商品評価ツイートからの属性語自動抽出手法の提案

駒田 康孝† 山名 早人‡§

†早稲田大学基幹理工学研究科 〒169-8555 東京都新宿区大久保 3-4-1
‡早稲田大学理工学術院 〒169-8555 東京都新宿区大久保 3-4-1
§国立情報学研究所 〒101-8430 東京都千代田区一ツ橋 2-1-2
E-mail: {ykoma, yamana}@yama.info.waseda.ac.jp

あらまし Web 上に存在する評判情報は，消費者の意思決定や企業のマーケティングにおいて有用な指標となっている．中でも近年利用者数が増加している Twitter は，有用な情報源として注目されている．しかし，既存のサービスでは評価に使う語のみをツイートの抽出対象としており，商品のどんな属性に対して評価しているかを把握することは難しかった．さらに，商品ごとに評価される属性は異なるため，属性語抽出の為に予め属性語辞書を用意するにはコストが大きい．また，既存の属性語抽出手法では，Twitter における多様な表現に対応しきれていないという問題があった．そこで本研究では，Twitter における表現の多様性に対応しつつ属性語，評価語の対を取得するために，基本的な評価語辞書のみを利用し属性語の自動的な抽出を行い，属性語辞書を構築する手法を提案する．

キーワード テキストマイニング，Twitter，情報抽出，評判情報

1. はじめに

近年，インターネット上の掲示板やレビュー投稿サイトなどに，ある商品に対する意見や評判を含む文章を投稿する動きが活発になっている．例えば，インターネットショッピングサービスである，Amazon[1]における商品レビューは以下の図 1 の様になっている．

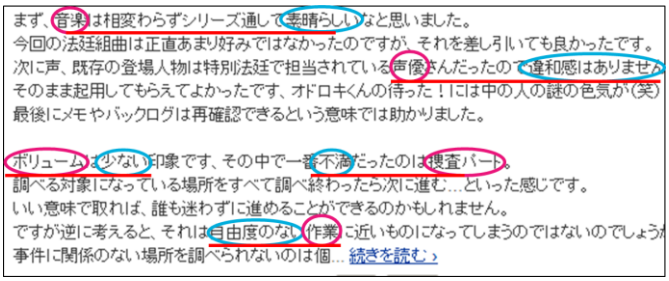

図 1 Amazon における商品レビュー例

図 1 の例では，赤枠に書かれた点に対して，青枠に書かれた意見が述べられている．このようにユーザが商品に対して感じた意見や感想を含む文章は，他のユーザが商品を購入する際の意思決定をする際に有用な指標となる．また，企業が自社商品に対する意見や感想を収集することで，マーケティングへの有用な指標とすることも可能である．しかし，インターネット上に存在する文章数は膨大であり，その全てに目を通すことは難しい．そのため，意見や感想が述べられた文章集合から，商品のどの側面がどのような評価を得られているのかという情報を，自動的に収集・解析する技術についての関心が高まっている．ここで，小林らの研究[3]に基づき対象とする商品名を対象語，商品の特徴を表している側面を示す語を属性語，その属性語に対する評価を表す語を評価語と定義する．

本稿では，マイクロブログの一つである Twitter[2]に投稿された文章(以下ツイート)の内，商品に対する意見や評判を含む文章(以下商品評価ツイート)から属性語自動的に抽出するタスクについて取り組む．既存研究では，レビューサイトに投稿された文章から属性語の抽出を行う研究や[3][4][5][7]，Web ページ集合を対象として属性語の抽出を行う研究[6]など，様々なデータを対象とした研究がなされてきた．このとき，レビューサイトを対象とした研究では，予め商品の評価視点やジャンルがある程度把握でき，文章も大きく崩れることが少なかった．しかし，Twitter においては予め商品に関する知識を取得することはできず，ツイートに関しては崩れた文章であることが多い．また，Weblog を対象とした研究と比較した場合，Twitter は 1 文章が 140 字までという制限があるため短文で投稿されることが多く，文章の長さという点で差異がある．このように，文章の性質や事前知識の差が生じるため，既存の手法をそのまま Twitter に適用しても有用な結果を得にくいのが現状である．

これらの問題を解決するため，提案手法では，係り受け解析を行う際に，ツイートに含まれる表現に適合するパタンを設定し，ノイズとなる表現のフィルタリングを行う．そして，一般的な評価語と係り受け関係にある語の抽出を行い，対象とする商品との関連度を計算することで，未知である属性語の同定を行う．同様にして，同定された属性語を元に，未知である評価語の同定を行う．これらの作業を繰り返して行うことで，Twitter に投稿された商品評価ツイートから，精度の良い属性語の自動抽出を実現する．

本稿では，次の構成をとる．まず，2 章で関連研究について述べ，3 章で提案手法を説明する．次に 4 章で提案手法についての実験と評価を行い，最後に 5 章

でまとめを述べる.

