Tweet の興味分析を用いたユーザ推薦システムの構築

2012SE169 森田拓也 2012SE184 野端直人 2012SE255 谷本雄哉
指導教員：河野浩之

1 はじめに

Twitter は世界で多くのシェアを誇り，1 日に Tweet される数は 5 億，毎日利用するアクティブユーザは 1 億人，1 ヶ月の間に利用するアクティブユーザは 2 億 1830 万人にものぼる [1]．Twitter ユーザ推薦システムの研究は多くあるが，ノイズの発生や興味の種類の少なさ [2]，スパムアカウントや bot の推薦，フォロワーの少ないユーザの推薦 [3] などの問題から，精度の高い推薦をすることは難しいという問題がある．

本研究では，文献 [4] よりシステムを利用するユーザの Tweet の TwitterAPI を用いて抽出し，はてなキーワード自動リンク API を用いて興味カテゴリを作成しそれぞれのカテゴリからコサイン類似度 [5] を用いて興味分野の近いユーザを推薦する．作成するカテゴリを多くすることで「興味の種類の少なさ」の問題を改善する．加えて，「スパムアカウントの推薦」,「フォロワーの少ないユーザの推薦」を改善するために，推薦されるユーザが格納されたデータベースを作成する際に，スパムアカウントや bot を除き，フォロワーの多いユーザを格納する．「ノイズの削除」に対しては，カテゴリ分類を用いて解決する．以上の 4 点を改善した推薦アルゴリズムの構築を目指す．

本研究論文は全 6 章で構成される．2 章ではカテゴリを用いたユーザ推薦システムについて解説し，3 章では先行研究に対する問題点の解決方法，提案する推薦システムの構成を述べる．4 章では TwitterAPI を用いた Tweet の収集とはてなキーワード自動リンク API を用いたカテゴリ分類を行い，コサイン類似度を用いて興味が近いユーザを推薦する．5 章で実験の評価，6 章で本研究をまとめる．

2 先行研究

2.1 興味領域を考慮した Twitter アカウント推薦 [1]

Twitter のユーザ推薦システムの開発において TF-IDF などを用いた一般的なユーザ推薦手法を用いると顔文字などの日常的に Tweet に登場する興味を示すものではないものがノイズとなってしまう．国外の Twitter ユーザ推薦の研究の現状として，Twitter ユーザ推薦が可能な Twitter クライアント [6] が開発が盛んになっている．

国内では，カテゴリに着目した研究が増えてきている．久米らの研究ではスポーツ・音楽などのキーワードのカテゴリに着目し，各キーワードがどのカテゴリに属するかを把握・管理することで顔文字などのどのカテゴリにも属さないキーワードを除外した．また，このカテゴリを一つの嗜好情報として取扱い，従来の推薦手法に加えることで，より利用者の嗜好に合った Twitter アカウント推薦の実現を目指した．

久米らは興味領域抽出機能，特徴語抽出機能，推薦フォローユーザ取得機能の 3 つの要素で構成される手法を提案した．また「Twitter4j」，形態素解析を行う「MeCab」，「SlotLib」，はてなキーワード自動リンク API の計 4 つのライブラリを用いて実装を行った．興味領域抽出機能では 0.9 という非常に高い適合率を記録した．また，特徴語抽出機能では上位 10 件では 0.66，上位 5 件では 0.68 を記録しており，概ね良好な結果を得た．標準偏差は興味領域抽出機能では 0.15，特徴語抽出機能では上位 5 件では 0.42，上位 10 件では 0.38 を記録した．推薦結果の適合率の比較結果は 5 段階中で，Twitter 公式おすすめユーザでは 5，提案手法では 1 となり，提案手法の方が低い適合率を示したアカウントの数が多い結果となった．

2.2 Twitter からのユーザ情報抽出及びプロフィール推定 [2]

的埜の研究では，マーケティング分野で利用することや共通点が多いユーザを推薦することで人々が繋がることを想定したプロフィールの推定を行った．具体的には，ユーザ属性のうち「居住地や出身地」「職業」が同じであれば共通の話題が多く生まれる可能性が高くなった．また，ユーザの興味を持っている内容が同じであれば共感できる可能性が高いと考え，「関連地」「職業」とユーザの「興味」3 点を推定し用いた．

