Vol.2011-IFAT-102 No.1
Vol.2011-DD-80 No.1
2011/3/28

ベイジアンフィルターを用いたTwitterにおけるツイートのハッシュタグ分類

竹 中 姫 子[†1]　古宮 嘉那子[†2]　小 谷 善 行[†2]

Twitter ではハッシュタグという，自分の投稿（ツイート）に則した内容のインデックスをつける機能が提供されている．本研究ではハッシュタグのついていないツイートにたいしてハッシュタグを推定することを目的とする．そこでハッシュタグのついたツイートを学習し，そしてあるツイートがどのハッシュタグに属するかの推定を行った．分類器としてベイジアンフィルターを使用し，それぞれのタグについて 2 値分類を行い，複数のハッシュタグの推定を行った．実験では 50 種類のハッシュタグのつきの約 4 万件のツイートを学習データとして使用した．ツイート文にベイジアンフィルターを適用する場合は既知語に限定して処理を行うことで良い結果が得られるとわかった．

Hashtag Classification of Tweets in Twitter using Bayesian Filtering

Himeko Takenaka,[†1] Kanako Komiya[†2]
and Yoshiyuki Kotani[†2]

In this paper, we propose a method of discovering hashtags, which are indexes in Twitter. We estimate hashtags of tweets without hashtags using tweets with hashtags. Binary classifier was developed for every tweet so as to they have more than one tags, and Bayesian filtering was used to classify. In the experiment, about 40,000 tweets with 50 kinds of hashtags are classified. The result shows Baysian filtering with limiting known words is effective in estimating hashtags of tweets.

1. はじめに

近年，マイクロブログサービスである Twitter が注目されている．Twitter[⋆1] とはマイクロブログの代表的な WEB サービスである．その規模は世界的であり，マイクロブログの普及の火付け役ともなっている．特徴として，

- 投稿（ツイート）は日本語も英語も 140 字以内と制限される．
- 自分が購読 (Follow) しているユーザーの発言をまとめてトップページ TimeLine で時系列順に読むことができる．
- 特定の発言に対して返信することができる．
- 自分が気に入った，自分の follower に知らせたい他人のツイートを引用 (リツイート) することができる．
- WEB API が公開されているため，大量のツイートを集めやすい．

などがあげられる．このためチャットや掲示板のように扱われることもある．

Twitter ではハッシュタグという自分のツイートにタグをつける機能が提供されている．ツイートにハッシュタグを埋め込むことによって，そのツイートにインデックスをつけることができる．ユーザーが発信したツイートがあるハッシュタグの則した内容だと思われるのものには，そのツイートの中にアルファベット・数字・アンダースコアの組み合わせで自由な文字列でハッシュタグを埋め込むことができる．ユーザーはハッシュタグを検索することでその内容の話題を検索することができる．これにより膨大なデータからユーザーが求める情報へと導くことができる．

ハッシュタグは一つのツイートにいくつでも埋め込むことができるが，ツイートの 140 字以内に収めなければならない．ユーザーの判断にまかされるため，ハッシュタグだけでもツイートは構成することができる．

ハッシュタグは発言するユーザーが決めるものであり，ユーザーが自由に生成できるため，同じ意味を持つハッシュタグでも別の文字列でハッシュタグが付けられることがある．そこで本研究ではツイートの内容に対して適切なハッシュタグをつけることを目的とする．

†1 東京農工大学大学院　工学府　情報工学専攻
Tokyo University of Agriculture and Technology, Graduate School of Engineering, Department of Computer and Information Sciences
†2 東京農工大学　工学研究院　先端情報科学部門
Tokyo University of Agriculture and Technology, Institute of Engineering
⋆1 Twitter,http://twitter.com

Vol.2011-IFAT-102 No.1
Vol.2011-DD-80 No.1
2011/3/28

以降，2 章で関連研究について述べ，3 章でベイジアンフィルターを用いた推定手法を提案する．4 章で評価実験の結果を示し，5 章で実験結果の考察を行う．6 章でまとめを述べる．

