ハッシュタグを用いたRandomForestによる tweet分類精度の向上

M2011MM038 小寺英爾
指導教員：石崎文雄

1 はじめに

近年, Social Networking Service やブログ等のソーシャルメディアの普及が拡大し，ユーザー間の情報のやりとりが大量に発生している．その情報の中には様々な種類の情報, 感想, 意見が内包されている．しかし，爆発的に情報量が増加しているソーシャルメディア内において自身が要求している情報を発見することは困難である．そこで，情報の整理・分類を行うことによる情報の容易な共有・発見が重要となる．Twitter の分類に関する研究は本岡らの研究 [2] や黒木らの研究 [3] 等多く行われているが，ユーザベースによる分類であるため精度が安定しない等の問題が存在した．

本研究では, ユーザベースによるユーザの嗜好性の影響を受けない分類を行うために Twitter から発生する tweet に注目し, Twitter の特徴であるハッシュタグを利用することで tweet 内容分類精度の向上を目的とする．具体的な手法としてはハッシュタグごとの特徴語を算出し，特徴語を利用することで RandomForest 等の分類木を作成する．作成した分類木を用いてハッシュタグを持たない tweet の類似タグへの分類を行う．また，分類結果の適合率による評価を行い，単純検索やユーザベースの既存研究手法との比較・評価を行う．

2 Twitter分類の先行研究

本章では，Twitter 等のマイクロブログに対する既存のサービスや分類研究に関する先行研究について述べる．また，先行研究から得た知見と本研究との相違点について述べる．

2.1 既存の Twitter サービス

Twitter の情報を扱う既存のサービスは多数存在する．Twitter から得られる情報にはいくつか種類があり，tweet 内容に関する情報, ユーザアカウントに関する情報, Twitter 経由の流入やコンバージョンなどの情報などに区分可能である．表 1 に具体例をいくつか示す.

表 1 既存のサービス

サービス名	サービス内容	区分
Tabetie	tweet されたグルメ情報を集約	tweet 内容
脳内スキャンβ	tweet した話題の傾向を分析	アカウント
tweet cloud	tweet を解析してタグクラウドを作成	アカウント
hashtagsjp	ハッシュタグの検索・分類	tweet 内容

2.2 ハッシュタグを用いた tweet の分類

ハッシュタグを用いた tweet の分類を行う研究は積極的に行われている．本岡らの研究 [2] では，ハッシュタグに注目して既知のイベントを示すハッシュタグからその類似イベントを示すハッシュタグを検索・発見することを目的としている．検索手法は次の様に行う.

- 入力ハッシュタグを使用しているユーザを Twitter から 1 人分取得し，そのユーザ集合が使用しているハッシュタグを取得
- 取得したハッシュタグ集合に対してイベント性判定
- 類似性に基づき，ランク付け
- ランクに基づき上位 n 件のハッシュタグ集合を出力

また，集中度を用いたフィルタリング関数を σ_{Con}，突発事象を取り除くフィルタリング関数を σ_{Pre}，取得ハッシュタグ集合を H とし，イベント判定のためのフィルタリング関数 $\sigma_{Eve}(H)$ を (1) 式のように定義している.

$$\sigma_{Eve}(H) = \sigma_{Con}(\sigma_{Pre}(H)) \quad (1)$$

本岡らの研究の問題点としてはハッシュタグを使用しているユーザ層によって類似イベントを示すハッシュタグの発見の精度が変動したことと，複数事象を表すハッシュタグに関してはうまく尺度が働かずうまく検索できないという点が存在した．また，この手法は，ハッシュタグをユーザデータを元に分析する点で本研究とは異なる.

2.3 ハッシュタグの内容説明

黒木らの研究 [3] では，ハッシュタグが含まれる tweet を時系列別にクラスタリングすることで，各クラスタの中から重要となる tweet を抽出し，ハッシュタグの内容説明を行う．具体的な手法を次に示す.

- 同一のハッシュタグが付いた tweet を抽出
- 取り出した tweet を MeCab を用いて形態素解析を行い名詞を抽出
- TF-IDF 値を計算して文書スコアを算出

黒木らの研究ではハッシュタグの内容要約のために特徴量を文書スコアとして使用していたが，本研究では同様に特徴量を利用することで類似 tweet の分類精度の向上を図った.

