2F3-1

商品カテゴリ情報に着目した教師データ収集による商品名抽出手法

Automatic product name recognition using product category information

渡辺 尚吾　乾 孝司　山本 幹雄
Shogo Watanabe　Takashi Inui　Mikio Yamamoto

筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻
Department of Computer Science, University of Tsukuba

Through some Web-based CMSs such as Blog and SNS, massive amount of subjective opinions have been generated day by day. Such information is beneficial for both product companies and users who are planning to purchase and use the products. In this paper, we proposed a method for extracting product names from freestyle un-structured text. The proposed method is based on the standard NE extraction method, but has two improvements. One is that in the proposed method, the category information of the target products is used for automatically collecting supervised labeled data. Another is that new chunking scheme is applied for enabling above-mentioned labeled data and realizing robust product name chunking.

1. はじめに

現在，Web 上には膨大な情報が蓄積され，すべての情報を人間の手で処理することは不可能である．情報はテキスト，画像など様々な形で存在するが，伝達に主に用いられるものはテキストである．そこで，高度な情報検索を行うために，テキストを情報抽出 (Information Extraction) などの技術により構造化することは重要な課題となっている．情報抽出の基礎技術に，人名，組織名，日付などの固有表現を抽出する固有表現抽出 (Named Entity Extraction) がある．固有表現を正しく抽出することが，情報抽出において重要であることが知られている．

また，テキストにおける書き手の主観的な意見を抽出する評判分析という技術がある．評判分析技術のひとつに，商品レビューなどのレビューテキストから商品についての情報を構造化する技術がある．しかし，それらの多くはレビューが商品と紐付けられているなど，対象とするレビュー文書がリッチな情報を持っていることを前提としてるものが多い．そういった技術は，blog 記事中などの，商品との対応付けがされていないレビューに対して用いることができない．しかし，blog はコンピュータに詳しくない一般的な主婦や学生による記事が多く，一般的な人々の率直な意見が多くテキスト情報として存在し，blog からの意見抽出は重要な課題である．blog 記事から商品についての意見を抽出するとき，その記事がどの商品のことについて意見を述べているかを判断する必要がある．そこで，本稿では blog 記事から商品名を抽出することに注目した．商品名は固有表現の一種なので，固有表現抽出の技術を使用することにより商品名抽出が可能であると考えられる．本稿で提案する商品名の抽出手法は，評判分析の基礎技術となることを目指している．

固有表現抽出は，機械学習による手法が一般的である．機械学習による手法では，教師データから固有表現の特徴を機械学習によって学習し，抽出器を作成する．しかし，学習のための教師データの作成は人手で行う必要があり，莫大なコストを要する．日本語の固有表現抽出では，IREX(Information Retrieval and Extraction Exercise) による固有表現抽出タスク [1] が行われており，人手によって作成された教師データが提供されているが，商品名は抽出対象として定義されていない．そのため，商品名抽出器学習のための教師データは存在せず，新しく作成する必要がある．しかし，人手で作る場合はあらゆる商品の知識を必要とし，コスト面において現実的ではない．そこで，本稿では商品のカテゴリ情報に注目し，blog から自動で教師データを収集する手法と，その教師データを用いて抽出器を学習し，商品名を抽出する手法を提案する．

連絡先: 渡辺尚吾，筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻，nabe@mibel.cs.tsukuba.ac.jp

表 1: 固有表現の種類 (IREX の定義)

固有表現の種類		例
固有名詞的表現		
ORGANIZATION	組織名	最高裁，筑波大
PERSON	人名	田中，クリントン
LOCATION	地名	太平洋，日本
ARTIFACT	固有物名	ノーベル賞，日米安保
時間表現		
DATE	日付表現	一昨日，1 月 1 日
TIME	時間表現	午後 5 時 25 分，正午
数値表現		
MONEY	金額表現	500 億円，$104,500
PERCENT	割合表現	20%，5 割

2. 日本語固有表現抽出

固有表現抽出は新聞記事などのテキストから，人名・組織名・日付表現などの固有表現部分を抜き出し，抜き出した部分がどの種類の固有表現であるかを分類するタスクである．固有表現抽出は，英語における MUC-7(Message Understanding Conference)[2] や日本語における IREX などで共通課題として扱われている．

