

## 評価属性を考慮した評判情報の可視化

谷本 融 紀<sup>†1</sup> 太田 学<sup>†1</sup>

近年ブログやレビューサイトなどに書かれた評判情報を商品購入やマーケティングの参考にすることが多くなっている。しかし、このような情報の有用性は高いが、情報が膨大なため、その内容把握には手間がかかる。そこで本稿ではテキストを属性ごとに点数化し、一見して評価が分かるように可視化する手法を提案する。このとき、点数化はレビューテキストを用いて自動生成した評価表現辞書を用いることで行う。実験ではレビューサイトにおける商品の5段階の点数評価と、提案手法により属性ごとに点数化した評価の比較を行う。

### Visualizing Reputations Focusing on Evaluative Attributes

YUKI TANIMOTO<sup>†1</sup> and MANABU OHTA<sup>†1</sup>

Online reputation is useful for choosing products and marketing research. However, it is not easy to grasp reputational contents at a glance because it usually involves a vast amount of textual information such as blogs and word-of-mouth information. As described in this paper, we propose a mode of visualizing reputation with respect to various evaluative attributes using evaluative expression dictionaries. To create the dictionaries automatically, we use a review site with review texts and ratings. We compare the ratings of each attribute of a product in the review site and the scores of the corresponding attribute that our system calculates based on the review texts in order to evaluate how appropriate our proposal is.

### 1. はじめに

評判情報は商品、およびサービスの選択において重要な情報である。近年では、評判情報がブログやレビューとして投稿されており、多数の意見情報を Web で閲覧することが可能である。すなわち、商品についての多くの情報を一度に取得することができる。しかし、通常これらはテキストであるため、大量の内容を把握するのは困難である。また、そのために特定のサイトのみを閲覧するようになれば、情報に偏りが生じる恐れもある。一方、レビューサイトの中には商品を「デザイン」など特定の評価属性毎に点数化し、この点数に基づいてレビューをソートして提示してくれるものもあり、これは大変分かりやすい。そこで本研究では、特定の商品について、評判情報のテキストに基づいて評価を属性毎に点数化し、可視化する手法を提案する。そのためにまず評価表現辞書を作成する。評価表現は「良い」「悪い」などの評価値、「デザイン」「操作性」などの評価属性、「かなり」「とても」などの評価副詞の三つから構成されるものとし、これらを組み合わせて評価表現辞書を構成する。また評価値は評価属性に依存すると考え、評価値は評価属性とのペアで辞書に格納する。

本稿では2節で関連研究について述べ、3節で評価表現辞書の具体的な構成、4節で評価表現辞書を利用した可視化手法について説明する。5節では作成した評判情報の可視化システムについての評価、6節では本稿のまとめを行う。

### 2. 関連研究

#### 2.1 評価表現の抽出と評価表現辞書

評判の抽出に関して、あらかじめ人手で評価表現辞書を作成し、この辞書を用いて意見を抽出し、提示する立石らの研究<sup>1)</sup>がある。評価表現辞書には「書籍」、「コンピュータ」などのドメイン毎に、評価値と「肯定」か「否定」の情報を登録する。一方評判抽出は次のように行う。はじめに Web から評判情報の書かれたテキストを取得する。テキストの文中で、ユーザの入力した商品名と、評価表現辞書に登録された評価表現が十分近い距離にあるか判定する。そして評価表現として適正かどうか判定するルールを用い、条件を満たした文を評価文とみなす。この評価文を肯定か否定に分類し、ユーザに提示する。立石らの研究では、辞書を人手で作成するので高品質な辞書の作成が可能であるが、あらゆるドメインごとに大規模な辞書を作成するのは容易でない。また提示する情報は、ユーザの登録した商品がテキスト上で高評価か低評価かの評価極性だけであり、評価属性ごとの評価も抽出していない。Web 上のレビューを肯定、否定に分類し評価表現辞書を自動で作成する研究<sup>2)3)</sup>も行わ

<sup>†1</sup> 岡山大学大学院自然科学研究科  
Graduate School of Natural Science and Technology, Okayama University

れている。藤村ら<sup>3)</sup>が利用した価格.com<sup>\*1</sup>のレビューには商品毎に「良」、「悪」のようなタグをつけることができる。藤村らはこのタグを基に評価表現を肯定か否定かに分類し、出現頻度に基づくスコア付けを行っている。本研究でも価格.comを利用するが、肯定か否定かの極性ではなく、価格.comの評点から評価値の点数を算出する点が異なる。また藤村らは商品の評価属性を考慮に入れていない。

評判情報を評価属性毎に分類し、提示するシステムを瀬藤らが提案している<sup>4)</sup>。彼らの研究ではシステムに入力された評判情報から評価属性候補を取得し、関連する商品紹介ページを用いて評価属性を絞り込み、評判情報を評価属性ごとに分類して提示する。評価属性候補の取得のために瀬藤らは「評価属性の出現頻度」「共起ボタン」「文内共起」の3つの指標を用い、優劣を比較している。「共起ボタン」とは評価表現と評価属性の特定ボタンの共起頻度である。「文内共起」は評価表現と評価属性の同一文内での共起頻度である。本研究における評価表現辞書の作成では、評価属性の候補取得のため彼らの「共起ボタン」に類似した指標を利用するが、評価属性の絞り込みは評価属性の類似表現と類語を取得して行う点が異なる。

