SVMを用いたハッシュタグを持つ Twitter日本語文書に対する皮肉文判別

M2012MM020 水野　陽介
指導教員：河野 浩之

1　はじめに

近年, インターネットの爆発的な普及により Web 上には多種多様な情報が存在している．これによって, 政治に関する意見や商品に対する評価であったりと本来ではアンケート等を実施して得なければならなかった情報が Web 上で得ることが可能となってきている．しかしながら, これらの情報は人の手では読み取ることが困難なほどの莫大な量が存在している．そこで機械学習を用いて, 文章をポジティブな意見とネガティブな意見に分類する方法が研究されてきている．機械学習とは普段人間が自身の経験から自然と行っている判断等をコンピュータに行わせるものである．これらを行うに当たって重要な項目として機械学習における分類精度が挙げられるが, 皮肉文の存在により分類精度の低下が少なからず起きてしまう. このため, 皮肉文を機械学習を用いて判別する研究が近年活発に行われてきている.

[1] では, 英語での皮肉文判別の機械学習を行う際, コンピュータに前もって与える訓練データをいかに正確なものに焦点を当て, マイクロブログである Twitter の機能の一つであるハッシュタグを用いた訓練データの作成を行っている. ハッシュタグの利用により, 訓練データとなる皮肉文の判別を発言者に委ねている．これは皮肉文は第三者からでも判別が難しいと考えているからである.

本研究では [1] の考えを元に日本語での皮肉文判別のハッシュタグの有用性について実験を行う．ハッシュタグを用いて

・皮肉文, ポジティブ文, ネガティブ文
・皮肉文, ポジティブ文
・皮肉文, ネガティブ文

以上 3 つの訓練データから皮肉文判別を行い, 日本語における皮肉文とポジティブ文, ネガティブ文との関係を考察する．また, 第三者によってツイートを皮肉分と通常分へ判別した訓練データを作成しハッシュタグを用いて作成した訓練データとの比較評価を行う．分類方法としては tf-idf 法を用いて各ツイートに重み付けを行うと共に, SVM (Support Vector Machine) による皮肉文と通常文への分類を行う．この時データの偏りによる不正確な分類精度が計算されることを防ぐため,K 分割交差検定を用いる.

本稿の流れは, 第 2 節では皮肉文に関する先行研究の説明, 第 3 節で分類までの流れを示し, 第 4 節で具体的な特徴付けおよび分類に用いるアルゴリズムの説明を行い, 第 5 節で実験結果の考察を行うと共に, 第 6 節で全体のまとめ及び今後の課題を記述する.

2　皮肉分判別における関連研究

2.1　日本語での皮肉文判別に関する研究

[3] では皮肉表現の解釈機構の認知モデル・計算モデルの提案し, 妥当性について示している. これは皮肉表現には話し手の期待, 期待と現実の不一致, 否定的態度を暗黙的に提示しているという考えの下選択方式で皮肉であるか否かを判別するものである.

ただし, この手法では選択方式になるため精度の向上を目指すにはシステム全体の規模が大きくなってしまうという問題点や, 皮肉表現には多種多様は表現が存在するため多くを解釈することは非常に難しい.

2.2　英語での製品レビューに関する皮肉分判別

[2] は通販サイト Amazon を用いた製品レビュー (本, 音楽プレーヤー, PC , 携帯電話などの 120 種類の製品から 66000 件のレビューを収集) における皮肉文判別を行っている. 訓練データの作成において, 語彙の特徴付けとして SASI アルゴリズムを提案し, 分類方法として k 近傍法を用いている．結果は, 精度 91.2 %, 再現率 75.6 %, F 値 82.7 %であった.

2.3　英語でのハッシュタグを用いた皮肉文判別

[1] では Twitter の機能の一つであるハッシュタグを利用してコンピュータに与える訓練データの質の向上を目指している．表 1 のように各ワードのハッシュタグよりポジティブな文, ネガティブな文, 皮肉文を集め (各 300 ツイートを収集) ポジティブネガティブ文から皮肉文の判別を行っている．特徴づけには [2] で用いられた Semi-supervised Algorithm for Sarcasm Identification, 分類方法では SVM を採用し, 再現率で比較を行っている．結果は, ポジティブ文と皮肉文で分類を行った時再現率 66.39 %, ネガティブ文と皮肉文の時再現率 68.56 %, ポジティブとネガティブを合わせた文と皮肉文の時 61.22 %であった.

表 1　各ワードのハッシュタグを用いたツイート収集一覧

ツイート	参照するハッシュタグ
皮肉文	#sarcasm, #arcastic
ポジティブな文	#happy, #joy, #lucky など
ネガティブな文	#sadness, #angry, #frustrated など

3　本研究の流れ

本節では研究のおおまかな流れおよび研究で使用するデータセットの説明を記す.

