

METHODOLOGY

Open Access



# Mining aspects of customer's review on the social network

Tu Nguyen Thi Ngoc<sup>1\*</sup>, Ha Nguyen Thi Thu<sup>1</sup> and Viet Anh Nguyen<sup>2</sup>

\*Correspondence:  
tunn.dhdl@gmail.com

<sup>1</sup> Department  
of E-Commerce, Vietnam  
Electric Power University,  
235 Hoang Quoc Viet, Hanoi,  
Vietnam  
Full list of author information  
is available at the end of the  
article

## Abstract

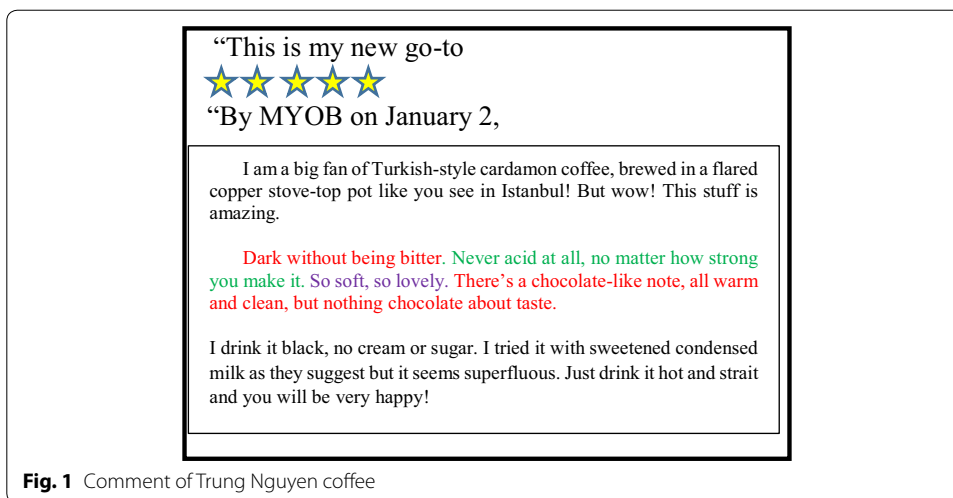
This study represents an efficient method for extracting product aspects from customer reviews and give solutions for inferring aspect ratings and aspect weights. Aspect ratings often reflect the user's satisfaction on aspects of a product and aspect weights reflect the degree of importance of the aspects posed by the user. These tasks therefore play a very important role for manufacturers to better understand their customers' opinion on their products and services. The study addresses the problem of aspect extraction by using aspect words based on conditional probability combined with the bootstrap technique. To infer the user's rating for aspects, a supervised approach called the Naive Bayes classification method is proposed to learn the aspect ratings in which sentiment words are considered as features. The weight of an aspect is estimated by leveraging the frequencies of aspect words within each review and the aspect consistency across all reviews. Experimental results show that the proposed method obtains very good performance on real world datasets in comparison with other state-of-the-art methods.

**Keywords:** Aspect extraction, Aspect rating, Aspect weight, Conditional probability, Core term, Naive Bayes

## Introduction

In recent years, a lot of people often express their opinions about things such as products and services on social networks and e-commerce web sites. These opinions or reviews often play significant role in improving the quality of products and services. However, the huge amount of reviews poses a challenge of how to efficiently mine useful information about a product or a service. To deal with this problem, much work has been introduced including summarizing users' opinions [1], extracting information from reviews [2–5], analyzing user sentiments [6–9], and so on. In this paper, we focus on the problem of extracting information from reviews. More specifically, this study aims at developing efficient methods for dealing with the three tasks: extracting aspects mentioned in the reviews of a product, inferring the user's rating for each identified aspect, and estimating the weight posed on each aspect by the users.

A user review often mentions different aspects, which are attributes or components of a product. An aspect is usually a concept in which the user's opinion is expressed in different level of *positivity* or *negativity*. For example, in the review given in Fig. 1, the user likes the coffee, manifested by a 5-star overall rating. However, positive opinions about

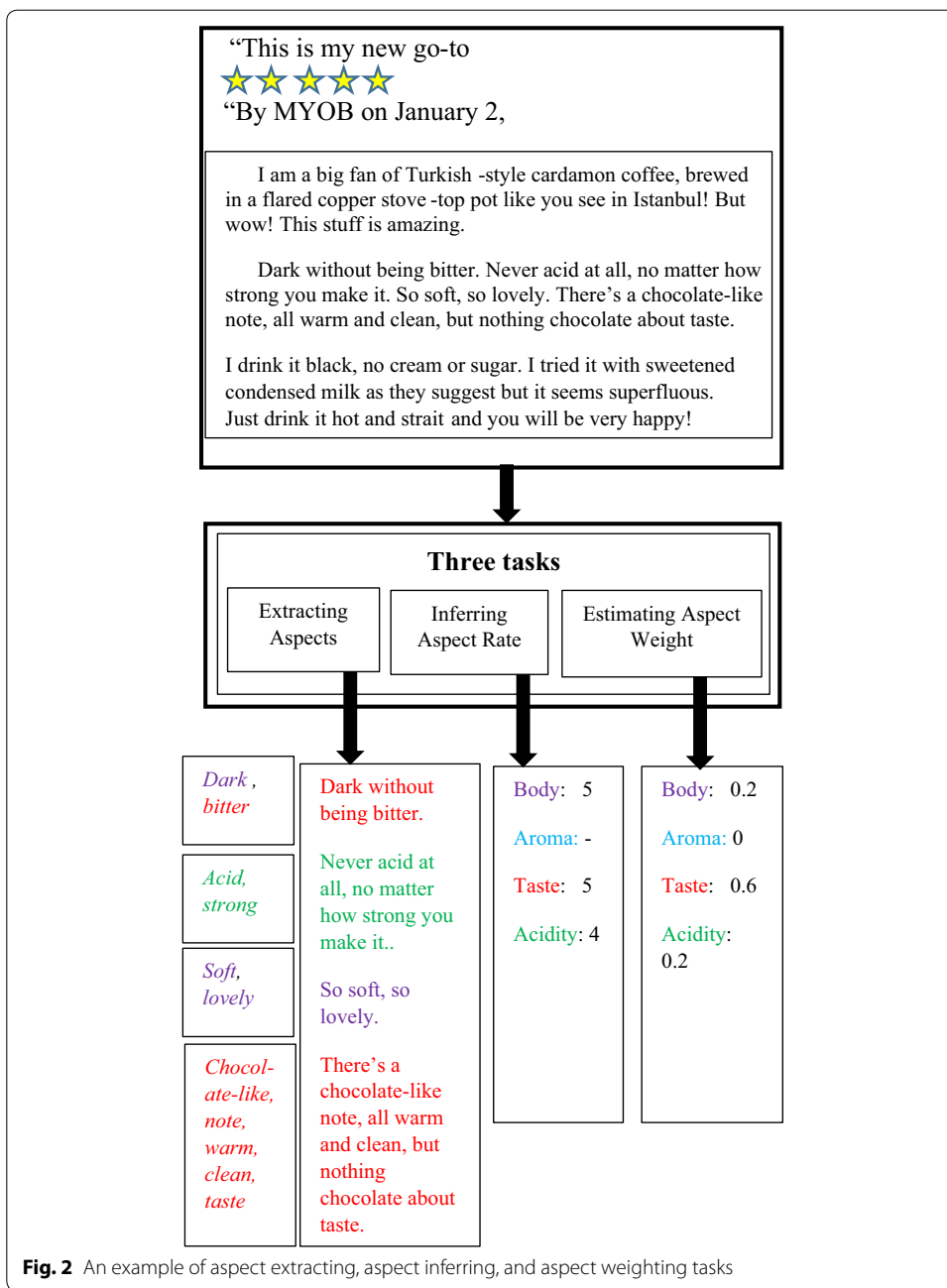


**Fig. 1** Comment of Trung Nguyen coffee

*body, taste, aroma* and *acidity* aspects of the coffee are also given. The task of *aspect extraction* is to identify all such aspects from the review. A challenge here is that some aspects are explicitly mentioned and some are not. For instance, in the review given in Fig. 1, *taste* and *acidity* of the coffee are explicitly mentioned, but *body* and *aroma* are not explicitly specified. Some previous work dealt with identifying explicit aspects only, for example [10]. In our paper, both explicit and implicit aspects are identified. Another difficulty of the aspect extraction task is that it may generate a lot of noise in terms of non-aspect concepts. How to minimize noise while still be able to identify rare and important aspects is also one of our concerns in this paper.

