Display results of opinion mining with tagclouds

Luole Qi

Abstract

Nowadays, many big commercial websites such as Yahoo shopping, amazon.com often ask their customers to write and post their reviews and hands-on experiences on products they have purchased. Unfortunately, reading through all customer reviews is time-consuming, especially for popular products, the number of reviews can be up to hundreds or even thousands. This makes it difficult for a potential customer to read them to make an decision. The system designed in this work aims to mine customer reviews of a product and extract product features on which reviewers express their opinions and use an effective way called Tagclouds to present them. This summarization task is different from traditional text summarization because we are only interested in the specific features of the product that customers have opinions on and also whether the opinions are positive or negative. We do not summarize the reviews by selecting or rewriting a subset of the original sentences from the reviews to capture their main points as in the classic text summarization. In this paper, a number of techniques are presented to mine such features and we also give the method to generate tagclouds.

1. Introduction

With the rapid expansion of e-commerce, more and more products are sold on the Web, and more and more people are buying products on the Web. In order to enhance customer satisfaction and their shopping experiences, it has become a common practice for online merchants to enable their customers to write and post reviews or to express opinions on the products that they buy. With more and more common users becoming comfortable with the Internet, an increasing number of people are writing reviews. As a consequence, the number of reviews that a product receives grows rapidly. Some popular products can get hundreds of reviews at some large merchant sites. This makes it very hard for a potential customer to read them to help him or her to make a decision on whether to buy the product. This paper aims to design a system that is capable of extracting, product features and opinion expressions from product reviews and present the result to users in tagclouds form.

The objective in our system is to answer the following questions: given a particular product, 1) how to extract potential product features from the reviews? 2) how to group the synonymous product features together? and 3) How to design the font size and weight of tags in tagclouds.

The rest of this paper is organized as follows: section 2 discusses related work. Section 3 describes in detail the system framework and each system component. We describe in section 4 the design of font size and weight of features and opinion words in tagcloud and present the result. In the end, we give our conclusions and our future work in section 5.

2. Related Work

.

Opinion mining has been studied by many researchers in recent years. The research maybe divided into two main research categories, say, document level opinion mining and feature level opinion mining. In document level, Turney et. al [1] proposed a method to determinw document's polarity by calculating the average semantic orientation of extracted phrases. So was computed by using pointwise mutual information (PMI) to measure the dependence between extracted phrases and the reference words "excellent" and "poor" by using web search hit counts. Littman ea. al [2] further expanded Turney et. al's work by using cosine distance in latent semantic analysis (LSA) as the distance measure. Dave, Lawrence and Pennock [3] classified reviews on Amazon by calculating scores using normalized term frequency on uni-gram, bi-gram and tri-gram with different smoothing techniques. Das and Chen [6] studied document level sentiment polarity classification on financial documents. Pang, Lee and Vaithyanathan [4] used several machine learning approaches to classify movie reviews and in [5], they further studied another machine learning approach based on subjectivity detection and minimum cuts in graphs for sentiment classification of movie reviews. Our work is different from these as their goal is to determine the sentiment of documents while ours is to perform extraction and classification on features. Another difference is they were not focused on features being commented on.

In feature level opinion mining, Hu and Liu [7] proposed a statistical approach capturing high frequency

feature words by using association rules. Infrequent feature words are captured by extracting known opinion words'adjacent noun phrases. A summary is generated by using high frequency feature words (the top ranked features) and ignoring infrequent features. Zhuang, Jing and Zhu [8] classified and summarized movie reviews by extracting high frequency feature keywords and high frequency opinion keywords. Feature-opinion pairs were identified by using a dependency grammar graph. However, it used a fixed list of keywords to recognize high frequency feature words, and thus the system capability is limited. Popescu and Etzioni [9] proposed a relaxation labeling approach to find the semantic orientation of words. However, their approach only extracted feature words with frequency greater than an experimentally set threshold value and ignored low frequency feature words. Ding, Liu and Yu [10] further improved Hu's system by adding some rules to handle different kinds of sentence structures. However all their work does not group synonymous features and do not present them to users in an effective way.

