2007－NL－177 (6)
2007／1／26

A Proposal for Extracting and Classifying Neutral Expressions from the Weblog

Guangwei Wang and Kenji Araki
Graduate School of Information Science and Technology, Hokkaido University
{wgw,araki}@media.eng.hokudai.ac.jp

Recently, the acquisition and analysis of opinions from the web has attracted more and more attention. Most research only focuses on how to classify the opinions into bipolar orientation i.e. either positive or negative opinion. There are in fact many neutral expressions (e.g., sentences with greetings, wishes or requests) in the opinion contents, especially in the Weblog. These expressions make the precision of opinion mining lower. This paper proposes an original idea to extract and classify the neutral expressions from Weblog, at both sentence and phrase level. Experiments show that the precision is improved 10% by introducing a process of neutral expression to opinion mining.

Weblogからの中立表現の抽出と分類について

王　広為　　荒木　健治
北海道大学大学院情報科学研究科
{wgw,araki}@media.eng.hokudai.ac.jp

近年，ウェブから意見を抽出し，分析を行う処理にますます注目が集まっている．多くの研究は肯定と否定カテゴリーに意見を分類する方法だけに研究の焦点があてられている．しかし，Weblogに記述されている意見情報の中には，意見が含まれていない中立表現（例：挨拶文，請求文，質問文など）が多数存在している．これらの表現は意見マイニングの精度を低下させている．そこで，本稿では文とフレーズ両方のレベルでWeblogから中立表現を抽出し、分類するための新しい手法を提案する．実験の結果，中立表現を処理するにより意見分類の精度が10%改善された．

1 Introduction

Nowadays, there are a lot of personal opinions about different products and services on the Web, e.g. customer reviews of products, forums, discussion groups and blogs. Here, we use the term Weblog for these sites. The enormous amount of this kind of information is beneficial for both product companies and users who are planning to purchase and use the products. However, these opinion data are actual and also mainly presented in textual form. It is very hard and costly to collect and analyze them manually. As a result, the problem of "opinion mining" has seen increasing attention over the last four years from [1,2] and many others.

Most related work has studied how to classify opinions into bipolar orientation i.e. either positive or negative opinions. These research results make it possible for users to reduce the time spent on reading threads of text and focusing more on analyzing summarized information. There are however also many neutral expressions (e.g., sentence with greetings, wishes or requests) in the opinion contents, especially in the Weblog. In our former study about opinion mining, we found that neutral expressions are one of the main causes that negatively affect the precision of opinion classification. Thus, dealing with neutral expressions is an important subtask in opinion mining.

This paper proposes a novel method for extracting and classifying neutral expressions from the Weblog, at both sentence and phrase level. At the sentence level, we present a method for detecting neutral sentences using a similarity algorithm, and determine an optimal similarity threshold. Since some neutral phrases are very difficult to detect at the sentence level, we propose a two classifications for them at the phrase level after structure parsing of Japanese language. These two classifications are feature classification using a supervised approach (Naive Bayes) and P/N (positive/negative) classification using an improved unsupervised approach (SO-PMI: Semantic Orientation Using Pointwise Mutual Information).

We review related work in Section **2**, and then explain a definition of neutral expressions in Section **3**. Section **4** presents the proposed method for identifying neutral sentences. Section **5** describes how to perform feature classification and P/N classification in detail. In section **6**, we evaluate our method and discuss the experimental results. Finally, section **7** gives concluding remarks.

2 Related Work

Much of the earlier research in automated opinion detection has been performed by Wiebe and colleagues [3,4,5], who proposed methods for discriminating between subjective and objective text at the document, sentence and phrase levels. Wiebe et al [5] report on document-level subjectivity classification, using a k-nearest neighbor algorithm based on the total count of subjective words and phrases within each document.

Turney [1] worked on opinion classification using a few semantically oriented words (namely, "excellent" and "poor") to label other phrases co-occurring with them as positive or negative. He then used the average semantic orientation of these phrases to automatically separate positive and negative movie and product reviews, with accuracy of 66-84%.

Yu et al. [6] presented several models for an opinion question answering system: distinguishing between opinions and facts at the document level, and classifying opinion sentences as positive, negative, or neutral. Pant et al [7] adopted a more direct approach, using supervised machine learning with words and n-grams as features to predict orientation at the document level with up to 83% precision.

3 What are Neutral Expressions?

Based on INUI and OUKUMURA's survey [8], we define Neutral Expressions at both the sentence and phrase level as shown in **Figure 1** (the two dashed ellipses).