評価属性を抽出し、それをディレクトリ構造として上位属性、下位属性に分類する手法として関口らの手法がある<sup>5)</sup>。これは商用検索エンジンに入力されたクエリの遷移を抽出し、絞り込み検索に用いている語句を下位属性として登録する手法である。具体的には「ジャンル」「地名」などクラスを用意し、「ジャンル」に対しては「パチンコ」「宝くじ」などシードインスタンスを用意する。シードを用いて絞り込もうとしていた単語がクラスに対して偏りなく出現していれば、これを下位属性とする。

評価文を分類する手法として Minqing らの手法<sup>6)</sup>や新井らの手法<sup>7)</sup>がある。Minqing らは NLProcessor<sup>\*2</sup>により取得した形容詞を評価値とし、それと共起する特徴語を評価属性としている。そして WordNet<sup>8)</sup>を用いて評価値の関連語を取得し、肯定的か否定的かタグ付けをすることで評価表現辞書を構築している。また新井らは評判情報の意見文とその関連文を評価属性別に分類する手法を提案している。このとき「評価属性」「関連語」「関連度」を登録した関連語辞書を予め作成しておく。この辞書を用いて、レビュー内の各文に対して評価属性毎に関連語の関連度を計算し、一番高い文を意見文とする。

\*1 <http://kakaku.com/>

\*2 NLProcessor Text Analysis Toolkit. 2000. <http://www.infogistics.com/textanalysis.html>

## 2.2 従来提案との関連

我々は WebDB Forum 2009 で本研究と同じ目的に利用する評価表現辞書を提案した<sup>9)</sup>。この評価表現辞書は本研究と同様に、価格.comのレビューテキストを係り受け解析し、評価値として形容詞、形容動詞を取得し、それらに点数を付加することで作成される。この評価表現辞書に格納した評価属性と評価値の約8割は適切なものであり、この評価表現辞書を用いてレビューテキストからそのスコアを算出できた。しかし、従来提案では評価属性の扱いが不十分で総合評価のスコアしか算出できなかった。そこで本稿ではそれを改善して評価属性ごとのスコアを算出できるようにしている。

## 3. 評価表現辞書

### 3.1 評価表現辞書の構成

本研究では、以下の要素により構成されるものを評価文と定義する。

- 評価主体 … 評価を行った主体を指す表現（レビュー）
- 評価対象 … 製品など特定のクラスの実体を指す表現（商品の種類）
- 評価表現 … 評価対象に対する評価を表す表現で構成要素は以下の三つ
  - － 評価値 … 評価対象や評価属性に対する評価の値を示す表現（「良い」など）
  - － 評価属性 … 評価対象の属性（側面）を示す表現（「音質」など）
  - － 評価副詞 … 評価値を強めたり弱めたりする表現（「かなり」など）

これを踏まえ、評価値、評価属性、評価副詞に関する評価表現辞書を作成する。

評価値の一部は評価属性に依存して評価が変わる。例えば、評価値「高い」について、評価属性が「音質」の場合は一般的に良い評価となり、評価属性が「値段」の場合は悪い評価となる。よって評価属性と評価値を組み合わせた辞書を本稿では提案する。また、評価属性の中には互いに関連するものが存在する。例えば、評価属性「音質」や「音」「声」などは関連があると考えられる。よって評価属性の関連も考慮する必要である。そこで、図1に示すようなディレクトリ構造をもつ評価属性-評価値辞書を提案する。

図1において、評価対象は「MP3 プレーヤー」であり、その直下の属性として「音質」、「バッテリー」などが登録されている。本稿ではこれら評価対象直下の属性を主属性と呼ぶ。さらに「音質」の評価値には「良い」「悪い」などがある。また「音質」の関連属性として、「音」「高音質」「低音質」などがあり、さらに「音」の評価値には「十分」などがあり、関連属性には「声」などがある。なお評価値に付されている数値は  $[-1, 1]$  の点数であり  $-1$  に近いほど不評、 $1$  に近いほど好評を表す。

