評判情報分析のための製品属性異表記辞書の自動構築

北陸先端科学技術大学院大学
情報科学研究科

LIU CHAOYU

2016 年 3 月

修 士 論 文

評判情報分析のための製品属性異表記辞書の自動構築

指導教員　白井清昭

審査委員主査　白井清昭
審査委員　東条敏
審査委員　飯田弘之

北陸先端科学技術大学院大学
情報科学研究科

1410047 LIU CHAOYU

提出年月: 2016 年 2 月

概 要

日本語では，送り仮名の違い，長音の有無，字種の違い，あるいは全く異なる表現など，同じ実体が異なる文字列で表現されることがよくある．このような表現を「異表記」または「表記ゆれ」と呼ぶ．本論文では，製品属性を対象とした評判情報分析のための基礎的な知識として，製品属性の異表記辞書を自動的に構築する手法について述べる．提案手法は大きく 2 つの処理に分けられる．

1 つ目の処理では，製品のウェブページなどに掲載されている仕様表から製品属性を抽出する．まず,「価格.com」というウェブサイトと製品メーカーのウェブサイトから製品の仕様を記載している表を取得し，さらにその仕様表から属性と属性値の組を抽出する．価格.com では仕様表のフォーマットが一意に決まっているため，仕様表の抽出も属性・属性値の組の抽出も容易である．一方，メーカーのウェブページについては，仕様表ならびに属性・属性値の組を抽出するためのルールを作成する．次に，同じ実体を表わす異表記の属性をひとつにまとめるために，凝集型クラスタリングアルゴリズムによって，属性・属性値の組のクラスタリングを行う．クラスタリング後，属性値を取り除いて，初期の製品属性異表記辞書を得る．

2 つ目の処理では，製品に対するレビュー文から異表記の製品属性を獲得する．まず，製品のレビュー文を「価格.com」から収集する．次に，パターンのテンプレートを用意し，それをレビュー文の集合に適用することで，製品属性抽出パターンの候補を得る．パターンは，(抽出するべき) 属性，評価語，およびその間に出現する単語の列として表現される．次に，パターンの候補に対してスコアを計算する．パターンを適用したときに初期の製品属性異表記辞書に含まれる属性をより多く抽出するもの対してより高いスコアを与える．パターンにマッチする文の数が 3 以上で，かつスコアが 0.5 以上の候補を最終的なパターンとして獲得する．そして，得られたパターンによるパターンマッチによってレビュー文から製品属性を得る．

最後に，仕様表から獲得した製品属性とレビュー文から獲得した製品属性を統合し，最終的な製品属性異表記辞書を得る．パターンを初期の製品属性異表記辞書のいずれかの属性と関連づけ，そのパターンによって獲得された属性は，パターンと関連づけられた属性の異表記として製品属性異表記辞書に追加する．

提案手法の評価実験では，まず仕様表からの属性抽出手法を評価した．仕様表抽出の精度と再現率は 0.89 と 0.82 であった．仕様表からの属性の抽出精度は 0.90 であった．属性・属性値の組のクラスタリングの Purity は 0.829 であった．これらの結果から，提案手法の有効性が示された．一方，レビュー文からの属性抽出手法を評価したところ，パターンによって獲得された属性の正解率は，評価対象とした製品カテゴリによって異なるが，0.07 から 0.34 となった．属性抽出の正解率は低く，抽出パターンの精緻化など，改善の余地が多く残されていることがわかった．

目 次

第1章　序論

1.1　研究の背景

日本語では，送り仮名の違い，長音の有無，字種の違い，あるいは全く異なる表現など，同じ実体が異なる文字列で表現されることがよくある．このような表現を「異表記」または「表記ゆれ」と呼ぶ．異表記の処理は自然言語処理における重要な課題のひとつである．

一方，近年ではオンラインショッピングがよく利用されており，ユーザが製品を使用した感想や評価を投稿できるウェブサイトも数多く存在する．このようなユーザによる製品レビューを対象とした評判情報分析は，メーカーと消費者の双方にとって有用である．メーカーの立場から見れば，評判情報を分析することで，自社の製品の設計，改良，販売戦略の立案などに役立てることができる．消費者の立場から見れば，製品を買う前に，他のユーザによるレビューや苦情をあらかじめ調べることで，自分のニーズに合った製品を購入することができる．ただし，このような評判情報が大量に存在する場合，その全てを閲覧するのは困難であるので，評判情報を分析した上でわかりやすくメーカーもしくはユーザに提示する技術が求められる．

製品を対象とした評判情報分析においては，製品そのものに対するユーザの評価や意見だけでなく，製品の属性に対する評価の分析が求められることが多い．例えば，パソコンの場合，価格，サイズ，メモリ，ディスク容量，拡張性，キーボードなどが製品の属性に該当し，「価格は安いがメモリが少ない」「サイズは手頃だがキーボードは打ちにくい」といったように製品の属性に対する意見を調べたいといった要望をメーカーやユーザーは持っていると考えられる．

製品属性を対象とし評判情報分析でしばしば問題となるのは異表記である．例えば，「価格」は「値段」「売り値」「購入金額」「コスト」といった様々な表現で表わされる．ある製品の価格に関する評判を網羅的に収集するためには，これらの表現が全て「価格」の異表記であり，同じ製品属性を指すものであることを認識する必要がある．

1.2　研究の目的

本論文では，製品属性を対象とした評判情報分析のための基礎的な知識として，製品属性の異表記辞書を自動的に構築する手法について述べる．ここでの製品属性の異表記辞書とは，「パソコン」「テレビ」「冷蔵庫」といった製品のカテゴリ毎に，製品の属性を異表記も含めて網羅的に収集した辞書と定義する．本研究で想定している製品属性異表記辞書

の例を図 (1.1) に示す．この辞書はパソコンの製品属性を記載している．それぞれの行が同一の属性を表わす異表記の単語に対応する．例えば,「値段」「価格」「コスト」の 3 つは互いに異表記の属性であり,「メモリー」「メモリ」「メモリ容量」もまた異表記の属性である．

[パソコン]
値段　価格　コスト　...
メモリー　メモリ　メモリ容量　...

図 1.1: 製品属性異表記辞書の例

製品属性異表記辞書の使用例を紹介する．今，パソコン A についてのユーザーのレビュー文として以下の 3 つが得られたとする．

1. アマゾンではパソコン A の価格が大変安くなっているので，お買い得だ．

2. パソコン A はスペックの割には値段が安いと思う．

3. パソコン A はシリーズの中でもコストを抑えたモデルといえる．

これら 3 つの文は全て「パソコン A」が安いという評判を示唆しているが，価格という属性が異なる表現で表わされているため，ナイーブな手法ではこれら 3 つの文が全て価格に関する意見であることを認識できない．一方，図 1.1 のような辞書があれば,「価格」「値段」「コスト」は全て同じ製品属性を指し，パソコン A の値段が安いという評判が多いことを知ることができる．

上記で述べた製品属性異表記辞書を構築するために，本研究では 2 つの知識源を用いる．ひとつはウェブ上に存在する製品ページの仕様表である．メーカーの公式サイトには製品の仕様が表でまとめられていることがあり，その仕様表から属性を表わす表現を収集できる．もうひとつはユーザによるレビューテキストである．ユーザーレビューでは製品の属性がしばしば言及されるため，製品属性を表わす語や句を収集することができる．2 つの知識源から製品属性を表わす表現を網羅的に収集し，また同一の実体を表わす表現をまとめることで，製品カテゴリに特化した異表記辞書を自動構築する．

1.3　本論文の構成

本論文は以下の章から構成される．2 章では, 異表記の語の自動獲得と製品属性の異表記に関する研究を紹介し，これら先行研究と本研究の違いについて述べる．3 章では，提案する製品属性の異表記辞書の自動構築手法について詳しく述べる．4 章では，提案手法で構築した異表記辞書に対する評価実験について述べ，その結果を考察する．最後に 5 章では，本研究のまとめと今後の課題について述べる．

第2章　関連研究

本章では関連研究について述べる．まず，2.1 節では，異表記の語をコーパスから自動獲得する先行研究を紹介する．次に，2.2 節では，製品属性の抽出ならびにその異表記の取り扱いに関する研究について述べる．最後に，2.3 節では，関連研究と比べたときの本研究の特色について論じる．

2.1　異表記の語の自動獲得に関する研究

異表記の語をコーパスから自動獲得する研究として，カタカナで表記された外来語や翻字の異表記を検出する手法がいくつか提案されている．まず Masuyama らの研究 [4] と Ohtake らの研究 [6] について述べる．これらの研究は，異表記の語の検出に 2 つの尺度が組み合わせて用いられる点が共通している．ひとつは 2 つの単語間の編集距離，もう一つは大規模コーパスにおける単語の周辺文脈の類似度である．

Masuyama らの手法 [4] では，まずは巨大なコーパスからカタカナ文字ならびにカタカナ語でよく出現する特徴的な記号 (「・」「ー」「—」) を含んでいる単語を候補として抽出する. 例えば，以下の例文 1 から「ルートウィヒ・エアハルト」，例文 2 から「ソ」「ルードウィッヒ・エアハルト」「ドイツ」を抽出する.

1. “ 奇跡の経済復興の父 ”といわれる故ルートウィヒ・エアハルト氏.
2. もしソ連や東欧諸国が統制志向を捨て，一九四八年に西欧のルードウィッヒ・エアハルトがとったような経済の自由化へと突き進めば，西ドイツのような奇跡の復興を遂げるかもしれない.