2. 関 連 研 究

文書分類は自然言語処理の分野では古くから研究されている．コンピュータの発達やインターネットの普及のに伴い電子書類が増加してからは，情報検索などの分野にも応用できるため文書分類は盛んに研究されるようになった．

情報検索におけるクラスタリングの代表的な文書分類方法として文書中の出現単語頻度から求められる tf/idf を使用する方法がある．たとえば徳永ら[1] では重み付き IDF（WIDF）という tf/idf を改良した手法で文書中の単語からインデックスとなるものを求め，文書分類に応用している．また，小熊ら[2] では文書ごとの単語の共起頻度から各単語の重要度を計算し，k-means 法によってクラスタリングを行う手法を提案している．一方で文書の内容だけでなく，WEB 文書においてハイパーリンクで参照された文書の内容も学習する手法を鈴木ら[3] で提案され，精度は従来の方法とあまり変化はないものの，コンテンツのない文書の分類を可能にした．

しかしこれらはある程度の長さを持つ文書を使用した研究であった．最近ではインターネットの消費者が生成したメディアである CGM を用いた研究も注目されている．CGM の代表としてロコミサイト，SNS，ネット掲示板などがあげられるが，特に最近ではマイクロブログを用いた研究も盛んになってきている．

マイクロブログを用いた研究としては，高村らのマイクロブログ記事でのあるトピックに関するエントリをまとめる研究[4] や，A.Ritter らの会話モデルを構築する[5] などが挙げられる．マイクロブログの特徴として投稿者の行動とのリアルタイム性が高いということがあげられるが，投稿文字数が制限されているため本来文書分類で重要な素性であった単語の含有数が少なくなってしまう．青島ら[6] で出現単語の前処理として単語間の類似度を求めたうえで非階層型クラスタリング CLWC 法にて制約付きクラスタリングを行っている．CLWC 法は記事間に must-link と cannot-link の属性を付与し，クラスタリングを行う手法である．しかし must-link と cannot-link の属性を明確に付与することは，ハッシュタグのついたツイートにおいては難しいと考えられる．

3. 推 定 手 法

3.1 ハッシュタグ推定の全体設計

Twitter のツイートのハッシュタグ分類の流れを図 1 に示す．全体の流れとして学習部で得られたデータを利用して推定部でハッシュタグの推定を行う．

図 1 の縞背景部分が学習部である．学習部ではまずハッシュタグ付きのツイートを入力する．入力したツイートに対して形態素解析を行い，単語とその出現数を数え上げる．そして入力ツイートが属するそれぞれのハッシュタグのデータに数え上げた単語を加える．いくつもツイートを入力することにより，ハッシュタグ別で各単語についての単語出現頻度表が作成できる．

図 1 の点背景部分が学習部である．推定部ではまず入力した一つのツイートに対して形態素解析を行う．そして学習部で作成した学習データから生成されたハッシュタグごとの分類器に入力し，そのツイートが属するハッシュタグを出力する．これは図 1 の縞背景内の太線枠部分にあたる．推定の対象となるハッシュタグが X 種類であれば X 個の分類器を用意する．分類器はそのハッシュタグに属するか・属さないかの 2 値分類を行う．分類器はベイジアンフィルターを用いる．

3.2 ベイジアンフィルター

ベイジアンフィルターはあるツイート T がハッシュタグ H に属するか属さないかの判定をする．ツイート T は N 個の形態素形態素 t からなると，以下のように定義する．

$$T = \{t_1, t_2, ..., t_N\} \tag{1}$$

ツイート T がハッシュタグ H に属する確率はベイズ推定を用いて以下のように表す．

$$p(H \mid T) = \frac{p(H)p(T \mid H)}{p(T)} \tag{2}$$

ツイート T は単語 t_i の列からなるとする．ハッシュタグ H が出現したときツイート T である確率は，ハッシュタグに属する事象 H と属さない事象 $\neg H$ を用いるとそれぞれ以下のように表せる．

$$p(T \mid H) \approx \prod_i p(t_i \mid H) \tag{3}$$

$$p(T \mid \neg H) \approx \prod_i p(t_i \mid \neg H) \tag{4}$$

3，4 をベイズ推定の式にあてはめると，ツイート T が出現したときにハッシュタグ H に

Vol.2011-IFAT-102 No.1
Vol.2011-DD-80 No.1
2011/3/28

属する確率と属さない確率はそれぞれ以下のように表せる．

$$p(H \mid T) \approx \frac{p(H)}{p(T)} \prod_i p(t_i \mid H) \tag{5}$$

$$p(\neg H \mid T) \approx \frac{p(\neg H)}{p(T)} \prod_i p(t_i \mid \neg H) \tag{6}$$

