Wikipediaのカテゴリ構造を特徴ベクトルに用いたRandom Forestによるショートメッセージ分類

武田 昌大[1]　小林 伸行[2]　椎名 広光[3]
[1] 岡山理科大学大学院 総合情報研究科 情報科学専攻
[2] 山陽学園大学 総合人間学部 生活心理学科
[3] 岡山理科大学 総合情報学部 情報科学科
i15im02tm@ous.jp[1], koba_nob@sguc.ac.jp[2], shiina@mis.ous.ac.jp[3]

1 はじめに

現在，Twitterでは1日に5億件を超えるショートメッセージ(Tweet)が投稿されており，多様な情報の源となっている．このことから，Twitterを対象としたテキストマイニング，ユーザの特性分析等の研究が盛んに行われている．特に，Tweetを用意したカテゴリに対して自動的に分類する研究は，サービスへの応用が効きやすい上，自然言語処理の発展の面でも重用である．

カテゴリ分類に関する手法の参考例としては，Naive Bayesを用いてTweetをトピック別のカテゴリに分類する研究[1]がある．Naive Bayesは学習や識別で高速に動作し，識別精度も高いことから，実用的な文章分類手法として広く使用されている．

しかし，ラベル付きデータを直接モデルの構築に用いる一般的な学習手法では，訓練データの作成に膨大なコストが掛かる懸念がある．さらに，Tweetのようなショートメッセージデータにおいては，得られる素性が限定的であるため，少ない特徴量に分類精度が大きく左右されてしまう．

そこで，本研究ではTweetの特徴ベクトルをWikipediaのカテゴリ構造を学習させたNaive Bayesにより生成し，さらに生成した特徴ベクトルをRandom Forestで学習させることで，素性の拡張を図り，少量のラベル付きデータから簡単かつ高精度なショートメッセージ分類を試みる．Random Forestは学習や識別が高速であり，ノイズに対して頑健であるというメリットを持つことから，物体認識や文字認識といった分野で幅広く使用されている．一方で，ランダム性を確保しつつ学習を収束させる必要があるため，スパースな特徴ベクトルを用いることが多い自然言語処理における使用例はあまり多くない．しかし，TweetがWikipediaの各カテゴリにそれぞれ所属する確率を特徴量としてベクトル表現できれば，素性情報が拡張されると共に，密な特徴ベクトルが生成され，十分な訓練データが得られると考えられる．さらに，豊富なカテゴリ情報を利用した決定木による弱学習機をRandom Forestにより大量に生成するため，たとえ一部誤った特徴を抽出してもそれらの影響が小さくなるように学習するモデルが構築できると考えられる．カテゴリの所属率は従来の研究でも多く用いられてきたNaive Bayesにより算出する．また，注目カテゴリに周辺カテゴリの確率平均値を加算した値を特徴量として特徴ベクトルを生成する．評価実験では，SVM等による精度と比較して，モデルの考察を行う．

2 関連研究

少量のラベル付きデータから大量のラベル付きデータを自動生成する方法としてdistant supervision[2]という学習の手法がある．これは，Wikipedia等の外部情報リソースを用いて知識ベースを構築し，ラベル無しコーパスに対して，知識ベースに保持しておいた関係に適合する文に自動的にラベル付けを行う手法である．distant supervisionによるTweetを対象とした研究も盛んに行われており，例えば，ラベル情報をリッチにすることで，Tweetを感情表現別に分類する研究[3]がある．また，Wikipediaの記事やカテゴリ情報を用いたTweetのカテゴリ分類に関する研究[4]もあり，少量のラベル付きデータから高精度のカテゴリ分類を実現している．これらの研究のように，Wikipediaをはじめとする外部情報リソースをうまく利用することで，訓練データの収集コストの低下と素性情報の拡張による精度の担保が可能になると考えられる．

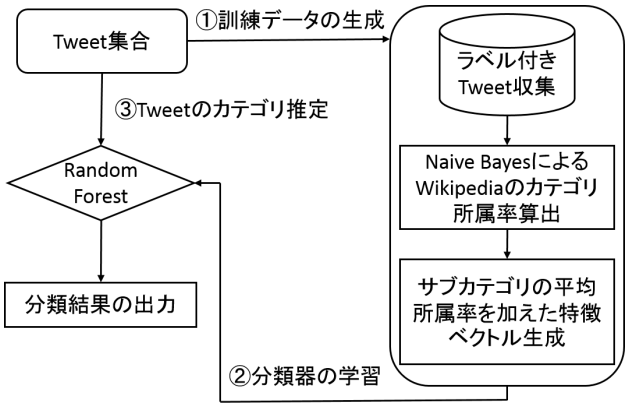

図 1: 提案手法の概要図

3 提案手法の概要

訓練データを生成するため，Tweet 集合から学習させる Tweet を収集し，クラスごとにラベル付けを行う．次に，Wikipedia の記事とカテゴリについて学習させた Naive Bayes を用いて，Tweet の Wikipedia における各カテゴリの事後確率を算出する．その後，各カテゴリのサブカテゴリにおける確率平均値を合算した値を特徴量として特徴ベクトルを生成し，訓練データとする．生成した訓練データを用いて Randam Forest による学習を行い，学習後のモデルを用いて未学習の Tweet のカテゴリを推定を行う (図 1)．