そして，抽出された候補単語のうち，カタカナ表記が似ている 2 つの単語の組を見つける. 言い換えれば,「文字列のペナルティ」の小さい単語の組を見つける．文字列のペナルティは，2 つの文字列に対して与えられる非類似度で，編集距離と同じように，一方の文字列を変形させてもう一方の文字列を得るまでに要する処理の数や種類によって定義される. 具体的には，文字列のペナルティは，編集距離で用いられる 3 つの文字列の変換操作 (挿入，削除，入れ替え) と，発音の類似度を考慮した著者らの独自のルールに基づいて計算される. 文字列のペナルティが閾値 4 以下の単語の組が次のステップにおける異表記の語の候補となる．最後に，異表記の候補となる 2 つの単語の文脈類似度を計算し，それが十分に大きいものを最終的な異表記の語として獲得する. 2 つの語が出現する文書を形態素

解析によって単語に分割し，ストップワードを除去した上で，残された主に自立語を素性とする単語ベクトルを作成し，そのベクトルのコサイン類似度を文脈類似度と定義する．文脈類似度の閾値を 0.05 と設定し，これを越える語の組を異表記とみなす．

Ohtake らは日本語の翻字 (外来語を発音によってカタカナで表記した語) の異表記を検出する手法を提案している [6]．彼らの研究も，Masuyama らと同様に，異表記を検出するための 2 つの基準を用いている．1 つは文字列の類似度であり，もう 1 つは対象となるカタカナ列を含む文の文脈類似度である．まず，与えられたコーパスに対して文節の係り受け解析を行い，カタカナ語ならびにその文脈ベクトルを抽出する．文脈ベクトルは，名詞，カタカナ語が係る動詞，助詞と動詞の組 [1] を素性とする．得られたカタカナ語の集合から，対象単語を 1 つ選択し，その対象単語に対して，少なくとも 1 つの文字を共有する別のカタカナ語を異表記候補単語とする．次に，対象単語と異表記候補単語の組に対して 3 種類のスコアを計算する．1 つ目は文字列間の編集距離，2 つ目はそれぞれの単語をローマ字表記に直した後で計算された編集距離，3 つ目は文脈ベクトルのコサイン類似度によって計算される文脈類似度である．最後に，これらの 3 つのスコアから対象単語と異表記候補単語が真に異表記であるかを判定する決定リストを人手で作成し，その決定リストを用いて異表記と判定された語の組を獲得する．

大前と黄瀬は，ウェブ上の表を解析し，属性と属性値を抽出する手法を提案している [5]．さらに，同じ実体を表わす属性を同定し，1 つのグループにまとめることを試みている．属性の類似度は，それに対応する属性値の類似性によって計算される．まず，与えられた属性に対し，属性値を素性，その属性値の出現頻度を重みとするベクトル (属性ベクトル) を作る．属性ベクトル A_i は式 (2.1) のように表わされる．ここで，i_j は属性値 i の出現頻度，n は属性値の数を表わす．

$$A_i = \{\, i_1,\ i_2, \cdots\ i_n\} \tag{2.1}$$

例えば，図 2.1 に示すように, 属性「開催都市」の属性値として「ワシントン」「大阪」「大阪」「ロンドン」「北京」があれば,「大阪」の重みは 2，それ以外の都市の重みは 1 となる．属性間の類似度は，属性ベクトルの Jaccard 係数とする．その定義を式 (2.2) に示す．

$$\alpha\ (A_x, A_y) = \frac{\sum_{i=1}^{n} x_i \cdot y_i}{\sum_{i=1}^{n} x_i^2 + \sum_{i=1}^{n} y_i^2 + \sum_{i=1}^{n} x_i \cdot y_i} \tag{2.2}$$

x_i，y_i は属性ベクトル A_x，A_y における i 番目の属性値の重みを表わす．最後に，類似度がある閾値以上の属性は同じ実体を表わす異表記の語とみなして統合する．

[1] 正確には，カタカナ語の直後に出現する助詞とカタカナ語が係る動詞の組．

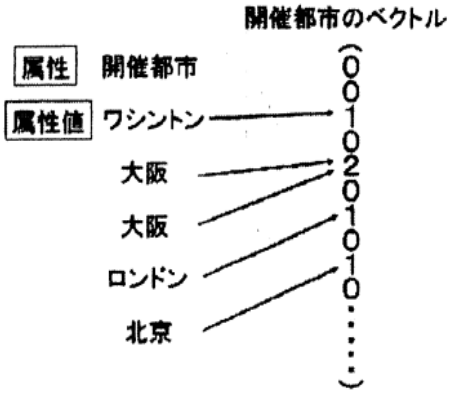

図 2.1: 属性ベクトルの例 (文献 [5] の図 4)

2.2 製品属性の異表記に関する研究

Shinzato と Sekine は，オンラインショッピングサイトにおける商品説明文から製品の属性・属性値を自動抽出する手法を提案している [7, 8].

まず，商品ページにおける表および箇条書きから属性と属性値の組を抽出する. ウェブページの表におけるヘッダ (th タグでマークアップされた語) を属性として抽出する．この際,「保存方法」「その他」「商品説明」「広告文責」「特徴」「仕様」などのストップワードをあらかじめ用意し，これらは属性として抽出しない. 次に，属性に対する属性値を著者らが作成したパターンを用いて抽出する. そのパターンを図 2.2 に示す．同図において，[ATTR] は属性，[ANY] は任意の文字列，[P] は prefix または開き括弧，[S] は suffix または閉じ括弧を表わす.

- ⋆ P1: <T(H|D)>[ATTR]</T(H|D)><TD>[ANY]</TD>
- ⋆ P2: [P][ATTR][S][ANY][P]
- ⋆ P3: [P][ATTR][ANY][P]
- ⋆ P4: [ATTR][S][ANY][ATTR][S]

図 2.2: 文献 [7] で使われた属性値を抽出するためのパターン

次に，商品説明文に対して属性と属性値を自動的にタグ付けし，属性・属性値抽出モデルを Conditional Random Field(CRF) で機械学習する．学習に用いた素性を図 2.3 に示す．最終的に，得られた CRF のモデルを用いて商品説明文から属性と属性値を抽出している.

基本素性: 単語の表層形，基本形，品詞，前 2 文字 (e.g., **シャ**トー)，後 2 文字 (e.g., シャ**トー**)，単語内の文字の種類 (平仮名，カタカナ，漢字，アルファベット，数字，その他).
文脈素性: 前後 3 単語の基本素性

図 2.3: 文献 [7] で使われた学習素性

Shinzato と Sekine は，表や箇条書きから抽出した属性の集合に対し，同じ属性値を持つ属性は異表記であるという考え方に基づいて同一の実体を表わす製品属性をまとめることを試みている．オンラインショッピングサイトにおいて，ある属性が出現するショッピングサイトの数を店舗頻度と定義し，店舗頻度が N を超える属性のうち，(1)2 つの属性が同じ属性値を持ち，(2)2 つの属性が同一の構造化データ (表または箇条書き) に含まていない，という条件を満たす属性の組は，同じ属性の異表記の語であるとみなす．閾値 N は式 (2.3) で定義している．

$$\mathrm{N} = max(2, M_s/100) \tag{2.3}$$

M_s は対象カテゴリにおいて構造化データを提供している店舗数を表わす．

2.3 テキストからの製品属性の抽出に関する研究

駒田と山名は，Twitter に投稿されたレビュー文において評価語の周辺に出現する語を製品の属性として抽出する手法を提案している [3]．彼らの研究は，まず，商品カテゴリに依らずに用いられる一般的な評価語の集合を初期の評価語辞書とする．次に，対象語と評価語を含むツイートを「商品評価ツイート」として抽出する．そして，URL，リプライ，リツイート，ハッシュタグなど，Twitter に特徴的なノイズを除去する．この前処理の後，CaboCha を用いて文節の係り受け解析を行う．次のステップでは，係り受け解析の結果を基に製品属性の候補を抽出する．まず，評価語辞書に含まれる評価語と係り受け関係になっている文節を抽出し，それらの文節の形態素解析を行う．形態素解析結果，品詞が「名詞-一般」「名詞-固有名詞」「名詞-サ変接続」「未知語」となる単語を属性の候補とする．ただし，図 2.4 に示すパターンに該当する単語列は一つの単語として扱う．さらに，図 2.5 に示すパターンにマッチするときは，文節の係り受け解析の結果に依らずに，マッチする単語列全てを属性の候補として抽出する．

つぎに，以上の処理により得られた属性の候補と対象製品の関連度を，属性の候補の出現頻度，評価語と属性の候補の共起頻度を基に算出する．関連度がある閾値以上の属性の候補を属性辞書に加える．その後，属性を抽出する手続きとほぼ同じ手続きにより，商品評価ツイートから評価語を抽出し，評価語辞書に加える．以上の手続きを繰り返し，属性辞書ならびに評価語辞書を漸進的に拡張する．

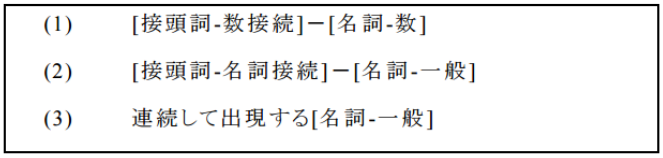

図 2.4: 一つの属性候補として抽出する品詞のパターン (文献 [3])

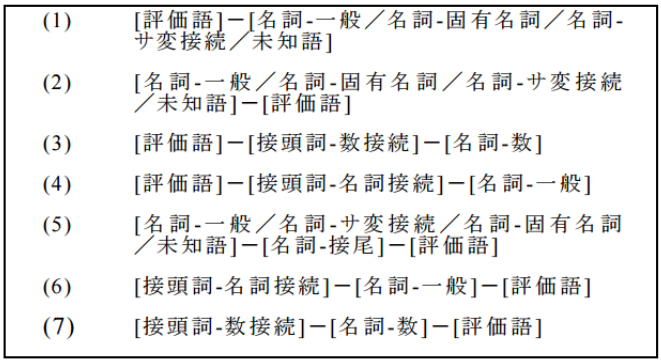

図 2.5: 属性候補の抽出パターン (文献 [3])