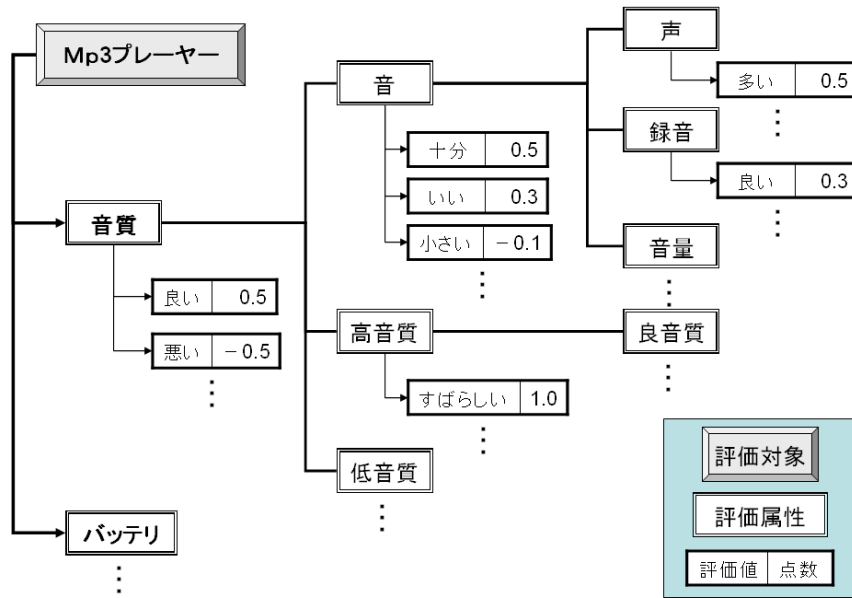


図1 ディレクトリ構造をもつ評価属性-評価値辞書の一例

Fig. 1 An example of the directory-based evaluative attribute-value dictionary.

本研究では評価副詞はこの評価値の点数を増減する表現とみなすので、評価副詞辞書には副詞とその重みを登録し、重みの範囲は [0, 2] とする。評価副詞は評価属性、評価値に比べ数は少なく、両者の依存関係も認められないため、評価副詞辞書は評価属性-評価値辞書とは別に手作業で作成する。

### 3.2 評価表現辞書の作成

評価表現辞書の作成には価格.comのレビューテキストとそれに付与されている評点を利用する。作成の流れを図2に示す。

図2の(1)の二つ組の作成手順は3.2.1項で(2)のディレクトリ構造を持つ辞書の構築方法については3.2.2項で(3)の点数計算方法は3.2.3項で説明する、

#### 3.2.1 二つ組の生成

図3は価格.comのあるiPodのレビューで、本研究で利用する「項目」と「評点」、「評価属性候補」と「評価値候補」を示している<sup>9)</sup>。このようなレビューを利用して評価属性と

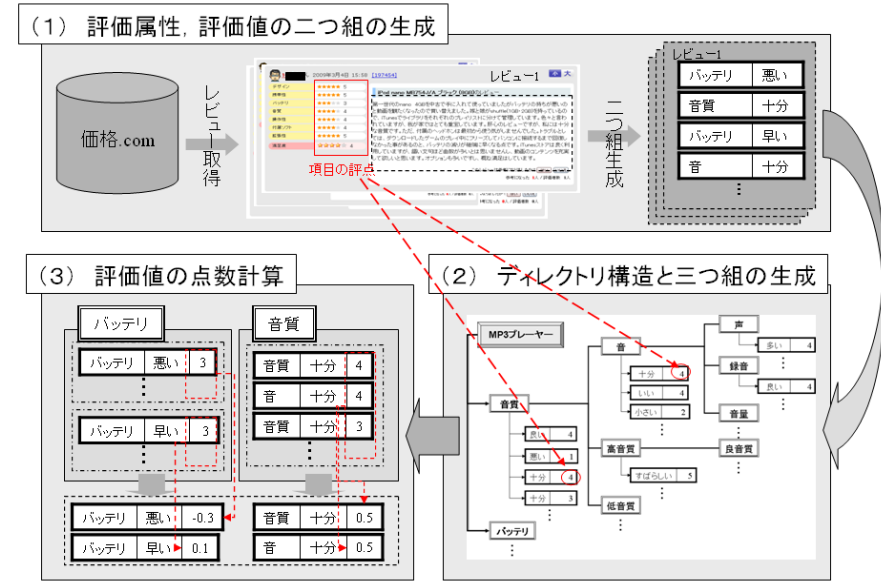


図2 評価属性-評価値辞書の作成

Fig. 2 Construction of the evaluative attribute-value dictionary.

評価値の二つ組、およびそれに点数を加えた三つ組を作成する。以下二つ組生成までの手順について説明する。

- (1) 価格.comの商品検索API<sup>\*1</sup>を利用してレビューテキストを取得
- (2) レビューテキストから評価値を取得
- (3) 係り受け解析を用いて評価値にかかる評価属性を取得し二つ組を生成

本稿における評価値にはCaboCha<sup>\*2</sup>で取得した形容詞と形容動詞を、評価属性には名詞を抽出し、評価属性、評価値の二つ組を生成する。抽出作業の際、CaboChaの情報をもとに記号、名詞および形容詞の連結処理を行う。連結処理は2語以上の形態素を一つに繋げる処理で、以下の条件を満たしたときに行う。