IREX の日本語固有表現抽出タスク [1] では，固有表現は表 1 に示す 8 種であるとし，それぞれの固有表現は重ならないと定義している．日本語固有表現抽出タスクでは一般的に IREX による定義が用いられているが，定義が 8 種類と少なく，特定のものだけが細分化されているとの考えから，関根らは幅広い分野で応用することを前提とし，200 種類の階層的カテゴリを持つ「関根の拡張固有表現階層 [3]」を定義している．しかし，以前として一般的な消費者が必要とするような商品名の定義はされていないため，本稿ではいずれの定義も用いていない．

表 2: 商品カテゴリ情報

商品カテゴリ情報	商品名インスタンス
エアコン	白くまくん 霧ヶ峰
カップ麺	カップヌードル どん兵衛

商品名インスタンス	暑いので、**うるるとさらら**を買いました。
商品カテゴリ情報	暑いので、**エアコン**を買いました。

図 1: 商品名インスタンスと商品カテゴリ情報

3. 提案手法

3.1 商品カテゴリ情報と教師データ

テキストからの商品名抽出を考えるとき，商品名を固有表現の一種と捉えることができる．商品名抽出の場合も固有表現抽出と同様に，次々と新しい表現が生み出されるため辞書を用いる方法や，人手による抽出規則の作成は難しく，教師データから機械学習で抽出規則を学習する手法が有効であると考えられる．IREX の定義に基づいた固有表現抽出器を作成する場合などは教師データが提供されているが，従来の定義では商品名という固有表現クラスがないため，新たに商品名抽出のための教師データを作成する必要がある．本稿では商品カテゴリ情報に注目し，自動で教師データを収集する手法を提案する．**商品カテゴリ情報**とは,「エアコン」,「カップ麺」などの商品が属するカテゴリを表す情報である．表 2 に，商品カテゴリ情報とそれに属する商品名 (**商品名インスタンス**) の関係を示す．

Web 上のテキストでは,「シーフードヌードル」や「うるるとさらら」のような商品名インスタンスが記述されている文章と,「カップ麺」や「エアコン」のような商品カテゴリ情報による記述がなされている文章が共に存在する．また，その中には図 1 で示すような，商品名インスタンスと商品カテゴリ情報の前後の文脈が同じ，あるいは似ている文章が存在することが考えられる．機械学習で抽出規則を学習するとき，商品名インスタンスにタグ付けされたコーパスがあれば理想的であるが，何度も言うように作成には大きなコストを要する．そこで，商品名インスタンスのかわりに商品カテゴリ情報にタグ付けを行い，教師データとすることで，商品名抽出規則を前後の文脈から学習できると考えられる．本稿ではこの直感をもとに，教師データの自動収集方法と商品名抽出器の学習手法を提案する．

提案手法では，図 2 で示すように，固有表現タグのかわりに商品カテゴリ情報を用いることにより，教師データの自動収集，商品名抽出を行う． 商品カテゴリ C についての教師データの作成は，以下の手順で行う．

1. 抽出したい商品群の商品カテゴリ情報 C で Web を検索し，C に関するテキスト (blog 記事) を収集する．
2. コンテンツ抽出や文分割，不要な文の除去などを行う．
3. 文中に現れた商品カテゴリ情報にタグを自動で付与する．

提案手法における教師データ自動収集システムでは商品カテゴリ情報のみを入力とし，実際の商品名インスタンスについての知識は一切不要である．

3.2 商品名抽出

3.2.1 抽出の流れ

固有表現抽出と同じように，提案手法も文をトークンに分割し，各トークン毎に商品名らしさを学習する．

固有表現抽出
昨晩は <PERSON:田中 > さんと食事した。 <ORGANIZATION:共産党 > のマニフェスト。
商品名抽出
今日は < #カップ麺:シーフードヌードル > が美味しい。 電器屋で < #エアコン:うるるとさらら > を買ってきたよ。