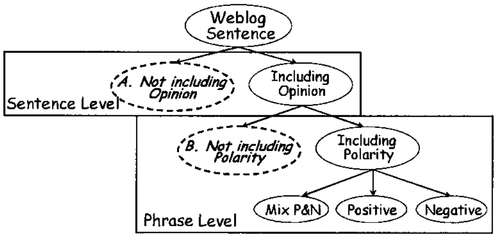

Figure 1: Definition of Neutral Expressions

First, at the sentence level (**A** dashed ellipse in **Figure 1**), a neutral expression means a sentence that dose not include any opinions. This type of sentence mainly includes greetings, wishes, appreciation and similar things (**Example [A]**). Especially, there are many honorific expressions used in Japanese.

For example [A]:

"はじめまして *(Nice to meet you)*"

"誰か試された方がいるなら教えてください *(If there is anyone who has used it, please tell me)*"

"アドバイスありがとうございました *(Thank you for advice)*"

"送料・代引き手数料無料 *(The postage / COD mail fee is free of charge)*"

As a result of investigating five types of IC Dictionaries and five types of MP3 players on the Weblog, we found 218 such neutral sentences manually among a total of 2,303 sentences, about a 9% ratio. In Section **4**, we propose a method for identifying similar neutral sentences by using several representative "seed sentences" and a similarity algorithm.

At the phrase level (**B** dashed ellipse in **Figure 1**), neutral expressions are the portions of opinion sentences that dose not include polarity i.e. either positive or negative opinion.

For example [B]:

"この欄のレビューを見て購入したが、とても使い難いのでがっかりです.

(I purchased it since seeing a review of this column, but I was very disappointed because it was hard to use.)"

The underlined part of this sentence (**Example [B]**) only depicts some facts, and does not include any opinions. However, it is very difficult to deal with this case at the sentence level. Therefore, we decided to extract these kinds of neutral expressions at the phrase level after the structure analysis.

4 Identifying Neutral Sentences

4.1 Similarity Algorithm

We used a simple *O(ND)* time and space algorithm [9] to calculate the SED (Shortest Edit Distance) of two strings. Let String $A = a1,a2... aN$ and String $B = b1,b2...bM$ be sequences of length N and M respectively. *N* of *O(ND)* is the sum of the lengths of String *A* and *B*. *D* of *O(ND)* is the size of the minimum edit script for String *A* and *B*. The algorithm for calculating the SED of String *A* and String *B* is shown in detail in Figure **2**.

```
Constant MAX ∈ [0,M+N]
Var V: Array [−MAX .. MAX] of Integer
V[1] ← 0
For D ← 0 to MAX Do
  For k ← −D to D in steps of 2 Do
    If k = −D or k ≠ D and V[k - 1] < V[k + 1] Then
      x ← V[k + 1]
    Else
      x ← V[k − 1]+1
    y ← x − k
    While x < N and y < M and a x + 1 = by + 1 Do
     (x,y) ← (x+1,y+1)
    V[k] ← x
    If x ≧ N and y ≧ M Then
      Length of an SES is D
      Stop
Length of an SES is greater than MAX
```

Figure 2: O(ND) algorithm

The Similarity Algorithm gives a similarity value for two strings between "0" and "1". "0" means that String *A* and *B* are entirely different strings. "1" means that String *A* and *B* are the same string.

4.2 Choosing Similarity Threshold

We propose a distinction method to identify similar neutral sentences by using several representative "seed sentences" (**Example [C]**) and the similarity algorithm.

For example [C]:
[Seed Sentence]: 教えてください *(Please tell me)*
[Similar Sentences]: 教えてね. 教えて下さい. 教えてほしい. *(Tell me. Please tell me. I want you to tell me.)*

The identification method calculates the SV (similarity value) between a sentence and a "seed sentence". If the SV exceeds a threshold, the sentence is considered to be a neutral sentence. Therefore, a threshold (between "0" and "1") should be determined.

We chose 20 representative "seed sentences" of the type presented in Section **3**. As a test corpus, we selected the 150 neutral sentences which were most similar to the "seed sentences" from the Weblog manually. Our identification method calculated every SV between the neutral sentences and the "seed sentences", and then achieved the recall, precision and F-measure curves shown in Figure **3**. If a sentence in test corpus is taken as appropriate "seed sentence", it would be one of sentences correctly extracted as neutral sentence.

