A Chinese Product Feature Extraction Method Based on KNN Algorithm

Biao Ma, Haiyan Chen

The Glorious Sun School of Business and Management, Donghua University, Shanghai, China
Email: 2160916@mail.dhu.edu.cn

Abstract
The product feature set of online reviews obtained by the current product feature extraction methods has a low coverage rate of review information. In order to solve this problem, this paper proposes a method of product feature extraction based on KNN algorithm. We establish the classification system of product feature set firstly. Then we extract part of product features as training set manually, and according to similarity between words and the classification system, the product features of all reviews are quickly classified and extracted. At last, the PMI algorithm is used to filter and supplement it to improve the correct rate and the review information coverage rate of product feature set. Through the examples of online clothing reviews data in the Taobao platform, we prove that this method can effectively improve the review information coverage rate of product feature set.

Keywords
Product Features, KNN Algorithm, Review Information Coverage Rate, Online Reviews

1. Introduction
With the rapid development of the Internet economy, the amount of data on the network review is growing. The huge amount of comment data makes it necessary for consumers to spend a lot of time to find interesting contents. Comment mining is an important tool for dealing with massive reviews [1]. And product feature extraction is the core part of Reviews’ mining. For taking advantage of the full information of comment features for further analysis, artificial extraction of product features is accurately and completely from reviews, but it will take a lot of time and efforts. Therefore, this paper first manually extracts part of the product features and then uses the KNN algorithm to extract all product feature
from reviews. This method can obtain higher product feature information at less time and labor cost.

Now the methods of product feature extraction are mainly divided into two categories: manual definition and automatic extraction. Song X. L., Wang S. G. and Li H. X. used supervised learning method to extract product features. Firstly, the text pattern defined manually constitute a collection of seeds, and then the product name and product characteristics are extracted by Bootstrapping algorithm [2]; Yao T. F., Nie Q. Y., Li J. C., Li L. L., Lou D.C., Chen K. et al. used ontology knowledge to establish the field of automotive product features set [3]; Titov, Ivan, McDonald and Ryan proposed a multi-granularity LDA model that avoids the problem that the topic meaning in the LDA model cannot be determined and applied it to the subject’s emotional digest generation system [4]; Liu, J., Wu, G. and Yao, J. proposed the extraction of opinion examples and established relevant domain knowledge to complete the simultaneous extraction of feature items and emotion words [5]; Hu M and Liu B. firstly proposed the association rule classification method-Apriori algorithm to extract the candidate set of product features, and then extracts the product features in English reviews [6]; Li S., Ye Q., Li Y. J. and Rob L. referred to Hu et al. research and proposed product feature extraction method of Chinese text review based on Apriori algorithm according to the Chinese language characteristics [7] [8]. To reduce the information noise generated by Li et al. method, Wang Y., Zhang Q. and Yang X. J. put forward a FP increasing algorithm which is much higher than Apriori algorithm. He considered the semantic relation between product and attribute to filter the candidate product feature set and then improved the accuracy rate [9]. The above methods did an improvement in the accuracy and recall rate, but these methods didn’t consider that the extracted product features cover the rate of comment information. This paper proposes a method of product feature extraction based on KNN algorithm, which can effectively improve the product feature set’s coverage rate of review information based on ensuring correct rate.

2. Proposed Method

This paper proposes a method of product feature extraction based on KNN algorithm, which can effectively improve the product feature set’s coverage rate of review information. Firstly, we extract part of product features as training set manually, and according to similarity between words and the classification system, the product features of all reviews are quickly classified and extracted. Finally, the PMI algorithm is used to filter candidate product feature set and get the final product features.