2.4 既存手法のまとめ

本岡らの研究ではハッシュタグの内容の類似性をユーザの嗜好性を元に判断し分類を行った．また，黒木らの研究ではハッシュタグの内容をハッシュタグを含む tweet の文書スコアを元に判断した．本研究では，ハッシュタグの内容を特徴語によって判断し，樹木モデルを作成して tweet の分類を行うことでユーザの嗜好性の影響を受けない分類を目指す．これらの既存手法と本手法をまとめて表 2 に記述する.

表 2 手法比較

研究者	対象	手法	特徴
本手法	ハッシュタグ	樹木モデル	ユーザの嗜好性が影響しない
本岡ら	ハッシュタグ	フィルタリング関数	ユーザの嗜好性が影響する
黒木ら	tweet 内容	文書スコア	話題の精度が安定しない

3 tweet 分類手法の概要

本章では，本研究で提案する分類手法について説明する．

3.1 分類を行う流れ

本研究で行う tweet データセット収集から分類までの流れを図 1 示す．

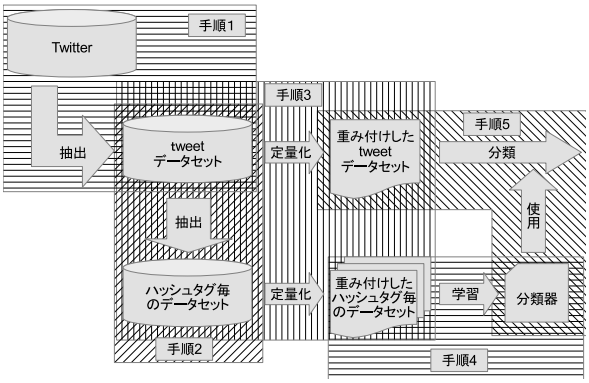

図 1 手法の流れ

1. Twitter 上からの tweet データの収集
2. 同一ハッシュタグを持つ tweet の抽出
3. 各 tweet データセットを MeCab を用いて形態素解析・特徴語の算出
4. 3 を元に tweet の分類木を作成
5. 4 で作成した分類木を元にデータセットを分類

本研究では日本語によって書かれた tweet を対象とする．tweet の形態素解析には MeCab を用いる．また，tweet 内のユーザ ID や URL は形態素解析の対象としない．MeCab による形態素解析の結果から名詞を抽出し TF-IDF 値を計算し，特徴語を算出する．最後にこの特徴語を用いて分類木を作成し tweet の分類を行う．

3.2 特徴語の算出

本節では，提案手法の特徴量算出アルゴリズムについての説明を行う．本研究では形態素解析に MeCab を使用した．MeCab は形態素解析エンジンの一つであり，文章の品詞付けなどを行うことが可能である．

特徴語については黒木らの研究 [3] で用いられている TF-IDF 法を採用した．TF-IDF 法とは，TF 法と IDF 法を組み合わせて，ある単語 t の特定の文書 d の重みを計算する手法であり，情報検索分野において索引語の重み付けに一般的に用いられている手法である．

TF 法 (Term Frequency) は，索引語の頻度をもとに重み付けする方法であり，文書中で出現頻度が高い単語をその文書において重要な単語とする．しかし，多くの文書に出現する単語は文書を特定する性質を持たないことが多い．

IDF 法 (Inverse Document Frequency) は，単語がどのくらい特定性を持つかを重み付けに反映して，単語重要度を評価する手法である．しかし，TF-IDF 法を tweet に適用するためには問題がある．論文のように，ある程度の長い文書ならば重要な単語が繰り返し出現するという TF 法の仮定が有効である．しかし，検索対象の文書が tweet のように繰り返しの少ない短文である場合，TF 法の結果がどれも同じになってしまい，IDF 法の単語の出現文書数だけで単語重要度を決めることになるため，TF-IDF 法では高精度な文書類似検索は期待できないという欠点がある．

そこで，本研究では tweet 単位ではなくハッシュタグ単位での IDF 値の算出モデルを提案し，ハッシュタグごとの特徴語の算出を行う．

3.3 樹木モデルについて

本節では，形態素解析によって重み付けされた tweet データセットの分類を行うアルゴリズムについて説明する．本研究では，分類手法として樹木モデルを採用している．樹木モデルとは非線形回帰分析，非線形判別分析の一つであり，単語の頻度等の説明変数を元に分類モデルを作成する．本研究では，CART と RandomForest を使用した．