ツイート T がハッシュタグ H に属する確率が属さない確率より大きければツイート T はハッシュタグ H に属すると言える．言い換えれば式 5÷ 式 6>1 が成り立つ．これを以下に表す．

$$\frac{p(H \mid T)}{p(\neg H \mid T)} \approx \frac{p(H)}{p(\neg H)} \prod_i \frac{p(t_i \mid H)}{p(t_i \mid \neg H)} > 1 \tag{7}$$

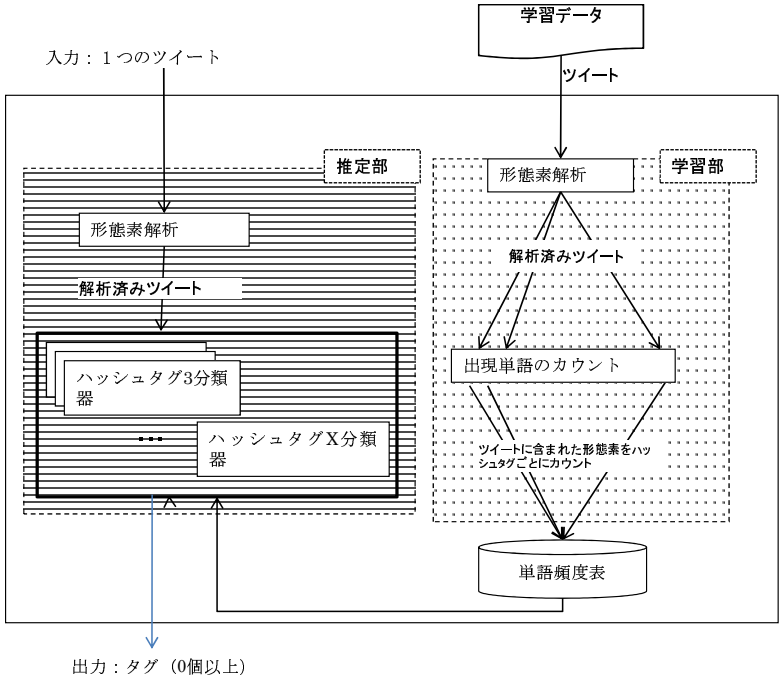

図 1 ハッシュタグ推定全体の設計

上式はベイジアンフィルターとも呼ばれ，本式を基本の分類器とする．

3.3 スムージング

単語 t が学習データに出現しなかった場合，t の出現確率は 0 となってしまう．しかし，使用するのは出現確率の積であり，出現確率が 0 の単語がツイートの中に出現した場合，そのツイート全体の出現確率が 0 となってしまう．この問題はゼロ頻度問題とも呼ばれている．ゼロ頻度問題に対応するために 2 つのスムージング手法を提案する．

- 加算法

全ての出現回数に定数 δ を加える方法である．一般に $\delta = 1$ にしたものをラプラス法という．よってベイジアンフィルターの式は以下のように書き換えることができる．

$$\log \frac{f(t,H)}{f(t,\neg H)} + \sum_i \log \frac{\frac{f(t_i,H)+\delta}{f(t,H)+\delta V(H)}}{\frac{f(t_i,\neg H)+\delta}{f(t,\neg H)+\delta V(\neg H)}} > 0 \tag{8}$$