2.1. Establishing Product Characteristic Classification System

In this paper, the product features are extracted from the product’s indexes defined by the manufacturer. First, this paper establishes the product characteristic classification system M. The product characteristic classification system M in this paper is only the characteristic system of the product itself. Of course, it can
also establish the characteristic system for other aspects (such as logistics, service and so on) and the process is same. The product characteristic classification system divides into two levels system. The first level is the product classification $P$. A product category can be an indicator, and multiple product classes can be merged into one indicator according to the specific business scenarios. The second level is the product’s indexes of manufacturer’s definition $S$, $S = \{s_1, s_2, \ldots, s_n\}$, where $n$ represents the number of indicators. We assume that the user’s descriptions of the product features are within the scope of the vendor’s description. If the customers’ comments about the product characteristics are not within the scope of the firm’s defined indexes when the training set is set up by using the KNN method, the second-level indexes will be supplemented accordingly. Take coat and trouser as an example, according to the manufacturer’s definition indexes including color, size and so on, so we can get the characteristic classification system of this product, as shown in Figure 1.

2.2. Extracting Product Feature Sets Based on KNN Algorithm

After establishing the classification system of product features, the product feature set is extracted by using the KNN algorithm. The central idea of the KNN algorithm is that if the majority of the $k$ most adjacent samples of one sample in the feature space belong to a particular category, this sample also belongs to this class. First, we classify part of the candidate feature sets manually according to the classification system, and classify the unclassified candidate feature sets by computing the similarity between the unclassified candidate product features and the already classified features. The feature set of the product is characterized by the noun. After the noun is extracted, the training set and the test set are established according to the product characteristic classification system $M$. We use the synonym forest of Harbin University of Engineering to calculate the similarity between words and class and extract product features. The specific process is as follows:

![Figure 1. Product characteristics classification system.](image-url)
The first step: word segment and POS tagging

We firstly use the Chinese Institute of Computing Technology Institute Chinese segmentation tool ICTCLAS for word segmentation and POS tagging of the original review corpus before feature extraction to build the set related with noun including {/an, /ng, /n, /nr, /ns, /nt, /nz, /vn}. This paper selects set N_0 of the nouns and gerunds {/n, /vn} for the subsequent experimental verification and then filter the stop words to get the candidate feature set I_0.

The second step: manually labeling training set and test set respectively

We randomly select some items from the candidate feature set I_0 = {word_1, word_2, ..., word_m} (M is the number of words in the candidate feature set) as training set I_1 = {word_1, word_2, ..., word_a} (a is the number of words in the training set), i.e. I_1 \in I_0 and i < m. According to the classification system of product characteristics established by 2.1, I_1 can be classified manually, so we can obtain:

\[ P = \{s_i(word_{i1},...,word_{im}),...,s_n(word_{n1},...,word_{nm})\} \]

where word_{im} denotes the m-th word of the i-th index and m_i denotes the number of words contained in the i-th index. Example 1: Selected training set I_{sam-1} = {Chi Cun (尺寸), Bu Liao (布料), Yan Se (颜色), Jia Wei (价位), Hao Ma (号码), Mian Liao (面料), Shi Hui (实惠), Jia Qian (价钱), Liao Zi (料子), Se Cha (色差), Xing Jia Bi (性价比)}. According to the previously established product characteristics classification system P, I_{sam-1} can be classified manually as shown in Table 1.

We also randomly select some items from the initial candidate feature set I_0 as test set I_2 = {word_{a1}, word_{a2}, ..., word_{ab}} (b is the number of words in the test set), i.e. I_2 \in I_0 and a + b < m. According to the classification system of product characteristics established by 2.1, I_2 can be classified manually. Example 1: Selected test set I_{sam-2} = {Cuo Ma (错码), Bao Lan Se (宝蓝色), Ku Liao (裤料), Biao Jia (标价), Zong Se (棕色), Ma Zi (码子), Yuan Jia (原价), Qian Se (浅色), Mian Liao (面料), Hou Liao (厚料)}. According to the previously established product characteristics classification system P, I_{sam-2} can be classified manually as shown in Table 2.

The Third step: Using the synonyms of Harbin Institute of Technology to classify words

Table 1. Manually classification results of training set in Example 1.