関連地推定手法，職業推定手法，興味推定手法の 3 つの手法を試み，結果としてユーザの「関連地」「職業」を高精度に推定することに成功し，実用的な値を出すことができた．ユーザの「興味」においては，アンケート調査では 5 段階評価で 4.4 という満足度を得たため，ある程度ユーザの「興味」あるものを推定できた．最終的には推定したいユーザの “ TwitterID ” と “ 抽出したい Tweet 数 ” を入れるだけでプロフィールを作成することに成功した．

2.3 先行研究の課題

久米らの研究の課題として，推薦フォローユーザの偏りの防止，企業・スパムアカウントの検出の 2 点がある．推薦結果に関しては利用者からの不満点が多く，以下のような意見が目立った．

・特徴語は適切に抽出できているが，それが活かしきれてない
・Tweet 数，フォロワー数が少ないものや商品の宣伝等としているだけのアカウントはフォローしたくない
・似たようなアカウントばかり推薦結果に含まれている

的埜の研究の課題として，ユーザの「興味」をあらかじめ 3 つに限定したためにでた不満の解消と，ノイズ除去をしたために興味語が減少するという 2 点がある．

3　ユーザ推薦システムの提案

3.1　問題点と改善方法

本節では，先行研究の問題点に着目し，その解決方法を提案する．先行研究の問題点として次の 4 項目が挙げられる．

I:Tweet 数，フォロワー数の少ないユーザが推薦される
II:スパムアカウントや bot が推薦される
III:ユーザの興味が 3 つに限定される
IV:未知語に対処できない

I と II はデータベースに格納するユーザを選別，III はカテゴリの数を拡大，IV は未知語に対応した API を使用することで解決を図る．

3.2　ユーザ推薦システムの構成図

本節では上記の問題点と提案を考慮し，新たなユーザ推薦システムの構築をする．このシステムを構成するために必要となる機能を以下のアーキテクチャ図 1 に示す．

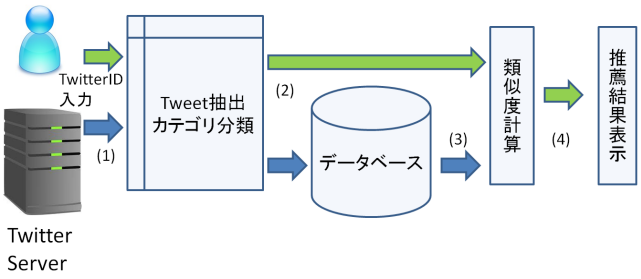

図 1　ユーザ推薦システムのアーキテクチャ

(1) システムを利用するユーザ TwitterID を入力し，Tweet 収集ツールを用いてそのユーザの Tweet を抽出する．それとは別に，推薦ユーザデータベースに格納するユーザの TwitterID を入力し，Tweet を抽出する．この際にユーザの選別を行う．
(2)(1) で抽出した Tweet を文書分析が可能なツールを用いてカテゴリ分類し，データベースへユーザ名と一緒に格納する．この際にカテゴリ数の拡大と未知語に対応した API の使用を行う．
(3) データベースに格納されているユーザの Tweet とシステムを利用するユーザの Tweet を類似度計算し，興味分野の近いユーザを算出する．
(4) 算出されたユーザの TwitterID と類似度を端末に表示し，システムを終了する．

3.3　Tweet 収集ツール

抽出を行うツールとして，「あつめるったー」，「ttc」，「TwimeMachine」，TwitterAPI の 4 つが候補に挙がった．「あつめるったー」はキーワード毎の Tweet を txt ファイルで抽出することができるフリーソフトである．抽出する時間や Tweet の量を指定することができるのが特徴であり，json，csv ファイルでの出力も可能である．「ttc」は Twitter から検索キーワードを含む Tweet を収集するフリーソフトであり，収集した Tweet は QUERY(検索ワード)，ID(Twitter 番号)，DATE(日付)，SCREEN NAME(TwitterID)，TWEET,FRIENDS(フォロー中)，FOLLOWERS の 7 つに分けられ，csv ファイルとして出力する．「TwimeMachine」は，TwitterID を入力することでの Tweet を一括で見ることができる web サイトである．最大 3200 件まで過去の Tweet を調べることが可能で，検索機能が付いている．しかし上記の 3 点のアプリケーションはプログラム内に組み込むことができない，または指定ユーザの Tweet のみを取り出すことができないため本研究では使用が困難である．