3. The Proposed Method

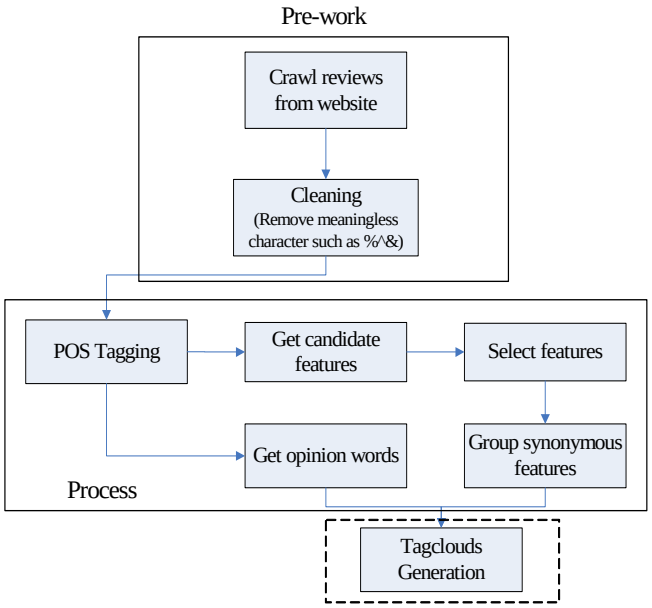

Figure 1: A general schedule for the Symposium

Figure 1 gives an architectural overview for our opinion mining system. The system performs the mining in three main steps: features selection, opinion words extraction and tagclouds generation. The inputs to the system are all semi-structured reviews for all the reviews of each product. The output is the tagclouds webpage of the reviews. Given the inputs, the system first downloads (or crawls) all the reviews from Yahoo shopping website by using the API it provide to developers, and save them as XML format files in the local server. Some cleaning work such as deleting some meaningless characters will be done as well. The feature extraction function, which plays a crucial role in our work, first extracts noun words from cons and pros parts of the reviews as candidate product features. And then, we will group synonymous candidate product features together and use a typical word to represent them. After that the opinion words will be extracted by using the POS tags and the semantic polarity of the opinion words will be determined through a dictionary manually made by Pang, B. and Lee, L [11]. In the end, all the results will be displayed in a tagclouds format. In Figure 1, POS tagging is the part-of-speech tagging from natural language processing. Below, we discuss each of the functions in feature extraction in turn.

3.1. Semi-structure of review

All the reviews in our work are from Yahoo shopping website. We got these data using the Yahoo shopping API and save them as static XML file in the local server. The review format is semi-structured and it consists of several parts:

Table 1: Each part of one complete review

Review	Contains each individual review.
Title	The title of the review as entered by the reviewer.
Reviewer	The name or Yahoo! ID that the reviewer chose to call them self.
CreateTime	The UNIX time when the review was written
HelpfulRecommendations	The number of people who found this review helpful
TotalRecommendations	The total number of people who have read this review.
Ratings	Container for sub-ratings / detailed components of the user review.
Rating	Sub-rating for detailed component of the user reivew. Has an attribute ratingType, which specifies the name of the sub-rating component
OverallRatings	The rating given by the reviewer for the product, out of 5.
Pro	The pros (positive attributes) of the product as per the reviewer.
Con	The cons (negative attributes) of the product as per the reviewer.
Posting	The body of the review.

3.2. Part-of-speech Tagging

As many researchers observed, product features are usually nouns or noun phrases in review sentences. Thus

the part-of-speech tagging is crucial. The task of POS tagging is the process of marking up the words in a text (corpus) as corresponding to a particular part-of-speech, such as noun and verb. We use the LBJPOS tool by natural language processing group of University of Illinois, Urbana-Champaign to produce the part-of-speech tag for each word in every review, the following shows the example of POS tags for one sentence from our reviews:

(PRP I) (VBD used) (NNP Olympus) (IN before) (, ,) (VBG comparing) (TO to) (NN canon) (IN in) (JJ general) (, ,) (PRP it) (VBD was) (DT a) (NN toy) (, ,) (NNP S3) (VBZ IS) (VBZ is) (RB not) (DT a) (JJ professional) (NN camera) (, ,) (CC but) (RB almost) (VBZ has) (NN everything) (PRP you) (VBP need) (, ,) (TO to) (PRP me) (, ,) (PRP it) (: ;) (VBZ s) (JJ professional) (, ,) (NN 6mb) (VBZ is) (JJ great) (, ,) (PRP I) (VB don) (: ;) (NN t) (VBP need) (DT a) (NN 10mb) (, ,) (VB zoom) (VBZ is) (JJ outstanding) (, ,) (NN night) (NNS snapshots) (VBP are) (RB really) (JJ good) (. .)