3.1　分類までの流れ

皮肉文分類の流れを図 1 を用いて説明する

(1)Twitter より各感情を意味するハッシュタグを含むツイートを収集

(2) ツイートを適切に分類出来るよう必要のない文章を削除する. また収集したツイートを第三者によって皮肉文, 通常文への分類分けを行う (ハッシュタグを使用したものとの比較を行うため)

(3) 収集した各ポジティブ, ネガティブ, 皮肉文のツイートを tf-idf によって特徴付けを行う

(4)SVM によって分類を行い, 精度, 再現率,F 値より結果を表示する

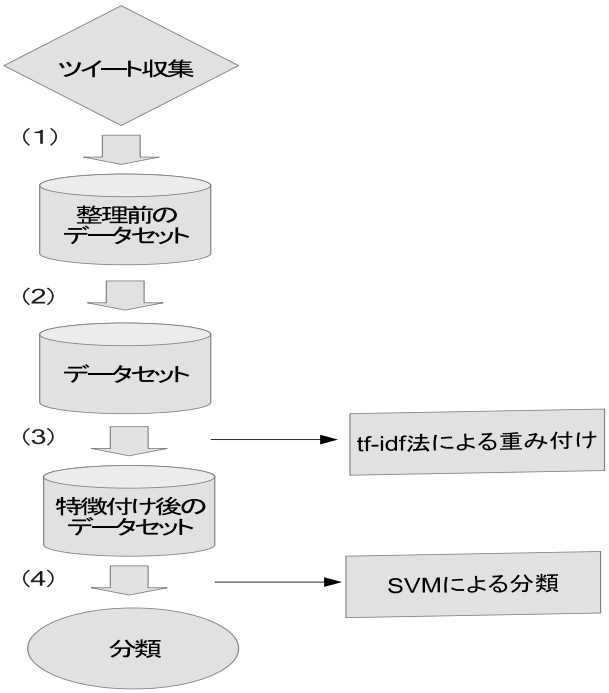

図 1　分類データフロー

3.2　ハッシュタグの利点

本研究では皮肉文判別に用いる訓練データの質の向上を目的としている. 例えば, 「あの政治家はホントに凄いなぁ」という文章があったとする. これを発言者以外の第三者が皮肉文か通常文かの判断を行うことは困難であると考えられる. 間違った判断をして訓練データを作成すれば, その分皮肉文の分類の精度も低くなるのではないかと考えられる. この時ハッシュタグを用いれば発言者に皮肉文か通常文かの判断を直接委ねることができ, より質の良い訓練データの作成が可能になるのではないかと考える.

3.3　データセット

本研究では Twitter よりハッシュタグで検索をかけ, 表 2 を参考に皮肉, ポジティブ, ネガティブなツイートをそれぞれ 100 ずつ集めた. 今回の実験ではツイート収集における必要情報はツイート内容のみであるため, 直接手作業で収集を行った. 収集の際, 明らかにこちらがハッシュタグに求めているものと関係のないものと判断できるものの収集は控えた. また, 各ツイート収集時に使用するハッシュタグは以下の表 2 を用いてツイートの検索を行った.

表 2　収集時に用いたハッシュタグ一覧

ツイート	参照するハッシュタグ
皮肉文	#皮肉
ポジティブな文	#ポジティブ, #楽しい
ネガティブな文	#ネガティブ, #悲しい

図 2 はツイートの整理を行った後の皮肉文ツイートの一部である.

	A	B	C	D	E	F	G	H	I
1	いやー、涼しくなったね～。死にかけるくらいに暑かったのにね～。								
2	自己開示すると良いことあるよね、色々。								
3	ほんとにあった話は怖いですね								
4	お客様は神様だな								
5	何が怖いだレバーでも食ってろ								
6	教育委員会が規制進めるから歴史という教科が無くなってしまうよ！ やったね学生！課題が減るよ！								
7	東北レゲエ祭行きたかったーー！！								
8	あ じゃあ、明日の朝にしようかな どうせ家族は俺を置き去りにして旅行だし～								
9	色白で毛深いのと、ナイスバディなブス								
10	タバコ吸わないでいるとここまで理不尽・差別的な思想になってしまうのか…。やだな、今からでもタバコ吸								
11	一般学部生って単位落とし放題の定額サービスだからいいよね。								
12	今年もラウパ開催決定した様だが、公式トップの"日本最大にして唯一のメタルフェス"にわろたwww								

図 2　皮肉文データセット一例

4　特徴付けと分類および評価

本節では分類前に行う前処理, 特徴付けの方法と分類および評価方法について述べる.