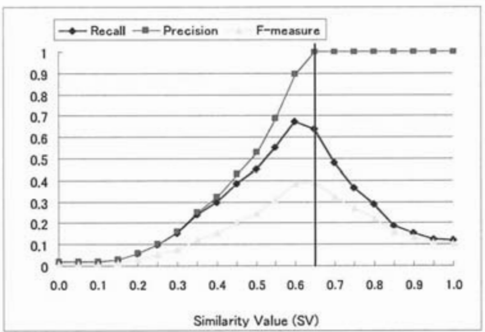

Figure 3: Experiment of Similarity Threshold

It is obvious that the identification method achieves the best performance and the highest F-measure when the threshold is "0.65". Hence we decided to set "0.65" as the similarity threshold. Using this similarity threshold, we separated the neutral sentences from real opinion contents in the Weblog and gained positive results, which are presented in Section **6.2**.

5 Classifying Neutral Phrases

There are many of neutral expressions (such as statements of facts) in the opinion sentences. It is hard to extract them at the sentence level. Therefore, we used a well known structure analyzer for Japanese called Cabocha [13] to analyze the Weblog content, and then classified neutral expressions and opinions at the phrase level.

Based on the syntactic analysis of Japanese and the results in related works [11, 12], we separated opinion phrases into attribute expressions (nouns) and reputation expressions (adjectives or verbs) using the POS (part-of-speech) of the words. Then two kinds of classification, feature classification and P/N classification are performed. The purpose of feature classification is to know what an opinion evaluates based on the attribute of the opinion. P/N (positive or negative) classification aims to understand what the opinion expresses about the feature. In the next section, we present how to process neutral expressions at the phrase level in these two classifications.

5.1 Feature Classification

Feature classification aims to classify attribute expressions into feature categories (e.g. price, design, function, software, battery and neutral) so we can know what an opinion evaluates.

Through analyzing various product review information on the Weblog, we discovered the following facts. Firstly, more than 90 percent of attribute expressions are nouns. Secondly, different categories of products have similar attributes (such as "cost", "size" etc.) and similar neutral expressions (such as "sale date", "delivery" etc.). Thirdly, the number of these kinds of expressions is limited. A sample of attribute expressions is shown in Table **1**.

Table 1 : Sample of product attributes expression

Feature	Attribute expression
Price	値段(price), 価格(cost), 料金(charge), 安値(low price)…
Design	サイズ(size), デザイン(design), 幅(width), キー(key) …
Function	再生(play), 辞書(dictionary), 発音(pronunciation), 機能(function) …
Software	ソフト(soft), 付属ソフト(attached software), パソコン(PC) …
Battery	充電時間(charge time), 電池交換(battery exchange), バッテリ(battery) …
Neutral	発売日(sale date), 配達(delivery), サイト(site), 場合(case) …

This investigation indicated that it would be easy to create a training corpus. Creating a classifier would not take much time and reasonably accurate classification results could be expected. Concretely, the training time was about 3 second; Precision was about 65% in our preliminary experiment. Therefore, we decide to use a supervised approach (Naïve Bayes) to perform this classification. The training corpus and experimental results are shown in Section **6.3**.

5.2 P/N Classification

P/N Classification here means classifying phrases into positive or negative meanings using SO (Semantic Orientation) values for understanding what the opinion expresses.

We modified the SO-PMI approach [1] as formula (1), (2) and (3) below to adapt it to Japanese semantic orientation. The SO-PMI approach can be used to estimate the semantic orientation of a phrase by measuring co-occurrences with semantically oriented words or phrases.

$$SO(phrase) = \log_2[A] + threshold \quad (1)$$

$$A = \frac{hits(phrase\,\mathrm{AND}\,"good\_basic") * hits("bad\_basic")}{hits(phrase\,\mathrm{AND}\,"bad\_basic") * hits("good\_basic")} \quad (2)$$

$$Threshold = \log_2\left[\frac{hits(bad\_basic)}{hits(good\_basic)}\right] * (-0.8) \quad (3)$$

"good_basic" = "いい | 好き | 良い | 魅力 | 大好き | 欲しい | 楽しい | 嬉しい | 面白い | 素敵 | 良かった | 良く | おもしろい | 満足 | 素晴しい | うれしい | よい | すばらしい"
(いい, 良く, 良かった, よい, 良い: good / 好き: like / 魅力: charm / 大好き: favorite / 欲しい: want / 楽しい: delightful / 嬉しい, うれしい: happy / 面白い, おもしろい: interesting / 素敵: lovely / 満足: satisfaction / 素晴しい, すばらしい: wonderful)