2.4 本研究の特色

本研究は，製品の仕様表とテキストの両方から異表記の製品属性を抽出する点で Shinzato と Sekine の研究と類似している．ただし，本研究は以下のような特色を持つ．

- Shinzato と Sekine の手法では，表から属性や属性値を抽出する際に，単一のオンラインショッピングサイトのみを処理の対象にしている．これに対し，本研究では複数のメーカーのウェブページの製品仕様表から属性の抽出を試みる．様々なメーカーのウェブページを抽出対象とした方がより多くの製品属性を獲得することが期待できる．

- 先行研究では商品説明文を属性抽出の対象テキストとしているが，本研究ではレビュー文を用いる．本研究では，自動構築した製品属性異表記辞書を評判情報分析に応用することを想定している．ユーザが製品を評価する文によく出現する製品属性を獲得するためには，商品説明文よりもレビュー文の方が適している．

- 先行研究では，異表記の製品属性を獲得する処理としては，表や箇条書きから抽出された属性値の類似性を基にした比較的単純な手法を採用している．本研究では，属性値の類似性に基づく手法に加え，レビュー文から製品属性の異表記の語を検出するパターンを自動獲得し，またそのパターンを用いて製品属性の異表記の語を獲

得する．すなわち，テキストから単に製品属性を抽出するのではなく，ある製品属性と同じ実体を表わす異表記の語に対象を絞って獲得する点に特徴がある．

第3章　提案手法

3.1　概要

ここでは,「パソコン」「冷蔵庫」のような製品カテゴリが入力として与えられたとき,そのカテゴリの製品の評価によく使われる属性を収録した製品属性異表記辞書 D を構築することを目的とする. 本研究では D を以下のように定式化する.

$$D = \{\cdots, A_i, \cdots\} \tag{3.1}$$
$$\text{但し}, A_i = \{\cdots, a_{ij}, \cdots\}$$

a_{ij} は製品の属性を表わす単語 (もしくは複合語) であり，集合 A_i は同じ実体を表わす異表記の属性の集合である．A_i を集めたものを D とする．

提案手法の概要を図 3.1 に示す．提案手法は大きく 2 つに分けられる．ひとつは，製品のウェブページなどから製品の仕様表を抽出し，さらに仕様表から製品属性を抽出して，初期の製品属性異表記辞書 (D_t と記す) を構築する処理である．この詳細は 3.2 節で説明する．もう一つは，製品に対するレビュー文を知識源とし，製品属性を抽出するパターンを獲得し，そのパターンを用いたマッチングによって異表記の製品属性を獲得する処理である．この詳細は 3.3 節で述べる．最後に，仕様表から獲得した製品属性とレビュー文から獲得した製品属性を統合し，最終的な製品属性異表記辞書 D を得る．この処理の詳細については 3.4 節で述べる．

3.2　仕様表からの異表記の製品属性の抽出

3.2.1　属性と属性値の組の抽出

ここでは製品の仕様表から製品属性を抽出する手法について述べる．本研究では，製品属性を獲得するための重要な情報源として価格.com(http://kakaku.com/) を用いる．価格.com は，複数のオンラインショップでの製品の販売価格を比較するウェブサイトである．価格の情報の他に，製品の仕様表や製品に対するユーザレビューなどの情報も掲載されている．

仕様表から製品属性を抽出する処理の流れを以下に示す.

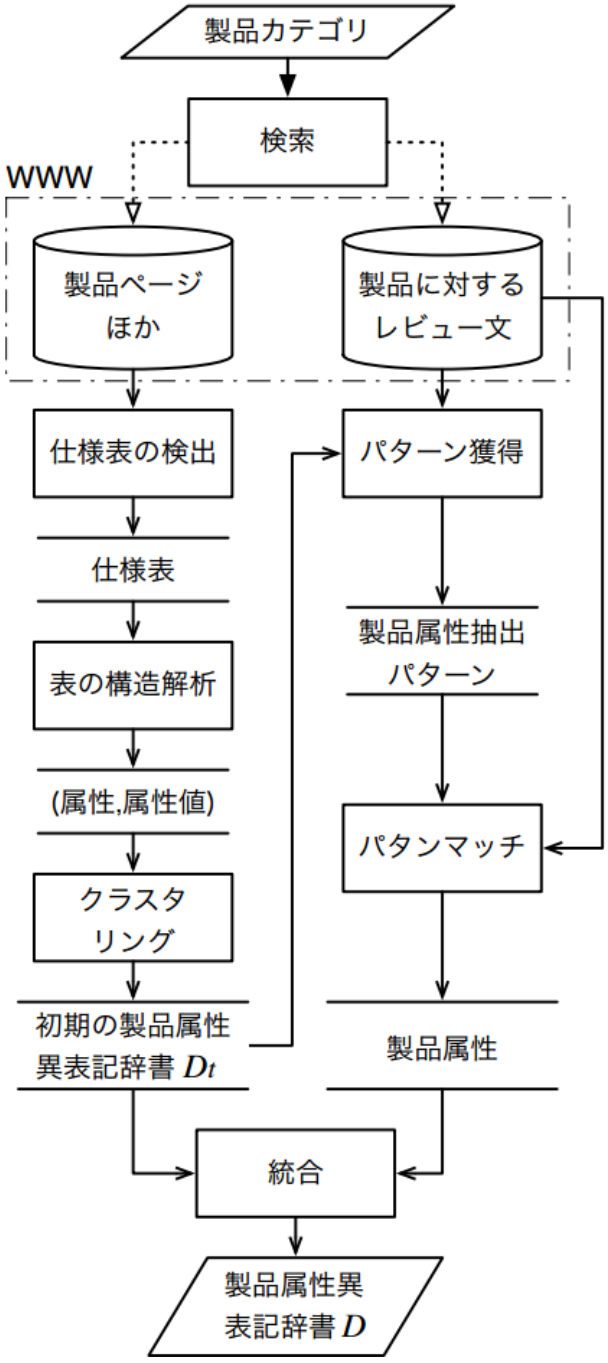
製品カテゴリ
検索
WWW
製品ページ
ほか
製品に対する
レビュー文
仕様表の検出
パターン獲得
仕様表
製品属性抽出
パターン
表の構造解析
(属性,属性値)
パタンマッチ
クラスタ
リング
初期の製品属性
異表記辞書 D_t
製品属性
統合
製品属性異
表記辞書 D

図 3.1: 提案手法の概要

1. 価格.com のサイトから，入力として与えられた製品カテゴリに該当する製品のページを取得する．
2. 価格.com の製品ページから仕様表を取得する．
3. 価格.com の製品ページの中から，その製品のメーカーのウェブサイトへの URL を検出する．
4. メーカーのウェブページ内に存在する table タグでマークアップされた表のうち，後述する条件を満たすものを仕様表として取得する．
5. ステップ 2. と 4. で取得した表から属性と属性値の組を取得する．

ステップ 1. では，価格.com において，与えられた製品カテゴリに該当する製品のページの HTML ソースファイルをダウンロードする．ダウンロードにはプログラミング言語 Python のライブラリである BeautifulSoup を用いた．後続のステップ 3 では，価格.com の製品ページからその製品のメーカーのページへのリンクを辿る．価格.com の製品ページには，メーカーへのリンクが存在するものとしないものがあるが，ここではリンクが存在する製品ページのみを取得する．また，取得する製品ページはメーカーの製品ブランドにつき 1 つとする．メーカーのウェブサイトに掲載されている仕様表のフォーマットはメーカーによって様々であるが，同じメーカーの同じブランドの製品であれば，仕様表のフォーマットや仕様表に掲載されている製品属性の集合は一意に定まると考えられる．ここでの目標は仕様表から製品属性を抽出することなので，仕様表はメーカーのブランドにつき 1 つが得られればよい．

ステップ 2. では，HTML のソースファイルから仕様表に該当する箇所を抽出する．価格.com では仕様表は図 3.2 のように ‘tblBorderGray’ で始まる class 属性を持つ table タグでマークアップされている．したがって，上記の table タグで囲まれた領域を仕様表として抽出する．

```
<table width="100%" border="0" cellpadding="0" cellspacing="0" class="tblBorderGray mTop15">
<col width="20%"><col width="30%"><col width="20%"><col width="30%">
<tr class="itemviewColor02b">
<th colspan="4"><strong>基本スペック</strong></th>
</tr>
```

図 3.2: 価格.com での仕様表の例

ステップ 3. では，製品のメーカーのウェブサイトへのリンクを検出する．メーカーのウェブページでは，製品の仕様をまとめた表が掲載されていることがある．本研究では，価格.com だけでなく，メーカーのウェブサイトに掲載されている仕様表からも製品属性を抽出する．価格.com では，メーカーのウェブサイトにおける製品ページへのリンクは「メーカー仕様表」というテキストで表わされているため，容易に検出できる．URL を検

出後，メーカーの製品ページの HTML ソースファイルをダウンロードする．ステップ 1. と同様に，ダウンロードには Python の BeautifulSoup を用いた．

ステップ 4. では，メーカーの製品ページから仕様表を検出する．仕様表は，メーカー毎に異なるフォーマットで記載されているため，検出は容易ではない．ここでは，以下の条件のいずれかを満たす table タグを検索し，その table タグで囲まれた領域を仕様表として抽出する．