TwitterAPI はユーザ毎の Tweet を抽出し，単語の頻度を表に出力することができる．TwitterAPI を利用するにはユーザアカウントを作成しメールアドレス認証と電話番号認証を完了する必要がある．TwitterAPI は指定ユーザの Tweet 抽出が可能かつシステム間の連結も容易な為，ユーザの興味抽出，及び推薦プログラム作成を行う本研究に適している．よって今回我々は Tweet 収集に Twitter-API を用いる．OAuth での認証が必要な為，容易に認証可能な Tweepy を用いて実装する．

3.4　文書分析ツール

文書分析を行うツールとして，「MeCab」，「ChaSen」，「CaboCha」，はてなキーワード自動リンク API の 4 つが候補に挙がった．「MeCab」は C++ によって書かれた形態素解析ツールである．未知語に対しては定義を変更可能なため，ノイズの削除が可能であり，「ChaSen」よりも平均 3-4 倍の解析速度で動くのが特徴である．「ChaSen」は「MeCab」を設計し直したもので，解析精度に変化はないが，辞書登録などが可能なためカテゴリ分類プログラム作成の手助けになると考えられる．「CaboCha」は高性能な係り受け解析器であり，柔軟な入力形式が可能．データを用意すればユーザ側で学習し再定義が可能となっている．そのため Tweet のような自由な文書を解析した際に誤った品詞に分けられる可能性があるが，「CaboCha」なら再定義により解決できる．これら 3 つの文書解析ツールは最新の未知語には対応していないことと，ノイズを自動で削除する機能が存在しないという欠点がある．

はてなキーワード自動リンク API は任意のテキストを送信することでテキストからはてなキーワードを抽出し，キーワード部分を自動的にリンクして返信する API である．そのため解析の度に最新の未知語に対応できる．さらにどのカテゴリにも分類されなかった単語はノイズとして削除できる．返信された情報には各キーワードのカテゴリが含まれており，それを取得し類似度計算に利用する．この API は会員登録が不要かつ行動履歴を必要としないため，実用性の高いユーザ推薦を目的としている本研究に適している．よって今回我々は Tweet のカテゴリ分類にはてなキーワード自動リンク API を用いる．

4　ユーザ推薦システムの実装・実験

4.1　実験の流れ

我々は今回，OS:linuxOS，メモリ:6GB，CPU:Intel(R)，Core(TM):i5-2520M，CPU@2.50GHz のスペックの PC を使用した．システムの作成に当たって Python ver2.7.10 と TwitterAPI ver1.1 を使用する．ユーザ推薦システムの作成は TwitterAPI のモジュールである Tweepy を用いて行う．はてなキーワード自動リンク API，Tweepy は共に無償で提供されているモジュールのため，比較的簡単に実装環境を整えることが可能であり，本研究に適していると判断した．また，Tweepy をインストールするにあたり，pip ver.2.6 を利用した．

我々は今回の実験のために 2 つのプログラムを作成した．Twitter ユーザ推薦プログラムは 126 行，データベース格納プログラムは 96 行のプログラムで構成されている．以下の図 2 に本研究でも用いるユーザ推薦システムの流れを示す．

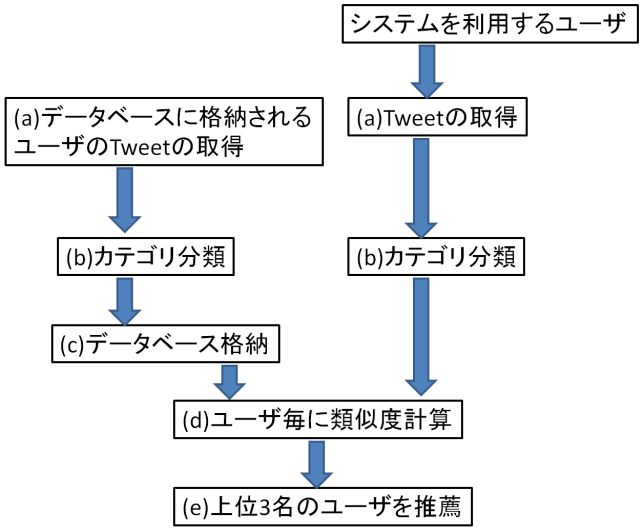

図 2　ユーザ推薦システムの流れ

(a)Tweet の収集には TwitterAPI を利用する．デベロッパーサイトからコンシューマーキー，アクセストークンなどを取得することで OAuth 認証を行うことで利用できる．本実験は 500Tweet を用いる．端末内で TwitterID を打ち込むことでそのユーザの Tweet が取得できる．TwitterAPI を使用したプログラムを図 3 に示す．26 行目では OAuth 認証を行い，29 行目でユーザのタイムラインを取得する．