\*1 <http://apiblog.kakaku.com/KakakuItemSearchV1.0.html>

\*2 <http://chasen.org/taku/software/cabocho/>

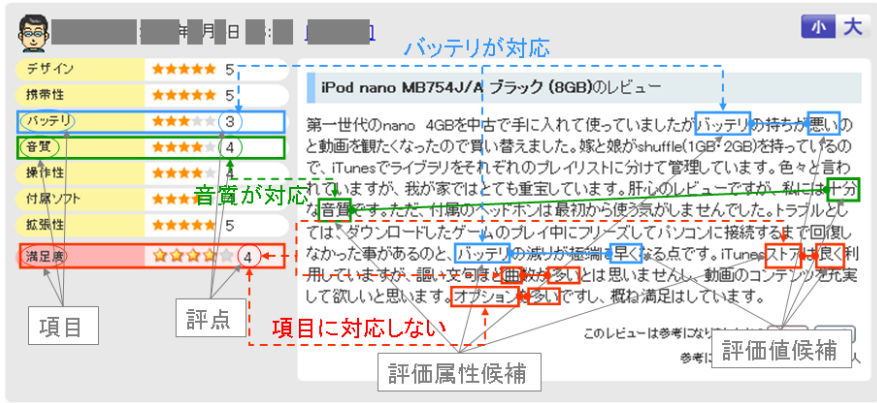


図 3 価格.com のレビュー  
Fig. 3 A review in kakaku.com.

- 記号の処理
  - 「記号」が連続
- 名詞の処理
  - 「名詞-一般」が連続
  - 「接頭詞-数接続」「名詞-数」の順で連続
  - 「接頭詞-名詞接続」「名詞-一般」の順で連続
- 形容詞の処理
  - 「動詞-自立」「形容詞-非自立」の順で連続
  - 「名詞-ナイ形容詞語幹」「助動詞 ナイ」の順で連続
  - 「名詞-ナイ形容詞語幹」「助詞-格助詞-一般」「形容詞-自立」の順で連続
  - 「形容詞-自立」「助動詞-ナイ」の順で連続
  - 「形容詞-自立」「助詞-係助詞」「助動詞 ナイ」の順で連続

この連結処理をすることで、例えば形容詞の処理で「扱う」「やすい」を「扱いやすい」にしたり、「難しい」「ない」を「難しくない」とすることができる。

3.2.2 ディレクトリ構造と三つ組の生成

二つ組の評価属性と図 3 にある項目を対応付け、図 1 の評価表現辞書のディレクトリ構造を構築する。以下にその構築方法を説明する。

- (1) 項目を主属性ノードとして新たにノードを生成し「mp3 プレーヤー」の子属性とする
- (2) 項目と評価属性の文字列が一致する二つ組を主属性ノードに対応付ける
- (3) 各主属性ノードを起点として以下の手順で子属性ノードを生成する
  - (a) 以下のいずれかを満たす二つ組の評価属性を子属性に割り当てる
    - (i) 現在のノードの文字列が評価属性の文字列に含まれる
    - (ii) 評価属性の文字列が現ノードの文字列に含まれる
    - (iii) 現ノードの類語と評価属性が一致
  - (b) (a) で割り当てた子属性のノードがない場合は新しくノードを生成し、そのノードに二つ組を対応付ける
  - (c) (a) で割り当てた子属性のノードがある場合はそのノードに二つ組を対応付ける
  - (d) 生成された各子属性ノードを起点として (a) ~ (c) を実行する

ここで類語とは、類語辞典 weblio\*<sup>1</sup>により取得した表現である。また終了条件について、現在は主属性ノードに対する孫属性ノードまで取得した段階で構築を終了している。

この構築手法を用い、評価属性に対するディレクトリ構造を構築したあと、評価属性の先祖の主属性ノードに対応する図 3 の項目の評点を二つ組に付加し、三つ組を生成する。また対応付けを行われなかった二つ組は図 3 の項目の「満足度」の評点を付加し、これらを独自属性<sup>9)</sup>の三つ組とする。この独自属性とは主属性以外で出現頻度が高い評価属性である。独自属性についても主属性と同様にして子孫属性を対応付ける。

3.2.3 評価値の点数計算

評価値の点数は主属性  $j$  ごとに計算され、各主属性の全子孫ノードに割り当てられた三つ組に対して行われる。

式 (1) で評価値  $i$  の平均点数  $average_i$  を計算する。ここで  $N$  は属性が主属性  $j$  自身、または主属性  $j$  を先祖に持つ属性で、評価値  $i$  を持つ三つ組の総数、 $point_{i,k}$  は同様の属性で、評価値  $i$  を持つ  $k$  番目の三つ組における評点であり、 $point_{i,k} \in \{1, 2, 3, 4, 5\}$  である。

$$average_i = \frac{1}{N} \sum_{k=1}^N point_{i,k} \tag{1}$$

この  $average_i$  を式 (2) を用いて  $[0, 1]$  に正規化する。式 (3) (4) で  $n$  は主属性  $j$  を先祖とする、属性にの評価値の異なり数である。