図 2: 商品カテゴリ情報によるタグ付けの例

8 月になってようやく < #エアコン:エアコン > を買った。
この時期に < #エアコン:エアコン > の修理を電器屋に頼んだ。

図 3: タグの自動付与例

ある商品カテゴリについて，提案手法により作成した教師データは，商品名インスタンスではなく，すべてその商品カテゴリを表す同一の単語 (商品カテゴリ情報) にタグ付けされている．例えば，エアコンカテゴリについての教師データはすべて図 3 のように，エアコンという文字列に < #エアコン:エアコン > というタグが自動付与される．固有表現抽出の場合は固有表現そのものにタグ付けが行われているため，各トークンについて固有表現の種類と固有表現中の位置を組み合わせた固有表現タグで表現することができる．そして，固有表現タグについての分類器を学習することにより，各トークンの固有表現の種類の判定と，トークンのまとめあげという二つの処理を同時に行うことができた．ところが，提案手法による教師データの場合，トークンの商品カテゴリ情報は学習することはできるが，同時にトークンの商品名中の位置を学習することができない．そこで，各トークンにおける商品カテゴリの判断と，トークンのまとめ上げ (Chunking) を，同時ではなく逐次的に処理する．そのために，各商品カテゴリ情報について，商品名チャンクを構成する**商品名構成タグ**として，商品名開始タグと商品名終了タグの 2 種類のタグを定義し，それぞれについて分類器を作成する．そして，これらの 2 種類のタグからトークンのまとめあげ処理を行う．表 3 に商品名構成タグからまとめ上げ処理を行い，商品名チャンクを構成する様子を示す．

3.2.2 商品名構成タグ分類器

本稿では，トークンとして形態素解析によって得られた単語を用い，分類器には固有表現抽出において高い精度を示している SVM[4] を用いた．

商品名開始タグを推定するためにはトークンの前の文脈，商品名終了タグを推定するためにはトークンの後ろの文脈が重要であると考えられる．また，商品名抽出では，述語やトークンに係る単語が抽出の精度に大きく関係すると考えられる．例えば，図 1 の文章を考えたとき,「買う」という述語，エアコンに係る「暑い」という単語を手がかりとすれば，精度の向上を期待することが出来る．そこで，分類器に用いる素性として，次のものを用いた．

- 商品名開始タグ分類器
  - トークン単語の前 n 単語の表層文字列
  - トークン単語の前 n 単語の品詞
  - トークン単語が係っている動詞 v の原形
  - 動詞 v に係っている単語の原形
  - トークン単語に係っている単語の原形
- 商品名終了タグ分類器
  - トークン単語の後 n 単語の表層文字列
  - トークン単語の後 n 単語の品詞
  - トークン単語が係っている動詞 v の原形

表 3: 商品名構成タグからのまとめ上げ例

トークン	商品名構成タグ 開始タグ	終了タグ	商品名チャンク
暑い	0	0	
ので	0	0	
うるる	1	0	チャンク
と	0	0	
さらら	0	1	
を	0	0	
買い	0	0	

表 4: 商品名チャンクの構成例

開始	終了	c	開始	終了	c	開始	終了	c
0	0	-	0	0	-	1	0	-
1	0	c	0	0	-	1	0	c
0	0	c	1	1	c	0	0	c
0	1	c	0	0	-	0	1	c
0	0	-	0	0	-	0	1	-

- 動詞 v に係っている単語の原形
- トークン単語に係っている単語の原形

本稿で採用した教師データでは，商品名インスタンスではなく，商品カテゴリ情報にタグ付けされている．そのため，対象トークン自身の表層文字列や品詞情報は素性として用いていない．SVM の学習には，上に示した素性を 0/1 の 2 値ベクトルに変換したものを用いる．

3.2.3 商品名チャンクのまとめ上げ

次に，商品名開始タグ・終了タグ分類器によって得られた 2 列のタグ列から，トークンをまとめ上げ，商品名チャンクを構成する．理想的な商品名構成タグは，表 3 のように商品名の先頭単語に開始タグ，終了単語に終了タグがふられている場合である．また，一つの単語に開始タグと終了タグが両方ふられている場合は，その単語のみで商品名チャンクが構成されると考えられる．

理想的な商品名構成タグの場合，次のようなアルゴリズムで商品名チャンクを構成することができる．*tokens* をトークン列，l をトークン列の長さとする．

normal chunking algorithm

```
start ← nil
for i ← 1 to l do
  if tokens[i] の開始タグ = 1 then
    if tokens[i] の終了タグ = 1 then
      tokens[i] は 1 単語から構成される商品名チャンク
    else
      start ← i
    end if
  else if start ≠ nil and tokens[i] の終了タグ = 1 then
    tokens[start] から tokens[i] は商品名チャンク
    start ← nil
  end if
end for
```

