセットの作成を行う．その際のパラメータ n および m は，手順 7 の終了まで統一する．具体的には 4.3 項で説明した実験用データセットを用意し，各P_iについてテストデータと学習データを作成する．

5. 評価実験

本節で説明する評価実験では，提案手法及び既存手法を用いて，マイクロブログのデータを用いた大規模候補者群に対する著者推定を行い，各手法の評価を行う．

5.1 実験環境

本項では，評価実験に使用したデータセット，実験に使用する辞書に追加したキーワードセット，及び実験の際に使用した形態素解析器についての説明を行う．

本稿では Twitter から収集した tweet をデータセットとして用いた．データセットの概要は以下の通りである．

- データ収集期間: 2012 年 10 月-12 月
- 総収集 tweet 数: 945 名×1,000 件

本実験で使用するデータセットに含まれるすべての tweet には，その tweet を投稿したユーザに固有の情報である，ユーザ ID が付随する．ここで，前処理としてデータセット内の tweet に含まれるメンション（@username），ハッシュタグ，他人の文章であるリツイート(RT)をデータセットから除去した．

次に，辞書に追加するキーワードセットの概要は以下の通りである．

1. はてなキーワードファイル
   - ファイル取得日時: 2013 年 5 月 30 日
   - 単語数: 375,806
2. 叫喚フレーズファイル
   - フレーズ抽出元データ: Twitter メッセージ
   - データ収集期間: 2012 年 1 月 1 日～12 月 31 日
   - 単語数: 1,686

本実験で使用するキーワードセットの 1 つであるはてなキーワードファイルは，株式会社はてなが提供するキーワード共有サービスから取得した．インターネット上のブログサービスであるはてなダアリー[b]のユーザにより単語が追加されるため，マイクロブログ特有の単語について十分に対応することができると考えた．また，叫喚フレーズファイルについては，評価実験に使用したデータセット以外のデータから叫喚フレーズを収集したものをキーワードセットとして使用することで，公平性を保てるものと考えた．また，評価実験では形態素解析器として Sen[c]を利用する．辞書については，先行研究で使用されている辞書である IPAdic2.6.0[d]を形態素解析に用いる基本の辞書とし，IPAdic2.6.0 に各キーワードセットを追加していく．その為，品詞体系は IPA 品詞体系に依存したものになる．

5.2 評価実験全体の流れ

4.3 項で作成した学習データとテストデータの組について，著者推定タスクにおける手順 2 と手順 3 の方法で文体相違度を算出する．文体相違度算出には，表 1 で示す 2 つの手法を用いる．ここで，提案手法で用いる文字列採用品詞群の決定は 4.4 項の手順で行った．その際に使用したデータについては，文字列採用品詞群がデータに依存したものにならないよう，n=100，m=100 である実験用データセットについて，4.4 項の手順を用いて作成した．また，算出する 2 つの文体相違度は別々に保持しておく．ここで，各手法における文体類似度算出方法は，式(3)のピアソンの積率相関係数を用いる．

表 1 評価実験の対象となる著者推定手法

手法名	文字列採用品詞群	頻度分布
提案手法	接続詞，感動詞，連体詞，接頭詞，名詞，フィラー その他，未知語	2gram 頻度分布
井上らの手法	動詞，接続詞，記号，副詞，形容詞，感動詞	2gram 頻度分布

次に，テストデータ中のすべての文章に対して，著者推定タスクにおける手順 4 より文体類似度順位を算出し，MRR および mean top-k call を算出する．

5.3 実験結果の評価

4.3 項の手順で n=900，m=10 として，n と m の組み合わせについてそれぞれテストデータ及び学習データを作成し，

b はてなダイアリー, http://d.hatena.ne.jp/
c 形態素解析システム Sen, http://www.mlab.im.dendai.ac.jp/~yamada/ir/MorphologicalAnalyzer/Sen.html
d IPAdic legacy, http://sourceforge.jp/projects/ipadic/downloads/24431/ipadic-2.6.0.tar.gz/

5.2 項の手順を用いて各手法に対し，以下の 2 つの評価の算出を行った．

1. MRR
2. mean top-k call

MMR は，文体類似度順位の累積相対度数分布を定量的に評価したものである．具体的には，すべてのテストデータにおいて文体類似度順位が高くなるときに，MMR の値も高くなる．よって，MMR が高くなる手法は高く評価される．また，mean top-k call は，著者推定を行い，結果として出力した文体類似度順位の k 位までに正解が存在したときは 1，そうでなければ 0 と考え，その平均をとったものである．MRR および mean top-k call の一部についての結果は表 2 のようになった．ここで，提案手法 1 は辞書を追加しないとき，提案手法 2 ははてなキーワードファイルを，提案手法 3 は叫喚フレーズを正規化した上で叫喚フレーズファイルを，提案手法 4 は叫喚フレーズを正規化した上で各キーワードファイルを辞書として追加したときの提案手法である．