Most of the earliest work to identify aspects are unsupervised model-based [11], in which statistics of relevant words are used. These methods do not require the labeled training data and have low cost. For example, frequency-based methods [10, 12, 13] consider high-frequent nouns or noun phrases as aspect candidates. However, frequency-based approaches may miss low-frequent aspects. Several complex filter-based approaches are applied to solve this problem; however, the results are not as good as expected because some aspects are still missed [14, 15]. Moreover, these methods face difficulty in identifying implicit aspects. To overcome these problems, some supervised learning techniques, such as the Hidden Markov Model (HMM) and Conditional Random Field (CRF) have been proposed. These techniques, however, require a set of manually labeled data for training the model and thus could be costly.

The problem of aspect extraction is solved by using aspect words based on conditional probability combined with the bootstrap technique. It is assumed that the universal set of all possible aspects for each product are readily available together with aspect words called *core terms* (terms that describe aspects). This assumption is practical because the number of important aspects is often small and can be easily obtained by domain experts. The aspect extraction task then becomes how to correctly assign existing aspects to sentences in the review. The main challenge here is that in many reviews, sentences do not contains enough core terms or even do not have any core term at all, and thus may be assigned with wrong aspects. This problem is solved by repeatedly updating



and enlarging the set of core terms to the set of *aspect words* by using the conditional probability technique combined with the bootstrap technique. This method leads to better results of aspect extraction as shown in “[Results and discussion](#)” section.

After the aspects are identified, inferring the user’s rating for them may bring more thorough understanding of the user’s satisfaction. A user usually gives an overall rating which express a general impression about a product. The overall rating is not always informative enough. However, it can be assumed that the overall rating on a product is weighted sum of the user’s specific rating on multiple aspects of the product, where

the *weights* basically measure the degree of importance of the aspects. Some previous work [16, 17] infer the user's rating for aspects and estimate the weight of aspects at the simultaneously based on regression methods and using only the review content and the associated overall rating. Different approach is applied to infer rating and weight of aspects. More specifically, the weight of an aspect is calculated by leveraging the aspect words frequency within the review and the aspect consistency across all reviews. Then, a supervised approach called the Naïve Bayes classification method is used to infer the user's rating for aspects. Despite the fact that the solution is relatively simple, its tested accuracy on different real-life datasets are comparable to much more sophisticated state of the art approaches as shown in "Results and discussion" section.

The Fig. 2 summaries the three tasks mentioned above. The methods for solving these tasks are discussed in details in "Method" section of this paper.

The rest of this paper is structured as follows. "Related work" section introduces related works. "Problem definition" and "Method" sections represent the proposed methodology. "Results and discussion" section show experimental and evaluation of the proposed method. Finally, "Conclusion" section concludes the paper and gives some future research directions.

## Related work

During the last decade, many researches work has been proposed in the opinion mining area. Researchers are paying increasing attention to methods of extracting information from reviews that indicates users' opinions of aspects about products. A survey on opinion mining and sentiment analysis [18] shows that two important tasks of aspect-based opinion mining are aspect identification and aspect-based rating inference. The survey also mentions some interesting methods for these tasks including frequency-based, lexicon-based, machine learning and topic modeling.

Most of the earliest researches to identify aspects are frequency-based ones [11]. In these approaches, nouns and noun phrases are considered as aspect candidates [10, 12–15]. Hu and Liu [10] uses a data mining algorithm for nouns and noun phrases identification and label assignment by the part-of-speech/POS [19]. Their occurrence frequencies are counted, and only the frequent ones are kept. A frequency threshold is used and can be decided via experimental. In spite of its simplicity, this method is actually quite effective. Some commercial companies are using this method with some improvements to increase in their business [11]. However, producing "non-aspect" is the limitation of these methods because some nouns or noun phrases that have high-frequency are not really aspects.

To solve these problems, some improved methods of this filtering approach have been proposed. [15] augments the frequency-based approach with an additional pattern-based filters to remove some non-aspect terms. A similar solution, [14] extracts aspects (nouns) based on frequency and information distance. Firstly, they find seed words for each aspect by using the frequency-based method. Secondly, they use the information distance in [20] to find other related words to aspects, e.g., for aspect price, it may find "\$" and "dollars". However, the frequency-based and rule-based approaches require the manual effort of tuning various parameters, which limits their generalization in practice.

To deal with the limitations of frequency-based methods, in recent years, topic modeling has emerged as a principled method for discovering topics from a large collection of texts. These researches are primarily based on two main basic models, pLSA (Probabilistic Latent Semantic Analysis) [21] and LDA (Latent Dirichlet allocation) [22]. In [4, 15, 23–25], the authors apply topic modeling to learn latent topics that correlate directly with aspects. [23] proposes a topic modeling for mining aspects. Firstly, they identify aspects using topic modeling and then identify aspect-specific sentiment words by considering adjectives only. Lin et al. [4] proposes Joint Sentiment-Topic (JST) and Reverse-JST. Both models were based on the modified Latent Dirichlet allocation (LDA). These models can extract sentiment as well as positive and negative topic from the text. Both JST and RJST yield an accuracy of 76.6% on Pang and Lee [7] dataset. While topic-modeling approaches learn distributions of words used to describe each aspect, in [24], they separate words that describe an aspect and words that describe sentiment about an aspect. To perform, this study use two parameter vectors to encode these two properties, respectively. Then, a weighted bipartite graph is constructed for each review, which matches sentences in review to aspects. Learning aspect labels and parameters are performed with no supervision (i.e., using only aspect ratings), weak supervision (using a small number of manually-labeled sentences in addition to unlabeled data), or with full supervision (using only manually-labeled data). Moghaddam and Ester [15] devised factorized LDA (FLDA) to extract aspects and estimate aspect rating. The FLDA method assumes that each user (and item) has a set of distributions over aspects and aspect-based ratings. Their work on multi-domain reviews reaches to 74% for review rating on TripAdvisor data set. In [26], the authors propose a new method called Aspect Identification and Rating model (AIR) for mining textual reviews and overall ratings. Within AIR model, they allow an aspect rating to influence the sampling of word distribution of the aspect for each review. This approach is based on the LDA model. However, different from traditional topic models, the extraction of aspects (topics) and the sampling of words for each aspect are affected by the sampled latent aspect ratings which are dependent on the overall ratings given by reviewers. Then, they further enhance AIR model to handle quite unbalance of aspects mentioned in short reviews.

Although topic modeling is an approach based on probabilistic inference and it can be expanded to many types of information models, it has some limitations that restrict their use in real-life sentiment analysis applications. For example, it requires a huge amount of data and a significant amount of tuning in order to achieve reasonable results. It is very easy to find those general and frequent topics or aspects from a large document collection, but it is hard to find those locally frequent but globally that is not frequent aspects. Such locally frequent aspects are often the most useful ones for applications because they are likely to be most relevant to the specific entities that the user is interested in. In short, the results from current topic modeling methods are usually not relevant or specific enough for many practical sentiment analysis applications [11].

Besides, some lexicon-based methods, which are also unsupervised approach, are proposed. Opinions are extracted with respect to each feature using the dictionary-based approach, which also yields polarity and strength. These methods use a dictionary of sentiment words and phrases with their associated orientations and strength. They are combined with intensification and negation to compute a sentiment score for each

document [8]. Xiaowen Ding, Minqing Hu use sentence and aspect-level sentiment classification [10, 27, 28]. Yan et al. [29] propose a method called EXPRS (An Extended PageRank algorithm enhanced by a Synonym lexicon) to extract product features. To do so, they extract nouns/noun phrases first and then extract dependency relations between nouns/noun phrases and associated sentiment words. Dependency relations included subject-predicate relations, adjectival modifying relations, relative clause modifying relations, and verb-object relations. The list of product features was extended by using its synonyms. Non-features nouns are removed on the basis of proper nouns, brand names, verbal nouns and personal nouns. Peñalver-Martinez et al. [30] developed a methodology to perform aspect-based sentiment analysis of movie reviews. To extract the movie features from the reviews, they make a domain ontology (Movie Ontology). SentiWordNet is utilized to calculate the sentiment score. However, the critical issue here is how to construct such a sentiment lexicon, due to the cost of time and money to build such dictionaries.