このアルゴリズムによって構成される商品名チャンクの例を，表 4 に示す．c が連結する部分トークン列が商品名チャンクを示す．

開始タグと終了タグの分類器はそれぞれ独立しているため，それらが推定した商品名構成タグは常に理想的とは限らず，上記のアルゴリズムでは商品名が抽出できない可能性も考えられる．そこで，開始タグしか推定できなかった場合や，その逆の場合でも商品名チャンクを構成できる方法を，**part** chunking アルゴリズムとして次に示す．このアルゴリズムは，開始タグ，終了タグを区別せず，連続してどちらかのタグが現れている部分を商品名チャンクとして構成する．

表 5: part chunking による商品名チャンクの構成例

開始	終了	c	開始	終了	c	開始	終了	c
0	0	-	0	0	-	0	0	-
1	0	c	0	0	-	1	0	c
1	0	c	0	1	c	1	0	c
0	0	-	0	1	c	0	1	c
0	0	-	0	0	-	0	1	c

part chunking algorithm

```
start ← nil
end ← nil
for i ← 1 to l do
  if tokens[i] の開始タグ = 1 or tokens[i] の終了タグ =
  1 then
    if start = nil then
      start ← i
      end ← i
    else
      end ← i
    end if
  else if start ≠ nil then
    tokens[start] から tokens[end] は商品名チャンク
    start ← nil
    end ← nil
  end if
end for
```

このアルゴリズムによって構成される商品名チャンクの例を表 5 に示す．

4. 評価実験

4.1 実験データ及び実験設定

本稿では，商品カテゴリとしてエアコンを選び，実験を行った．

教師データの収集には google を利用し，61,260 記事，4,826,326 文の blog 記事を収集した．そのうち，教師データとして得られた文は 30,848 文で，平均 30.3 単語となった．

評価データは，表 6 に示す代表的なエアコンの商品名を含む記事を Yahoo!ブログ検索 API を利用して収集し，記事中の商品名インスタンスを人手で見つけ，タグ付けを行った．それぞれの商品名について，評価データの作成に用いた記事数と，最終的な評価データ中のインスタンス数を表 6 に示す．最終的な評価データは，商品名を含むものが 1,040 文，含まないものが 1,042 文，合計 2,082 文となった．

各モデルの精度の比較には固有表現抽出の分野で広く用いられている，適合率 (precision)，再現率 (recall)，適合率と再現率の調和平均である F 値を用いた．また，本稿では，システムが抽出した商品名チャンクの一部でも正解商品名チャンクと一致した場合，正解として評価した．

形態素解析器には MeCab[5, 6] を使用した．係受け解析には，構文解析器 CaboCha[7] を使用した．また，SVM の学習ツールに，TinySVM[8] を使用し，SVM のカーネル関数には固有表現抽出において最も精度が高いとされている 2 次の多項式カーネルを使用した．

4.2 実験結果

実験結果を表 7 に示す．n は素性として考慮する文脈単語数を示す．

本実験では，すべての n について normal chunking が最も

表 6: 評価データに使用した商品名

商品名	記事数	インスタンス数
白くまくん	1,337	218
キレイオン	336	42
異風人	49	0
うるるとさらら	1,458	293
nocria	390	51
霧ヶ峰	2,130	258
大清快	1,129	186
四季彩館	372	34
合計	7,201	1,082

表 7: 商品名抽出結果

n	chunking	適合率	再現率	F 値
2	normal	**40.39**	1.941	3.704
	part	8.297	3.512	4.935
3	normal	26.03	1.756	3.290
	part	8.514	4.344	5.753
4	normal	21.95	1.664	3.093
	part	10.24	5.823	**7.425**
5	normal	29.03	1.664	3.147
	part	9.176	4.529	6.064

良い適合率を示した．この結果より，理想的な商品名構成タグがふられた場合は商品名である可能性が高いことがわかる．再現率は part chunking で高い精度がみられた．part chunking で高い再現率が得られた理由として，開始タグと終了タグを区別しないため，どちらかのタグの推定が失敗していても商品名チャンクを構成できることが考えられる．また，$n = 4$，part chunking のときに最も高い F 値 7.425 が得られた．