CART（Classification and Regression Trees）とは 2 分岐の決定木によってクラス，連続値，生存時間を予測する手法であり，不純度を表す *Gini* 係数によって分類を行う．具体的には以下のような手順で分類を行う．

1. tweet ごとに目的変数に対する *Gini* 係数を計算
2. ルートノードの *Gini* 係数と分類したノードの *Gini* 係数の差を計算
3. 全ての変数に対してその差を計算して，差が最大のものを分岐条件とする
4. 1. から 3. を繰り返す

Gini 係数は，クラス $c \in C$ のサンプルの，全体に対する割合を p_c と表すと式 (2) のように表すことができる．

$$Gini\ 係数 = 1 - \sum_{c \in C} {P_c}^2 \tag{2}$$

RandomForest は，集団学習法の一つであり，精度の不安定な複数の結果を組み合わせることで精度の向上を行う手法である．具体的には次の手順で分類木を作成する．

1. 与えられたデータセットから N 組のブートストラップサンプルを作成
2. 各ブートストラップサンプルデータを CART の *Gini* 係数などの基準により最良の特徴を選択
3. 全ての結果を統合し，新しい予測・分類器を構築

4 tweet の分類実験・評価

本章では Twitter での実データ収集・分類実験について記述を行う．

4.1 データセット

本研究で使用するデータセットとして Twitter 上から 10 月に収集した 5980073 件の tweet された実データを収集した．データ内容は，tweetID, tweet 時間，tweet 内容の 3 項目を収集した．収集したデータの一例を図 2 に示す．

id string primary key	time string	status string
y_k_h26	2012-08-16T16:56:29+09:00	@rikopin801 本当に本当にスイマセンでし
724685	2012-08-16T16:56:29+09:00	あの新ロゴのグッズ欲しい！>「【プレゼント
thorn_1989	2012-08-16T16:56:29+09:00	こりゃ録り終わるでよ
maesh1n	2012-08-16T16:56:29+09:00	園内の同じ1歳児クラスで一人だけとても
hikari_420	2012-08-16T16:56:29+09:00	@revolater0522 私もあっぷうさんのお名
rudent_dent	2012-08-16T16:56:29+09:00	今超ナチュラルに横入りされてバス乗る順
simona_com	2012-08-16T16:56:29+09:00	Btooom! chiuso! sotto con Blood Lad 2
goro1982	2012-08-16T16:56:29+09:00	りゅうさん…
lazuogozen	2012-08-16T16:56:29+09:00	FEAって結局、考えうる要素を全て網羅する
kaerubungei	2012-08-16T16:56:29+09:00	@donkou 盆休み最終日効果ですかね？
chi_jhy	2012-08-16T16:56:29+09:00	@ayaachaaaaaaaaa え、わたし(((;ﾟДﾟ))))!
shizya555	2012-08-16T16:56:29+09:00	@S4toimo ふむ じゃあ今度大阪行くときに

図 2 データセットの一例

4.2 収集方法

収集方法としては Twitter の公式 API を使用して日本語 tweet をフィルタリングして収集した．Twitter API は REST API，Streaming API 等の複数の API を提供している．REST API はツイートの更新や参照を行う最も基本的な API であり，URL にクエリを渡してレスポンスを得る REST を使用している．一方で Streaming API はタイムラインの変更をリアルタイムに受け取ることが可能である．また，API には 1 時間に同じ IP からの使用制限があるため本研究では Streaming API を使用して収集を行った．

4.3 データ前処理

本節では，実験を行う前処理としてデータセットの標準化を行う．内容としては日本語以外の tweet の削除，URL の削除，リプライなどのユーザ ID の削除を行った．これは，Kevin らの研究 [4] によって URL が正確な分類を阻害することが示されていたためである．この処理によって日本語のみの 737928 件の tweet データセットを作成した．また，tweet の日本語及びハッシュタグを含む数を表 3 に示す．

表 3 データセット内容

言語	全体量	ハッシュタグを含む tweet
全体	5980073	701567
日本語	737928	44848

次に，データセットの定量化を行う．前処理を行った日本語のみのデータセットに対して名詞の抽出を行い，重みを付与する．名詞の抽出には，python を使用して MeCab を用いて形態素解析を行い名詞を抽出するプログラムを作成した．作成したプログラムを図 3 に示す．