Sentiment classification can be performed using machine learning approaches which often yield higher accuracy. Machine learning methods can be further divided into supervised and unsupervised ones. For supervised methods, two sets of annotated data, one for training and the other for testing are needed. Some of the commonly applied classifiers for supervised learning are Decision Tree (DT), SVM, Neural Network (NN), Naïve Bayes, and Maximum Entropy (ME). In paper Asha et al. [31], propose a Gini Index based feature selection method with Support Vector Machine (SVM) classifier for sentiment classification for large movie review data set. The Gini Index method for feature selection in sentiment analysis has improved the accuracy. Another research, Duc-Hong Pham and Anh-Cuong Le [32] design a multiple layer architecture of knowledge representation for representing the different sentiment levels for an input text. This representation is then integrated into a neural network to form a model for prediction of product overall ratings. These techniques, however, require a set of manually labeled data for training the model and thus could be costly.

**Problem definition**

A user review  $i$  on some product is assumed containing two parts: the review’s text denoted by  $d_i$ , and the review’s overall rating denoted by  $y_i$ . Each review’s text  $d_i$  can contain multiple sentences. Furthermore, each sentence contains multiple words coming from the universal set of all possible worlds  $\mathcal{V} = \{w_k | k = \overline{1, P}\}$ , called a word dictionary.

It is assumed further that for a product, the set of all possible  $K$  aspects is already known together with topic words, called *core terms* that describe each aspect of the product.

**Definition 1. Aspect** An aspect is a feature (an attribute or a component) of a product. For example, *taste*, *aroma*, and *body* are some possible aspects of the product “coffee”. We assume that there are  $K$  aspects mentioned in all reviews, denoted by  $\mathcal{A} = \{a_j | j = \overline{1, K}\}$ . An aspect is represented by a set of words and denoted by  $a_j = \{w | w \in \mathcal{V}, A(w) = j\}$ , where  $a_j$  is the name of the aspect,  $w$  is a word from the set  $\mathcal{V}$ , and  $A(\cdot)$  is a operator that maps a word to the aspect. For example, words such as “taste”, “aftertaste”, and “mouth feel” can characterize the *taste* aspect of the product coffee.

**Definition 2. Aspect rating** Given a review  $i$ , a  $K$ -dimensional vector  $\mathbf{r}_i \in \mathbb{R}^K$  is used to represent the rating of  $K$  aspects in the review’s text  $d_i$ , denoted by  $\mathbf{r}_i = (r_{i_1}, r_{i_2}, \dots, r_{i_K})$ , where  $r_{i_j}$  is a number indicating the user’s opinion assessment on aspect  $a_j$ , and  $r_{i_j} \in [r_{\min}, r_{\max}]$  (e.g., the range of  $r_{i_j}$  can be from 1 to 5).

**Definition 3. Aspect weight** Given a review  $i$ , a  $K$ -dimensional vector  $\alpha_i \in \mathbb{R}^K$  is used. The vector is denoted as  $\alpha_i = (\alpha_{i_1}, \alpha_{i_2}, \dots, \alpha_{i_K})$  where  $\alpha_{i_j}$  is a number measuring the degree of importance of aspect  $a_j$  posed by the user,  $\alpha_{i_j} \in [0, 1]$ , and  $\sum_{j=1}^K \alpha_{i_j} = 1$ . A higher weight means more emphasis is put on the corresponding aspect.

**Definition 4. Aspect core terms** Given an aspect  $a_j$ , the set of associated core terms for  $a_j$  is denoted by  $\mathcal{C}_j = \{w_{j1}, w_{j2}, \dots, w_{jN}\}$  where  $w_{jk}$  is a word that describes the aspect  $a_j$ . The core terms can be provided by the user or by some field experts.

Major notations used throughout the paper are given in Table 1.

**Extracting aspect**

The goal of this task is to extract aspects mentioned in a review. It is assumed that each aspect is a probability distribution over words. It is also assumed that each sentence in a review’s text can mention more than one aspect. Therefore, our method to extract aspects is based on conditional probability of words such that each sentence can be assigned with multiple labels.

**Inferring aspect rate**

This task is to infer the vector  $\mathbf{r}_i$  of aspect ratings (defined in Definition 2) given a review  $d_i$ . Rating of an aspect reflects the user’s sentiment on the aspect which is often expressed in positive or negative words. The more positive words the user use, the higher rating he/she want to pose on the aspect. This research adopts a supervised learning method, the Naive Bayes method, to learn the aspect ratings in which sentiment words are considered as features.

**Table 1 Notations used in this paper**

Notation	Description
$\mathcal{D} = \{d_i   i = \overline{1, Q}\}$	The set of reviews’ text, where $Q$ is the number of reviews
$\mathcal{Y} = \{y_i   i = \overline{1, Q}\}$	The set of overall rating, $y_i$ is overall rating corresponded with $d_i$
$\mathcal{A} = \{a_1, a_2, \dots, a_K\}$	The set of aspect, where $K$ is the number of aspects
$\mathcal{C}_j = \{w_{j1}, w_{j2}, \dots, w_{jN}\}$	The set of associated core terms for aspect $a_j$ , where $N$ is the number of words
$\mathcal{V} = \{w_k   k = \overline{1, P}\}$	The corpus of words, where $P$ is the number of words
$\mathcal{S}_j = \{s_{j1}, s_{j2}, \dots, s_{jM}\}$	The set of sentences are assigned aspect $a_j$ , where $M$ is the number of sentences
$\mathcal{T}_j = \{w_{j1}, w_{j2}, \dots, w_{jT}\}$	The set of aspect words are aspect expressions, where $\mathcal{T}_j$ is the expression for aspect $a_j$ , and $T$ is the number of words
$i_j \in \mathbb{R}^K$	The aspect rating inferred from review $d_i$ over $K$ aspect, $\mathbf{r}_i = (r_{i_1}, r_{i_2}, \dots, r_{i_K})$
$\alpha_i \in \mathbb{R}^K$	The aspect weights user places on $K$ aspect within reviews’ text $d_i$ $\alpha_i = (\alpha_{i_1}, \alpha_{i_2}, \dots, \alpha_{i_K})$
$y_i \in \mathbb{R}^+$	The overall rating of review $d_i$
$r_{ij}$	The aspect rating on $j$ -th aspect of review $i$ , $r_{ij} \in [1, 5]$
$\alpha_{ij}$	The aspect weight of $j$ -th aspect of review $i$ , $\alpha_{ij} \in [0, 1]$

### Estimating aspect weight

This task is to estimate non-negative weights  $\alpha_i$  that a user places on aspect  $a_{ij}$  of review  $i$ . Weight of an aspect essentially measures the degree of importance posed by the user on the aspect. It is observed that people often talk more on aspects that they are interested in a same review. Besides, the idea that an aspect is important is often shared by many other people. Based on these observations, a formula is devised to calculate aspect weight. The formula takes into account the occurrences of words discussing the aspect within a review and the frequency of text sentences discussing the same aspect across all reviews.

## Method

### Extracting aspect

The goal of this task is to assign a subset of aspect labels from the universal set of all aspect labels of a product to every sentence in a review. Aspect label is determined based on the set of relevant words called aspect words or terms. Each aspect in the universal label set is provided with some initial core terms. The main challenge here is that many reviews contain very few core terms or even do not contain any term at all. This results in incorrect labels being assigned to sentences. Therefore, it is required to expand the core terms to a richer set of aspect words based on the given data (the reviews). In some existing methods, the set of aspect words is built based on Bayes or Hidden Markov Model. Our method use conditional probabilistic model [33] combined with the Bootstrap technique to generate aspect words. Figure 3 illustrates four aspects of a coffee product represented by their corresponding aspect words, in which the symbol O represents core terms, the symbol X represents words appearing in the corpus. For this coffee product four aspects *body*, *taste*, *aroma*, and *acidity* are already known.