\*1 <http://thesaurus.weblio.jp/>

$$normal_i = \begin{cases} \frac{average_i - m}{M - m} & (if M > m) \\ \frac{average_i - 1}{4} & (if M = m) \end{cases} \quad (2)$$

$$M = \max_{i=1, \dots, n} average_i \quad (3)$$

$$m = \min_{i=1, \dots, n} average_i \quad (4)$$

そして式 (5) により評価値の点数  $value_i$  を定義する .

$$value_i = \begin{cases} shift_i & (if shift_i > 0) \\ shift_i / 5 & (otherwise) \end{cases} \quad (5)$$

ただし  $shift_i$  については以下の式で計算される .

$$shift_i = 6 \cdot normal_i - 5 \quad (6)$$

4. 評価表現の可視化

レビューテキストの可視化の概略を図 4 に示す . 可視化に用いるレビューの取得には価格.com 商品検索 API や Yahoo!JAPAN のウェブ検索 WebAPI\*1を用いる . 取得するレビューの件数はユーザが指定し , 図 4 においては  $n$  件としている . そして各テキストについて属性ごとの点数を計算し , ランキングする .

実装中のプロトタイプシステムによる評判情報可視化の流れは図 4 と以下で説明する .

- (1) テキストの係り受け解析により評価値 , 評価属性 , 評価副詞の組を取得
- (2) 評価表現辞書を用いて主属性に対するスコアを計算
- (3) 「主属性 , 評価属性 , スコア」の三つ組を生成
- (4) 各主属性のスコアを三つ組を用いて計算
- (5) レビューの総合評価を (4) で取得した全スコアの平均として算出
- (6) 各主属性のスコア , または総合評価ごとにレビューをランキングして表示

評価値 , 評価属性の取得は 3.2.1 項と同様に行い , 連結処理も同時に行う . 評価副詞は評価値に係る副詞を探索することで取得する . ここで評価属性が主属性でない場合 , 主属性の取得は評価属性-評価値辞書における先祖属性を探索することで取得する . そして評価属性-

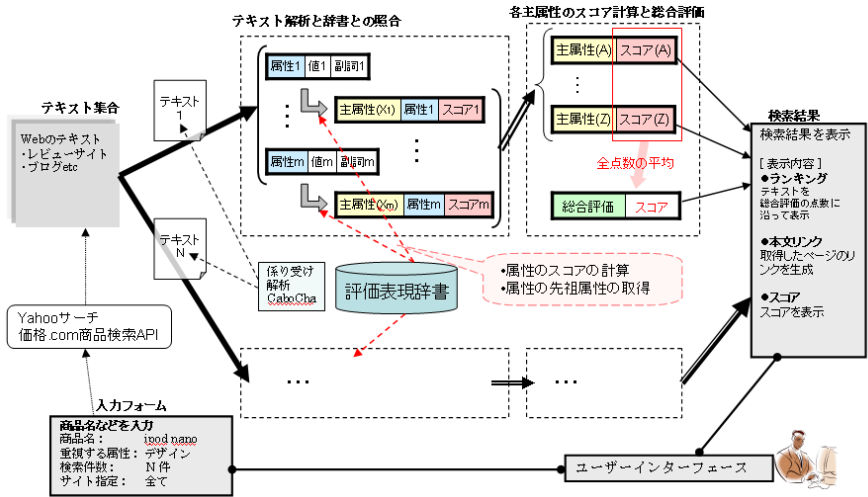


図 4 レビューテキストの可視化  
Fig. 4 Visualization of review texts.

評価値辞書と評価副詞辞書の両方を用いて各評価属性のスコアを計算し , 「主属性 , 評価属性 , スコア」の三つ組を取得する . このスコアの値は評価属性-評価値辞書の点数と評価副詞辞書の重みの積とする . 副詞が存在しない場合は評価属性-評価値辞書の点数をそのままスコアとする . そして三つ組を用いて各主属性の点数の平均を計算し , その主属性のスコアとする . さらにこれらの平均を計算することで総合評価のスコアとする .

これらにより取得した全スコアの平均をレビューテキストの総合評価とし , 可視化に利用する . 図 5 はレビューテキストの可視化の例である . このプロトタイプシステムでは , 商品の種類を示すドメインと検索対象とする商品名 , 検索件数 , サイトの指定を行い , レビューを検索する . ここで重視する属性にチェックを入れると , チェックを入れた主属性とその子孫の属性の重みを大きくことが出来る . そしてテキストを解析し , ランキングした結果を画面右側に出力する . 図 5 の出力結果の左側には属性ごとの点数 , 右側にはもとのレビューテキストを表示している . 属性は上から順に主属性 , 独自属性 , 全主属性の点数を平均した総合評価となっている . また独自属性とは主属性に分類されなかった評価属性である .