"bad_basic" = "あまり | あんまり | 悪い | 嫌 | 不安 | 怖い | 不良 | 嫌い | 苦手 | 悪く | まずい | 悲しい | 危険 | だめ | 辛い | 不満 | 不快 | 最悪"
(あまり, あんまり: not good / 悪い, 悪く, 不良, まずい: bad / 難しい, 苦手: hard / 嫌, 嫌い: dislike / 不安: worry / 怖い: fearful / 不満: dissatisfaction / 危険: risk / だめ: useless / 悲しい: sad / 辛い: painful / 不快: dissatisfaction / 最悪: worst)

The reference words "good_basic" and "bad_basic" are used, and then SO is positive when a phrase is more strongly associated with "good_basic" and negative when a phrase is more strongly associated with "bad_basic". In a former study [10], we determined good sets of reference words ("good_basic" and "bad_basic"). These are basic semantically oriented words for positive and negative opinions in Japanese, and all have a comparatively steady occurrence ratio as the most common opinion words in the Weblog.

Given the SO value, how can we determine the orientation (**P**: positive or **N**: negative or **U**: neutral) of a phrase? We use the following **Threshold Rules** to perform this classification. The threshold values for ***ta***, ***tb*** and ***tc*** are obtained from a small, hand-labeled subset of phrases. The phrase will be **U** if it is strongly or faintly associated with both "good_basic" and "bad_basic". This means that this phrase has an ambiguous connection with both "good_basic" and "bad_basic". Therefore, it is considered a neutral phrase.

Threshold Rules:

*If (hits(phrase AND good_basic) > **ta** AND hits(phrase AND bad_basic) > **ta**) phrase is **U***
*elsif (hits(phrase AND good_basic) < **tb** AND hits(phrase AND bad_basic) < **tb**) phrase is **U***
*elsif (| hits(phrase AND good_basic) – hits(phrase AND bad_basic) | < **tc**) phrase is **U***
*elsif (SO(phrase) = 0) phrase is **U***
*elsif (SO(phrase) > 0) phrase is **P***
*elsif (SO(phrase) < 0) phrase is **N***

6 Experimental Performance Evaluation

6.1 Test Data

Table **2** shows the five types of IC Dictionaries and five types of MP3 players that were used in the experiments. These products were popular types from the ranking of kakaku.com on 2006/10/19.[1] We extracted the opinion information of these products from the Japanese Weblog by the following steps [10]:

[Information Search]: Using the Google Search Engine to collect all the relevant information on the Internet. The search range is restricted by URL type (e.g. blog, bbs, and review)

[Weblog Content Extraction]: Extracting the Weblog content from each of the collected pages according to the URL type.

Table 2: Product List for the test

Product Category	Product Type
IC Dictionary	T1: CASIO XD-ST6200
	T2: CASIO XD-GT6800
	T3: SHARP Papyrus PW-V8100
	T4: SHARP Papyrus PW-A8410
	T5: SEIKO SR-E10000
MP3 Player	T6: SONY NW-E005
	T7: SONY NW-A1000
	T8: APPLE iPod MA002J/A
	T9: APPLE iPod nano MA005J/A
	T10: PANASONIC D-Snap Audio SV-SD750V-A

6.2 Sentence Level Experiment

As mentioned above, Weblog content about IC Dictionaries and MP3 players was extracted. Then all the sentences were separated by end of sentence punctuation mark (such as "?", ".", "。", "!" etc). This resulted in 2,303 Weblog sentences. We found 218 neutral sentences manually among these sentences as illustrated in Section **3**. We used 20 seed sentences and a similarity threshold as was demonstrated in Section **4** to do the identification of neutral sentences. Table **3** shows the experimental results.

Table 3: Sentence Level Experiment

	Value	Notes
Seed Sentence	20	Representative Neutral Sentence
Weblog Sentence	2303	Neutral Sentence: 218
Similarity Threshold	0.65	Demonstrated in Section 4
Precision	**90%**	161/178
Recall	**74%**	161/218

Using the opinion information from the Weblog, we achieved the 90% precision and 74% recall. The precision was similar and the recall was better than in the preliminary experiments which we used to decide the similarity threshold. Thus, it is

[1] Kakaku.com: Http://www.kakaku.com/

feasible to identify neutral sentences using our method.

6.3 Phrase Level Experiment

We used a supervised approach (Naïve Bayes) to perform feature classification. For a supervised approach, the most important thing is to train the classifier on an appropriate corpus. We built the training corpus manually using the Weblog content on the products T1, T2, T3, T4, T6, T7, T8 and T9 shown in Table **2**. T5 and T10 are used for testing.