2. 関連研究

本節では，評判情報を扱う研究の中でも，属性語，評価語の抽出について扱った関連研究について述べる.

属性語，評価語の抽出に関する研究として，まず小林ら[3]の研究があげられる. 小林らは対象語, 属性語, 評価語の共起パタンに基づく，属性，評価表現の半自動的収集方法を提案した．共起パタンは，「“属性語”が/は/も/に/を“評価語”」というようなテンプレート表現をなすものとしている．小林らの手法では，日本語係り受け解析器の CaboCha[8]を用いた係り受け解析を用いることで，商品の評価の基準となる属性語，属性語に対する評価語を取得する．しかし，小林らの手法では，取得した属性語，評価語のうち，どの属性語，評価語が適したものであるかを人手で判別する必要があり，辞書生成のコストが大きい．同様に平山ら[4]，菊池ら[5]，丸山ら[6] も，CaboCha を用いた係り受け解析を用いて，独自に設定した共起パタンと適合する語を抽出し,属性語･評価語の同定を行っている．また谷本ら[7]は，レビューサイトに投稿された文章を用いて評価表現辞書を自動的に作成し，文章の極性判定を行う手法を提案した．属性語の抽出には，レビューサイトで予め定められた評価語とその他の語の関連度を語の共起頻度によって定義し，関連度が高い語を属性語と同定した.

3. 提案手法

本節では, 提案手法である Twitter に投稿された商品評価ツイートから属性語を自動的に抽出する手法について述べる．既存研究においては，主に次に挙げるような問題が存在した.

- 評判情報抽出の精度を向上するために対象とする商品に関する語を人手で収集するのは，コストが大きい．また，Twitter に投稿されるような多様な表記の属性語に対応できない.
- Twitter には特有な表現や, 崩れた文章が多く存在するため，既存の係り受け解析，形態素パタンによる属性語抽出をそのまま適用しても精度が低くなってしまう.

これらを踏まえた上で提案手法では，人手による辞書構築のコストを避けるため，初期評価語辞書として商品に依存しない一般的な評価語群を利用し，評価語と属性語の共起頻度を元に商品に関係が深い属性語の抽出を行う．また，Twitter に特有な表現や，崩れた表現に多く出現するノイズの除去を行うことで，属性語抽出精度の向上を狙う. 提案手法の流れを以下に示す.

Step1. 商品評価ツイートの収集を行う.

Step2. Step1 で収集したツイート集合中のノイズ除去を行う.

Step3. Step2 より得られたツイート集合に対し係り受け解析を行う.

Step4. Step3 より得られた結果から，属性語候補を選択する.

Step5. Step4 より得られた属性語候補に対して，対象とする商品との関連度を計算する.

Step6. Step5 で算出した関連度が閾値以上のものを対象の商品の特徴を表す属性語と同定する.

Step7. Step4 で選び出す結果を評価語候補として，同様に Step6 まで実行する.

Step8. Step1 から Step7 を n 回(n は任意)繰り返す.

ここで本手法では，乾ら[9][10]が公開している日本語評価極性辞書(用言編)に含まれている評価属性を持つ語を評価語における初期辞書として保持する. 以下，順に詳細を説明する.

3.1. 商品評価ツイートの収集

まず，属性語を抽出する対象となる商品評価ツイートの収集を行う．具体的には，Twitter に投稿されたツイートの内，対象語を含むツイートの抽出を行う．そして，対象語を含むツイートの内，評価語をもつツイートを商品評価ツイートと定義し，抽出を行う.

3.2. 商品評価ツイートのノイズ除去

3.1 で収集した商品評価ツイートについて，ノイズとなる表現を除去することで，係り受け解析における精度向上を目指す．除去を行う対象となる表現を以下に示す.