4 Wikipedia のカテゴリ構造を用いた特徴ベクトルの生成手法

Tweet の素性情報を Wikipedia のカテゴリ構造を用いることで拡張する手法について説明する．

Wikipedia では，原則として主題について説明された各ページ (記事) に対して，1 つ以上のカテゴリが割り当てられている．さらに，1 つのカテゴリは関連度の高い複数のカテゴリと紐付けがなされており，全体としてカテゴリのネットワーク構造を形成している．以上のような性質をうまく活用することで，Tweet の意味情報の拡張が可能になると考えられる．

本研究では，分類対象の Tweet を Wikipedia の各カテゴリに対する所属率の特徴ベクトルに変換するため，まず Naive Bayes を用いてカテゴリの所属率を算出する．Naive Bayes では，予め Wikipedia のカテゴリ及びカテゴリに包含される記事の文章を学習させたモデルを分類器として利用する．さらに，Wikipedia のカテゴリリンクを用いて，より適切な特徴量をもった特徴ベクトルを生成する．

4.1 Naive Bayes による所属率の算出

Wikipedia のカテゴリ情報について学習させる Naive Bayes を以下に定義する．

$$P(c|d) = \frac{P(c)P(d|c)}{P(d)} \propto P(c)P(d|c). \quad (1)$$

Naive Bayes では，文書 d が与えられたとき，カテゴリ c が得られる事後確率を算出する．ここで，c は Wikipedia のカテゴリ，d は Wikipedia のカテゴリに包含される記事を表す．事前確率 $P(c)$ は Wikipeida の各カテゴリ c の文章数の総文章数に占める割合であり，以下のように定義する．

$$P(c) = \frac{\text{対象カテゴリ } c \text{ の文章数}}{\text{分類に用いる全カテゴリの総文章数}} \quad (2)$$

尤度 $P(d|c)$ はカテゴリ c が与えられたとき，文書 d を固有名詞の集合モデルとして仮定することにより生成される確率である．文章 $d = (w_1, w_2, \ldots, w_{|n|})$ とし，尤度 $P(d|c)$ を次に定義する．

$$P(d|c) = P(w_1, w_2, \ldots, w_k|c) = \prod_{i=1}^{k} P(w_i|c). \quad (3)$$

$P(w_i|c)$ は，カテゴリ c に出現する固有名詞 w_i の割合を表し，カテゴリ c に含まれる固有名詞 w_i の頻度を用いて以下のように定義する．$N(c, w_i)$ は c に出現する固有名詞の総数である．

$$P(w_i|c) = \frac{N(c_i, w_i)}{\sum_i N(c_i, w_i)}. \quad (4)$$

Tweet T 中の単語をベクトル化して，$T = (w_1, w_2, \ldots, w_{|n|})$ とし，学習した Naive Bayes を用いて，Wikipedia の各カテゴリに対する Tweet T の所属率を以下の定義を用いて算出する．

$$P(c|T) = P(c)P(T|c). \quad (5)$$

4.2 カテゴリリンクを用いた特徴量算出

Naive Bayes で求めた事後確率によるベクトルは一部のカテゴリ c に存在している単語 w_i に共起したスパースなベクトルであり，うまく学習が収束できない懸念がある．さらに，所属率が単語の共起のみによって表されているため，カテゴリに対する Tweet の関連性までを含めるには不十分である．. そこで，Wikipedia のカテゴリ構造を利用して，1 つのカテゴリにおける