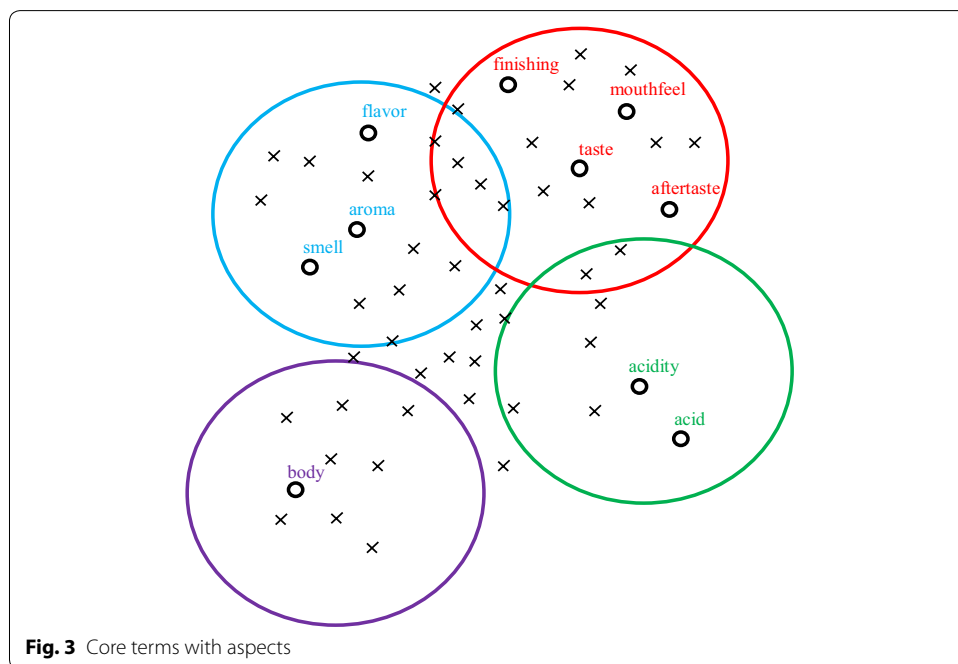


Fig. 3 Core terms with aspects



The sets of core terms corresponding to these aspects are {body}, {taste, aftertaste, finishing, mouthfeel}, {aroma, smell, flavor} and {acid, acidity}, respectively. Core terms are then enlarged by inserting words that have high probability to appear in the same sentences that they occur. Sets of aspect words are represented by the four circles. These circles may overlap, indicating that some aspect words may belong to different aspects.

Suppose that  $\mathcal{A} = \{a_1, a_2, \dots, a_K\}$  is the set of  $K$  aspects. For each  $a_j$ , a set of words that appear in sentences labeled with aspect  $a_j$  such that their occurrences exceed a given threshold is obtained. The set of words of two aspects can overlap, such that some terms may belong to multiple aspects. First, sentences that contain at least one word in the original core terms of the aspect are located. Then, all words including nouns, noun phrases, adjectives, and adverbs that appeared in these sentences are found. Words that occur more than a given threshold  $\theta$  are inserted to the set of aspect words. Words with maximum number of occurrences in the set of new-found aspect words are added to the set of core terms. The new set of aspect words with core terms excluded is used to find new sentences. The above-mentioned process is repeated until no more new words are found.

The procedure for updating aspect words for an aspect  $a_j$  is given below.

---

**Update( $\mathcal{C}_j, \mathcal{T}_j$ )    Update aspect words**

---

Input:

- $\mathcal{V}$  : Set of all words in the corpus
- $\mathcal{C}_j$  : Set of core terms regarding aspect  $a_j$
- $\mathcal{S}_j$  : Set of sentences containing aspect  $a_j$
- $\theta$ : a threshold given by the user ( $0 \leq \theta \leq 1$ )

Output:

- $\mathcal{C}_j$  : Set of updated core terms
- $\mathcal{T}_j$  : Set of aspect words regarding aspect  $a_j$

---

Initialization: set of candidate aspect word  $\mathcal{T}_j = \emptyset$ ;  
 $n=0$ ;  $N=0$ ;

**For** each word  $w_i \in \mathcal{S}_j$  **do**

- If**  $w_i \in \mathcal{V}$  **then**  $n(i) \leftarrow n(i)+1$ ;

$N=\max(n(i))$ ;

**For** each  $w_i \in \mathcal{S}_j$  **do**

- If**  $w_i \in \mathcal{V}$  &  $n(i) == N$  &  $w_i \notin \mathcal{C}_j$  **then**

$\mathcal{C}_j \leftarrow w_i$ ; // update the core term set

- If**  $w_i \in \mathcal{V}$  &  $w_i \notin \mathcal{C}_j$  &  $\Pr(w_i | \mathcal{S}_j) \geq \theta$  **then**

$\mathcal{T}_j \leftarrow w_i$ ; // update the aspect word set  $a_j$  with new word  $w_i$  that exceeding threshold  $\theta$ ,

---

A bootstrapping algorithm to assign labels to sentences in the reviews is given below.

---

**Aspect Extraction Algorithm**

---

Input:

$\mathcal{D}$  : Set of reviews' text,  $\mathcal{D}=\{d_1, d_2, \dots, d_Q\}$   
 $\mathcal{A}$  : Set of aspects,  $\mathcal{A}=\{a_1, a_2, \dots, a_K\}$   
 $\mathcal{C}$  : Set of aspect core terms,  $\mathcal{C}=\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_K\}$ ,  
 $\mathcal{C}_j$  is a set of core term in aspect  $a_j$   
 $\mathcal{V}$  : Dictionary (corpus of words)  $\mathcal{V}=\{w_1, w_2, \dots, w_p\}$   
 Loop : Number of loops  
 $\theta$  : Threshold

Output:

$\mathcal{S}_j$  : Set of labeled sentence with aspect  $a_j$

---

Step 1: Initialize the variables  $\mathcal{S}=\emptyset$ ;  $\mathcal{S}_j=\emptyset$ ;  $\mathcal{T}_j=\emptyset$ ; Loop =0; //  $\mathcal{S}$  is the set of all sentences in corpus;  $\mathcal{S}_j$  is the set of sentences are assigned aspect  $a_j$ ;  $\mathcal{T}_j$  is the set of aspect words of aspect  $a_j$ ;  $j=1 \div K$ ;

Step 2: **For** each  $d_i \in \mathcal{D}$  **do**  $\mathcal{S} \leftarrow \text{Segment}(d_i)$ ; // Split all reviews' text into sentences saved in the set  $\mathcal{S}$

Step 3: **For** each  $s_i \in \mathcal{S}$  **do**  
     **For**  $j=1$  to  $K$  **do**  
         **For** each  $w_{ik}$  **do** //  $w_{ik}$  is  $k$ -th word in  $s_i$   
             **If**  $w_{ik} \in \mathcal{C}_j$  **then**  $\{s_i \leftarrow \text{label}(a_j); \mathcal{S}_j \leftarrow s_i\}$  // Assign all possible aspect labels  $a_j$  to a sentence if the sentence contains a core term from  $\mathcal{C}_j$  of  $\mathcal{C}$ .