① 「URL」を含むツイート全体
② 「リプライ」「リツイート」「ハッシュタグ」を含むツイートの当該表現部分
③ 特定の記号を含むツイートの当該記号
④ 対象の商品に関わらず，頻出する語の除外

ここで③の対象となる記号は，具体的には

(1) ”！”,”？”,”・”
(2) ”『』””「」”,”［］”,”【 】”,”（）”,
(3) ”。”

である. このうち(2)に関しては全て除去を行い，(1)に関しては全て”。” に置換した後，連続して出現する(3)を一つに集約する. また，④の一般的によく使われる表現の除去としては，無差別に取得したツイート群中に現れる頻度の高い”名詞”，”未知語”を本手法で除外の対象としている.

3.3. 商品評価ツイートに対する係り受け解析

3.2 の処理により得られるツイート集合に対して係り受け解析を行う. 係り受け解析器には CaboCha を用いる．このとき，係り受け解析の際に用いられる形態素解析器 MeCab[11]が用いる辞書について，はてな Keyword リスト[1]，Wikipedia タイトルリスト[1]を用いて語彙の拡張を行う.

3.4. 属性語候補の選択

3.3 の処理により得られた係り受け解析結果から，属性語の候補を選択する．具体的には，まず評価語と係り受け関係になっている文節の抽出を行う．次に，抽出した文節に対し MeCab を用いて形態素解析を行う．形態素解析の結果，文節中に品詞が「名詞-一般」「名詞-固有名詞」「名詞-サ変接続」「未知語」となる語が含まれていた場合，それらを属性語候補とする．ただし，以下に示す形態素の並びは，一つの語として属性語候補に加える．

(1)	[接頭詞-数接続]－[名詞-数]
(2)	[接頭詞-名詞接続]－[名詞-一般]
(3)	連続して出現する[名詞-一般]

また，係り受け解析の結果によらず，以下のパタンに当てはまる形態素列を全て属性語候補とする．

(1)	[評価語]－[名詞-一般／名詞-固有名詞／名詞-サ変接続／未知語]
(2)	[名詞-一般／名詞-固有名詞／名詞-サ変接続／未知語]－[評価語]
(3)	[評価語]－[接頭詞-数接続]－[名詞-数]
(4)	[評価語]－[接頭詞-名詞接続]－[名詞-一般]
(5)	[名詞-一般／名詞-サ変接続／名詞-固有名詞／未知語]－[名詞-接尾]－[評価語]
(6)	[接頭詞-名詞接続]－[名詞-一般]－[評価語]
(7)	[接頭詞-数接続]－[名詞-数]－[評価語]

3.5. 属性語候補の商品に対する関連度の計算

3.4 の処理により得られた属性語候補が，どの程度対象とする商品と関連があるのかを算出する．関連度の計算には，属性語候補と評価語の共起頻度，属性語候補の出現頻度を用いる．対象商品を示す対象語 x に対する属性語候補 i の関連度R_{xi}を，以下の式(1)の様に表す．

$$R_{xi} = \frac{f_{xi}}{N_x} \times \frac{c_{xi}}{M_x} \qquad (1)$$

ここで，N_xは対象語 x を含むツイート集合，M_xは対象語 x を含む商品評価ツイート中の，評価語辞書に含まれるいずれかの評価語へ係り受ける文節の総数とする．また，N_xにおける属性語候補 i の出現頻度をf_{xi}，M_xにおける属性語候補 i の，評価語辞書に含まれるいずれかの評価語との共起数をc_{xi}とする．こで求めたR_{xi}の値が，任意に定める値 D 以上であった場合，属性語候補 i を対象 x の特徴を示す属性語として，属性語辞書に追加を行う．

3.6. 商品評価ツイートからの評価語の抽出

属性語辞書への追加が終了した後，3.4 項の処理と同様に係り受け解析を用いて評価語候補を抽出し，関連度の計算，評価語辞書への追加を行う．抽出する表現パタンを以下に示す．

