カテゴリ情報を利用したblog記事からの商品名自動抽出

渡邊 尚吾　　乾 孝司　　山本 幹雄
筑波大学大学院システム情報工学研究科
nabe@mibel.cs.tsukuba.ac.jp, {inui, myama}@cs.tsukuba.ac.jp

1　はじめに

現在 Web 上では blog や SNS などのサービスの普及によって，様々な人々が多くの情報を発信するようになった．そのうちの有益な情報のひとつに，商品のレビューがある．レビューサイトを利用すると，レビューが商品ごとに分類・構造化されているため知りたい商品のレビューを容易に得ることができる．レビューサイト以外でも blog などにレビューは多く存在し，その中にはレビューサイトにない貴重な情報も含まれているが，blog 上のレビューは構造化されていないため，発見するためには検索サイトを経由し目視による選別が必要となる．そのため，blog などのレビューを自動で構造化し，レビューサイトと同様に扱えるようにすることが望まれている．

blog 上のレビューを自動で構造化するためには，blog 記事がレビューかどうかを判定し，レビューであれば対象としている商品を判断することが必要となる．レビューが対象としている商品を判断するとき，商品名が非常に有効な手がかりになると考えられる．このような背景のもと，我々は blog テキストから商品名を自動抽出する手法を検討した．商品名は一種の固有名詞であり，固有名詞などをテキストから抽出する固有表現抽出（NER: Named Entity Recognition）と同様の問題と考えることができる．NER には Support Vector Machine（SVM）など，機械学習による手法 [8] を用いることが一般的である．機械学習には主に IREX[6] の定義や関根の拡張固有表現階層 [5] に基づき整備された教師データが利用されるが，それらを商品名抽出に利用することはできないため，新たに商品名抽出の教師データを作成する必要がある．また，商品名は形態素解析で未知語となることが多いため解析誤りとなりやすく，商品名の境界と解析結果の単語境界が一致せず抽出が困難となる問題がある．

我々は，これらの問題に対して次のように対処した．通常であれば教師データの作成は人手によって行われるがその作業は非常に負荷がかかるものであるため，商品のカテゴリ情報を利用して疑似教師データを自動生成する．固有表現とは異なり，商品のカテゴリ名は商品名と同様の文脈で扱われることが多く[1)]，カテゴリ名にタグ付けされた教師データで抽出器を学習することで商品名抽出が可能であると考えられる．提案手法ではこうした考えのもと，カテゴリ情報による教師データの自動収集を行う．単語境界が一致しない問題については，単語分割に制約を導入することで抽出前に境界を修正する．

提案手法による抽出器の学習は，次のような流れとなる．

1. カテゴリ情報による疑似教師データの収集
2. 単語分割制約による単語分割を基本とした学習・評価事例の選択
3. SVM による抽出器の学習

2　疑似教師データの自動生成

カテゴリ情報とは，商品が属するカテゴリを示す情報である．本稿では商品を表す文字列を商品名，カテゴリ情報を表す文字列をカテゴリ名と呼ぶ．表 1 にそれぞれの関係を示す．商品名抽出はカテゴリごとに抽出することを目的とし，NER で “人名”，“地名” などを区別して抽出するように，商品名を “#スマートフォン”，“#ビール” のように区別して抽出する．

表 1: カテゴリ情報と商品名の関係

カテゴリ情報	カテゴリ名	商品名
#スマートフォン	スマートフォン スマホ	iPhone 4S Xperia acro HD
#ビール	ビール beer	ザ・プレミアム・モルツ 琥珀エビス

抽出器を学習するとき，人手で商品名に正解タグが付与された教師データを使用することが望ましいが，提案手法による自動収集ではカテゴリ名に自動でタグが付与された教師データ（**カテゴリ教師データ**と呼ぶ）となる．図 1 にそれぞれの教師データの例を示す．以下において，カテゴリ教師データが低コストで自動収集可能な理由，自動収集したカテゴリ教師データを用いて商品名抽出が可能である理由，および具体的な収集手法について述べる．

[1)] 例えば,「今日の夕飯はカップ麺」と「今日の夕飯はカップヌードル」.