- table タグの class 属性が文字列 'spec' を含む．この条件を満たす table タグの例を図 3.3 に示す．
- HTML の DOM において，table タグの直前に出現する兄弟ノードのタグの class 属性が文字列 'spec' を含む．この条件を満たす table タグの例を図 3.4 に示す．
- table タグの直前に出現する兄弟ノードが仕様表を示唆するキーワードを含む．仕様表を示唆するキーワードは,「仕様」「概要」「性能」のいずれかである．この条件を満たす table タグの例を図 3.5 に示す．
- そのページにおける唯一の table タグである．
- ウェブページ内に同一の class 属性を持つ table タグが 10 回以上出現する (10 回以上出現する全ての table タグを仕様表として抽出する). この条件を満たす table タグの例を図 3.6 に示す．

最後の条件について説明する．メーカーのページの中には，１つのウェブページに同じフォーマットで書かれた仕様表が複数個出現することがある．特に，カメラの製品ページでは，複数の表が使われることが多い．このようなウェブページでは，仕様表を表わす table タグの class 属性の値は全て同じである．図 3.6 の例では，'table1' という class 属性を持つ table タグが 10 個以上存在するので [1]，これら全てを仕様表として抽出する．

ステップ 5. では，仕様表を解析し，属性を表わすセルならびに属性値を表わすセルを特定し，属性と属性値の組を抽出する．価格.com から取得した仕様表はフォーマットが決まっているため，属性と属性値は容易に抽出できる. 価格.com から獲得した仕様表の例を図 3.7 に示す．ここで，class 属性が 'itemviewColor03b textL' である th タグは属性セルとし，その後ろの td タグは属性値セルとして，これらのセルから属性と属性値の組を抽出する．ただし，抽出した文字列の中に含まれる ' '(スペースを表わすメタ文字) は削除する．

一方，メーカーのウェブページに掲載されている仕様表からは，以下の手続きにしたがって属性と属性値の組を取得する．

- 表の行に th タグが 0 個，td タグが 2 個以上ある場合

  最初の td から属性，残りの td から属性値を抽出する．抽出の例を図 3.8 に示す．

[1] 図では「記録形式」と「撮影素子」の 2 つの表以外は省略されている．

```
        <table class="tech-specs-table" border="0" cellspacing="0" cellpadding="0">
<tr><td class="title" colspan="2">オペレーションシステム</td></tr>
                    <tr><td class="name-tech">オペレーションシステム</td>
                        <td>Windows 8.1</td>
                    </tr>
```

図 3.3: メーカーのウェブページにおける仕様表の例 (その 1)

```
<div class=s5-header1__heading>仕様表</div>
<!-- /.s5-header1 --></div>
<div class="s5-specTable">
<table>
<col style="width:15%">
```

図 3.4: メーカーのウェブページにおける仕様表の例 (その 2)

```
<h2 class="nakamidashi-wc">仕様</h2>
<table width="530" border="0" cellpadding="0" cellspacing="5" summary="レイアウトテーブル">
  <tr>
    <td width="30%"><span class="txt-m01">最大炊飯容量(L)</span></td>
    <td><span class="txt-m01">1.0 (約1～5.5合)</span></td>
  </tr>
```

図 3.5: メーカーのウェブページにおける仕様表の例 (その 3)

```
<h2>記録形式</h2>
<table class="table1">
<tbody>
<tr>
<th>記録フォーマット</th>
<td>DCF2.0</td>

<h2>撮像素子</h2>
<table class="table1">
<tbody>
<tr>
<th>形式</th>
<td>CMOSセンサー</td>
```

図 3.6: メーカーのウェブページにおける仕様表の例 (その 4)

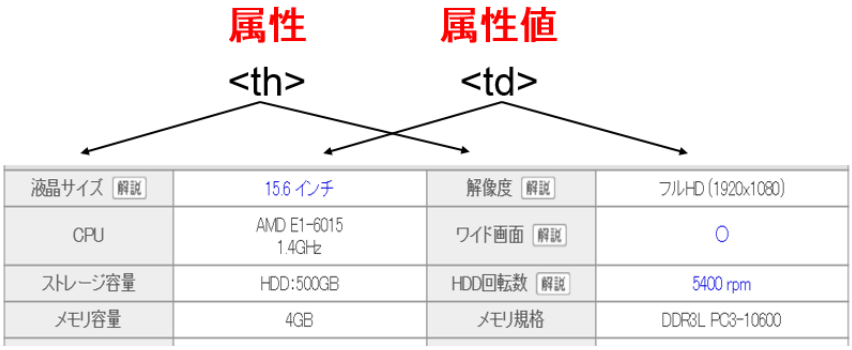

図 3.7: 価格.com での仕様表における属性と属性値の例

- 表の行に th タグが 1 個，td タグが 1 個以上ある場合
  th から属性，td から属性値を抽出する．抽出の例を図 3.9 に示す．
- 表の行に th タグが 2 個，td タグが 1 個ある場合
  2 つの th から属性，td から属性値を抽出する．抽出の例を図 3.10 に示す．
- 表の行に th タグが 2 個あり，td タグが存在しない場合
  最初の th から属性，次の th から属性値を抽出する．抽出の例を図 3.11 に示す．

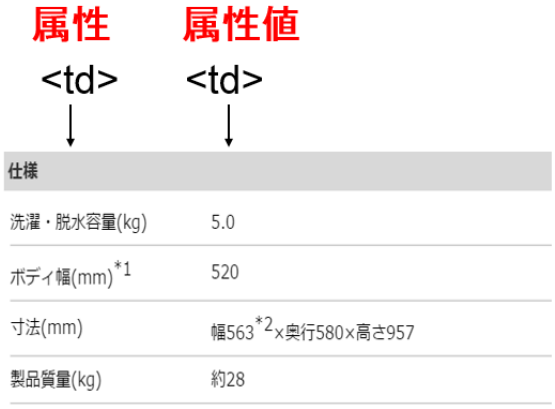

図 3.8: メーカーのページの仕様表における属性と属性値の例 (その 1)

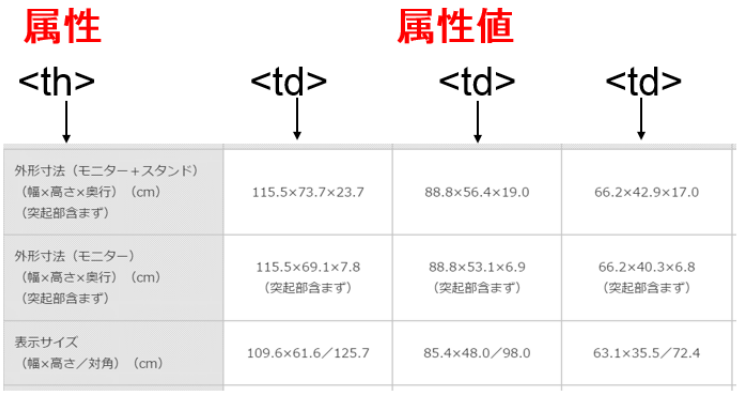

図 3.9: メーカーのページの仕様表における属性と属性値の例 (その 2)

図 3.10: メーカーのページの仕様表における属性と属性値の例 (その 3)

図 3.11: メーカーのページの仕様表における属性と属性値の例 (その 4)

3.2.2 属性のクラスタリング

仕様表から属性と属性値の組を抽出した後，同じ実体を表わす異表記の属性をひとつにまとめるために，属性のクラスタリングを行う．ただし，属性の類似性だけでなく，属性値の類似性も手がかりとしたいため，属性と属性値の組に対してクラスタリングを行う．

まず，属性・属性値の組を素性ベクトルで表現する．素性ならびにその重みの一覧を表 3.1 に示す．A は属性，V は属性値から取得される素性を表わす．属性から取得される素性は，属性そのもの，もしくは属性の部分文字列 (3-gram, 2-gram, 1-gram) である．属性が同じ単語である場合には同じ実体を指すことは明らかなので，これらをひとつにまとめるために,「属性そのもの」の素性に対しては高い重みを与える．部分文字列の素性については，文字列の長さに応じて重みを決める．一方，属性値から取得される素性は，属性値の部分文字列とする．ただし，ここでは文字の n-gram を素性とするのではなく，属性値を区切り文字 (スペース, 括弧,「,」,「：」,「、」,「/」) で区切って得られる部分文字列を素性とする．また，属性値内に出現する数字列は ⟨N⟩ というシンボルに置き換える．属性値ではしばしば数字が使われるが，数字が異なる場合でも類似しているとみなせる属性値が存在する．例えば，パソコンのメモリ容量の属性値として「1GB」「2GB」「8GB」が得られたとき，これらは異なる文字列ではあるものの，全てメモリの容量を表わしているという点では似ている．数字列を ⟨N⟩ で置き換えることで，これら 3 つの属性値は「⟨N⟩GB」という同じ文字列で表現される．また，属性値内の部分文字列の素性の重みは，‘⟨N⟩’ もしくは ‘⟨N⟩+文字列’ というパターンにマッチするときには少し高く設定する．後者は「1GB」のように ‘数字+単位’ という表現であるとみなしている．なお，表 3.1 における素性の重みは直観により決めている．

表 3.1: 属性，属性値の組の素性ベクトル

素性	重み
A　属性そのもの	10
A　文字の 3-gram	3
A　文字の 2-gram	2
A　文字の 1-gram	1
V　属性値内の部分文字列 (⟨N⟩)	2
V　属性値内の部分文字列 (⟨N⟩+文字列)	2
V　属性値内の部分文字列 (その他)	1

属性・属性値の組の素性ベクトルの例を図 3.12 に挙げる．同図において，矢印の左は属性と属性値の組，矢印の右は抽出された素性のリストを表わす．また，[ ] 内は素性に対する重みを示している．