**Repeat**

Step 4: **For**  $j=1$  to  $K$  **do**  $\text{Update}(\mathcal{C}_j, \mathcal{T}_j)$ ; // Update core term and aspect word sets

    Loop ++;

Step 5: **For** each  $s_i \in \mathcal{S}$  **do**  
     {**For**  $j=1$  to  $K$  **do**  
         {  $m(j) = 0$ ;  
             **For** each  $w_{ik} \in s_i$  **do**  
                 { **If**  $w_{ik} \in \mathcal{C}_j$  **then**  
                     {  $s_i \leftarrow \text{label}(a_j); \mathcal{S}_j \leftarrow s_i$ ; } **else**  
                     { **if**  $w_{ik} \in \mathcal{T}_j$  **then**  $m(j) \leftarrow m(j)+1$ ; }  
                 }  
             }  
         }

**If**  $m(j)=\text{argmax}(m(j))$  **then** {  $s_i \leftarrow \text{label}(a_j)$ ;  
 $\mathcal{S}_j \leftarrow s_i$ ;

    } // Re-assign aspect  $a_j$  label to sentences

**Until**  $\text{Update}(\mathcal{C}_j, \mathcal{T}_j)$  is not change || Loop  $\geq L$ ;

---

The proposed Aspect Extraction Algorithm works as follows. First all reviews' texts are split into sentences (step 2). Then, aspect labels from the set  $\mathcal{A}$  of all labels are assigned to every sentence of the set  $\mathcal{D}$  of reviews' text based on the initial aspect core terms (step 3). Based on this initial aspect labeling, the set of aspect core terms and the set of aspect words for every aspect are updated (step 4). The labels for all sentences are updated using the new core terms and the aspect words sets (step 5). Step 4 and step 5 are repeated until no more new aspect word set are found or the number of iterations exceeds a given threshold.

### Inferring aspect rating and estimating aspect weight

Aspect ratings often reflect the user's satisfaction on aspects of a product. Meanwhile, aspect weights measure the degree of importance of the aspects posed by the user. Given the overall rating on a product, it is assumed that the overall rating is the weighted sum of rating on multiple aspects of the product. Following this assumption, some regression-based methods [16, 17, 34] have been proposed to estimate the two parameters by solving the following equation:

$$y_i = \sum_{j=1}^K r_{ij} \alpha_{ij} \tag{1}$$

where  $r_{ij}$  and  $\alpha_{ij}$  are the rating and the weight of  $k$ -th aspect of the review  $i$ , respectively.

There are linear regression methods [35] which estimate only the aspect weight and require that the aspect ratings are available. Some other methods [17, 34] estimate both aspect's rating and weight at the same time. The key point of these methods is to use sentiment words, more specifically the polarity of sentiment words, to calculate ratings and weights. Even though sentiment words can usually correctly reflect the user's rating for each aspect, they do not always reflect the user's opinion about an aspect's weight.

Aspect rating and aspect weight of an aspect are estimated separately. An important point in our method is that aspect rating and aspect weight are calculated based on the review content only, without the requirement of knowing the user's overall rating. However, in "Results and discussion" section, Eq. (1) is still used to test our method. It is shown experimentally that our results conform well to the assumption that the overall rating is the weighted sum of rating on multiple aspects.

The aspect rating problem is treated as the problem of multi-label classification, in which ratings (from 1 to 5) as considered as labels, and sentiment words are used as features. In most sentiment analysis work, adjectives and adverbs are used as candidate sentiment words. Adjectives and adverbs are detected based on the well-known Part of Speech technique (POS). It is recognized that some phrases can also be used to express sentiments depending on different contexts. For example, in the following two sentences "we have big problem with staff", and "we have a big room", the two noun phrases "big problem" and "big room" convey opposite sentiments, negative vs. positive, while both phrases contain the same adjective "big". Some *fixed syntactic patterns* in [9] as phrases of sentiment word features are used. Only fixed patterns of two consecutive words in which one word is an adjective or an adverb and the other provides a context are considered.

**Table 2 POS labeled rules [9]**

	The first word	The second word	The third word (non extracted)
1.	JJ	NN or NNS	Any word
2.	RB, RBR, or RBS	JJ	Not NN nor NNS
3.	JJ	JJ	Not NN nor NNS
4.	NN or NNS	JJ	Not NN nor NNS
5.	RB, RBR, or RBS	VB, VBD, VBN, or VBG	Any word

Two consecutive words are extracted if their POS tags conform to any of the rules in Table 2 in which JJ tags are adjectives, NN tags are nouns, RB tags are adverbs, and VB tags are verbs. For example, rule 2 in this table means that two consecutive words are extracted if the first word is an adverb, the second word is an adjective, and the third word (which is not extracted) is not a noun. As an example, in the sentence “Quite dry, with a good grassy note”, two patterns “quite dry” and “good grassy” are extracted as they satisfy the second and the third rules, respectively. Then, conditional probability of word features in the corpus is determined. Label (scoring) for each aspect is predicted based on Naïve Bayes method.

Given a review’s text  $d_p$ , the rating of an aspect  $a_j$  with  $q$  extracted features is inferred based on the probability  $r_{ij}$  that the rating label belongs to class  $c \in C = \{1, 2, 3, 4, 5\}$ . The probability is as:

$$P(r_{ij} \in c | f_1, \dots, f_q) = \frac{P(f_1, \dots, f_q | r_{ij} \in c) P(r_{ij} \in c)}{P(F_1, \dots, F_q)} \tag{2}$$

It is assumed that the features are independent, then (2) is transformed into:

$$P(r_{ij} \in c | f_1, \dots, f_q) = \frac{\prod_{k=1}^q P(f_k | r_{ij} \in c) P(r_{ij} \in c)}{\sum_{k=1}^q P(f_k)} \tag{3}$$

in which:  $P(f_k | r_{ij} \in c) = n_{aj}(f_k, c) / n_{aj}(c)$  is the probability that feature  $f_k$  belongs to the class  $c$ ,  $n_{aj}(f_k, c)$  is the number of sentences labeled as  $c$  of the aspect  $a_j$  which contains the feature  $f_k$ , and  $n_{aj}(c)$  is the number of all sentences containing the aspect  $a_j$  and has class label  $c$ ,

$P(r_{ij} \in c) = n_{aj}(c) / n_{aj}$  is the probability that the rating  $r_{ij}$  belongs to the class  $c$ ,  $n_{aj}(c)$  is the number of sentences labeled as  $c$  of aspect  $a_j$ , and  $n_{aj}$  is the number of all sentences containing the aspect  $a_j$ ,

$P(f_k)$  is the probability of feature  $f_k$ .

For smoothing (3), Laplace transformation is used. We get:

$$P(f_k | r_{ij} \in c) = \frac{n_{aj}(f_k, c) + 1}{n_{aj}(c) + |V| + 1} \tag{4}$$

in which,  $|V|$  is number of word features regarding the aspect  $a_j$ .

The rating  $r_{ij}$  is the label  $c$  that maximize  $P(r_{ij} \in c | f_1, \dots, f_q)$ .

$$\hat{c} = \operatorname{argmax}_{c \in C} \prod_{k=1}^q P(f_k | r_{i_j} \in c) P(r_{i_j} \in c).$$

Now the method to estimate aspect weight is given. By doing research carefully throughout the reviews, it can be seen that if a user care more about an aspect (showing that the aspect is important to the user), he/she will mention more about it in the review. Moreover, the idea that an aspect is important is often shared by many other users. Following this observation, we estimate aspect weights by calculating two components: the weight measure of aspect  $a_j$  within the reviews' text  $d_i$ , denoted by  $ED_{ij}$ , and the weight measure of the aspect across all reviews, denoted by  $EC_j$ . Note that in this way, the polarity measures of sentiment words are not used as in some other approaches. Instead, probability measures of words and sentences regarding an aspect in the review and the corpus are considered. This idea is similar to the idea of using tf/idf for measuring word importance to some extent.

Given a review  $i$ , the weight component of the aspect  $a_j$ ,  $ED_{ij}$ , is calculated as:

$$ED_{ij} = \frac{\sum_{k=1}^{N_i} w_{ijk}}{N_i}. \tag{5}$$

In which:  $w_{ijk}$  is the  $k$ -th word in the aspect words of aspect  $a_j$ , and  $N_i$  is the number of aspect words that occur in the review's text  $d_i$  for all aspects.

The weight component  $EC_j$  is calculated as:

$$EC_j = \frac{\sum_{k=1}^M s_{jk}}{M} \tag{6}$$

In which:  $s_{jk}$  is the  $k$ -th sentence in the corpus labeled by the aspect  $a_j$ , and  $M$  is the number of all sentences in the corpus.