理想（人手が必要）：通勤中には <#スマートフォン >iPhone 4S</#スマートフォン > でネットをしている。
提案（自動化可能）：通勤中には <#スマートフォン > スマートフォン </#スマートフォン > でネットをしている。

図 1: 理想教師データと提案教師データの例

まず，低コストで自動収集可能な理由について説明する．blog からカテゴリ教師データを自動収集するとき，カテゴリ名を含む文が多く blog に存在することが条件となるが，我々は日常で何か商品のことを話題とするとき，具体的な商品名のかわりにカテゴリ名を使用することが多い．そのことから blog 中にはカテゴリ名を含む文が多く存在していることがわかっている．また，カテゴリ教師データを自動収集する際に具体的な商品名の知識は不要であるため，理想的な教師データと比べ，低コストでかつ大量のカテゴリ教師データが収集可能となる．

仮に，IREX で定義された「人名」や「地名」などの固有表現抽出を考えた場合，これらのカテゴリ教師データは，「昨日は < 人名 > 人名 </人名 > さんと食事に行った」や「GW は < 地名 > 地名 </地名 > に遊びに行く」といったものとなる．商品カテゴリ名の場合と比べてこのような文が存在することは非常に希であるため，従来の固有表現抽出においてカテゴリ情報による教師データの収集は困難であると考えられる．つまり本手法は blog から商品名を抽出するという問題に特化した手法であると言える．

次に，カテゴリ教師データで商品名が抽出可能な理由について説明する．図 1 をみてわかるように，カテゴリ名を含む文は商品名を含む文と似た文脈をもって現れることが多い．そのため，カテゴリ教師データから文脈情報を学習することによって，文脈情報から商品名を抽出する抽出器が構築できると考えられる．一方で，理想的な教師データを使用した場合は，文脈情報に加えて具体的な商品名の文字列自体がもつ辞書的な情報を学習することができるが，カテゴリ教師データではカテゴリ名しか学習できない．そのため，提案手法ではカテゴリ名の文字列そのものは学習せず，文脈情報のみを学習して商品名抽出器を構築する．

以下に，本稿の実験で使用した具体的なカテゴリ教師データ自動収集の流れを示す．

1. Web からカテゴリ名を含む blog 記事を検索し，収集する．
2. 収集された各記事に対し，本文部分のみを抽出し，文分割を行う．
3. 上記の文から 5 単語未満，51 単語以上の文，動詞を含まない文を除外する．また，カテゴリ名を含まない文を除外する．
4. カテゴリ名にタグを付与する．

ここまでは，議論を単純化するため，商品名に関する知識が全くない状況を想定した．しかし，実際には既に知っている商品名があるという状況も考えられる．そこで，このような商品名を既知商品名と呼び，既知商品名を利用して教師データを自動収集することも考える．具体的な手続きは，カテゴリ情報を用いて教師データを自動収集するときと同様で，カテゴリ名のかわりに既知商品名を用いて教師データ（**既知教師データ**と呼ぶ）を自動収集する．既知教師データは，カテゴリ名とは異なり実際の商品名に正解タグが付与されるため，カテゴリ教師データより高品質な教師データとなることが予想される．

3 商品名抽出

3.1 抽出対象の制限

IREX 等の従来の固有表現クラスに対する NER では，固有表現を表す文字列そのものを学習できたため，どのトークン（形態素や文字）が固有表現のどの部分であるかという情報を学習できた．しかし，カテゴリ教師データでは商品名の文字列そのものは学習できず，固有表現と同様の手法により抽出を行うことは不可能である．そのため商品名についてトークンのまとめ上げを行うためには異なる手法が必要となる．また，従来のようなトークンを基本とした手法では，商品名の境界と形態素解析による単語分割の境界が異なる場合は抽出が不可能であり，商品名抽出の場合は形態素解析器にとって商品名が未知語となり解析誤りを導きやすく，特に問題となる．

本研究では形態素解析時の単語分割に制約を設け，商品名境界と単語分割境界が一致するように事前に調整する．具体的には，正規表現により商品名を 1 つの単語として制約した状態で形態素解析を行う．つまり，通常の形態素解析では「今日/は/iPhone/4/S/を/買っ/た」となっていた文を，単語分割制約により「今日/は/iPhone 4S/を/買っ/た」のように分割する．

単語分割制約により，トークンのまとめ上げを行う必要がなくなるため，すべてのトークンを分類対象とせず，単語分割制約により制約された単語（**制約単語**）のみについて商品名であるかどうかの分類を行う．つまり，あるカテゴリについての商品名抽出は制約単語についての 2 値分類問題に単純化される．また，学習時についても制約単語のみについて学習事例を生成し，タグ付けされた制約単語を正例，タグ付けされていない制約単語を負例とする．このように単語分割制約を導入することで，単語分割やまとめ上げの解析誤りの影響を極力排除できる．

3.2 素性抽出

提案手法で機械学習に用いる素性は，NER の先行研究でよく用いられている素性を基本としている．た

だし，商品名抽出において必要がないと判断したものについては省略している．例えば，NER における文字素性や文字種素性は固有表現を表す文字列そのものを表す素性として有効であったが，提案手法では商品名そのものが学習できないため用いていない．