全ての属性・属性値に対する素性ベクトルを得た後，凝集型クラスタリングアルゴリ

(液晶サイズ,15.6 インチ)	→	液晶サイズ [10], 液晶サ [3], 晶サイ [3], サイズ [3], 液晶 [2], 晶サ [2], サイ [2], イズ [2], 液 [1], 晶 [1], サ [1], イ [1], ズ [1], ⟨N⟩ インチ [2]
(解像度,Full HD 1080p)	→	解像度 [10], 解像度 [3], 解像 [2], 像度 [2], 解 [1], 像 [1], 度 [1], Full[1], HD[1], ⟨N⟩p[2]

図 3.12: 属性・属性値の素性ベクトルの例

ズムによってクラスタリングを行う．クラスタリングのツールとして CLUTO[2] を用いた．ここでは CLUTO の入力ファイルを作成する過程を簡単に説明する．まず，クラスタリングの対象とする属性・属性値の組の集合から取得された全ての素性のリストを作成する．リストのフォーマットを図 3.13 に示す．␣ は半角スペースを表わす．また，各素性にはユニークな識別番号を付ける．

1␣素性 1␣重み
2␣素性 2␣重み
……
……
N␣素性_n␣重み

(1,2,...,N は素性の識別番号を表わす)

図 3.13: 素性と重みのリスト

次に，属性・属性値の組に対する素性ベクトルのリストを作成する．そのフォーマットを図 3.14 に示す．各行が 1 つの属性・属性値を表わす．また，素性は文字列ではなく図 3.13 における素性の識別番号で表わされている．

[2]http://glaros.dtc.umn.edu/gkhome/cluto/cluto/download/

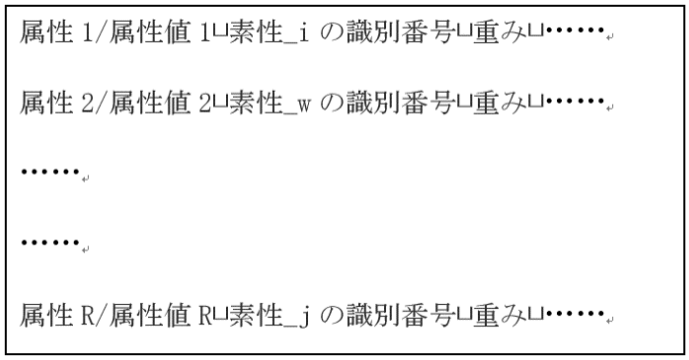

図 3.14: 属性・属性値と素性ベクトルのリスト

最後に，CLUTO の入力ファイルを作成する．そのフォーマットを図 3.15 に示す．このファイルでも，各行は属性・属性値の組の素性ベクトルを表わすが，素性の識別番号 (feature_i) と重みのみが表記されている．

図 3.15: CLUTO の入力ファイル

凝集型クラスタリングでは，クラスタ数をあらかじめ定義する必要がある．ここでの理想的なクラスタ数とは，同一実体を表わす属性をひとつにまとめたときの属性の異なり数に相当する．しかし，この数を事前に推測することは困難である．本研究では，クラスタ数はクラスタリングの対象とする属性・属性値の組の総数の 90%と設定する．クラスタ数の最適な設定方法を探究することは今後の課題である．

クラスタリング後，属性値を取り除いて，初期の製品属性異表記辞書 D_t を得る．このとき，式 (3.1) において，クラスタが A_i に，クラスタ内に属する属性が a_{ij} に該当する．

3.3 レビュー文からの属性抽出

本節では，製品カテゴリに関する製品のレビュー文から異表記の製品属性を獲得する手法について述べる．

3.3.1 レビュー文の収集

まず，与えられた製品カテゴリに属する製品に関するレビュー文を収集する．本研究では，価格.com に掲載されているレビュー文を用いる．価格.com では，製品毎に，ユーザがその製品に対するレビューを書き込むことができる．図 3.16 は，価格.com に投稿されたノートパソコンに関するユーザのレビューの例である．このようなユーザレビューのウェブページの HTML ソースファイルをダウンロードする．価格.com では，ユーザのレビュー文は，図 3.17 に示すように，class 属性が ‘revEntryCont’ である p タグによってマークアップされている．そこで，`<p class="revEntryCont">` というタグを検索し，このタグで囲まれたテキストをレビューテキストとして抽出する．

価格.com では，1 つの製品に対するユーザーレビューが複数のウェブページに掲載されていることがある．このとき，図 3.16 で示すようなウェブページの下部には，次のページへのリンクが存在する．HTML ソースでは，次ページへのリンクは，図 3.18 に示すように，class 属性が ‘arrowNext01’ である a タグによってマークアップされている．したがって，`<a href="..." class="arrowNext01">`というタグを検索し，その href 属性に記載されている URL を取得して，次のユーザレビューのページをダウンロードする．ダウンロードしたウェブページから，先ほどと同じ処理でレビューテキストを取得する．この操作を再帰的に繰り返し，レビューテキストの集合を得る．

3.3.2 属性抽出パターンの候補の獲得

獲得したレビューテキストに対して以下の前処理を行う．

- `<br>`タグなどの HTML タグを除去する．
- テキストを文に分割する．レビューテキストの中に出現する「.」「。」「\n(改行記号)」という記号をスペースに置き換える．次に，テキストをスペースで分割する．分割して得られたテキストの段片を文とみなす．
- 形態素解析を行う．形態素解析ツールとして MeCab[3] を使用した．

本研究では，パターンマッチによってレビュー文から異表記の製品属性を抽出する．以後，製品属性を抽出するためのパターンを「製品属性抽出パターン」と呼ぶ．レビューテキストに対して上記の前処理を実行した後，製品属性抽出パターンの候補を獲得する．

図 3.16: 価格.com におけるレビュー文の例

フルHDでコストパフォーマンスがよかったので購入しました。

このスペックを必要とする使い方はしていませんが、処理速度は十分です。

フルHDのきれいな画面で作業するのは、やっぱり気持ちがいいですね。

とくに不満はありませんが、シルバーのボディを選べればよかったと思います。

図 3.17: 価格.com におけるレビュー文の例 (HTML ソース)

次のページへ

図 3.18: 価格.com におけるレビューページへのリンクの例 (HTML ソース)

[属性] w_1 [評価語]
[属性] w_1 w_2 [評価語]
[属性] w_1 w_2 w_3 [評価語]
[属性] w_1 w_2 w_3 w_4 [評価語]
[属性] w_1 w_2 w_3 w_4 w_5 [評価語]
[評価語] w_1 [属性]
[評価語] w_1 w_2 [属性]
[評価語] w_1 w_2 w_3 [属性]
[評価語] w_1 w_2 w_3 w_4 [属性]
[評価語] w_1 w_2 w_3 w_4 w_5 [属性]

図 3.19: 製品属性抽出パターンのテンプレート

製品属性抽出パターンのテンプレートを図 3.19 のように定義する．

図 3.19 のテンプレートにおいて，[属性] は初期の製品属性異表記辞書に含まれる属性 (仕様表から獲得した属性)，[評価語] は,「よい」「悪い」「素晴しい」など，ある事物に対する評価を表わす単語である．本研究では，日本語評価極性辞書 (用言編)[1, 2] に登録されている語を [評価語] と定義する．一方，w_i は [属性] と [評価語] の間に出現する任意の単語にマッチする変数である．図 3.19 に示すようにパターンは 10 種類あり，5 つは [属性] の後に [評価語] が出現するパターン，もう 5 つは [評価語] の後に [属性] が出現するパターンである．仕様表から抽出した属性と評価語が 5 語以内の範囲で出現するとき，[属性] と [評価語] の間の単語を変数 w_i に当てはめ，属性を抽出するパターンの候補を作成する．レビュー文，ならびにそれから取得される製品属性抽出パターンの例を図 3.20 に示す．

レビュー文 A の形態素解析の結果:

キーボード　の　タッチ　も　いい　意味　で　ＤＥＬＬ　っぽくっ　て

「キーボード」は初期の製品属性異表記辞書に含まれる属性である．
「いい」は日本語評価極性辞書に登録されている評価語である．

パターンのテンプレートを適用し，以下のパターンの候補を得る．
パターン P: [属性] の タッチ も [評価語]

図 3.20: パターンの候補の獲得例

製品属性抽出パターンを用いてレビュー文から製品属性を抽出する過程を説明する．与えられたレビュー文に対し，製品属性抽出パターンにおける [属性] と [評価語] の間にある単語列 $w_1 \cdots w_l (1 \leq l \leq 5)$ を検索する．単語列 $w_1 \cdots w_l$ が見つかったとき，その後の単語が評価語であるかをチェックし，評価語である場合には $w_1 \cdots w_l$ の前に出現する単語を製品属性として抽出する．ここで，単語列の前に複合名詞が存在するときは，複合名詞全体を１つの属性として抽出する．具体的には，$w_1 \cdots w_l$ の前に出現する単語の品詞が名詞であり，その前にも名詞が出現するときは，名詞以外の品詞の単語が出現するまで前方に単語を辿り，これらの名詞の連続を複合名詞の製品属性として抽出する．また，単語列 $w_1 \cdots w_l$ の前の単語が評価語であるときには，$w_1 \cdots w_l$ の後に出現する単語を製品属性として抽出する．先ほどと同様に，$w_1 \cdots w_l$ の後に連続する名詞が出現しているときは，それら全てを複合名詞とみなし，その複合名詞を製品属性として抽出する．製品属性抽出パターンによって製品属性を抽出する例を図 3.21 に示す．

[3]http://taku910.github.io/mecab/

パターン P: [属性] の 感触 も [評価語]

パターン P にマッチする文:

タイピング の 感触 も 安っぽい 音 が せ ず

単語列「の 感触 も」が見つかり,「も」の後の「安っぽい」は評価語であるので,「の」の前に出現する「タイピング」を製品属性として抽出する.