ただし，H で出現した単語の種類数を $V(H)$ とする．今回，δ の適切な値は実験で決定する．

- 加算法+既知語限定処理

学習データに出てこない単語が出てきた場合の単語における出現確率はハッシュタグ全体の単語数に依存してしまう恐れがある．そこで学習データで得られた単語のみ計算を行う手法を以下に示す．

$$\log \frac{f(t,H)}{f(t,\neg H)} + \sum_i B(t_i,H) > 0 \tag{9}$$

ただし $B(t_i,H)$ は以下のように定義する．

$$B(t_i,H) = \begin{cases} \log \frac{\frac{f(t_i,H)+\delta}{f(t,H)+\delta V(H)}}{\frac{f(t_i,\neg H)+\delta}{f(t,\neg H)+\delta V(\neg H)}} & (\ f(t_i) \neq 0\ ) \\ 0 & (f(t_i) = 0) \end{cases} \tag{10}$$

これにより学習データに出てこない単語については全く計算を行わないことになる．既知語のみ計算を行うため既知語限定処理と呼ぶ．既知語限定処理は加算法を元にしているため，加算法+既知語限定処理とする．

4. 評 価 実 験

3 章で提案した，ベイジアンフィルターに加算法もしくは加算法+既知語限定処理を用いたツイートのハッシュタグの推定の評価を行う．

Vol.2011-IFAT-102 No.1
Vol.2011-DD-80 No.1
2011/3/28

4.1 実験設定

- 実験データ

hashtagjp[⋆1]で紹介されていた 1 月 16 日 3 時（日本時間 1 月 16 日 12 時）時点での日本語ツイートでの人気タグ 50 件について Twitter の検索 API[⋆2]を通じて取得した．1 月 16 日 3 時（日本時間 1 月 16 日 12 時）から 1 月 16 日 20 時（日本時間 1 月 17 日 5 時）の間に，各タグにつき 1 時間毎に最新 200 発言まで取得し，合計 80912 件取得した．取得したデータの中には公式でリツイートされたツイートは含まれない．取得したデータを 2 つにわけ，学習データとテストデータとした．

人気タグ 50 件の中で検索できた発言の中には人気タグ以外のタグも含まれるが，今回はそれらの推定は行わない．学習データやテストデータの中には日本語以外の言語で書かれた文も含まれている．日本語以外の言語で書かれた割合はハッシュタグごとに異なるが，1% から 95% 前後であった．

また，取得したデータからハッシュタグは全て除去した．よって学習部や推定部ではハッシュタグの情報は一切使用しない．

- 形態素解析

形態素解析には Mecab[7] を用いる．今回は表層形（出現形）を一つの単語として使用する．

- 評価方法

テストデータのツイートに関する正解のハッシュタグは，もともと付与されていたハッシュタグとする．実験の評価は適合率，再現率，F 値から求めるものとする．

4.2 実験結果

- 加算法

加算法のみを用いた推定の結果を表 1 に示す．傾向として δ が小さいほど再現率は低く，適合率は高くなっていた．F 値は $\delta = 1.0 \times 10^{-12}$ の時，最高値の 0.501 が得られた．

- 加算法+既知語限定処理

加算法+既知語限定処理を用いた推定の結果を表 2 に示す．加算法と同様に，傾向として δ が小さいほど再現率は低く，適合率は高くなっていた．F 値は $\delta = 1.0 \times 10^{-4}$ の時，最高値の 0.573 が得られた．

加算法のみのスムージングを行った結果と，加算法と既知語限定処理を行った F 値の結果を比較した図を図 2 に示す．

4.3 考　　察

- 適切な δ の値について

$\delta = 1$ はラプラス法と呼ばれるディスカウンティング手法である．$\delta = 1$ の時，再現率 0.989，適合率 0.038，F 値 0.074 であった．これはほとんど全てのツイートに対してほとんどのハッシュタグがつけられていたと考えられる．式 8 において単語に関する数値は