(1)	[動詞-自立]－[形容詞-非自立]
(2)	[名詞-ナイ形容詞語幹]－[助動詞-ナイ]
(3)	[名詞-ナイ形容詞語幹]－[助詞-格助詞-一般]－[形容詞-自立]
(4)	[形容詞-自立]－[助動詞-ナイ]
(5)	[形容詞-自立]－[助詞-係助詞]－[助動詞-ナイ]
(6)	[属性語]－[形容詞／名詞-形容動詞語幹／名詞-ナイ形容詞語幹]
(7)	[形容詞／名詞-形容動詞語幹／名詞-ナイ形容詞語幹]－[属性語]

3.7. 繰り返しによる辞書の拡張

3.6 の処理が完了し，属性語，評価語の辞書追加を行った結果を元に，3.1 から 3.6 までの処理を n 回(n は任意)繰り返し行うことで辞書拡張を行う．

4. 実験・評価

実験では，以下の 4 項目について検証を行った．

- ノイズ除去の有無における精度変化
- 既存研究[2]との精度比較
- 商品評価ツイート数の増減による精度変化
- 繰り返し回数 n の変化による抽出語彙数の変化

ここで精度は，提案手法を n 回繰り返した時点での属性語辞書に存在する属性語の関連度の内，上位 k 個を選択した場合における正答率を人手により判別し，Precision@k として算出した．

4.1. 実験データ

本研究では，商品評価ツイートとして，3 種類のゲームタイトルを含むツイートを，当該ゲームタイトルの発売日から 1 ヵ月間収集を行い，実験に利用した．実験に利用したツイートの総数は，18,400 ツイート．その内商品評価ツイートとして利用したツイートの総数は，4,273 ツイートであった．

4.2. 評価実験

評価実験として設定した 4 項目について，抽出した属性語の関連度上位 50 個について精度検証を行った結果を以下の図 1，図 2，図 3，表 1 に示す．

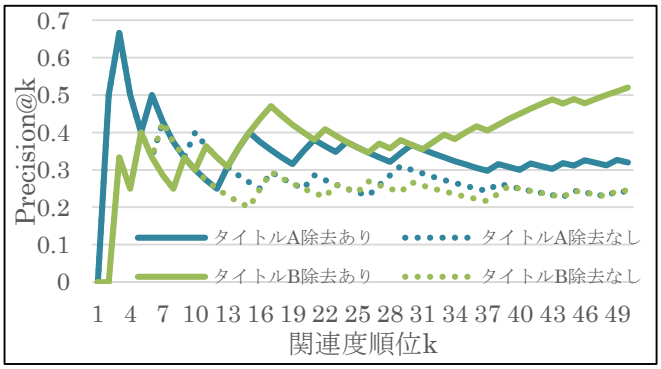

図 2 ノイズ除去における精度検証

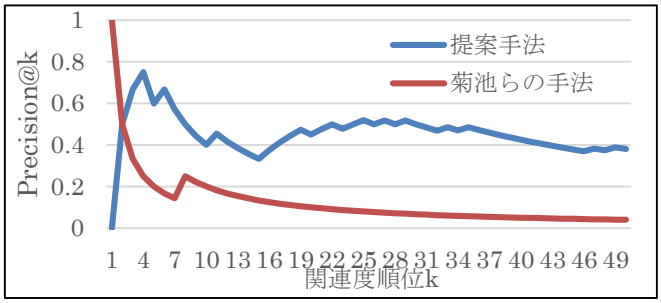

図 3　既存手法との精度比較

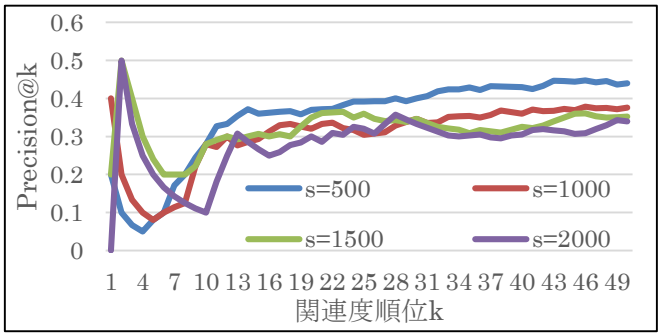

図 4 商品評価ツイート数 s の変化による精度比較

表 1　繰り返し回数 n による抽出語彙変化の調査

ゲームタイトル	A	B	C
n=1 属性語数	**951**	**377**	**438**
n=1 かつ R＞D	**269**	**284**	**166**
n=2 **属性語数**	**1490**	**586**	**577**
n=2 **かつ R＞D**	**263**	**167**	**160**
n=3 **属性語数**	**12**	**0**	**3**
n=3 **かつ R＞D**	**18**	**1**	**6**