図 3.21: 製品属性抽出パターンによって製品属性を抽出する例

製品属性抽出パターンは,[評価語] をパターンに含む.これは,製品に対する何らかの評価を表わしている文には製品属性が出現しやすいという仮定に基づいている.すなわち,「良い」「悪い」のようなユーザの評価を表わす語が文中に出現したときは,製品の属性に言及している可能性が高いと考えられる.また,製品属性抽出パターンに [属性] と [評価語] の間に出現する単語列を加えているのは,製品属性が出現する典型的な文脈を学習するためである.

3.3.3 製品属性抽出パターンの選別

前項で説明した手法で獲得した製品属性抽出パターンの候補の中には,パターンとして適切なものもあればそうでないものも存在する.ここでは,製品属性抽出パターンの候補の妥当性を検証し,適切なパターンのみを選別する.そのため,製品属性抽出パターンの候補のスコアを算出し,スコアが高いものを最終的な製品属性抽出パターンとして採用する.

製品属性抽出パターンの候補を P_i とおき,そのスコア $S(P_i)$ を式 (3.2) のように定義する.すなわち,初期の製品属性異表記辞書 D_t の中に含まれる属性を多く抽出できるパターンほど信頼性が高いとみなす.

$$S(P_i) = \frac{\text{パターン } P_i \text{ にマッチしかつ } D_t \text{中の属性が抽出される文の数}}{\text{パターン } P_i \text{ にマッチする文の数}} \tag{3.2}$$

製品属性抽出パターンのスコアの計算例を図 3.22 に示す.パターン P にマッチする文は 5 つあり,下線が引かれた単語が製品属性として抽出される.これらのうち,「キーボード」は初期の製品属性異表記辞書 D_t に登録されているが,「キー」は登録されていない.したがって,パターン P にマッチする 5 つの文のうち,3 つの文からは D_t に登録されている属性が抽出されるため,スコアは 0.6 となる.

パターン P: [属性] の タッチ も [評価語]

パターン P にマッチする文:

キーボード のタッチもいい意味で D E L L っぽくって
キーボード のタッチも安っぽいですが、使用に困ることはありません
全体の使用感ですが、安っぽさはなく キー のタッチもまずまずです
キーボード のタッチも良い
キー のタッチも良いです

スコア:

$$S(P) \;=\; \frac{3}{5} \;=\; 0.6$$

図 3.22: 属性抽出パターンのスコアの計算例

本研究では，以下の 2 つの条件を満たすパターンの候補を最終的な製品属性抽出パターンとして採用する．

1. マッチする文の数が 3 以上である．

2. $S(P_i)$ が 0.5 以上である．

1. の条件は，レビューテキストで少ない回数しかマッチしないパターンは有効ではないという考えに基づいて設定した．

スコアが 1 であるとき，そのパターンによって抽出できる属性は，全て初期の製品属性異表記辞書に含まれている．したがって，そのパターンからは新しい製品属性は獲得できないことになる．しかしながら，本研究では，計算時間を短縮するため，スコアの計算に用いるレビュー文は，3.3.1 項でダウンロードしたレビュー文の一部である．すなわち，収集したレビュー文集合の部分集合に対して式 (3.2) のスコアを計算している．したがって，スコアが 1 のパターンを，スコアの計算に用いたレビュー文以外に適用した場合には，D_t に含まれない新たな属性を獲得できる可能性がある．また，本研究で収集していないレビュー文に適用したときも，新しい属性を獲得できる可能性がある．そのため，$S(P_i) = 1$ となるパターンの候補 P_i も最終的な製品属性抽出パターンとして獲得する．

3.4 製品属性異表記辞書の構築

3.2 節で獲得した初期の製品属性異表記辞書 D_t と，3.3 節の方法で獲得した製品属性を統合して，最終的な製品属性異表記辞書を得る．統合は，パターンマッチで獲得した属性を D_t に併合することで実現する．パターン P_j によって得られる属性のうち，D_t に既に

含まれている属性の集合を K_j，含まれていない属性の集合を U_j とおく．K_j の要素は D_t における属性集合 A_i のいずれかに属する．K_j の中で出現頻度が最大の A_i を求め，P_j は A_i の属性の異表記を抽出するためのパターンとみなす．そして，U_j を A_i に追加する．

製品属性抽出パターンによって得られた属性を製品属性異表記辞書に統合する例を図 3.23 に示す．この例では，パターン P にマッチする文は 8 個あり,「キーボード」「タイピング」「キー」が属性として抽出される．このうち,「キーボード」は初期の製品属性異表記辞書 D_t に登録されている語であるので，K の要素とする.「タイピング」と「キー」は D_t に登録されていないので，U の要素を「キーボード」の異表記として製品属性異表記辞書に登録する．

パターン P: [属性] の 感触 も [評価語]

パターン P にマッチする文:

キーボード の感触も浅いですが
キーボード の感触も良いと思います
タイピング の感触も安っぽい音がせず
キー の感触もいい感じです
キーボード の感触も良い
キーボード の感触もいいですが
キー の感触もいい感じ
キーボード の感触も満足でした

キーボード　∈ K
タイピング, キー　∈　U

製品属性異表記辞書: A={ タイピング，キー，キーボード }

図 3.23: 製品属性異表記辞書を構築する例

仮に，図 3.23 に示したパターン P から以下のような属性が抽出されたとする．括弧内は属性が抽出された回数とする．

キーボード (4 回)∈ K
コーティング (1 回)∈ K
タイピング (1 回)∈ U
キー (1 回)∈ U

K の 2 つの要素「キーボード」と「コーティング」は異なる製品属性であるが，抽出回数が多いのは「キーボード」である．したがって，パターン P は「キーボード」の異表記

を獲得するためのパターンとみなし，U の要素である「タイピング」と「キー」は「キーボード」の異表記として辞書に登録する．

第4章　評価実験

本章では，提案手法の評価実験について述べる．まず，4.1 節で実験で使用したデータについて説明する．次に，4.2 節で実験結果について述べる．4.2.1 項では仕様表から属性を抽出する手法，4.2.2 項では属性のクラスタリング手法，4.2.3 項ではレビュー文から属性を抽出する手法を評価する．最後に 4.3 節では実際に獲得された異表記の属性の例を示す．

4.1　実験データ

提案手法では，製品カテゴリを入力として与え，その製品カテゴリに関連する製品の属性の異表記辞書を自動構築する．評価対象として設定した 9 つの製品カテゴリを図 4.1 に示す．これらは価格.com における製品カテゴリの名称と一致する．なお，図 4.1 における括弧内の単語は，これ以降で実験結果を示すときに用いる製品カテゴリの略称である．

ノートパソコン (PC)，デジタル一眼カメラ (カメラ)， テレビ，腕時計，冷蔵庫，炊飯器，洗濯機， 電子レンジ (レンジ)，エアコン

図 4.1: 評価対象とした製品カテゴリ

表 4.1 は，各カテゴリ毎に，仕様表を抽出するために用いたメーカーのウェブサイトの数，ならびに収集したレビュー文の数を示している．

表 4.1: 実験データ

製品カテゴリ	PC	カメラ	テレビ	腕時計
メーカー数	6	4	8	2
レビュー文数	240690	327544	280790	109980

製品カテゴリ	冷蔵庫	炊飯器	洗濯機	レンジ	エアコン
メーカー数	3	6	4	3	1
レビュー文数	44042	45547	73951	31424	37450

本論文の提案手法では，3.2 節で述べたように，メーカーのウェブサイトの仕様表から属性を抽出する．表 4.1 に示すように，製品カテゴリによってばらつきはあるが，平均 4 つ

表 4.2: 仕様表の抽出結果

	価格.com	メーカーページ	両方
PC	6	12	18
カメラ	4	54	58
テレビ	8	16	24
腕時計	2	1	3
冷蔵庫	3	5	8
炊飯器	6	6	12
洗濯機	4	5	9
レンジ	3	3	6
エアコン	1	1	2
合計	37	103	140

のメーカーのウェブサイトの仕様表から属性を獲得することを試みた．一方，提案手法ではレビュー文からも属性の抽出を試みる．3.3 節で述べたように，レビュー文は価格.com のサイトから収集した.「ノートパソコン (PC)」「デジタル一眼カメラ (カメラ)」「テレビ」についてはおよそ 25 万から 35 万文程度の比較的大量のレビュー文を取得できた．一方，それ以外の製品カテゴリについては，取得されたレビュー文の数は 5 万文以下である場合が多い．これらの製品カテゴリについては，将来は価格.com 以外のショッピングサイトからもレビュー文を取得し，レビュー文の数を増やすことが必要である．

4.2 実験結果

4.2.1 仕様表からの属性抽出の評価

ここでは，3.2 節で提案した，仕様表から製品属性を抽出する手法を評価する．

価格.com ならびにメーカーページから抽出した仕様表の数を表 4.2 に示す．価格.com よりもメーカーページの方がより多くの仕様表が抽出されている．これは，メーカーページでは 1 つの製品ページにつき 2 つ以上の仕様表が掲載されていることがあるためである．特に,「デジタル一眼カメラ」については，メーカーは多くの仕様表を載せる傾向がある．