(b) はてなキーワード自動リンク API を用いて興味語を取り出し，16 種類のカテゴリに分類する．はてなキーワード自動リンク API は任意のテキストを送信すると、はてなキーワードを抽出し、キーワード部分を自動的にリンクして返信する API である．今回の実験では「音楽」や「スポーツ」などのキーワード毎のカテゴリが必要だったため，mode：lite オプションを用いることで自動リンクに使われるキーワードとカテゴリ一覧を取得する．カテゴリ分類のプログラムを図 4 に示す．59 行目ではてなサーバーを呼び出し，60 行目で lite モードで実行する．

```
24.auch = tweepy.QAuthHandler(CONSUMER_KEY,CONSUMER
   _SECRET)
25.auth.set_access_token(ACCESS_TOKEN,ACCESS_SECRET)
26.api = tweepy.API(auth)
27.name = raw_input(‘ ID を入力してください＞ ’)
29.for status in api.user_timeline(screen_name=name,
   count=500)[::-1]:
30.print(’----------------------------------------’)
31.print(’name:’ + status.user.name)
32.print(status.text)
```

図 3　TwitterAPI を利用した Tweet 収集

```
59.server = xmlrpclib.ServerProxy(’http://d.hatena.
   ne.jp/xmlrpc’)
61.result = server.hatena.setKeywordLink({’body’:
   data1,’mode’:’lite’,})
```

図 4　API を利用したカテゴリ分類

(c) 分類したカテゴリはデータベースへ格納する．格納するユーザは 150 人と小規模な情報なため，Python ver2.7.10 に元から入っていることから利用が比較的容易な SQlite3 を利用した．カテゴリ内には book，music，muvie，animal，food，sports，game，anime，comic，art，science，web，elec，society，geography，idol が格納されている．データベース格納プログラムを図 5 に示す．121 行目でユーザテーブルにユーザ名と 16 種のカテゴリを格納している．

```
120.con = sqlite3.connect("tweetdata.db")
121.sql = u"insert into ユーザ values
(’%s’ 中略’%s’)"%(name,b_words 中略,idol_words)
122.con.execute(sql)
123.con.commit()
```

図 5　データベースの格納

(d) 本研究の推薦システムに使用するアルゴリズムとしてコサイン類似度を扱う．$\vec{q}$と$\vec{d}$をそれぞれ任意の文書ベクトルとする．ここでの文書ベクトルとは，文書中の単語の重要度を利用して文書をベクトルとして表現したものである．コサイン類似度とは，ベクトル空間モデルにおいて文書同士を比較する際に用いられる類似度計算手法である．コサイン類似度は，ベクトル同士の成す角度の近さを表現するため，1 に近ければ類似しており，0 に近ければ類似していないことになる．

$$\cos(\vec{q},\vec{d}) = \frac{\vec{q}\cdot\vec{d}}{|\vec{q}||\vec{d}|} = \frac{\vec{q}}{|\vec{q}|}\cdot\frac{\vec{d}}{|\vec{d}|} = \frac{\sum_{i=1}^{|V|} q_i d_i}{\sqrt{\sum_{i=1}^{|V|} q_i^2}\cdot\sqrt{\sum_{i=1}^{|V|} d_i^2}} \quad (1)$$

(e) 研究生の Tweet をコサイン類似度を用いてユーザ間の類似度を計算し上位 3 名を推薦する．推薦プログラムを図 6 に示す．144 行目から 145 行目では計算された類似度の値が入る部分を降順にして並び替える．146 行目から 147 行目では，今回は上位 3 名を表示するということなので，降順になったものを 3 回繰り返し表示する．

```
143.vector1= [] i = 0
144.for key, value,in sorted(cos.items(),
    key= lambda x:x[1], reverse=True):
145.    vector1.append((key,value))
146.for i in range(3):
147.    print "ユーザ",i+1,">",vector1[i]
```