提案手法では各制約単語自身について，係り先単語（係り先の文節の主辞），係り先単語の原形を，制約単語の前後について，表層単語，単語の原形，NE タグ，係り先単語，係り先単語の原形，表層文字を素性として用いる．NE タグはチャンクタグを取り除いた IREX の定義の固有表現種類を用いる．表層文字については，対象単語の前にあるか，後ろにあるか，ということのみを区別した．

3.3 Transductive SVM の利用

カテゴリ教師データはカテゴリ情報のみを用いて収集されるため，ノイズとなる事例が含まれている可能性がある．例えば,「人気の <#スマートフォン > スマートフォン </#スマートフォン > である Xperia が…」のような文では “スマートフォン” が正例となり，“Xperia” が負例となってしまう．しかし，正しく抽出器を学習するためには “Xperia” が正例となるべきであり，このような誤ったタグ付けを修正することが望まれる．

タグ付けの修正には Transductive SVM（TSVM）[2] を用いる．TSVM はバッチ評価が可能な際に，評価データの情報をあらかじめ学習に利用する SVM の変種である．この TSVM の学習機構を利用すると，ラベル付きデータとラベルなしデータの分布に基づき，ラベルなしデータのラベルを自動推定しながら抽出器を構築する半教師あり学習が可能となる．実際，教師データが多く確保できない状況において，多数のラベルなしデータを用意し，TSVM で学習することで分類精度が向上することが報告されている [7].

商品名抽出においては，既知教師データをラベル付き教師データ，カテゴリ教師データの正事例をラベルなしデータとして扱うことにより，カテゴリ教師データ中で間違ってタグが付与されているものについて，学習アルゴリズム内で自動的にラベル修正が実現される．カテゴリ教師データ中の正事例のみを扱う理由は，負例は既知教師データ中にも十分存在するためである．また，TSVM を使用する際はラベルなしデータの正負の分布をパラメータとして与える必要があるが，今回の場合のラベルなしデータは元々正例であると仮定できるため，実験では P(正例の割合) $= 0.995$ で固定した．

4 評価実験

4.1 実験データ及び実験設定

スマートフォンカテゴリについて評価実験を行った．データの作成には Web から収集した blog 記事約 2,300 万件を使用し，51,834 文のカテゴリ教師データ，2,506 文の既知教師データを得た．既知教師データの作成にはテストデータが含む商品名から blog 検索数が多い 10 商品名を既知商品名として扱った．テストデータは同じ blog 記事から人手によって作成した 1,639 文を用いた．

単語分割制約には，以下の規則に則るような正規表現を用いた．

- 制約単語は 1 つ以上の要素語で構成される．
- 要素語間には空白が含まれていても良い．
- 要素語は次のいずれかで構成される．
  - カタカナ要素語（KW）：カタカナと長音記号（ー）で構成される要素語．ただし，長音記号は先頭になれない．例,「アローズ」「アイフォン」.
  - アルファベット要素語 1（AW1）：アルファベットと数字とハイフンで構成される要素語．ただし，ハイフンは先頭と末尾になれない．また，数字は先頭になれない．例,「Xperia」「SO-01B」.
  - アルファベット要素語 2（AW2）：アルファベットと数字とハイフンとピリオドで構成される要素語．ただし，ハイフンとピリオドは先頭と末尾になれない．例,「4S」「10.1」.
- 制約単語構成規則
  - KW は AW1，AW2 の後に現れない．
  - AW2 は先頭になれない．
  - つまり，制約単語は以下の要素語の組み合わせとなる．

    規則 1 /{KW}+({AW1}|{AW2})*/

    規則 2 /{AW1}{AW2}*/

形態素解析には MeCab[3]，構文解析には CaboCha[4] を用いた．SVM，TSVM の実装は SVM^{Light}[1] を用いた．また，SVM では RBF（Radial Basis Function）カーネルを使用し，各種パラメータはカテゴリ教師データと既知教師データについてグリッドサーチによって最適なものを求めた．

4.2 評価方法

評価には 2 種の指標を用いた．1 つ目には，NER で広く用いられている適合率と再現率の調和平均である F 値を用いた．これは単純な抽出数（token 数）を基にしているので，token 評価と呼ぶことにする．商品名抽出の応用としてレビューサイトの構築を考えたとき，より多くの種類の商品名を抽出できることが望ましい．そこで，抽出した token 数ではなく種類数 (type 数）を基とした評価として，list 評価を行う．list 評価は次のようにして評価を行う．

1. 分類対象となった単語 type すべてについて，正例と分類された数（商品名であるとされた数，*pos*）

と負例と分類された数（商品名でないとされた数，neg）をカウントする．

2. $pos > neg$ である単語 type は商品名であると判断して，商品名リストを作成する．
3. テストデータから作成した正解リストと比較し，リスト適合率，リスト再現率，リスト F 値を計算する．