\*1 <http://developer.yahoo.co.jp/webapi/search/websearch/v1/websearch.html>

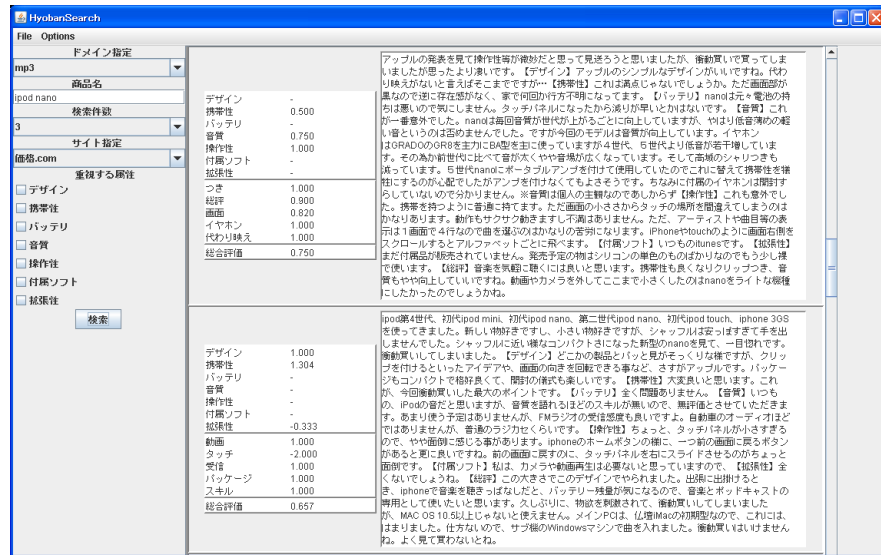


図 5 プロトタイプシステムの実行画面  
Fig. 5 Screenshot of the prototype system.

表 1 実験で使用したレビューテキスト

Table 1 Review texts used in the experiments.

キーワード	カテゴリ	取得したページ数	解析したテキスト数
mp3	All	1000	1839

## 5. 評価実験

評価実験では作成した評価属性-評価値辞書の評価と、可視化したレビューテキストのスコアを評価する。辞書作成のためのコーパスとして用いたレビューテキストは2010年8月に価格.comの商品検索APIを利用して取得した。このAPIに入力したキーワード、カテゴリ、取得したページおよびレビューテキストの数を表1に載せる。また可視化の実験でも、表1に示したレビューテキストを可視化の対象とした。

### 5.1 評価属性-評価値辞書の評価

ディレクトリ構造を持たせた評価属性-評価値辞書の評価を行うため、各主属性の子孫属

表 2 各主属性の子孫属性の正解率

Table 2 Accuracy of descendant attributes of each main attribute.

主属性	デザイン	携帯性	バッテリー	音質	操作性	付属ソフト	拡張性	全主属性
子孫属性の数	31	3	17	83	11	11	2	158
正解率 (rigid)	0.452	1.000	0.529	0.530	0.364	0.455	1.000	0.513
正解率 (relaxed)	0.968	1.000	0.647	0.904	1.000	0.455	1.000	0.867

表 3 評価属性の分類例

Table 3 A classification of evaluation attributes.

主属性	デザイン	携帯性	バッテリー	音質	操作性	付属ソフト	拡張性
子孫属性 (rigid)	形 外観	携帯 持ち運び	バッテリー 電池	声 低音	操作 扱い	付属 ソフト	拡張 拡大
子孫属性 (relaxed)	変形 姿勢	-	寿命 容量	程度 基本音質	運用 工夫	-	-
子孫属性ではない	言い方 -	-	ノイズ ホワイト	思い ただ	-	あとソフト ソフト事態	-

性の評価と、各評価属性に関連付けられた評価値の評価を行う。5.1.1項ではこの子孫属性の適切性、5.1.2項では評価属性と評価値、点数の三つ組の適切性を評価する。

#### 5.1.1 主属性の子孫属性の適切性の評価

ここでは主属性の子孫属性について評価する。表2は主属性に対する子孫属性の正解率である。「子孫属性の数」は表1のレビューテキストを解析することで取得した子孫属性の総数である。「正解率 (rigid)」は主属性に対し、評価属性が明らかに子孫の属性として適切であるものだけを正解とした正解率である。「正解率 (relaxed)」は正解判定を緩め、文脈によっては子孫属性になる可能性があるものも正解とした正解率である。表3は評価属性の分類例を示す。

「バッテリー」には「ホワイトノイズゆバッテリー」というノイズが子属性として含まれたため、結果としてその子属性が全て誤りとなっていた。これに対処するには、CaboChaの解析において、名詞の連結の方法やCaboChaの用いている辞書の内容を再考する必要がある。また評価属性には本来子属性として入るべきものが含まれていないこともあった。例えば「付属ソフト」には、Sony社製のwalkmanの付属ソフトである「Sonic Stage」のような具体的な付属ソフト名が子孫属性として含まれてもよい。そのためには、文章内での主属性との共起頻度による子属性の取得が必要であると考えている。

表 4 「評価属性, 評価値, 点数」の三つ組の正解率

Table 4 Accuracy of extracted triplets of evaluative attribute, value and its score.