Finally, the weight  $\alpha_{ij}$  for an aspect  $a_j$  of review  $i$  is calculated as:

$$\alpha_{ij} = \frac{ED_{ij} * EC_j}{\sum_{j=1}^K ED_{ij} * EC_j} \tag{7}$$

The denominator  $\sum_{j=1}^K ED_{ij} EC_j$  is to normalize the value of  $\alpha_{ij}$  to the range  $[0,1]$ .

## Results and discussion

In this section, experiments to evaluate the proposed methods are conducted.

### Data set

The experiments are carried out using three different data sets including a data set for hotel review collected from Tripadvisor.com [17], one data set for beer review used in [24], and a data set for Trung Nguyen coffee review collected by our self from the Amazon web site.

The Hotel data set contains seven different aspects that are room, location, cleanliness, check-in/front desk, service and business services. The beer data set has five distinct aspects that are aroma (or smell), palate (or feel), taste, appearance (or look), and overall. This data set is quite big with millions of reviews. A subset of 50,000 beer reviews is

**Table 3 Summary of the Data Set**

	Hotel dataset	Beer dataset	Coffee dataset
#Reviews	193,661	50,000	1200
#Sentences	1,790,880	509,320	5289
#Avg. sentences per review	9.25	10.19	4.41

**Table 4 Seed word for main aspects**

Category	Aspect	Seed words
Hotel	Value	Value, price, worth
	Room	Room, rooms
	Location	Location
	Cleanliness	Dirty, smelled, clean
	Check in/front desk	Staff
	Service	Service, breakfast, food
Beer	Business service	Internet, wifi
	Appearance	Appearance, color, colors, coloring, head, foam
	Aroma	Aroma, aromas, smell, smelling
	Palate	Palate, mouth, feel, mouth feel
	Taste	Taste, tastes, aftertaste, in the end, finish, finishing
Coffee	Overall	Overall
	Aroma	Aroma, aromas, smell, smelling, flavor, flavors
	Taste	Taste, tastes, aftertaste, finish, finishing, mouth feel
	Acidity	Acid, acidity
	Body	Body, aged, vintage

used in the experiment. The coffee data set contains 1200 reviews belongs to 17 different kinds of coffee. Table 3 gives some statistics of the three data sets.

**Inferring aspect rating task**

Note that each review may be assigned with different labels. This means that sentence level, not review level is considered. Testing sets of 2500, 2000, and 500 sentences are selected randomly from the hotel data set, beer data set, and coffee data set, respectively. The rest of sentences are used as the training sets.

Table 4 gives initial core terms for the three data sets.

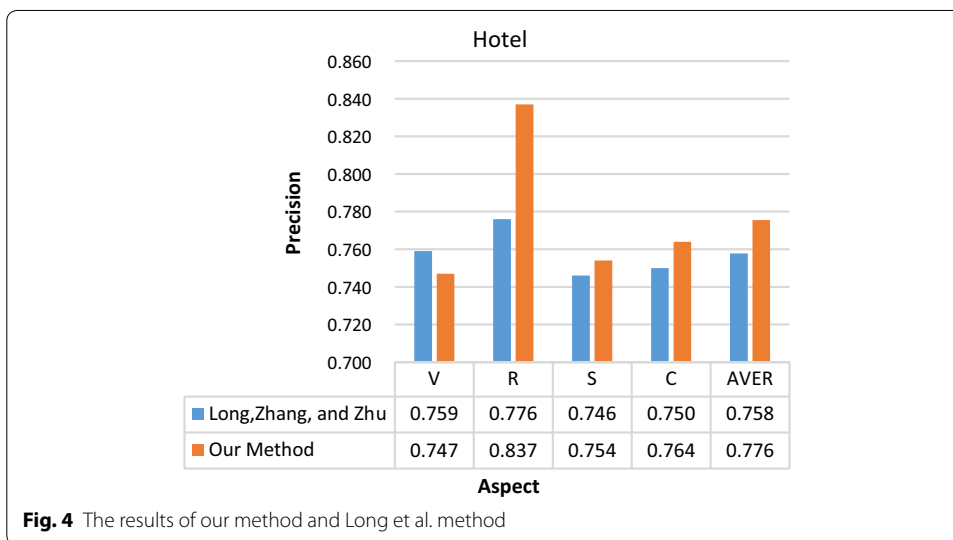
The precision measure is used to evaluate the experimental results:

$$P = \frac{|extracting\ Aspect \cap True\ Aspect|}{|extracting\ Aspect|} \tag{8}$$

Table 5 shows the performance of our method on three data sets for the aspect extraction task. Our method yields up to average precision of 0.786, 0.803 and 0.653 for hotel data set, beer data set and coffee data set, respectively. Our method obtains good performance on the hotel and beer data set. However, for the coffee data set, the result is not as good as expected. This is because in the coffee data set, users often give only general view about a product, and moreover, the data set contains mostly

**Table 5 Aspect Identification results**

Category	Aspect	Precision
Hotel	Value	0.747
	Room	0.837
	Location	0.814
	Cleanliness	0.764
	Check in/front desk	0.850
	Service	0.754
	Business service	0.737
	Average	0.786
Beer	Appearance	0.750
	Aroma	0.857
	Palate	0.857
	Taste	0.848
	Overall	0.704
Coffee	Average	0.803
	Aroma	0.667
	Taste	0.677
	Acidity	0.667
	Body	0.600
	Average	0.653

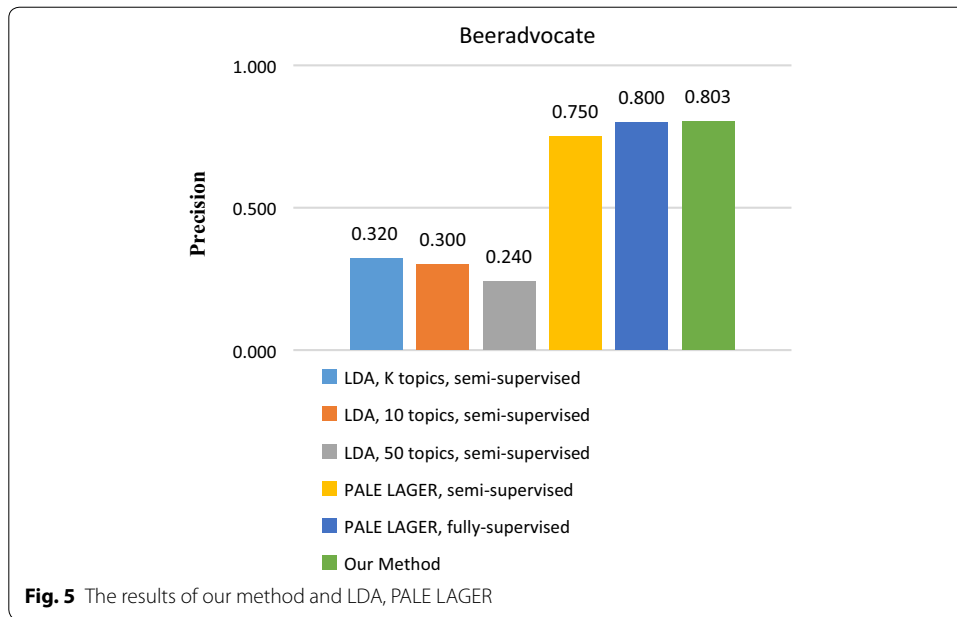


**Fig. 4** The results of our method and Long et al. method

very short reviews, with average number of sentences of 4.5, compared to 10 and 9 of the hotel data set and the beer data set.

Our method is compared with other works. First, our method is compared with the frequency-based method in [14] on the hotel dataset. Figure 4 shows that our method outperforms Long’s in room (R), service (S), and cleanliness (C) aspects. But Long’s method outperforms us in detecting the value (V) aspect.

Our method is compared with two topic modeling-based methods in [22] and in [24] on the beer data set. The method in [22] is a semi-supervised method, called



LDA. In [24], the authors give 3 different methods, namely, unsupervised, semi-supervised, and fully supervised methods. As our method can be considered as a semi-supervised method, it is compared with PALE LAGER, a semi-supervised method, and with PALE LAGER, a supervised method given in [24].