次に，仕様表から抽出した属性・属性値の組の数を表 4.3 に示す．価格.com とメーカーページでは抽出される属性・属性値の組の数に大きな差はない．価格.com における仕様表には基本的な属性が一通り書かれているためであると推察できる．デジタル一眼カメラ (カメラ) だけは例外で，メーカーページからの抽出数は価格.com の約 2 倍である．これは，デジタル一眼カメラのメーカーページにはカメラの詳細な機能や属性が細かく記載されているためである．

表 4.3: 属性・属性値の組の抽出結果

	価格.com	メーカーページ	両方
PC	261	219	480
カメラ	150	316	466
テレビ	288	194	482
腕時計	4	9	13
冷蔵庫	64	41	105
炊飯器	76	83	159
洗濯機	56	72	128
レンジ	47	52	99
エアコン	26	41	67
合計	972	1027	1,999

表 4.4: メーカーページから抽出された仕様表ならびに属性・属性値の組の評価

	仕様表	属性・属性値
精度	0.89	0.90
再現率	0.82	–

次に，抽出された仕様表を評価する．価格.com では，仕様表は常に class 属性が tblBorderGray である table タグでマークアップされているため，抽出された仕様表は全て正しいとみなす．一方，メーカーページから抽出された仕様表については，取得したメーカーページを全て人手でチェックし，仕様表に該当する table タグを決定した．これらを正解データとみなしたときの仕様表の精度と再現率を求めた．精度と再現率の正確な定義は式 (4.1) と式 (4.2) の通りである．

$$\text{精度} = \frac{\text{提案手法によって抽出された正しい仕様表の数}}{\text{提案手法によって抽出された仕様表の総数}} \tag{4.1}$$

$$\text{再現率} = \frac{\text{提案手法によって抽出された正しい仕様表の数}}{\text{メーカーページに存在する仕様表の総数}} \tag{4.2}$$

精度と再現率を表 4.4 の「仕様表」の列に示す．精度，再現率とも十分に高いことがわかる．

次に，抽出された属性・属性値の組を評価する．但し，本研究では，製品属性を抽出することを目的とするため，属性・属性値の組を評価する際には，属性のみが適切であるかを判定する．価格.com では仕様表のフォーマットが決まっているため，抽出された属性・属性値は全て正しいとみなす．一方，メーカーページから取得した属性・属性値については，全ての製品カテゴリを対象に抽出されたものの中からランダムに選択した 100 個

の属性を人手で調べ，正解率 (精度) を求めた．なお，今回の実験では再現率は評価しなかった．属性・属性値の抽出精度を表 4.4 の「属性・属性値」の列に示す．また，ランダムに選択した 100 個の属性の候補とそれに対する判定の一部を図 4.2 に示す．精度は 0.90 となり，実用的な観点からも十分に高いことがわかった．ただし，精度を算出するために調べた属性の数は少なく，再現率も示していないため，十分な評価であるとは言い難い．提案手法をより精密に評価することが今後の課題となる．

4.2.2 属性のクラスタリングの評価

提案手法では，3.2.2 項で述べたように，属性・属性値の組を抽出した後，同じ実体を表わす属性を 1 つにまとめるためにクラスタリングを行う．ここではクラスタリングを評価する．本実験では，式 (4.3) に示す Purity を評価基準とする．majority(A_i) は属性のクラスタ A_i の中で同一実体を指す属性の最大値である．また，C は，作成されたクラスタのうち，$|A_i| > 1$ であるクラスタの集合である。すなわち，要素数 1 のクラスタは評価から除外した。

$$\text{Purity} = \sum_{A_i \in C} \frac{|A_i|}{|C|} \cdot \frac{\text{majority}(A_i)}{|A_i|} \tag{4.3}$$

図 4.3 は，実際に得られたクラスタを表わす．表の各行が一つの属性・属性値の組を各セルが一つのクラスタを表わす．属性と属性値の組は「/」で区切って表示している．また，各クラスタの Purity も掲載した．例えば,「ノートパソコン」の 3 番目のクラスタでは,「ポインティングデバイス仕様」と「ポインティングデバイス」は同一の属性を表わす．一方,「キーボード」は別の属性を表わすので，Purity は 2/3=0.66 となる．

クラスタリングの評価結果を表 4.5 に示す．表 4.5 では，製品カテゴリ毎に，得られたクラスタの総数，そのうち要素数 $|A_i|$ が 1 より大きいクラスタの数，ならびに Purity を示している．一方，表 4.6 は，全ての製品カテゴリ，ならびに 9 つの製品カテゴリの平均について，表 4.5 と同じ指標を示したものである．

クラスタリングの Purity は，全体で 0.829 と高い．表 4.5 に示すように，製品カテゴリ毎に見ても,「腕時計」で 0.500,「エアコン」で 0.667 であるが，それ以外のカテゴリでは 0.75 以上と高い値が得られている．一方，要素数が 1 より大きいクラスタの数が少ないことから，異表記の属性が同じクラスタにまとめられていない可能性がある．クラスタリングの際に設定するクラスタの数を小さくすればより大きなクラスタを構築できるが，Purity は低下するだろう．現在はクラスタ数は全属性数の 90%と設定しているが，今後は最適化なクラスタ数を決める方法を探究する必要がある．

属性の候補	属性であるかどうか
画像位置自動調整	○
インストール OS	○
光学式ドライブ光学式ドライブ	×
ミニ AC アダプター	○
アスペクト比	○
製品重量	○
予約タイマー 1〜24 時間後	×
輝度	○
有効画素数	○
使用可能湿度	○
露出補正	○
AF モード	○
画角	○
オーブン温度調節範囲	○
方式	×
年間消費電力量	○
電源仕様	○
多重枚数	○
応答速度	○
スピーカー出力	○
映像入力端子	○
音声入力端子	○
使用環境	○
リモコン端子	○
参考上代	×
畳数の目安	○
ヘッドフォン端子	○
HDMI ミニ出力端子	○
電池情報	○
質量	○
大きさ	○
使用レンズ	○
商品名	○
テレビチューナー	○
パネルサイズ	○

図 4.2: 抽出された属性とそれに対する判定

「ノートパソコン」

番号	属性/属性値	Purity
1	電源 AC アダプター/入力 AC100V〜240V ± 10 % ミニ AC アダプター/入力：AC100V〜240V	100%
2	キーボードバックライト/あり バックライト/LED	100 %
3	ポインティングデバイス仕様/タッチパッド キーボード/タッチパッド ポインティングデバイス/タッチパッド	66.67 %
4	オーディオ機能インターフェース/インテル HighDefinitionAudio 準拠 インターフェース/USB3.0 ポート× 2	50 %
5	イーサーネットポート/あり イーサーネット/GigabitEthernet	100 %

「デジタル一眼カメラ」

番号	属性/属性値	Purity
1	ファインダー倍率/10.95 ファインダー形式/ペンタプリズム	50%
2	Wi-Fi/ Wi-FiDirect 対応/	100 %
3	表示言語/日本語 メニュー表示言語/日本語、英語	100 %

「テレビ」

番号	属性/属性値	Purity
1	地上デジタルチューナー/ 110 度 CS デジタルチューナー/ BS デジタルチューナー/	33.33%
2	液晶パネル/LED パネル 液晶パネル方式/IPS 方式直下型 LED パネル	100 %

図 4.3: 属性・属性値の組のクラスタリング結果の例

表 4.5: 製品属性のクラスタリングの評価 (製品カテゴリ別)

製品カテゴリ	PC	カメラ	テレビ	腕時計	
クラスタ数	183	254	157	8	
$\|A_i\| > 1$ のクラスタ数	16	29	18	2	
Purity	0.771	0.845	0.889	0.5	

製品カテゴリ	冷蔵庫	炊飯器	洗濯機	レンジ	エアコン
クラスタ数	44	69	67	53	45
$\|A_i\| > 1$ のクラスタ数	6	8	8	8	6
Purity	0.75	0.9375	0.875	0.875	0.667

表 4.6: 製品属性のクラスタリングの評価 (全ての製品カテゴリ)

	全カテゴリ	平均
クラスタ数	880	97.8
$\|A_i\| > 1$ のクラスタ数	101	11.2
Purity	0.829	0.790

4.2.3 レビュー文からの属性抽出の評価

ここでは，3.3 節で述べた手法の評価について述べる．すなわち，レビュー文から獲得されたパターン，ならびにそれを用いたパターンマッチによってレビュー文から獲得された異表記の属性を評価する．今回の実験では，表 4.1 に示すように，比較的大量のレビュー文を収集することのできた「ノートパソコン (PC)」「デジタル一眼カメラ (カメラ)」「テレビ」の 3 つの製品カテゴリのみを評価対象とした．

表 4.7 は，獲得されたパターンの数，パターンによって抽出された属性の数，そのうち初期の製品属性異表記辞書 D_t に登録されていない (新たに獲得できた) 属性の数，そのうち正しい異表記の属性とみなせるものの数，及び抽出精度を示している。抽出精度の定義は式 (4.4) の通りである。

$$\text{抽出精度} = \frac{D_t\text{に含まれない属性のうち異表記とみなせるものの数}}{D_t\text{に含まれない属性数}} \tag{4.4}$$

抽出精度は低く，改善の余地がある．現時点では，パターンのテンプレートは，図 3.19 に示すように，属性と評価語およびその間の単語という単純なものしか採用していない．属性抽出のための条件をより精密に定義できるようなテンプレートを用意することで，抽出精度を改善できると考えられる．

今回の実験では,「ノートパソコン」「デジタル一眼カメラ」「テレビ」以外の 6 つの製品カテゴリについては，十分な量のレビュー文が収集できず，パターンの自動獲得が困難

表 4.7: パターンマッチによる属性抽出の評価

	PC	カメラ	テレビ
パターン数	31	16	5
抽出属性数	108	64	65
D_t に含まれない属性数	69	45	58
異表記とみなせる属性数	26	3	7
抽出精度	0.34	0.067	0.12

であると考えられたため，評価対象とはしなかった．今後，これらの製品カテゴリについてより多くのレビュー文を収集し，提案手法の評価を行う予定である．

4.3 獲得されたパターンと属性の例

「ノートパソコン」「デジタル一眼カメラ」「テレビ」のカテゴリにおいて，提案手法で獲得された製品属性抽出パターンをそれぞれ図 4.4, 図 4.5, 図 4.6 に示す．それぞれの製品カテゴリのレビュー文でよく使われると思われる言い回しがパターンとして学習されていることがわかる．図 4.5 において，‘】’ を含むパターンが多く学習されているのは，以下の例のように，属性を ‘【’ と ‘】’ で囲んで各属性に対するコメントを箇条書きのように記述するレビュー文が多かったためである．

【デザイン】個人的には良いと思います.
【画質】写りは悪いです.