4.3 負例削減の効果

再現率が低い原因のひとつに，正例に比べ負例が遙かに多いことが考えられる．自動生成した教師データをそのまま用いた場合，負例の数は正例の約 28 倍であった．そこで，学習時に負例を削減し，正例と負例のバランスを考慮し作成した抽出器で実験を行った結果を表 8 に示す．ただし，素性として用いる文脈単語数は，負例削減していないモデルで最も高い F 値を示した $n = 4$ とした．「名詞以外削除」は，対象トークンの品詞が名詞でない負例は学習しないとしたもので，「1:1」は，そこから負例をランダムに削り，正例・負例の比を 1:1 にしたモデルである．負例を削減した場合，再現率が大きく向上することがわかった．F 値は，正例:負例を 1:1 にして学習したモデルの normal chunking で 17.38 の精度を得ることができた．

4.4 既知の商品名による教師データを用いた抽出

提案手法は，ある商品カテゴリについて商品名の知識がまったくないことを前提として教師データの自動収集，抽出器の学習を行う手法である．しかし，ある商品カテゴリを見たときそのカテゴリの代表的な商品名を得られる可能性は十分考えられる．そこで，エアコンカテゴリについて，商品名「霧ヶ峰」が既知であるとして，「霧ヶ峰」に関する文 881 文で抽出器を学習して実験を行った．評価データは未知の商品名「nocria」に関する文で作成し，適合率 34.90，再現率 24.88，F 値 29.05 の精度で抽出できた．

5. まとめと今後の課題

本稿では，テキストからの商品名抽出において，教師データを自動収集し抽出器を学習する手法を提案した．提案手法では，商品カテゴリ情報を固有表現抽出における固有表現タグのように使用することにより，商品名の知識がなくても教師データを自動収集し，商品名抽出器を作成することができた．

表 8: 負例削減時の抽出結果

負例削減	chunking	適合率	再現率	F 値
名詞以外削除 (1:10)	normal	**21.69**	3.789	6.452
	part	9.505	8.688	9.078
1:1	normal	11.38	36.78	**17.38**
	part	9.269	**44.55**	15.35

提案手法では辞書を用いずに抽出器を作成し，精度が F 値 17.38 であった．固有表現抽出において，土田ら [9] は従来のような教師データに比べ，作成コストの低い小規模な固有表現辞書を作成し，辞書とタグ付けされていないコーパスから，教師データを自動で作成する手法を提案している．我々の提案手法とはデータが異なるため厳密な比較はできないが，土田ら手法では F 値 31.1 という精度を示している．商品名抽出の場合も，Web から商品名インスタンスの小規模な辞書を作成することは比較的容易だと考えられるので，今後は，辞書を併用した手法などを検討していきたい．また，宇佐美ら [10] は，あるドメインについて入手可能な語彙辞書を利用し，タグ付けされていないコーパスから教師データを作成し，そのドメインについての固有表現抽出器を学習する手法を提案している．今後，提案手法と宇佐美らの手法の差異について考察を進めたい．

参考文献

[1] IREX 実行委員会 (編)．1999．IREX ワークショップ予稿集．

[2] DARPA．1998．In *the Proceedings of the 7th Message Understanding Conference (MUC7)*.

[3] 関根聡．関根の拡張固有表現階層 -7.1.0-. http://sites.google.com/site/extendednamedentityhierarchy/.

[4] 山田寛康，工藤拓，松本裕治．2002．Support Vector Machine を用いた日本語固有表現抽出．情報処理学会論文誌，Vol. 43，No. 1，p. 44-53.

[5] Kudo，T.，Yamamoto，K. and Matsumoto，Y．2004. Applying Conditional Random Fields to Japanese Morphological Analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP-2004)*，pp．230-237.

[6] 工藤拓．2009．MeCab: Yet Another Part-of-Speech and Morphological Analyzer. http://mecab.sourceforge.net/.

[7] 工藤拓，松本裕治．2002．チャンキングの段階適用による係り受け解析．情報処理学会論文誌，Vol. 43，No. 6，pp．1834-1842.

[8] 工藤拓．2002．TinySVM: Support Vector Machines. http://chasen.org/taku/software/TinySVM/.

[9] 土田正明，水口弘紀，久寿居大，大和田勇人．2009．辞書とタグ無しコーパスを用いた固有表現抽出器の学習法．第 23 回人工知能学会全国大会，2009.

[10] 宇佐美佑，Han-Cheol Cho，岡崎直観，辻井潤一．2011．自動構築した大規模訓練データを用いた固有名抽出．自然言語処理学会第 17 回年次大会，pp．782-785.