4.3 実験結果

提案手法との比較には，ベースラインとして辞書マッチによる手法を用いた．辞書マッチ法は，商品名辞書に含まれる商品名の文字列がテストデータ中の文字列とマッチするとき，その文字列を商品名として抽出する．商品名辞書には，価格.com[2)]の製品データベースから抽出した商品名 150 個を利用した．

提案手法では，カテゴリ教師データと既知教師データでそれぞれ学習を行ったモデル，それらを単純にあわせた教師データで学習を行ったモデル（カテゴリ+既知），TSVM によって学習したモデル（transductive）を構築し，比較を行った．

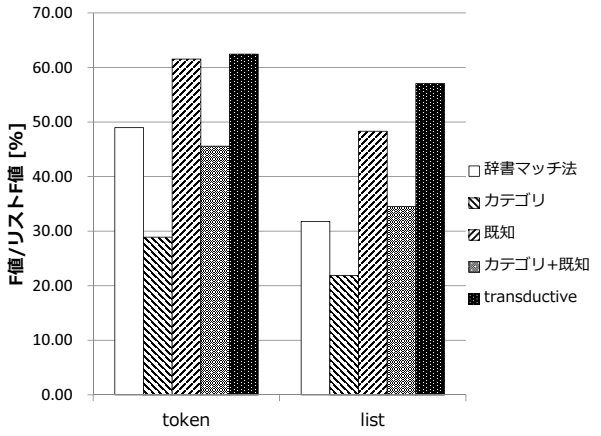

図 2: 実験結果

実験結果を図 2 に示す．token 評価，list 評価ともに transductive モデルが最も良い精度となった．特に，list 評価では大きく差が出ている．既知教師データはデータ量が少ないため，適合率（正確性）は高いが再現率（網羅性）が低いという特徴があった．transductive モデルでは再現率が低いという問題を，データ量の多いカテゴリ教師データを上手く利用し，補うことにより精度が向上している．カテゴリ教師データ単体ではノイズが多く，精度がベースラインより低い結果となった．また，2 つの教師データを単純にあわせたカテゴリ+既知モデルはノイズの影響を大きく受ける形となり，既知モデルより精度が低かった．

5 おわりに

我々は商品名抽出においてカテゴリ情報を用いて疑似教師データを自動生成する手法を提案し，実験で疑似教師データが商品名抽出に有効であることを示した．提案手法では，商品名とカテゴリ名が同様の文脈で現れるという商品名抽出における特徴を利用し，低コストな疑似教師データを自動生成可能とした．また，それを利用した商品名抽出器の構築手法についても述べた．さらに，2 種類の教師データから半教師あり学習を行うことでより高性能な抽出器を構築する手法についても説明し，実験では半教師あり学習が有効であることを確認した．

今後は，多様なカテゴリでの実験やカテゴリの粒度の調査が必要であると考えている．

謝辞

本研究で利用した製品データベースは，株式会社カカクコム様より提供して頂きました．ここに深謝の意を表します．この製品データベースは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による「大規模情報コンテンツ時代の高度 ICT 専門職業人育成事業」の支援により提供されたものです．

参考文献

[1] T. Joachims. 2002. Learning to Classify Text Using Support Vector Machines. Dissertation, Kluwer.

[2] T. Joachims. 1999. Transductive Inference for Text Classification using Support Vector Machines. In *Proceedings of the 16th International Conference on Machine Learning (ICML-99)*, pp. 200–209.

[3] T. Kudo, K. Yamamoto and Y. Matsumoto. 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.

[4] T. Kudo and Y. Matsumoto. 2003. Fast Methods for Kernel-based Text Analysis. In *Proceedings of the 41st Annual Meeting of The Association for Computational Linguistics*, pp. 24–31.

[5] S. Sekine and C. Nobata. 2004. Definition, dictionaries and tagger for Extended Named Entity Hierarchy. In *Proceedings of 4th International Conference on Language Resources on Language Resources and Evaluation (LREC2004)*, pp. 1977–1980.

[6] S. Sekine and H. Isahara. 2000. IREX: IR and IE Evaluation project in Japanese. In *Proceedings of 2nd International Conference on Language Resources on Language Resources and Evaluation (LREC2000)*.

[7] 嶋田 和孝, 林 晃司, 遠藤 勉. 2005. SVM および Transductive SVM を用いた製品スペック情報の抽出. 自然言語処理, Vol. 12, No. 3, pp. 43–66.

[8] 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, pp. 44–53.

[2)] http://kakaku.com. 株式会社カカクコム