4.1　ツイートの整理

4.1.1　不要な内容の削除

分類を行う際に不必要な情報が含まれているため, 内容と関係のない以下の様な内容を省く.

・ハッシュタグ

・URL

・@user(Twitter の機能の一つで, 特定のユーザの返信を行える)

・顔文字

etc

4.1.2　ハッシュタグを使わない分類

ハッシュタグを用いる場合, そのツイートが皮肉文か否かの判別を行う必要はない. しかし人間に訓練データを作成してもらう場合, そのツイートが皮肉文であるかどうかの評価を付けて置く必要がある. ここでは第三者に皮肉文であるか通常文であるかの分類を行ってもらった. 評価に使用するデータセットはハッシュタグとの比較時に評価しやすいよう, ハッシュタグより集取したデータの中からランダムに選択を行った. 本研究では, 三名を対象に実験を行っている.

4.2 特徴付け

本節では, 分類における特徴付けに用いるアルゴリズムについて説明する. 本研究では形態素解析 MeCab にて使用可能な tf-idf を用いる.

tf-idf とは tf 法と idf 法を組み合わせることによって, ある単語 t における特定の文書 d の重みを計算する手法であり, 情報検索やテキストマイニング, 機械学習などの分野で利用される.tf 法と idf 法の意味に関して以下に示す.

tf(term frequency) は単語出現頻度を示し, その文章中で特定の単語が出現した回数を表す. 文章中に頻出頻度が多いということは, それだけその単語が重要であるということがわかる.

idf(inverse document frequency) は逆文書出現頻度を示し, その訓練データ内のその文章を含む文書数の自然対数を示す.

重み付けの計算式は式 1 のようになる.

$$w_{t,d} = tf_{t,d} \cdot \log \frac{N}{df_t} \tag{1}$$

ここで,N は訓練データ全体のデータである. また,$tf_{t,d}$ は文書 d 中に t が現れた数で,df_t は N 個の文書中に単語 t が現れた数を示す.

例えば, 検索エンジンに当てはめると

tf = 検索した時のヒットページ数

idf = $\log_2$(検索エンジンの総ドキュメント数/df)

と考えることが出来る.

本研究では, R ソフトに MeCab のパッケージのインストールを施し特徴付けを行った.

4.3 分類

4.3.1 Support Vector Machine

SVM とは機械学習であり, パターン認識能力において, 最も優秀な学習モデルの一つである. ここでは本研究で行う 2 種類のデータの分類を行うアルゴリズムについて説明を行う.

□と○の 2 種類のデータを式 2 が存在するとする.

$$x = [x_1, x_2]^T \tag{2}$$

式 2 のような 2 次元データが存在する時, 与えられたデータを用いて SVM が分類方法を学習している. 図 3 のように分離平面と 2 種類のデータとの間の距離 (マージン) が最大となるような分離超平面にすることで, 最も汎用性の高い超平面となる. マージンの最大化を定式化することによって最適解を唯一に定めることが可能となり, 所的な最適解に陥ることがないため正確な分類が可能となる.

4.3.2 K 分割交差検定

一度の試行を行った時, 学習用データの訓練用データに偏りが生じていた場合, 不適切な結果が生じてしまう可能性が考えられる. このような状況を対象する為に用いられる手法として K 分割交差検定が存在する. これはデータセットを K 個に分割し, その一つを学習用データ, 残りの K-1 個を訓練用データとする. それらを K 回検証を行い, 得られた K 回のデータの平均を推定結果とするものである. 表 3 が K=3 の例である. 尚, 本研究では先行研究の論文で行われた K=5 をそのまま利用する.

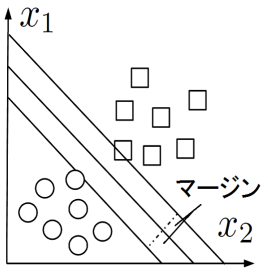

図 3 SVM におけるマージンの最大化

表 3 K 分割交差検定 (K=3 の時)

	A	B	C
一回目	評価用	学習用	学習用
二回目	学習用	評価用	学習用
三回目	学習用	学習用	評価用

4.4 評価

実行結果の評価は先行研究で使用されていた精度, 再現率, F 値の 3 つより行う. 各評価方法の意味と計算法を以下で示す.

図 4 は分類された結果を表示する図となる. 例えば予測が正 (皮肉文であると予想) だった時真の結果が正 (実際にそのツイートが皮肉文である) という結果が出たツイートの数は TP 個となる.