図 3 のプログラムでは，外部テキストから入力した文章に対して形態素解析を行い，形態素・素性をそれぞれ格納する．その後「名詞」を素性に持つ単語のカウントを行い，カウンタを用いてソートを行う．

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

import MeCab, sys
#1
mecab = MeCab.Tagger()
counter = {}
#外部ファイルから入力
for line in sys.stdin:
    node = mecab.parseToNode(line)
    node = node.next
#単語が名詞の場合カウント+1
    while node:
        if node.feature.startswith("名詞,"):
            if not node.surface.isdigit():
                counter[node.surface] = counter.get
(node.surface, 0) + 1
        node = node.next
#カウントでソート
for surface, count in sorted(counter.items(),
key=lambda x:x[1], reverse=True):
    print count, surface
  }
}
```

図 3 名詞抽出プログラム

上記のプログラムで抽出された名詞を用いて各 tweet に対して図 4 の様に重み付けを行う．

	笑	今日	人
@grun_wald あぁありがたやーありがたやー	0	0	0
@somafixer うおおおおおおおおあああああああああああああああああ	0	0	0
今日はとても充実ーっ(｀_´)ゞあざす！	0	1	0
@yomi9xx3x あwwwwwwwwwwやめろwwwwwwwwwwwwwwwwwwww	0	0	0
@haaruumiikii いいなあ(*´ω`*) うちも、そこ止まっちゃったんだけど確か「	0	0	0
@htβ2933 あ、ホテル2つあったんだ!?w 同じだったらやばかったね！(//∇	0	0	0
意外とあいねさまと紀華嬢サマライ進んでるみたいです(震え声)	0	0	0
@Tsukasa_Kikuo はあ。	0	0	0
@kululu0322 いいねー*¥(^o^)/*運転しちゃおうかー(≧∇≦)！笑 バーリナ	0	0	0
あ！ウインクや！	0	0	0
あ、20110321は春分の日でした。学校に行っている訳がない。	0	0	0
おやすみなしあ	0	0	0
@Tanaka35 あ、やっぱり！わくぶよ調べてみたけど、このアルルも好きだー	0	0	0
@aki0326m やっぱ(￣▽￣)長いなーあ(*^_^*)♪ 今回は続くでね(笑)	1	0	0
あ、でも以外に10万くらいだ(;_;)でも時期的にもお金的にも行けないかも。バ	0	0	0
@mayataka_8624 あ、先にやられた?(笑)	1	0	0
@inoooouma あ	0	0	0
自分を変えてくれる人に出会ってしまったら、今の 全てを投げ出す気がする。	0	0	1

図 4 重み付けした tweet

4.4 ハッシュタグ選定

本節では，本研究で使用する人気ハッシュタグの選定を行う．選定の参考に hashtagjp にて人気のハッシュタグとして登録されているハッシュタグを使用した．選定の手順は以下の様に行った．

1. hashtagjp を参考に人気ハッシュタグを検索
2. 収集したデータセットに含まれている事を確認
3. 選定したハッシュタグ同士が類似内容でない事を確認

その結果，次の 5 つのハッシュタグ「#amazon」「#相互」「#2ch」「#akb48」「#学校」を人気ハッシュタグとして選定して分類実験を行う．また，表 4 に収集したデー

表 4 人気ハッシュタグが含まれる数

ハッシュタグ	含まれる数
#amazon	400
#相互	1364
#2ch	410
#akb48	405
#学校	340

タセット内に含まれる人気ハッシュタグの数を示す.

4.5 tweet 分類実験

本節では，4.3 節にて作成したデータセットを元に分類実験を行う．実験手順としては以下の様に行った.

1. tweet データセットにハッシュタグのタグを付加
2. R にデータセットを読み込む
3. データセットを元に手法毎の分類木の作成