表 2　MRR および mean top-k call の結果

手法名	MRR	Mean top-k call		
		k=1	k=5	k=10
井上らの手法	0.200	0.08	0.312	0.469
提案手法 1	0.124	0.003	0.317	0.486
提案手法 2	0.129	0.007	0.312	0.512
提案手法 3	0.114	0.008	0.238	0.437
提案手法 4	0.113	0.008	0.24	0.412

表 2 から，提案手法はどれも MMR では井上らの手法より劣っているが，mean top-k call で評価した際，k=5，もしくは 10 の時ではほぼ違いがないことがわかる．提案湯法 2 は，k=10 としたときの mean top-k call の値について，井上らの手法よりも高い値を得た．このことから，twitter 特有の表現である叫喚フレーズを正規化し，叫喚フレーズを特徴量として使用することで，著者推定精度を向上させられるといえる．

6. まとめ

本稿では，既存の著者推定で取り扱ってこなかった，マイクロブログのデータを用いた大規模候補者群に対する著者推定について，推定手法の提案を行った．本稿で提案した著者推定手法を用いることで，マイクロブログのデータを用いた大規模候補者群に対する著者推定において，高精度の推定が行えることがわかった．これは，マイクロブログのデータを用いた著者推定で顕著化する「同一話題文章収集の困難化」「マイクロブログ特有の単語・表現の頻出」の 2 つの問題に，提案手法が各々対応できるためである．

本研究の課題点として，文体定量化の際，文字列と品詞以外で文体定量化を行う手法について検討していないことが挙げられる．そのため，今後の研究では文体定量化方法を模索しつつ, 推定精度を向上していくことが求められる．

謝辞 本研究の一部は科研・基盤（B)（No.25280113）によるものである。

参考文献

1) 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文の著者推定”, 計量国語学, Vol.22, No.6, pp.1-9, 2000.
2) 安形輝 : “圧縮プログラムを応用した著者推定”, J. of Library and Information Science, 三田図書館・情報学会, No.54, pp.1-18, 2005.
3) 金明哲, 村上征勝: “ランダムフォレスト法による文章の書き手の同定”, 統計数理, Vol.55, No.2, pp.255-268, 2007.
4) 石川尚季, 西村涼, 渡辺靖彦, 村田真樹, 岡田至弘: “コミュニケーションサイトに投稿されたメッセージに対する著者の推定”, 信学技報(NLC) ,Vol.109, No.142, pp.79-84, 2009.
5) 佐藤進也, 原田昌紀, 風間一洋: “文字列出現頻度比較による情報源間の類似性判定”, 情処研報(DD), Vol.2002, No.28, pp.119-126, 2002.
6) 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.
7) 井上雅翔, 山名早人: “大規模候補者群に対する著者推定手法の提案と評価”, DEIM2013, C6-6, 2013.
8) 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”, 日本データベース学会論文誌, Vol.10, No.3, pp.7-12, 2012.
9) 服部峻, 亀田弘之: “Web テキストにおける未知語の頻度調査”, 電子情報通信学会技術研究報告, Vol.110, No.63, pp.7-12, 2010.
10) 浅井洋樹, 秋岡明香, 山名早人: “きたあああああああああああああああああ！！！！！１１：マイクロブログを用いた教師なし叫喚フレーズ抽出”, DEIM2013, A4-1, 2013.
11) フリードマン, リチャード・エリオット 著, 松本 英昭 訳: “旧約聖書を推理する：本当は誰が書いたのか”, 海青社, p.355, 1989.
12) 村上征勝, “著者を探る古文書の計量分析”, 信学誌, Vol.85, No.3, pp.158-161, 2002.
13) 細江光: “谷崎の作品ではなかった 偽作「誘惑女神」をめぐって, 国文学 解釈と教材の研究”, 学灯社, Vol.33, No.8, pp.134-137, 1988.
14) Stamatatos, E. : “A Survey of Modern Authorship Attribution Methods, J. of the American Society for Information Science and Technology”, Vol.60, No.3, pp.538-556, 2009.
15) 坪井祐太, 松本裕治: “異なるタイプのドキュメントに対する著者推定”, 情処研報(NL), Vol.2002, No.20, pp.17-24, 2002.
16) N.V. Chawla, N. Japkowicz and A. Kotcz : “Editorial: special issue on learning from imbalanced data sets”, J. of the ACM SIGKDD Explorations Newsletter, Vol.6, No.1, pp.1-6, 2004.
17) 形態素解析システム Sen, http://www.mlab.im.dendai.ac.jp/~yamada/ir/MorphologicalAnalyzer/Sen.html (accessed on 2013/06/11)