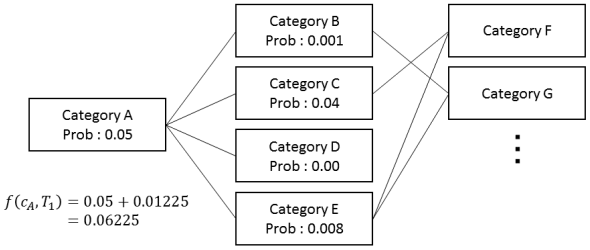

図 2: サブカテゴリを用いた特徴量の算出例

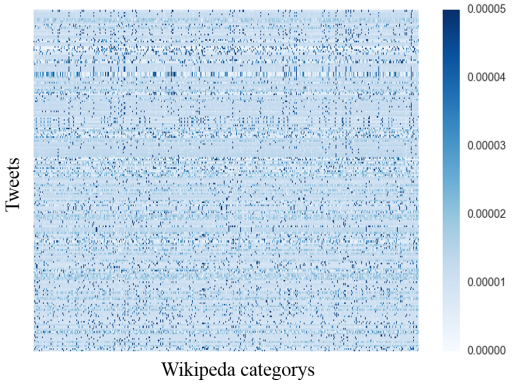

図 3: カテゴリ所属率のヒートマップ

サブカテゴリの事後確率平均値を加算した特徴量を算出する (図 2). カテゴリ c にリンクするサブカテゴリの集合 $\boldsymbol{C(c)}$ として，Tweet T に対するカテゴリ c の特徴量 $f(c,T)$ を次のように定義する．

$$f(c,T) = P(c|T) + \frac{1}{|\boldsymbol{C(c)}|} \sum_{c_i \in \boldsymbol{C(c)}} P(c_i|T). \quad (6)$$

特徴量の変化の具体例として，「遊園地」について話題としている Tweet 集合に対して，Naive Bayes により算出した Wikipedia のカテゴリへの所属率とサブカテゴリの確率平均値を加算した特徴量を表したヒートマップを図 3,4 に示す．

ヒートマップ (図 3) における所属率は，カテゴリを独立したものと扱っているため，カテゴリリンク間の関係は考慮されていない．それゆえ，Tweet に出現するキーワードが存在する一部カテゴリの濃度が際立って高く，その他多くのカテゴリ所属率は均一かつ希薄なベクトルになっていることがわかる．

一方，ヒートマップ (図 4) では，サブカテゴリの確率平均値を加算した特徴量になっているため，密な特徴ベクトルが生成されていることがわかる．特に，Wikipedia の各カテゴリを表す縦軸に縞模様が現れており，Tweet のカテゴリに対する特徴がより顕著に表現されていると考えられる．

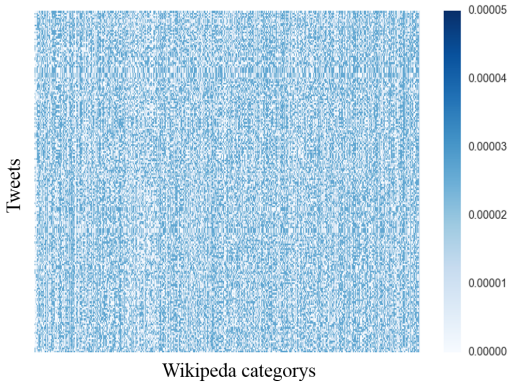

図 4: サブカテゴリ所属率の確率平均値を加算したヒートマップ

5 Random Forest による分類手法

本研究では，カテゴリの所属率の算出より生成した特徴ベクトルデータをブートストラップサンプリングすることによって複数のサブセットを生成し，サブセットごとに多様な決定木を構築する．各決定木の予測値は異なるため，分類問題には各リーフの持つヒストグラムを集計して平均値を求めることにより最終的な予測値を得る．以下に学習処理の手順を示す．

Step 1: 生成した特徴ベクトルのデータ集合 S からブートストラップサンプリングにより B 個のサブセットを生成
Step 2: 特徴量をランダムに選択
Step 3: 決定木 T_b が末端ノードに達するか指定した高さの階層に達するまで次のステップを繰り返す

(i) B の中で情報利得を評価して最適な分割点を選択
(ii) ノードを 2 つの子ノードに分割

Step 4: 学習が未完了なら Step 2 へ戻る
Step 5: 決定木集合 T_B を出力

得られた決定木集合 T_B を用いて，以下の最尤法によりクラスの識別を行う．ここで，定義 (7) は未学習の Tweet T をそれぞれの決定木に与えたときに T_B の確率平均値を返すことを意味する．また，定義 (8) は Tweet T の属性値で最大の値を出力する．