表 1　加算法を用いたハッシュタグ推定の実験結果

delta	再現率	適合率	F 値
1.0×10^{-0}	**0.980**	0.036	0.070
1.0×10^{-1}	0.936	0.090	0.165
1.0×10^{-2}	0.865	0.202	0.328
1.0×10^{-3}	0.803	0.291	0.427
1.0×10^{-4}	0.752	0.339	0.468
1.0×10^{-5}	0.716	0.367	0.485
1.0×10^{-6}	0.688	0.384	0.493
1.0×10^{-7}	0.666	0.395	0.496
1.0×10^{-8}	0.649	0.404	0.498
1.0×10^{-9}	0.634	0.410	0.498
1.0×10^{-10}	0.622	0.416	0.499
1.0×10^{-11}	0.612	0.421	0.499
1.0×10^{-12}	0.604	0.427	**0.501**
1.0×10^{-13}	0.598	0.430	0.500
1.0×10^{-14}	0.592	0.432	0.500
1.0×10^{-15}	0.587	**0.435**	0.500

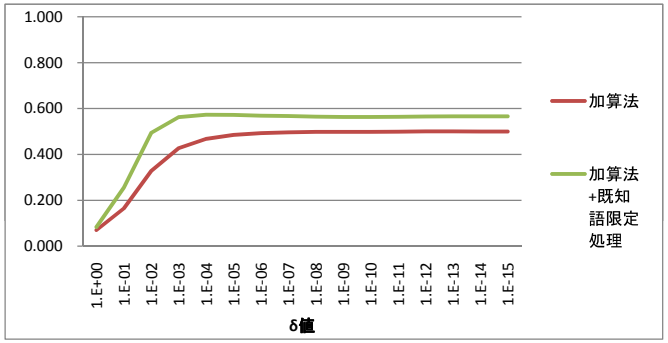

図 2　加算法と加算法+既知語限定処理の結果比較

⋆1 hashtagsjp,http://hashtagsjp.appspot.com/
⋆2 TwitterSearch,http://search.twitter.com/

Vol.2011-IFAT-102 No.1
Vol.2011-DD-80 No.1
2011/3/28

２項目の

$$\sum_i \log \frac{\frac{f(t_i,H)+\delta}{f(t,H)+\delta V(H)}}{\frac{f(t_i,\neg H)+\delta}{f(t,\neg H)+\delta V(\neg H)}} \tag{11}$$

である．よって上式において学習データに出現しなかった単語についての値は

$$\frac{\frac{\delta}{\delta V(H)}}{\frac{\delta}{\delta V(\neg H)}}$$

となる．ここで，$\delta V(H) < \delta V(\neg H)$ であることから，$\delta = 1$ の時分母より分子が大きくなり，結果として１よりも大きい値をとることになる．そのため分全体の評価も正の値を取りやすくなり，再現率が高く，適合率が低い結果となったと考えられる．

- ハッシュタグごとの結果について

ハッシュタグごとの結果に大きな違いがあり，推定しやすいハッシュタグと推定しにくいハッシュタグがあった．

加算法のみの場合において，ハッシュタグごとの結果においてF値が一番高かったものは

表 2 加算法+既知語限定処理を用いたハッシュタグ推定の実験結果

delta	再現率	適合率	F 値
1.0×10^{0}	**0.963**	0.044	0.084
1.0×10^{-1}	0.893	0.149	0.255
1.0×10^{-2}	0.810	0.356	0.494
1.0×10^{-3}	0.748	0.451	0.563
1.0×10^{-4}	0.703	0.483	**0.573**
1.0×10^{-5}	0.673	0.498	0.572
1.0×10^{-6}	0.650	0.506	0.569
1.0×10^{-7}	0.633	0.514	0.567
1.0×10^{-8}	0.618	0.520	0.565
1.0×10^{-9}	0.607	0.526	0.563
1.0×10^{-10}	0.597	0.533	0.563
1.0×10^{-11}	0.589	0.541	0.564
1.0×10^{-12}	0.582	0.550	0.566
1.0×10^{-13}	0.578	0.555	0.566
1.0×10^{-14}	0.573	0.559	0.566
1.0×10^{-15}	0.570	**0.563**	0.566