主属性	デザイン	携帯性	バッテリー	音質	操作性	付属ソフト	拡張性	合計
評価対象の組数	184	98	89	465	234	83	95	1248
正解率 (rigid)	0.647	0.561	0.528	0.563	0.620	0.723	0.663	0.602
正解率 (relaxed)	0.810	0.837	0.697	0.770	0.842	0.819	0.884	0.795

### 5.1.2 評価属性, 評価値, 点数の組の評価

ここでは評価属性に結び付けられた評価値の点数が妥当かどうかという観点から、「評価属性, 評価値, 点数」の三つ組を評価する。評価対象は 5.1.1 項において, rigid 判定で正解となる評価属性のみとする。

表 4 は取得した三つ組の正解率である。「評価対象の組数」とは取得した三つ組の評価属性が明らかに主属性の子孫属性であるものの数である。「正解率 (rigid)」はその属性と組み合わせたととき, 一般的に良いと考えられる評価値の点数が正, 悪いと考えられる評価値の点数が負であれば正解とした正解率である。「正解率 (relaxed)」は評価者によっては正解となるものも正解とした正解率である。例として「音質, いい, 0.49」という三つ組を考える。ここで評価属性「音質」と評価値「いい」を組み合わせると, あきらかにこれらは良い評価を表している。そして点数は正の値である。よってこの三つ組は rigid 判定で正解とする。しかし「デザイン, ダサイ, 1.0」の場合, 「デザイン」と「ダサイ」は明らかに悪い評価であるのに点数が正の値であるのでこれは誤りである。一方で「音, 近い, -0.4」などこれらだけでは判断できないものは relaxed 判定で正解としている。

今回の実験では relaxed 判定で正解とした「評価属性, 評価値」の組の中には, 評価者によっては rigid 判定で正解となるものも存在した。例えば「高音, 強い」は点数が正なら, 高音が強い方が好みである人にとっては rigid 判定で正解となるが, 弱い方が好みである人にとっては誤りであると考えられる。relaxed 判定の正解にはこのような例が含まれる。

### 5.2 可視化したレビューテキストのスコアの評価

提案した評価表現辞書と, それを利用した提案システムによるレビューテキストの可視化の妥当性を確認する。レビューテキストについて提案システムが属性ごとに計算したスコアと, レビューテキストに付随している項目の評点の比較を行った。これらの相関を図 6~図 12 に示す。これらの図中の一つの点が, 一つの「主属性, 属性, スコア」の三つ組のスコアと三つ組が得られたレビューの評点に対応している。また横軸は図 3 の価格.com におけるレビューの評点, 縦軸は提案するシステムにより計算したスコアであり, 直線は回帰

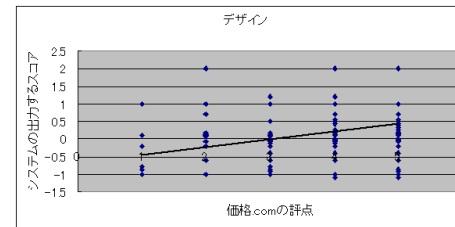


図 6 「デザイン」における評点とスコアの相関

Fig. 6 Correlation between the original review scores and the calculated scores w.r.t. “design”.

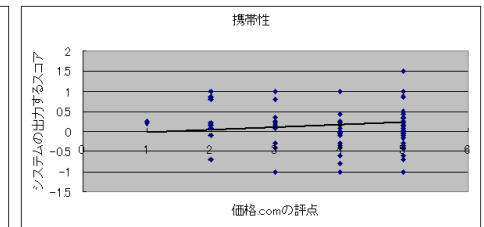


図 7 「携帯性」における評点とスコアの相関

Fig. 7 Correlation between the original review scores and the calculated scores w.r.t. “portability”.

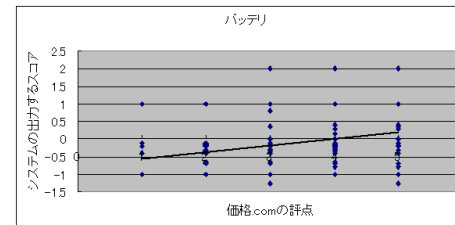


図 8 「バッテリー」における評点とスコアの相関

Fig. 8 Correlation between the original review scores and the calculated scores w.r.t. “battery”.

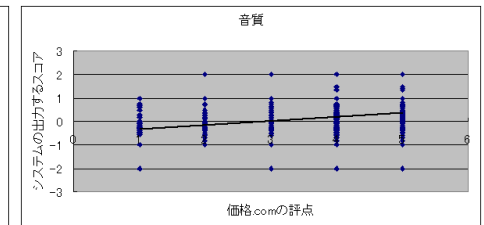


図 9 「音質」における評点とスコアの相関

Fig. 9 Correlation between the original review scores and the calculated scores w.r.t. “sound quality”.