図 6 ユーザ推薦プログラム

図 6 のプログラムを実行することで，図 7 のようにコサイン類似度によって出力された上位 3 名の興味分野の近いユーザが表示される．出力されたスクリーン名と一緒にコサイン類似度も算出される．

```
ユーザ 1 > ('norio_hangover', 0.96277523293357781)
ユーザ 2 > ('Dr_yandel', 0.96055862361479205)
ユーザ 3 > ('Issikiiiiiii', 0.94815989580843463)
```

図 7 推薦結果

5 ユーザ推薦システムの評価

ユーザ推薦システムの実行結果に対して，アンケート調査を行った．ユーザにはシステムを実行してもらい，自分の TwitterID を入力し，表示された上位 3 名のユーザの満足度をアンケート用紙に解答してもらった．アンケートを行ったユーザは 20 人であり，抽出結果が自分の嗜好に合ったものかどうか評価してもらい，提案手法の有効性を検討した．評価基準は以下の 3 点で，推薦された上位 3 名のそれぞれのアカウント毎に 7 段階中最も近いと感じた箇所に印をつけてもらった．

1:プロフィール，Tweet 両方を読んでも興味を持てなかった/フォローしてみようと思わなかった．

2:プロフィールだけでは興味を持たなかったが，Tweet を読んで興味を持った/フォローしてみようと思った．

3:プロフィール，Tweet を見ておもしろそうだと思った/フォローしてみようと思った．

推薦結果の満足度は表 1 のようになり，上位ユーザから順に 4.1, 3.9, 3.8 とある程度の満足度が得られた．これはてなキーワード自動リンク API を用いたカテゴリ分類とデータベースに格納するユーザを選別した結果だと考えられる．以下に提案手法を利用した被験者の感想を示す．

・サッカーのことを頻繁に呟いているユーザが推薦されたが，自分の好きなスポーツはテニスなのでフォローしたいとは思わなかった．

・頻繁に Tweet をしているため自分のタイムラインの邪魔になりそうなのでフォローしたくない．

表 1 アンケート結果

上位 3 名	平均満足度
ユーザ 1	4.1
ユーザ 2	3.9
ユーザ 3	3.8

・ご飯の画像を頻繁にアップロードしているユーザが推薦された．自分もよくアップロードしているため興味分野は近いが，別にフォローしたいとは思わない．

平均満足度の値よりある程度の精度が保証されたシステムの構築に成功したと考えられるが，カテゴリ分類をすることによって発生してしまった問題が多々発見された．

6 おわりに

本研究では，Twitter の公式ユーザ推薦システムがユーザにとって興味分野の異なるユーザを推薦してしまうという問題をカテゴリ分類を利用することで解決した．TwitterAPI を用いて 150 名のユーザと，そのユーザの Tweet を 500 件データベースに格納した．はてなキーワード自動リンク API のカテゴリ分類を利用することでノイズが発生するという問題と，未知語をカテゴリ分類することに成功した．また，データベースに格納されているユーザとシステムを利用するユーザの類似度をコサイン類似度を用いて算出し，端末に上位 3 名まで表示するシステムの構築をした．アンケート結果より，7 段階中平均 4 の満足度の得られ，ある程度の精度が保証されたユーザ推薦システムの構築に成功したと言える．

参考文献

[1] GaiaX SocialMedia Lab <http://gaiax-socialmedialab.jp/socialmedia/368>(2016 年 1 月 7 日閲覧)

[2] 的埜孝宏，“Twitter からのユーザ情報抽出及びプロフィール推定，” 法政大学大学院紀要 (情報科学研究科編)，Vol.9，pp.149-154 ，2014.

[3] 久米雄介，打矢隆弘，内匠逸，“興味領域を考慮した Twitter アカウント推薦，” 情報処理学会研究報告知能システム (ICS)，Vol.179 ，No.1，pp.627-632，2015.

[4] 関洋平，原田賢一，“tf/idf 重み付けに基づく動的文書作成，” 情報処理学会研究報告デジタルドキュメント (DD)，Vol.31，pp.22-31，2001.

[5] 岡崎直観，辻井潤一，“集合間類似度に対する簡潔かつ高速な類似文字列検索アルゴリズム，” 言語処理学会自然言語処理，Vol.18，pp.89-117，2011.

[6] John Hannon，Mike Bennett，Barry Smyth，“Recommending twitter users to follow using content and collaborative filtering approaches，”RecSys ‘10 Proceedings of the fourth ACM conference on Recommender systems，pp.199-206，2010.