4. 分類木を用いてデータセットを分類
5. 特徴量閾値を用いて説明変数の数を変更して 2. に戻る

データセットとして前節にて選定した 5 つの人気ハッシュタグを含む 897 個の tweet を使用して，手法に CART と RandamForest を用いて分類実験を行った．実験では特徴量閾値を用いて説明変数の数を変更して行い，適合率と検出結果を表 5 に示す.

表 5 CART と RandomForest による分類実験

手法	特徴量閾値	説明変数	適合率	検出数
CART	0	1000	95.7%	21
CART	0	39	91.9%	12
CART	10	39	96.4%	46
CART	30	19	95.0%	23
RandomForest	0	1000	96.4%	19
RandomForest	0	39	89.0%	16
RandomForest	10	39	95.2%	43
RandomForest	30	19	94.3%	50

表 5 の結果の様に特徴量閾値に従って，説明変数の数を変更して分類木を作成した場合，分類率の変動は少なく検出数の向上を確認した．また，特徴量閾値 30 で作成した RandomForest で行った分類が最も高い検出数を示した.

4.6 Twitter 実データでの分類結果

前節の結果を踏まえて，実際に収集した tweet データセットからランダムサンプリングした 5 万件の tweet に対して人気ハッシュタグ 5 つを用いて閾値 30 の RandomForest で作成した分類木を用いて分類実験を行った．また，分類結果に対してアンケートによる適合率の評価を行った．分類結果と評価結果を表 6 に示す.

表 6 分類結果

ハッシュタグ	説明変数	単純検索	検出数	適合率
#2ch	11	14	23	34.8%
#akb48	32	15	97	74.2%
#amazon	17	8	62	17.7%
#学校	22	349	499	73.5%
#相互	33	96	626	66.6%

4.7 評価結果の考察

表 6 の分類結果から「#akb48」「#学校」「#相互」の 3 つのハッシュタグは単純検索よりも高い精度を示したが,「#2ch」「#amazon」の 2 つのハッシュタグでは精度が安定しなかった．そこで，図 5 の様なハッシュタグごとの単語の共起ネットワーク図を作成してハッシュタグの性質と精度の考察を行った．精度の高い 3 つのハッシュタグでは共起関係が強く全体的にハッシュタグの内容を説明していた．それに対して，精度が安定しない 2 つのハッシュタグでは，共起関係が少なくハッシュタグに対して複数の話題を持つ集合となっているため精度が不安定な値を示したと考えられる.

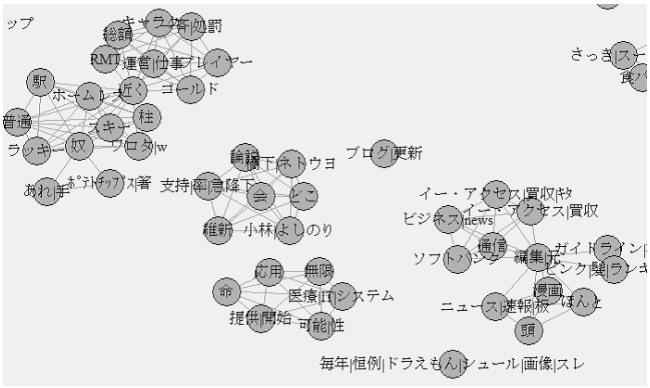

図 5 共起ネットワーク図

5 おわりに

本研究では，ハッシュタグの特徴語を利用した tweet 分類精度の向上を行った．収集した tweet データをハッシュタグに分類・重み付けを行うことでハッシュタグ毎の分類器を作成し，ハッシュタグによる分類精度の変動や分類アルゴリズムによる変化の実験・検証を行った.

評価実験の結果，提案手法による分類では単純検索時より 480% 検出数が向上し，ユーザベースによる分類の先行研究より平均で 7%の適合率が向上をするという結果を得られた．また，ハッシュタグの持つ特徴により精度・検出数に変動が起きることが判明した．今後の課題として共起性が低く複数の話題を持つハッシュタグに対する改善手法の考案が挙げられる.

参考文献

[1] Twitter, http://twitter.com/ (accessed 2013.1)

[2] 本岡亮, 湯本高行, 新居学, 高橋豊, 角谷和俊,“ Twitter ハッシュタグを用いた類似イベント検索, ”データ工学と情報マネジメントに関するフォーラム, A1-5, 2011.

[3] 黒木陽介，倉門浩二，大石哲也，越村三幸，藤田博，長谷川隆三,“ Twitter 発言の時系列解析に基づくハッシュタグの内容説明, ”情報処理学会第 73 回全国大会, 2N-9, pp.695-696，2011.

[4] Kevin Dela Rosa, Rushin Shah, Bo Lin, Anatole Gershman, Robert Frederking,“ Topical Clustering of Tweets, ”ACM SIGIR:SWSM, 2011.