$$P_{ave} = \frac{1}{B} \sum_{b=1}^{B} P_b(c|T). \quad (7)$$

$$C_t = argmax_{c_i}(P_{ave}(c_i|T)). \quad (8)$$

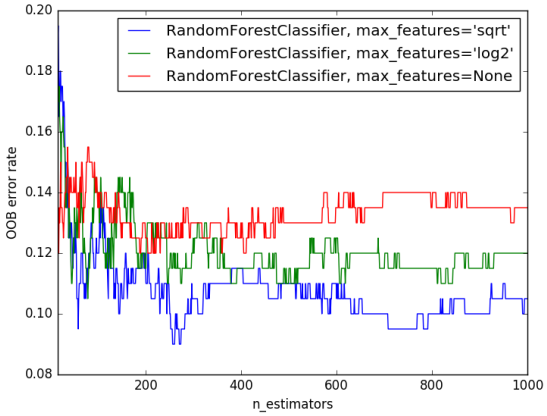

図 5: Random Forest 間の比較

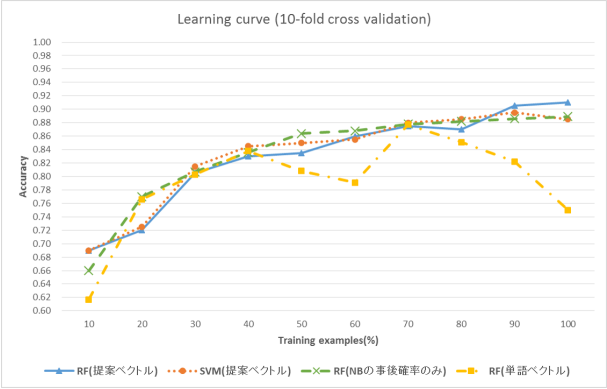

図 6: 精度比較

6 精度実験

提案手法の分類精度について検証を行う．検証用データは「IT」「家電」「映画」「スポーツ」「ニュース」のカテゴリ別にそれぞれ 500 件を人手でラベル付けした Tweet を使用する．ラベル付きデータの作成方法としては，Tweet に付与されたハッシュタグから各カテゴリに関連するものを網羅的に収集し，そこからさらに単なる広告や他サイトへの誘導などのカテゴリとして相応しくない内容で投稿されている Tweet を除外することにより作成する．まず，提案手法における Random Forest の out-of-bag(OOB) 誤り率について検証する．図 5 は，15 から 1000 までの決定木を生成したときの OOB 誤り率の推移である．3 本のグラフはそれぞれ各決定木に用いる最大の特徴量 $\sqrt{B}$, $\log B$, None を表している．図 5 では決定木が 300 を超えたあたりから OOB も安定しており，学習が収束していることがわかる．さらに，OOB が低くなるパラメータとしては，最大の特徴量が $\sqrt{B}$，生成する決定木を 300 程度に設定すればよいことがわかる．次に Cross-validation による精度評価を行う．ここでは提案手法の精度の他に，比較対象として，線形カーネルを用いた SVM，Naive Bayes の事後確率のみを特徴ベクトルとした Random Forest，単語の頻度ベクトルによる Random Forest の精度を図 6 に示す．図 6 の学習曲線から最終的な分類精度として，提案手法による Random Forest がいずれの学習手法よりも精度が上回っている．特に単語ベクトルを用いた場合では，提案手法に比べて精度が安定していない．これは，単語ベクトルは収集した訓練データから直接モデルを構築するため，学習の収束に十分なデータ量を得られなかったことが要因である．一方，提案手法は少数のラベル付きデータからでも Wikipedia のカテゴリ構造を用いた密な特徴ベクトルで学習させることで，分類精度の高い分類器を得られていることが確認できる．

7 まとめ

本研究では，Tweet に対して，Wikipedia のカテゴリの所属率をサブカテゴリまで考慮して算出した特徴ベクトルにした上で，Random Forest による分類を行った．提案した手法は，比較実験において高い精度が得ることができた．今後は，より多様なショートメッセージデータを適用した分類器の構築によって，実用面を考慮した発展をさせていきたい．

参考文献

[1] Kathy Lee, Diana Palsetia, Ramanathan Narayanan, Md. Mostofa Ali Patwary, Ankit Agrawal, Alok Choudhary: Twitter Trending Topic Classification, ICDMW '11, Proc. of the 2011 IEEE 11th International Conference on Data Mining Workshops, pp.251–258, 2011.

[2] Mike Mintz, Steven Bills, Rion Snow, Dan Jurafsky: Distant supervision for relation extraction without labeled data, ACL2009, Vol.2, pp.1003-1011, 2009.

[3] Alec Go, Richa Bhayani, Lei Huang: Twitter sentiment classification using distant supervision, CS224N Project Report: Stanford Vol.1, pp.12-18, 2009.

[4] Masumi Shirakawa, Kotaro Nakayama, Takahiro Hara,Shojiro Nishio：Wikipedia-Based Semantic Similarity Measurements for Noisy Short Texts Using Extended Naive Bayes, IEEE Transactions on Emerging Topics in Computing: Vol.3, pp.205-219, 2015.