パターンマッチによって獲得された異表記の属性の例を獲得に用いたパターンとともに図 4.7 に示す．*が付いている属性は D_t にも含まれている属性を表わす．最初の例では,「キーボード」の異表記として,「キー」や「タイピング」が得られている．これらを含む意見文はキーボードに対する評価を表わすとみなせる．2 番目の例では,「バッテリー」の異表記として「バッテリ」や「電池」が得られている．3 番目の例では,「サイズ」の異表記として「大きさ」が得られている．

[属性] の もち も [評価語]
[属性] の 持ち は かなり [評価語]
[属性] の 打 感 は [評価語]
[属性] も 打ち やすく [評価語]
[属性] の タッチ 感 は [評価語]
[属性] の 音質 も [評価語]
[属性] の 持ち も [評価語]
[属性] の もち が [評価語]
[属性] の 音 は [評価語]
[属性] の 音 も [評価語]
[属性] の 作り が [評価語]
[属性] の 打ち やす さ は [評価語]
[評価語] 日本 の [属性]
[属性] の 持ち が [評価語]
[属性] の 持ち は [評価語]
[属性] の 音 が [評価語]
[属性] の タッチ が [評価語]
[属性] の 劣化 を [評価語]
[属性] の タッチ は [評価語]
[属性] の 位置 に [評価語]
[属性] の もち は [評価語]
[評価語] ぎりぎり の [属性]
[属性] も 使い やすく [評価語]
[評価語] 場所 で も [属性]
[属性] の 反応 が 若干 [評価語]
[属性] の 持ち は 非常 に [評価語]
[属性] の 感触 も [評価語]

図 4.4: 「ノートパソコン」カテゴリの製品属性抽出パターン

[属性] レス の [評価語]
[属性] 】 個人 的 に は [評価語]
[属性] 】 すごく [評価語]
[属性] 】 画質 は [評価語]
[属性] レス で [評価語]
[評価語] の 高 [属性]
[属性] 】 抜け の [評価語]
[属性] 】 凄く [評価語]
[属性] 】 これ は 、 [評価語]
[属性] 特性 が [評価語]
[評価語] ！ 高 [属性]
[属性] そのもの は [評価語]
[属性] 】 写り は [評価語]
[属性] に 大変 [評価語]

図 4.5: 「デジタル一眼カメラ」カテゴリの製品属性抽出パターン

[属性] の 操作 性 は [評価語]
[属性] の 反応 は [評価語]
[属性] の 応答 は [評価語]
[属性] は どう でも [評価語]
[属性] の 反応 も [評価語]

図 4.6: 「テレビ」カテゴリの製品属性抽出パターン

パターン: [属性] の 感触 も [評価語] 製品属性: キーボード*、キー、タイピング
パターン: [属性] の 持ち も [評価語] 製品属性: バッテリー*、バッテリ、電池
パターン: [評価語] ぎりぎり の [属性] 製品属性: サイズ*、大きさ

図 4.7: 獲得された異表記の製品属性の例

第5章　結論

5.1　まとめ

本論文では，製品属性を対象とした評判情報分析のための基礎的な知識として，製品の仕様表ならびにレビュー文から製品属性を抽出し，また同一の実体を表わす製品属性を認識することで，異表記の属性を集めた製品属性異表記辞書を自動構築する手法について述べた．提案手法は 2 つの段階に分けられる．

第一段階では，価格.com およびメーカーページの仕様表から属性を抽出した．まず，入力として与えられた製品カテゴリに該当する製品の仕様表を抽出した．メーカーページからは，仕様表が満たすべき条件を精査し，その条件を満たす表を抽出した．そして，抽出した仕様表から属性・属性値の組を獲得した．次に，表記は異なるが同じ属性を表わす語をひとつにまとめるため，獲得した属性・属性値の組のクラスタリングを行った．属性・属性値の組から素性を抽出し，その重みを決定し，属性・属性値の組を素性ベクトルで表現した．この素性ベクトルを用いて凝集型クラスタリングアルゴリズムによってクラスタリングを実行した．クラスタ数は属性・属性値の組の総数の 90%と設定した．最後に，クラスタリングの結果から属性値を削除して，初期の製品属性異表記辞書を得た．評価実験の結果，仕様表からの属性抽出の精度ならびにクラスタリングの Purity は十分高く，初期の辞書を構築する手法として有望であることがわかった．

第二段階では，価格.com におけるレビュー文の集合から，異表記の属性を抽出するパターンを自動獲得し，それらを用いたパターンマッチングによって異表記の属性を抽出した．まず，価格.com のサイトからレビューテキストを大量に収集した．取得したテキストを文単位に分割し，MeCab で形態素解析を行った．形態素解析されたレビュー文の集合に対して，属性抽出パターンのテンプレートを適用し，製品属性を抽出するパターンの候補を獲得した．獲得したパターンの候補に対し，初期の製品属性異表記辞書 D_t の中に含まれる属性を多く抽出できるパターンほど信頼性が高いという仮定の下，そのスコアを計算した．スコアがある閾値以上のパターンを獲得し，それらをレビュー文の集合に適用し，製品属性を獲得した．最後に，獲得した新たな製品属性は，既存の初期の製品属性異表記辞書の中のいずれかの属性の異表記とみなして，最終的な製品属性異表記辞書を得た．ただし，レビュー文から獲得されたパターンによる属性抽出の精度は低く，今後の課題として残された．

5.2 今後の課題

提案手法では，メーカーページから仕様表を抽出する際には，あらかじめ設定した条件を満たすtableタグでマークアップされている表を取得した．しかし，実際のメーカーページでは，tableタグ以外のタグで記述された仕様表も存在する．今後，そのような仕様表も抽出の対象とすることで，より多くの製品属性を抽出したい．一方，属性のクラスタリングにおいて,クラスタ数は全属性数の90%と設定しているが，今後は最適なクラスタ数を決め，同じ実体を表わす異表記の製品属性を正確に認識する手法を探究する予定である．

本論文の評価実験では,「ノートパソコン」「デジタル一眼カメラ」「テレビ」以外の6つの製品カテゴリについては，十分な量のレビュー文が収集できず，パターンの自動獲得が困難であると考えられたため，パターンマッチによる属性抽出手法の評価対象とはしなかった．今後，これらの製品カテゴリについて，他のショッピングサイトからより多くのレビュー文を収集し，提案手法を適用し，評価を行う必要がある．一方,「ノートパソコン」「デジタル一眼カメラ」「テレビ」の製品カテゴリについては，レビュー文からの属性抽出の精度は低かった．属性抽出のための条件をより精密に定義できるようなテンプレートを用意することで，抽出精度を改善できると考えている．

最後に，今後の重要な課題として，提案手法の実用的な評価が挙げられる．すなわち，実際に提案手法で構築した製品属性異表記辞書を用いて評判情報分析を行い，製品属性異表記辞書が評判情報分析の性能向上にどれだけ貢献するかを検証する．

謝辞

本研究に際して、様々なご指導を頂きました白井清昭准教授に深謝いたします。研究において貴重な意見を頂いた東条敏教授，飯田弘之教授に感謝致します。また, 本研究の趣旨を理解して頂き, 議論などを通じて意見を頂いた白井研究室に所属する学生の皆様に感謝致します。

参考文献

[1] 東山昌彦, 乾健太郎, 松本裕治. 述語の選択選好性に着目した名詞評価極性の獲得. 言語処理学会第 14 回年次大会発表論文集, pp. 584–587, 2008.

[2] 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.

[3] 駒田康孝, 山名早人. 商品評価ツイートからの属性語自動抽出手法の提案. 第 12 回日本データベース学会年次大会, 2014.

[4] Takeshi Masuyama, Satoshi Sekine, and Hiroshi Nakagawa. Automatic construction of japanese KATAKANA variant list from large corpus. In *Proceedings of International Conference on Computational Linguistics*, pp. 1214–1219, 2004.

[5] 大前信弘, 黄瀬浩一. Web の表を対象とした属性の自動識別. 情報処理学会研究報告, 2006-NL-171, Vol. 2006, No. 1, pp. 43–48, 2006.

[6] Kiyonori Ohtake, Youichi Sekiguchi, and Kazuhide Yamamoto. Detecting transliterated orthographic variants via two similarity metrics. In *Proceedings of International Conference on Computational Linguistics*, pp. 709–715, 2004.

[7] 新里圭司, 関根聡. 商品説明文からの属性・属性値の自動抽出. 言語処理学会第 19 回年次大会発表論文集, pp. 7–10, 2013.

[8] Keiji Shinzato and Satoshi Sekine. Unsupervised extraction of attributes and their values from product description. In *Proceedings of International Joint Conference on Natural Language Processing*, pp. 1339–1347, 2013.