精度:予想したデータのうち, 実際に正であるものの割合を示す. 式は 3 となる.

$$\text{精度} = \frac{TP}{TP + FP} \tag{3}$$

再現率:実際に正であるもののうち, 正であると予測されたものの割合を示す. 式は 4 となる.

$$\text{再現率} = \frac{TP}{TP + FN} \tag{4}$$

F 値:精度と再現率の調和平均を示す. 式は 5 となる.

$$F\text{ 値} = \frac{2 \cdot \text{再現率} \cdot \text{精度}}{\text{再現率} + \text{精度}} \tag{5}$$

表 4 評価図

		データセット	
		正	負
予測結果	正	TP	FP
	負	FN	TN

5　評価

本節ではハッシュタグを用いた皮肉文判別の分類結果について述べる.

5.1　実験環境

・OS:Windws7 64bit
・CPU:Intel Core i5-2500K
・メモリ:4.00GB
・ハード R version3.0.0(MeCab,kernlab)

MeCab では tf-idf による文章の特徴づけを行い, kernlab で SVM による皮肉文の分類を行った.

5.2　分類するデータセット

表 5 では皮肉文とポジティブ文, ネガティブ文の皮肉文判別に使用するデータセットである. 皮肉±ではポジティブ文, ネガティブ文からランダム選んだ 150 ツイートを使用している.

表 5　皮肉, ポジティブ, ネガティブによるデータセット

データセット	内容
皮肉＋	皮肉文 300, ポジティブ文 300
皮肉文－	皮肉文 300, ネガティブ文 300
皮肉文±	皮肉文 300 ポジティブ文 150 ネガティブ文 150

表 6 ハッシュタグと第三者によって定義された訓練データを用いた皮肉文判別に用いるデータセットである. 皮肉 (#) は皮肉±と同じものであり第三者にはこのデータを用いて皮肉文と通常文への仕分けを行ってもらっている.

表 6　第三者との比較によるデータセット

データセット	内容
皮肉 (#)	皮肉文 300 ポジティブ文 150 ネガティブ文 150
皮肉 (人 1,2,3) －	皮肉文 300, 通常文 300

5.3　分類結果

3 つの訓練データを分類した結果の値を表 7 に表す. 皮肉±が皮肉＋, －に比べて精度が低い結果となっている. 皮肉文判別がポジティブ, ネガティブ文の混合によって分類の低下が起きていることが分かる. これは皮肉文がポジティブやネガティブな言い回しが使われているからだと思われる. これにより分類先にポジティブ, ネガティブ文が混ざっていることにより誤った分類がされていると考える.

次に第 3 者 3 名に分類してもらった訓練データの皮肉文, 通常文の内訳が表 8 で分類結果の比較が表 9 となる.

表 8 より皮肉文 (#) つまりハッシュタグを使った分類が最も分類精度が高い結果となった. 第三者に判別してもらったデータをみるとどちらかに偏りが出ているもの程精度が低いようになっている.

表 7　実行 1 結果

データセット	精度	再現率	F 値
皮肉＋	0.655	0.634	0.642
皮肉－	0.663	0.641	0.653
皮肉±	0.601	0.576	0.580

表 8　第三者の仕分け結果

データセット	皮肉文	通常文
皮肉 (#)	300	300
皮肉 (人)	185	415
皮肉 (人 2)	254	346
皮肉 (人 3)	437	163

表 9　実験 2 結果

データセット	精度	再現率	F 値
皮肉 (#)	0.601	0.576	0.580
皮肉 (人)	0.551	0.526	0.543
皮肉 (人 2)	0.589	0.564	0.571
皮肉 (人 3)	0.537	0.502	0.516

6　まとめと今後の課題

本研究では, ポジティブ文とネガティブ文が皮肉文判別に影響が出ていることがわかった. また, ハッシュタグを用いることでの皮肉文判別における訓練データの質の向上を示すことが出来た. 今後の課題として, 皮肉文判別の分類精度自体は高いとは言えないためポジティブやネガティブのような発言を用いた皮肉文の傾向を調査し, それらを反映した皮肉文の特徴付けモデルを作成することが考えられる.

参考文献

[1] Smaranda Muresan, Nina Wacholder, “ Identifying sarcasm in Twitter: a closer look. ” Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT 2011), pp.581-586, 2011.

[2] Dmitry Davidov, Oren Tsur, Ari Rappoport, “Semi-supervised recognition of sarcastic sentences in Twitter and Amazon, ” Proceedings of the Fourteenth Conference on Computational Natural Language Learning, pp.107-116, 2010.

[3] 内海彰, “ アイロニー解釈の認知・計算モデル, ” 情報処理学会論文誌, Vol.41, No.9, pp.2498-2509, 2000.