直線である。さらに取得した三つ組の総数と相関係数を表 5 にまとめる。「対象数」は評価対象となった三つ組の総数である。これらの相関係数が高いほど, 本稿の提案は妥当であると考えられる。

表 5 より, 評価属性により相関に大きく差があるのが分かる。「バッテリー」や「操作性」, 「付属ソフト」や「拡張性」は相関係数が 0.3 以上であり, 他の評価属性より比較的相関が大きいといえる。また「バッテリー」「音質」にもそれらにつぐの正の相関があった。一方で「携帯性」の評価属性は相関があまりないという結果となっている。この理由については, 係り受け解析の失敗や, 連結処理が不適切なものが多いことが原因の一部にあると考えている。

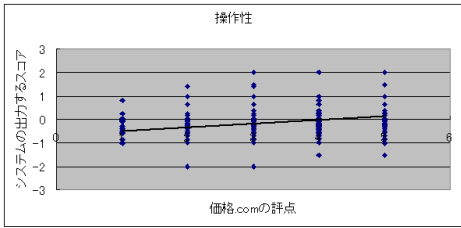


図 10 「操作性」における評点とスコアの相関

Fig.10 Correlation between the original review scores and the calculated scores w.r.t. "user-friendliness".

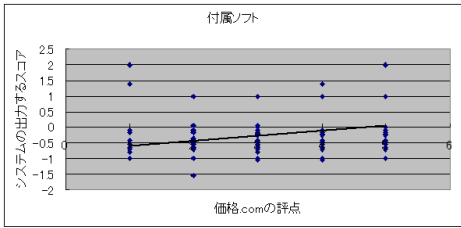


図 11 「付属ソフト」における評点とスコアの相関

Fig.11 Correlation between the original review scores and the calculated scores w.r.t. "attachment".

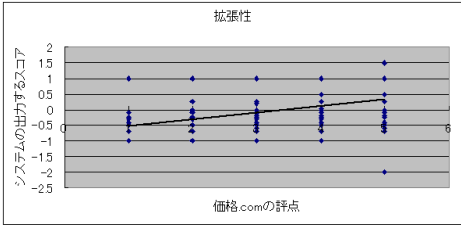


図 12 「拡張性」における評点とスコアの相関

Fig.12 Correlation between the original review scores and the calculated scores w.r.t. "extensibility".

表 5 主属性別の相関係数

Table 5 Correlation coefficients w.r.t.main attributes.

評価属性	デザイン	携帯性	バッテリー	音質	操作性	付属ソフト	拡張性
対象数	1106	581	295	2810	1052	281	188
相関係数	0.264	0.102	0.329	0.284	0.305	0.319	0.436

6. ま と め

本稿では、価格.com のレビューテキストを CaboCha を用いて係り受け解析して、評価値と評価属性、評価属性に対応する項目の評点を取得し、これらを用いて評価属性の階層構

造を持つ評価表現辞書を作成する方法を提案した。さらにこの評価表現辞書を用いて実際にレビューテキストを解析し、評価属性ごとのスコアを計算し、スコアを基にレビューテキストの可視化を行った。

実験では、価格.com の項目ごとの評点と提案システムで計算した主属性ごとのスコアの相関図を示し、相関係数を計算した。その結果、評価属性の一部については相関係数が 0.3 以上の正の相関が認められた。一方で、相関があまりない評価属性も含まれていた。そのため、今後辞書の構築方法や、点数化の方法についてさらに検討したい。また Amazon やその他のブログなど、価格.com 以外のレビューテキストについても提案した評価表現辞書を用いて可視化し、その評価を行う予定である。また結果の表示方法については、検索したレビューテキスト全てを総合したスコアの表示等を検討している。

参 考 文 献

- 1) 立石健二, 石黒義英, 福島俊一: インターネットからの評判情報検索, 情報処理学会研究報告 NL144-11, pp.75-82 (2001).
- 2) Dave, K., Lawrence, S. and M.Pennock, D.: Mining the Peanut Gallery: Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews, *International World Wide Web Conference(WWW2003)*, pp.519-528 (2003).
- 3) 藤村 滋, 豊田正史, 喜連川優: 電子掲示板からの評価表現および評判情報の抽出, *JSAI2004 3F1-03* (2004).
- 4) 瀬藤 亮, 佐藤哲司: 商品説明ページを用いた評価視点別評判情報提示システム, *DEIM2009 C6-5* (2009).
- 5) 関口裕一郎, 田中智博, 内山 匡, 藤村 滋, 望月崇由, 鈴木智也: 検索クエリログのセッション情報を利用した属性語句抽出, *DEIM2010 A2-3* (2010).
- 6) Hu, M. and Liu, B.: Mining and summarizing customer reviews, *In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.168-177 (2004).
- 7) 新井智也, 佐藤哲司: 評価視点別の言及度を用いた意見文の分類手法の提案, *DEIM2010 A2-2* (2010).
- 8) Miller1, G.A., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K.J.: Introduction to WordNet: An On-line Lexical Database, *International Journal of Lexicography*, pp.235-312.
- 9) 谷本融紀, 太田 学: 評価表現辞書を用いた評判情報の可視化, Web とデータベースに関するフォーラム (WebDB Forum 2009)1B-1 (2009).