Each sentence is saved in different folders of the local server according to pros part, cons part and post part. Then some cleaning work is also performed which mainly focus on the misspelling checking.

3.3. Product feature extraction

3.3.1 Extraction from pros and cons

We firstly extract product features from pros part and cons part of each review. The reason why we put our focus on these two parts is that they are usually simple sentences or just some words segments sometimes. In addition, they are kind of like the summary of posting made by the users themselves, thus few non-feature noun words would be mentioned in them. In other words, most of the noun words in pros part or cons part tend to be product features based our observation. So, we find all the single noun and noun phase which consist of some consecutive noun words as the selected features for the product. Figure 2 shows the pro and con part of digital camera Nikon D90 of one user.

<Pro>Ease of use,Daylight Photo Quality,Video</Pro>
<Con>Battery life,Photo Quality Gegrades when Zoom is Used</Con>

Figure 2: A general schedule for the Symposium

However, due to the difficulty of natural language understanding, some types of sentences are hard to deal with, even they are simple sentence or just some words segments. Let us see a sentence from the pro part of one review for Nikon D90:

Picture quality is superb.

In this sentence, we could easily find the "Picture quality" is the feature of this product, and the adjective word "superb" describe how good the camera's picture quality is. The feature is mentioned in this sentence, so we know this is a product feature. This kind of feature is defined as explicit feature by many other researchers. But, in many sentences, no feature words would be mentioned let us see this example:

It's not good in low light.

The user is also talking about the picture quality of this camera in a dark environment, however, the product feature words "picture quality" does not appear in this sentence. This kind of feature is defined as implicit feature. In our work, we only focus on explicit features.

3.3.2 Extraction from posting

Different from pros and cons part, posting part usually contains some complicated sentences, which include many unrelated noun words. Thus, just extracting noun word or noun phrase would cause a lot of errors. For example, in this sentence that also is from one review of Nikon D90:

I bought the D90 over half a year ago and have been happy with my purchase.

In this sentence, D90 and purchase are both noun words, however, actually, neither of the two words is product feature. To solve this problem, we adopt a way to prune those un-feature words. Based on our observation, a product feature word is usually going with an adjective word. We define the step of any two consecutive words is 1, and then we find out all the noun words and noun phrases as candidate product features. For each candidate product feature, if there is an adjective word within 3 steps from the candidate, this word or phrase would be selected as product feature. Of course, if there is a colon, stop or any other punctuation within 3 steps, it can not be seen as a feature.

3.4. Grouping synonymous features

In this step, we are trying to find out all the synonymous features and represent them in a typical word to make our summary work more precise, because people usually use different words to describe the same feature, for example, "photo", "picture" and "image" all refers to the same feature in digital camera reviews. Thus, it is important to group features with similar meaning together. We use a simple method. The basic idea is to employ WordNet [12] to check if any synonym groups/sets exist among the features. For a given word, it may have more than one sense, i.e., different synonyms for different senses. However, we cannot use all the synonyms as they will result in many errors. For example, movie and picture are considered as synonyms in a sense, or in a synset (defined in WordNet). This is true when we talk about Hollywood movies. But, in the case of a digital camera review, it is not suitable to regard picture and movie in one synset, as picture is more related to photo

while movie refers to video. To reduce the occurrence of such situations, we choose only the top two frequent senses of a word for finding its synonyms. That is, word A and word B will be regarded as synonyms only if there is a synset containing A and B that appear in the top two senses of both words.

3.5. Opinion Words Extraction

For those feature words extracted from pros and cons part, we already know their opinions expressed by users, either positive or negative. However, only knowing the sentiment of them can not satisfy many users, they usually want to see the detail descriptions, that is, how other user think about this product or what words they use to express their opinion. Thus we are trying to find out those words to present them to users in this step. For each feature in pros or cons part, we looking forward and back from the position it's in the context. All the adjective words within 3 step are it's opinion words, and also, if there is a colon, stop or any other punctuations within 3 steps which is more near the feature word, the adjective words can not be seen as an opinion word. For example,

The image is good, Nice high ISO.