図 1 から，ノイズ除去の行った場合，精度は平均して 10%程度向上しており，適切にノイズの除去が行えた事が示されている．また，図 2 より既存研究との精度比較結果をみると，既存手法との精度差は最大で 30%程度ある．これは，既存手法は Twitter に特有な表現への対処を行っておらず，表現パタンがレビューサイトのテキストに最適化されていたため，Twitter に対して手法を適用した場合には有効な結果が得られなかったものと考えられる．図 3 では，商品ツイート数を 500 から 2000 まで 4 段階に分けて精度を比較した結果を示した．しかし，結果としてはツイート数を変化させた場合にも精度は大きく変わらなかった．本来，ツイート数を増加するにつれ精度が向上することが望ましい．そのため，更に大量のツイートに対して手法を適用した場合の精度を調査することが今後の課題となる．また，表 1 をみると，n=2 の試行において，新たに抽出した属性語の数が大幅に増加しており，n=1 において抽出されなかった語を幅広く収集できていると言える．しかし，収集した属性語が増加したにも関わらず，関連度が閾値 D 以上の属性語数はほぼ変化していなかった．これは，関係が無いと判断される語も多く抽出してしまっていることが理由としてあげられる．

5. まとめ

本稿では，商品評価ツイートからの属性語自動抽出手法の提案を提案した．提案手法では，ツイート中に存在するノイズ除去を行うことで，精度低減を防いだ．また，属性語，評価語の出現パタンを設定し，抽出した候補語の関連度を定義することで，対象とする商品に特徴的な属性語の自動抽出を行った．実験の結果，従来手法を Twitter に適用した場合と比較して約 30%の精度改善が見られ，39%の精度で属性語を抽出することが出来た．今後の課題として，より多くのツイートを用いた場合の精度調査，抽出した属性語，評価語対の集約・可視化を行う必要がある．

参　考　文　献

[1] Amazon.co.jp, http://www.amazon.co.jp/（2014 年 1 月 6 日アクセス）．

[2] Twitter, http://twitter.com/ （2014 年 1 月 6 日アクセス）．

[3] 小林のぞみ，乾健太郎，松本裕治，立石健二，福島俊一，“テキストマイニングによる評価表現の収集”，情報処理学会研究報告，NL154-12, pp.77-84, 2003.

[4] 平山拓央，湯本高行，新居学，高橋豊，”属性評価モデルに基づく商品評価の抽出と提示”，DEIM Forum 2011, F2-5, 2011.

[5] 菊池悠太，高村大地，奥村学，”属性-評価ペアを単位とした評判情報の要約”，情報処理学会研究報告，NL-206, No.1, 2012.

[6] 丸山宏，鈴木健之，中村太一，“評判表現のための仕様表現辞書の構築手法” 電子情報通信学会技術研究報告，Vol.108, No.65, pp.35-40, 2008.

[7] 谷本融紀，太田学，”評価属性を考慮した評判情報の可視化”，情報処理学会研究報告，DBS-151, No.12, 2010.

[8] CaboCha/南瓜：Yet Another Japanese Dependency Structure, https://code.google.com/p/cabocha/.

[9] 公開資源/日本語評価極性辞書 - 東北大学 乾・岡崎研究室，http://www.cl.ecei.tohoku.ac.jp/index.php? 公開資源%2F 日本語評価極性辞書

[10] 小林のぞみ，乾健太郎，松本裕治，立石健二，福島俊一，” 意見抽出のための評価表現の収集”，自然言語処理，Vol.12, No.3, pp.203-222, 2005.

[11] MeCab: Yet Another Part-of-Speech and Morphological Analyzer, http://mecab.googlecode.com/svn/trunk/mecab/doc/index.html

[12] はてなキーワード一覧ファイル - Hatena Developer Center, http://developer.hatena.ne.jp/ja/documents/keyword/misc/catalog

[13] Index of /jawiki/latest/, http://dumps.wikimedia.org/jawiki/latest/