The results in Fig. 5 shows that our method outperforms LDA with a large margin, and slightly outperforms PALE LAGER (a semi-supervised method) and PALE LAGER (a supervised method).

We then search for the best threshold  $\theta$  at which our method performs the best. The results are shown in Fig. 6, where the threshold  $\theta$  of about 0.15 is the best one.

### Aspect ranking prediction

Unlike the evaluation of the aspect extraction task that is done based on the sentence level, in this task, the result based on the review level is evaluated.

The mean square error measure (named  $\Delta_{aspect}^2$ ) is used for evaluating methods of mining aspect rating.

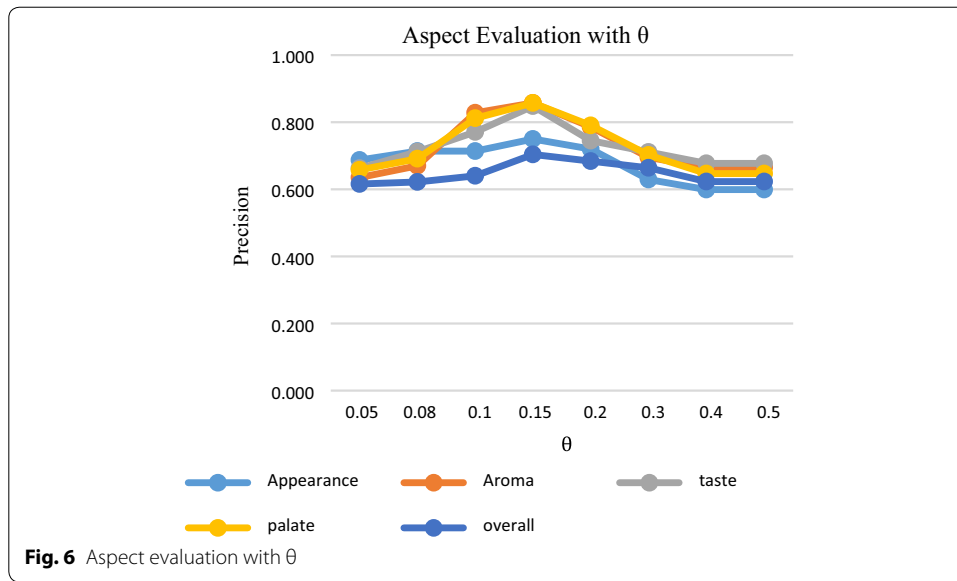
$$\Delta_{aspect}^2 = \frac{\sum_{i=1}^Q \sum_{j=1}^K (r_{ij} - r_{ij}^*)^2}{Q \times K} \tag{9}$$

where K is the number of aspects, Q is the number of reviews, and  $r_{ij}^*$  is the true ratings for aspect  $a_j$  within review's text  $d_i$ .

To evaluate how well the predicted aspect ratings can preserve their relative order within a review given the true ratings, the aspect correlation measure (named  $\rho_{aspect}$ ) is used:

$$\rho_{aspect} = \frac{\sum_{i=1}^Q \rho_{r_i, r_i^*}}{Q} \tag{10}$$





where Q is the number of reviews, and  $\rho_{r_i, r_i^*}$  is the Pearson correlation between two vectors  $r_i$  and  $r_i^*$  of the inferred and the true ratings, respectively.

The two measures above are for evaluating the results for each review. The results on the whole set of reviews are evaluated by using the so called *aspect correlation across reviews* measure ( $\rho_{review}$ ):

$$\rho_{review} = \frac{\sum_{j=1}^K \rho(\vec{r}_j, \vec{r}_j^*)}{K} \tag{11}$$

where  $\rho(\vec{r}_j, \vec{r}_j^*)$  is the Pearson correlation between two vectors  $\vec{r}_j$  and  $\vec{r}_j^*$  of the inferred and rating.

Our method is also compared with Long’s [14] and Wang’s [17]. Long proposed two methods based on the SVM classifier and the Bayesian Network classifier. Wang’s method is called Latent Rating Regression (LRR) which infers aspect ratings and aspect weights simultaneously.

The performance results are shown in Table 6. Our method performs much better than Long’s method and Wang’s method on all three measures.

**Estimating aspect weight**

For evaluating the correctness of estimated weights by our method, the overall rating is calculated and compared with the true overall rating given by the user. The estimated overall rating is given by the following formula:

$$\hat{y}_i = \sum_{j=1}^K r_{ij} \alpha_{ij} \tag{12}$$

where  $r_{ij}$  is the rating of the  $j$ -th aspect of the review  $i$  and  $\alpha_{ij}$  is the estimated weight.

**Table 6 Comparison with other models for referring aspect ratings**

Method	$\Delta^2_{aspect}$	$\rho_{aspect}$	$\rho_{review}$
Long et al. with SVM	0.286	0.557	0.708
Long et al. with BN	0.441	0.429	0.591
LRR	0.896	0.464	0.618
Our method	0.101	0.583	0.757

**Table 7 MSE of overall rating prediction**

Method	Product datasets		
	Hotel	Beer	Coffee
LRR	0.905	0.856	1.234
The proposed method	0.1456	0.1423	0.1904

Our method is compared with Wang’s method [17] based on the  $\Delta^2_{overallrating}$ . Table 7 presents the mean square errors of overall rating for the three data sets. As can be seen in the table, our results are comparable to Wang’s.

**Conclusion**

This paper dealt with three important sub-tasks of the opinion mining problem, that are (1) extracting aspects mentioned in the reviews of a product by using conditional probability of words, (2) inferring the user’s rating for each identified aspect based on Naïve Bayes classifier, (3) estimating the weight placed on each aspect by the users by using the occurrences of word that discuss the aspect within a review and the frequency of text sentences that discuss the same aspect across all reviews.

Our method does not require to know the overall ratings and is as not complicated as some other previous methods. However, it still works very well on real world datasets in comparison with other state of the art methods.

In the future, the problem of aspect mining from unlabeled data will be considered. In addition, the proposed model will be applied to other domains such as movie, digital camera businesses to validate its generalized effectiveness.

**Abbreviations**

pLSA: Probabilistic Latent Semantic Analysis; LDA: Latent Dirichlet allocation; HMM: Hidden Markov Model; CRF: Conditional Random Field.

**Authors’ contributions**

TNTN proposed method and performed experiments, HNTT supervised the programming and wrote draft manuscript. VAN wrote a part of the manuscript and corrected after received reviews. All authors read and approved the final manuscript.

**Author details**

<sup>1</sup> Department of E-Commerce, Vietnam Electric Power University, 235 Hoang Quoc Viet, Hanoi, Vietnam. <sup>2</sup> Institute of Information Technology, Vietnam Academy of Science and Technology, Hanoi, Vietnam.

**Acknowledgements**

This research is funded by the project “Building a System for Prediction and Management of Information Spreading in Social Networks in Vietnam” under Grant VAST01.01/17-18

**Competing interests**

Data mining, big data, machine learning and natural language processing.

**Funding**

Not applicable.

**Availability of data and materials**

All data used in this study are publicly available and accessible in the source Tripadvisor.com.

**Appendix**

See Tables 8, 9, 10.

**Table 8 Aspect word set of Hotel data**

Aspect	Aspect words
Value	Hotel, charge, cost, discount, dollars
Room	Bathroom, bathrooms, bed, beds, bath, floor, floors, chair, chairs, balcony, shower, lobby, noise, pool, queen, couple, Sheraton, coffee, desk, hotel, suite, tv, view, water, window, carpet, closet, doors, furniture, king, pillows, sink, toilet, tub, toiletries, ...
Location	Airport, area, center, downtown, hotel, market, place, places, restaurant, shop, shops, shopping, show review, street, view, views, neighborhood, square, waterfront, ...
Cleanliness	Hotel, floor, shelf, desk, chair, bag, door, lobby, stairs, ...
Check in/front desk	Desk, clerk, lounge, luggage, reception, checkout, ...
Service	Bar, bars, coffee, concierge, food, park, parking, restaurant, wine, buffet, ...
Business service	Tv, television, wireless, hotwire, cable, computer, connection, free, freeway, ...