In this sentence, "good" and "nice" are both adjective words within 3 steps from the position of product feature "image" in the context, but only "good" describe the "image", and "nice" does not.

We adopt the similar way for the features words which extracted from posting part, the only difference is that we need do determine the polarity of their opinion words. Here for each of them we decide their sentiment using a lexicon tagging more than 8000 adjective words with the key words positive or negative. (This lexicon is made by made by Pang, B. and Lee, L).

4. Tag clouds

After the steps described above, we find out the product features and their opinion words, that is, the opinion-feature pair. The traditional way to present them to users is simply listing them. However, this way can not catch the eyes of user at the first glance and it also can not express the information directly. To make users get the information in a straightforward way, we design Tag clouds to show the result. A tag cloud or word cloud (or weighted list in visual design) is a visual depiction of user-generated tags, or simply the word content of a site, typically used to describe the content of web sites. Tags are usually single words and are normally listed alphabetically, and the importance of a tag is shown with font size or color.[13]

4.1. Font size and weight of tags

In this step, we are trying to show the importance of tags by defining their font size and weight based on the times of one tag appears in a review and ratings given by the reviewer. Firstly, we will define some variables:

Table 2: The variables we defined

Variable names	Notation
HelpfulRecommendations	H_{ij}
TotalRecommendations	T_{ij}
Rating	R_{ijk}
OverallRatings	O_{ij}
Feature	F_{ijk}
Frequency	f_{ijk}
Importance	I_{ik}

In table 2, H_{ij} means the number of people who found the jth review of ith product helpful. T_{ij} means the total number of people who have read the jth review of ith product. R_{ijk} means sub-rating for the kth feature in jth review of ith product. O_{ij} means The rating given by the reviewer for the ith product in the jth review. F_{ijk} means the kth feature which appears in the jth review of the ith product. f_{ijk} means the appearing times of the kth feature in the jth review of the ith product. I_{ik} means the number of users who mention the kth feature of the ith product .

Thus, the impact of the rating value for the kth feature of the ith product (assume there are N_i reviews for the ith product):

$$\frac{R_{ijk}}{\frac{\sum_{j=1}^{N_i} R_{ijk}}{N_i}} = \frac{N_i R_{ijk}}{\sum_{j=1}^{N_i} R_{ijk}}$$

The impact of the overall rating value of any feature in the the jth review of the ith product:

$$\frac{O_{ij}}{\frac{\sum_{j=1}^{N_i} O_{ij}}{N_i}} = \frac{N_i O_{ij}}{\sum_{j=1}^{N_i} O_{ij}}$$

The impact of the number of users who mention the kth feature of the ith product:

$$1+\frac{I_{ik}}{N_i}$$

The impact of the frequency of the kth feature in the jth review of the ith product:

$$\ln(f_{ijk})$$

So the impact of the frequency of the kth feature in the ith product could be defined as (adding the impact of the HelpfulRecommendations and TotalRecommendations):

$$\prod_{j=1}^{N_i}[(\ln(f_{ijk})\cdot(1+\frac{H_{ij}}{T_{ij}})]$$

The font size of the kth feature of the ith product:

$$F_{ik}(size) =$$

$$(1+\frac{I_{ik}}{N_i})\cdot\prod_{j=1}^{N_i}[(1+\frac{f_{ijk}}{\max(f_{ijk})})\cdot(1+\frac{H_{ij}}{T_{ij}})]$$

The weight of the kth feature of the ith product:

$$F_{ik}(weight) =$$

$$(1+\frac{I_{ik}}{N_i})\cdot\prod_{j=1}^{N_i}[(\frac{N_i R_{ijk}}{\sum_{j=1}^{N_i} R_{ijk}})\cdot(1+\frac{H_{ij}}{T_{ij}})\cdot\frac{N_i O_{ij}}{\sum_{j=1}^{N_i} O_{ij}}]$$

All the font size and weight will be normalized from the smallest size to the biggest size.

The opinion words would employ the same rule, and the only difference is the whole scale of font size and weight.

4.2. Result

Figure 3 shows the result of opinion mining for the product Canon Powershot S3. You could see every feature word is following some opinion words. The font size of weight of all the words are based on the formula described above. The times of the feature word is also on the right down corner.