In our previous experiments on feature classification, we found that almost 80 percent of all opinions were of the "Function" class. Therefore, here we used only the "Function" feature category to evaluate the classification precision when detecting neutral expressions. For P/N Classification, the classification precision of positive phrases and negative phrases we achieved are shown in the Table **4**. "Preliminary experiment" means that the neutral expressions were not detected. "After Improvement" is the results when our method for dealing with neutral expressions is used.

Table 4: Phrase Level Experiment

		Preliminary Experiment (%)	After Improvement (%)
Feature Classification	**FP**	70	**79**
P/N Classification	**PP**	67	**73**
	NP	65	**74**

FP: The precision for Function feature

PP: Positive Precision **NP**: Negative Precision

After improving the classification method by taking neutral expressions into account, FP (the precision for Function Feature) and NP (Negative Precision) improved about 9%. PP (Positive Precision) improved only about 5%. This is because a lot of neutral expressions were classified into the negative category in the preliminary experiment.

We also compared the results between IC Dictionaries and the MP3 players. Since there are similar neutral expressions in the Weblog content for these products, we obtained similar classification results at both the sentence and phrase level.

7 Conclusions

We proposed an original idea for extracting and classifying neutral expressions from the Weblog to improve opinion mining. Our method works both on the sentence and phrase levels. First, we presented an identification method using a similarity algorithm to detect neutral expressions on the sentence level. At the phrase level, we improved feature classification by adding a neutral feature category and also improved P/N classification by using some threshold rules to identify neutral phrases.

The experimental results showed that this proposal improved the performance in opinion mining. The classification precisions were improved from 60% to 70%. Obviously, it was useful to extract neutral sentences in advance. Experimental results also show that the proposal for extracting and classifying neutral expressions form the Weblog in this paper has definite generality and effectiveness. It could be applied to mining opinions of various products from Weblog.

In the future, we will continue to improve our opinion mining system in mainly two directions. First, for the feature classification, we will study how to build a better training corpus to classify features more effectively. Secondly, for completing our opinion mining system, we will create a module that compares the mining results for several products of the some category and then recommend the one that seems most suitable to the user.

References

[1] Peter D. Turney, "Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews", In *Proc. ACL*, pp. 417-424 (2002)

[2] Mingqing Hu and Bing Liu, "Ming and Summarizing Customer Reviews", In *Proc. KDD*, pp. 168-177 (2004)

[3] Rebecca Bruce and Janyce Wiebe, "Recognizing subjectivity: A case study in manual tagging", Natural Language Engineering, 5 (2) (1999)

[4] Vasileios Hatzivassiloglou and Janyce Wiebe, "Effects of adjective orientation and gradability on sentence subjectivity", In *Proc. CONLING* (2000)

[5] Janyce Wiebe, Theresa Wilson, Rebecca Bruce, Matthew Bell and M. Martin, "Learning subjective language", Technical Report TR-02-100, Department of Computer Science, University of Pittsburgh, Pittsburgh, Pennsylvania (2002)

[6] Hong Yu and Vasileios Hatzivassiloglou, "Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences", In *Proc. EMNLP* pp.129-136 (2003)

[7] Bo Pang and Lillian Lee and Shivakumar Vaithyanathan, "Thumbs up? Sentiment Classification using Machine Learning Techniques", In *Proc. EMNLP*, pp.79-86 (2002)

[8] T. Inui and M.Okumura, "A Survey of Sentiment Analysis", Natural Language Processing, Vol.13, No.3, pp.201-241 (2006), In Japanese.

[9] E.W.Myers, "An O(ND) difference algorithm and its variations", Algorithmixa, 1 , pp.251-266 (1986)

[10] Guangwei Wang and Kenji Araki, "A Decision Support System Using Text Mining Technology", IEICE, SIG Notes, WI2-2006-6, pp.55-56 (2006)

[11] N. Kobayashi, K. Inui, Y. Matsumoto, K. Tateishi, and S. Fukushima, "Collecting evaluative expressions by a text mining technique", IPSJ SIG NOTE, Vol 154, No 12 (2003), In Japanese.

[12] Taku Kudoh and Yuji Matsumoto, "Applying Cascaded Chunking to Japanese Dependency Structure Analysis", IPSJ Academic Journals, Vol 43, No 6, pp. 1834-1842 (2002), In Japanese.

[13] Computational Linguistics Laboratory, Graduate School of Information Science, Nara Institute of Science and Technology, "CaboCha: A Dependency structure analyzer", http://www.chasen.org/~taku/software/cabocha/ (2005)