**Table 9 Aspect word set of Beer data**

Aspect	Aspect words
Appearance	Black, Body, brown, bubble, bud, copper, lace, lacing, dots, drip, dust, back, finger, fizzy, fluff, golden, half, layer, orange, straw, surface, top, white, yellow...
Aroma	Bacon, banana, basil, caramel, cheese, cream, dry citrus, fruitiness, honey, light, malt, malts, meat, mint, nose, pear, perfume, pill, pine, roast, sandalwood, smoke, smoky, spice, sweet, sweetness, yeast...
Palate	Alcohol, body, carbonation, cream, Drinkability, dry, hoppy, light, round, spring, summer, ...
Taste	Alcohol, avalanche, balance, bitterness, body, bread, burn, caramel, carbonation, cheese, chocolate, clear, cocoa, coffee, complexity, flavors, flavors, fruit, fruitiness, ginger, grains, malt, matiness, meat, Medium-dry, oats, pear, roast, smoke, smoothness, spring, subtleties, summer, sweet, throat, toffee, tongue, vanilla, wood, ...
Overall	Beer, beers, bottle, drink, beverage, style, glass, sense, quality, brewery, level, lace, brewpub ...

**Table 10 Aspect word set of Coffee data**

Aspect	Aspect words
Aroma	Bran, brew, butter, charr, chocolate, citrus, fruit, honey, love, lover, organic, press, quality, smooth, lemon, smoke, stuff, ...
Taste	Bitter, bitterness, chocolate, honey, salt, freshness, brew, love, lover, mild, organic, press, quality, roaster, smooth, soft, sour, stuff, sweet, sweetness, syrup, ...
Acidity	Acidic, acidness, sourness, ...
Body	Love, press, smooth, richness, thick, thin, soft, ...

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 12 December 2017 Accepted: 14 February 2019

Published online: 28 February 2019

## References

- Park S, Lee K, Song J. Contrasting opposing views of news articles on contentious issues. In: Proceedings of the 49th annual meeting of the association for computational linguistics (ACL-2011). 2011.
- van den Camp M, van den Bosch A. The socialist network. *Decis Support Syst.* 2012;53:761–9.
- Li SK, Guan Z, Tang LY, et al. Exploiting consumer reviews for product feature ranking. *J Comput Sci Technol.* 2012;27(3):635–49. <https://doi.org/10.1007/s11390-012-1250-z>.
- Lin C, He Y, Everson R, Ruger S. Weakly supervised joint sentiment-topic detection from text. *IEEE Trans Knowl Data Eng.* 2012;24(6):1134–45.
- Zhan J, Loh HT, Liu Y. Gather customer concerns from online product reviews—a text summarization approach. *Expert Syst Appl.* 2009;36:2107–15.
- Dang Y, Zhang Y, Chen H. A lexicon-enhanced method for sentiment classification: an experiment on online product reviews. *IEEE Intell Syst.* 2010;25(4):46–53.
- Pang B, Lee L. A sentiment education: sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the 42nd annual meeting on association for Computational Linguistics. 2004. p. 271.
- Taboada M, Brooke J, Tofloski M, Voll K, Stede M. Lexicon-based methods for sentiment analysis. *Comput Linguistics.* 2011;37(2):267–307.
- Turney PD. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: ACL '02 Proceedings of the 40th annual meeting on association for computational linguistics. p. 417–24.
- Hu M, Liu B. Mining and summarizing customer reviews. In: Proceedings of the Tenth ACM SIGKDD international conference on knowledge discovery and data mining, KDD'04, New York: ACM; 2004. p. 168–77.
- Liu B. Sentiment analysis and opinion mining. *Synth Lect Human Lang Technol.* 2012;5(1):1–67.
- Popescu AM, Etzioni O. Extracting product features and opinions from reviews. In: HLT '05 Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing. p. 339–46.
- Zhu J, Wang H, Tsou BK, Zhu M. Multi-aspect opinion polling from textual reviews. In: Proceedings of ACM international conference on information and knowledge management (CIKM-2009). 2009.
- Long C, Zhang J, Zhut X. A review selection approach for accurate feature rating estimation. In: Proceedings of Coling 2010: Poster volume. 2010.
- Moghaddam S, Ester M. Opinion digger: an unsupervised opinion miner from unstructured product reviews. In: Proceeding of the ACM conference on Information and knowledge management (CIKM-2010). 2010.
- Chen Li, Wang Feng. Preference-based clustering reviews for augmenting e-commerce recommendation. *Knowl Based Syst.* 2013;50:44–59.
- Wang H, Lu Y, Zhai C. Latent aspect rating analysis on review text data: a rating regression approach. In: Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining, KDD'10, New York: ACM; 2010. p. 783–92.
- Ravi K, Ravi V. A survey on opinion mining and sentiment analysis: tasks, approaches and applications. *Knowl Based Syst.* 2015;89:14–46.
- Santorini B. Part-of-speech tagging guidelines for the Penn Treebank Project, University of Pennsylvania, School of Engineering and Applied Science, Dept. of Computer and Information Science. 1990.
- Cilibrasi RL, Vitanyi PMB. The google similarity distance on Knowledge and Data Engineering, *IEEE transactions.* 2007; 370–83.
- Hofmann T. Probabilistic latent semantic indexing. In: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information SIGIR'99. New York: ACM; 1999. p. 50–7.
- Blei DM, Ng AY, Jordan MI. Latent dirichlet allocation. *J Mach Learn Res.* 2003;3:993–1022.
- Brody S, Elhadad N. An unsupervised aspect-sentiment model for online reviews. In: Human language technologies: the annual conference of the north american chapter of the association for computational linguistics, HLT'10, Stroudsburg; 2010. p. 804–12.
- McAuley J, Leskovec J, Jurafsky D. Learning attitudes and attributes from multi-aspect review. In: International conference on data mining (ICDM). 2012.
- Sauper C, Barzilay R. Auto-matic aggregation by joint modeling of aspects and values. *J Artif Int Res.* 2013;46(1):89–127.
- Li H, Lin R, Hong R, Ge Y. Generative models for mining latent aspects and their ratings from short reviews. In: 2015 IEEE international conference on data mining. p. 241–50.
- Ding X, Liu B, Yu PS. A holistic lexicon-based approach to opinion mining. In: Proceedings of the conference on web search and web data mining (WSDM-2008). 2008.
- Kim SM, Hovy E. Determining the sentiment of opinions. In: Proceedings of international conference on computational linguistics (COLING-2004).
- Yan Z, Xing M, Zhang D, Ma B. EXPRS: an extended pagerank method for product feature extraction from online consumer reviews. *Inf. Manage.* 2015.
- Penalver-Martinez I, Garcia-Sanchez F, Valencia-Garcia R, Rodriguez-Garcia MA, Moreno V, Fraga A, Sanchez-Cervantes JL. Feature-based opinion mining through ontologies. *Expert Syst Appl.* 2014;41(13):5995–6008.

31. Manek AS, Shenoy PD, Mohan MC, Venugopal KR. Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier. *World Wide Web*. 2017;20(2):135–54.
32. Pham DH, Le AC. Learning multiple layers of knowledge representation for aspect based sentiment analysis. *Data Knowl Eng*. 2017;114:26–39.
33. Dao TT, Thanh TD, Hai TN, Ngoc VH. Building Vietnamese topic modeling based on core terms and applying in text classification. In: *Proc. of the fifth IEEE international conference on communication systems and network technologies*. 2015. P. 1284–88.
34. Yu J, Zha ZJ, Wang M, Chua TS. Aspect ranking: identifying important product aspects from online consumer reviews. In: *Proceedings of the 49th Annual meeting of the association for computational linguistics: human language technologies*. Volume 1, HLT'11, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. p. 1496–505.
35. Archak N, Ghose A, Ipeirotis PG. Show me the money!: Deriving the pricing power of product features by mining consumer reviews. In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD'07, New York: ACM; 2007. p. 56–65.

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

---

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)

---