Through this picture we could see what features of this product mentioned by most users and the opinion words which describe them clearly. However, there is still some errors in the results, how to decrease them is what we should do in the next step. Also, in our future work, in order to prove our method is better than the traditional ways of present results, we would evaluate our system and conduct a comparison among similar systems by other researchers.

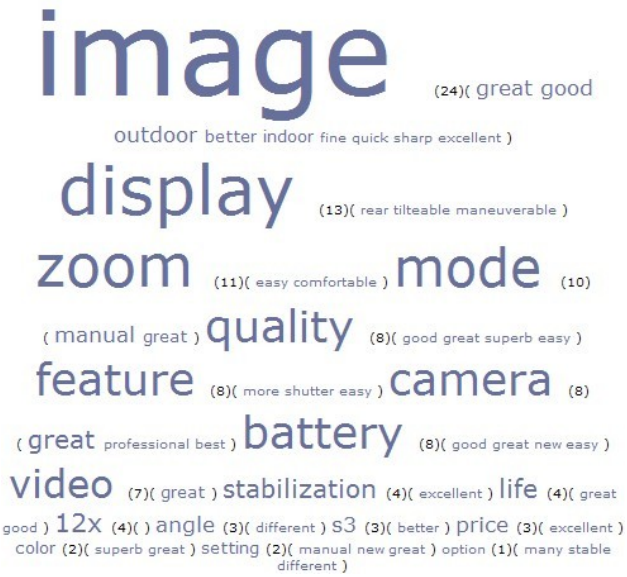

Figure 3: the result of opinion mining

5. Conclusion

In this paper, we proposed a method to extract features and opinion words from a semi-structured review and design a tagcloud to display the results. The output of our system successfully generates features and opinion words tagclouds for products which is more straightforward than the traditional ways of listing result.

In our future work, we plan to improve and refine our system in several ways, such as extracting features words more precise, finding out the opinion words which are not expressed in adjective words and so on. We also will conduct to some work to evaluate our system and make comparisons with other researchers' systems. We believe that this problem will become increasingly important as more people are buying and expressing their opinions on the Web. Mining reviews is not only useful to sellers, buyers, but also crucial to product manufacturers.

References

[1] Turney, P. D. 2002. Thumbs up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02), 417-424.

[2] Turney, P. D. and Littman, M. L. 2003. Measuring praise and criticism: Inference of semantic orientation from association. ACM Trans. On Information Systems, 21, 4 (2003), 315-346.

[3] Dave, K., Lawrence, S., and Pennock, D. M. 2003. Mining the Peanut Gallery: Opinion Extraction and Semantic

Classification of Product Reviews. In Proceedings of the 12th international conference on World Wide Web (WWW'03), 519-528.

[4] Pang, B., Lee, L., and Vaithyanathan, S. 2002. Thumbs up? Sentiment classification using machine learning techniques. In Proceedings of 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP'02), 79-86.

[5] Pang, B. and Lee, L. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In Proceedings of the 42th Annual Meeting of the Association for Computational Linguistics (ACL'04), 271-278.

[6] Das, S. and Chen, M. 2001. Yahoo! for Amazon: Extracting market sentiment from stock message boards. In Proceedings of the 8th Asia Pacific Finance Association Annual Conference (APFA'01).

[7] Hu, M. and Liu, B. 2004. Mining and Summarizing Customer Reviews. In Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'04), 168-177.

[8] Zhuang, L., Jing, F., and Zhu, X. 2006. Movie Review Mining and Summarization. In Proceedings of the International Conference on Information and Knowledge Management (CIKM'06), 43-50.

[9] Popescu, A. and Etzioni, O. 2005. Extracting Product Features and Opinions from Reviews. In Proceeding of 2005 Conference on Empirical Methods in Natural Language Processing (EMNLP'05), 339-346.

[10] Ding, X., Liu, B., and Yu, P. S. 2008. A Holistic Lexiconbased Approach to Opinion Mining. In Proceeding of the international conference on Web Search and Web Data Mining (WSDM'08), 231-239.

[11] http://www.cs.cornell.edu/home/llee/.

[12] Fellbaum, C. 1998. WordNet: an Electronic Lexical Database, MIT Press.

[13] http://en.wikipedia.org/wiki/Tag_cloud.