qanow であった．このタグはウェブサービス「Q&A なう」[⋆1]に関するツイートであり，ツイートの特徴としてサイトへのアドレスが貼り付けられていた．そのドメインが頻出するために，良い結果であったと考えられる．F 値が低かったものは itunes でオープンテストでは再現率 0.750，適合率 0.115，F 値 0.199 であった．しかしクローズドテストでは再現率 1.00，適合率 0.920，F 値 0.964 と他と比べてもかなり高かった．これはもともと itunes のハッシュタグがつけられたものは日本語以外の言語が混じっていることが原因であると考えられる．他のハッシュタグでは日本語が多く，itunes ハッシュタグがついたツイートは英語やその他の言語で構成されるため，itunes タグでしか出現しない単語が多い．そのためクローズドテストでは結果が良く，オープンテストでは結果が悪くなってしまったと考えられる．しかし加算法+既知語限定処理ではハッシュタグ itunes は F 値 0.570 になっていた．とくに適合率が 0.115 から 0.511 になった．これはそもそも未知語が多かったハッシュタグのため，既知語のみの計算によって改善されたと言える．

- 加算法と加算法+既知語限定処理の比較

加算法のみの結果と加算法+既知語限定処理の結果の比較を行うと，表 2 からもわかるように，$\delta > 0.1$ の時，加算法より加算法+既知語限定処理の方が全て F 値が高かった．再現率と適合率の差を見てみると，再現率にあまり差は表れていないが，適合率に大きな差があることがわかった．加算法のみの手法は未知語の計算をしていたため，ハッシュタグに属する方向へひっぱられていたが，既知語限定処理を追加することで改善できたと考えられる．

5. おわりに

本論文では Twitter におけるツイートのハッシュタグの推定を行った．入力を一つのツイートとし，そのツイートに対するハッシュタグを推定する手法としてベイジアンフィルターを用いた．各ハッシュタグごとに２値分類を行い，複数のハッシュタグの推定に対応した．

学習データに出現しない単語の確率が 0 になってしまうゼロ頻度問題に対応するために，加算法・加算法+既知語のみを計算する手法の二つを試した．加算法では，全ての単語の出現数に対して δ を加算した．実験によると $\delta = 1.0 \times 10^{-12}$ で再現率 0.604，適合率 0.427，F 値 0.501 であった．加算法に加え，学習データに全く出現しない単語の場合は計算を行わ

⋆1 Q&A なう:http://qa-now.com/

Vol.2011-IFAT-102 No.1
Vol.2011-DD-80 No.1
2011/3/28

ないという既知語限定処理で実験を行ったところ，$\delta = 1.0 \times 10^{-4}$ で再現率 0.703，適合率 0.483，F 値 0.573 という結果になった．

Twitter のツイートにおけるベイジアンフィルターを利用したハッシュタグの推定は，加算法+既知語限定処理 ($\delta = 0.000001$) で推定を行うことが有効であった．

参 考 文 献

1) 徳永 健伸, 岩山 真. “ 重み付き IDF を用いた文書の自動分類について ”, 自然言語処理研究会報告 94, pp33-40, 1994
2) 小熊 淳一, 内海 彰. “ 語の共起情報を用いた文書クラスタリング ”, 人工知能学会全国大会（第 19 回）, 2005
3) 鈴木 祐介, 松原 茂樹, 吉川正俊 . “ ハイパーリンクを用いた Web 文書の自動分類 ”, NLP2005,B1-7, 2005
4) 高村 大也, 横野 光, 奥村 学. “ Summarizing microblog stream ”, 人工知能学会第 22 回 SWO 研究会, SIG-SWO-A1001-03, 2010
5) Alan Ritter, Colin Cherry, Bill Dolan. “ Unsupervised Modeling of Twitter Conversations ”,Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL, pp172-180, 2010
6) 青島傳隼, 福田直樹, 横山昌平, 石川博. “ マイクロブログを対象とした制約付きクラスタリングの実現 ”, 第 2 回データ工学と情報マネジメントに関するフォーラム, 2010
7) 京都大学情報学研究科-日本電信電話株式会社コミュニケーション科学基礎研究所共同研究ユニットプロジェクト：形態素解析エンジン mecab