<table>
<thead>
<tr>
<th>Level1 index</th>
<th>Level 2 index</th>
<th>Product features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coat</td>
<td>Size</td>
<td>Chi Cun (尺寸), Chi Ma (尺码), Ma Shu (码数)</td>
</tr>
<tr>
<td>Fabric</td>
<td>Bu Liao (布料), Mian Liao (面料), Liao Zi (料子)</td>
<td></td>
</tr>
<tr>
<td>Prize</td>
<td>Jia Wei (价位), Jia Qian (价钱), Xing Jia Bi (性价比), Shi hui (实惠)</td>
<td></td>
</tr>
<tr>
<td>Color</td>
<td>Yan Se (颜色), Se Cha (色差)</td>
<td></td>
</tr>
</tbody>
</table>
Table 2. Manually classification results of test set in Example 1.

<table>
<thead>
<tr>
<th>Level 1 index</th>
<th>Level 2 index</th>
<th>Product features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coat</td>
<td>Ma Zi (码子), Cuo Ma (错码)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ku Liao (裤料), Hou Liao (厚料), MianLiao (棉料)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Biao Jia (标价), Yuan Jia (原价)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bao Lan Se (宝蓝), Qian Se (浅色), Zong Se (棕色)</td>
<td></td>
</tr>
</tbody>
</table>

After setting up training sets $I_1$, we calculate the similarity $d$ between the words $word_i$ in the test set $I_2$ and the words $word_{im_j}$ in the training set $I_1$. The calculation of similarity is as follows:

$$d_{im_j} = Sim(word_{im_j}, word_i)$$

where $word_{im_j} \in I_1$, $word_i \in I_2$, $d_{im_j}$ indicates the similarity between $word_{im_j}$ and $word_i$, $Sim(a, b)$ indicates the similarity between word $a$ and word $b$. If the maximum similarity between the word $word_i$ and all the words in the training set is greater than the corresponding threshold value $f$, i.e. $\max(d_{im_j}) > f$, the words $word_i$ belongs to the product feature words and the index $i$ of the word $word_i$ is same with the index of the word $word_{im_j}$ in the training set that have the maximum similarity with the word $word_i$. If the maximum similarity between the word $word_i$ and all the words in the training set is lower than the corresponding threshold value $f$, i.e. $\max(d_{im_j}) < f$, the word $word_i$ does not belong to the product feature words. The method is used to judge whether each word in the test set belongs to the product feature set. If it belongs to the product feature word, this method can be classified directly. Then we can get the initial product feature set $I_3$.

In this paper, we use the similarity calculation method based on synonym word forest proposed in [10] to calculate the similarity between words. The similarity of words is expressed by $Sim$, and is calculated as follows:

1) If the two items are not on the same tree

$$Sim(A, B) = 0.1$$

2) If the two items are on the same tree
a) If the two items are on the second branch

$$Sim(A, B) = 0.65 \times \cos \left( n \times \frac{\pi}{180} \left( \frac{n-k+1}{n} \right) \right)$$

b) If the two items are on the third branch

$$Sim(A, B) = 0.8 \times \cos \left( n \times \frac{\pi}{180} \left( \frac{n-k+1}{n} \right) \right)$$
c) If the two items are on the fourth branch

$$Sim(A, B) = 0.9 \times \cos \left( n \times \frac{\pi}{180} \left( \frac{n-k+1}{n} \right) \right)$$
d) If the two items are on the fifth branch
Sim(A, B) = 0.96 × \cos \left( n \times \frac{\pi}{180} \left( \frac{n-k+1}{n} \right) \right)

where n is the total number of nodes in the branch layer, and k is the distance between the two branches.

Through the above method, the test set \( I_2 \) uses the training set \( I_1 \) to classify. By modifying the training set, we classify the candidate feature sets \( I_0 \) and then generate the initial product feature set \( I_1 \). In example 1, the classification results by using the synonyms of Harbin Institute of Technology are consistent with those in the second step.

2.3. Filtering \( I_3 \) by the PMI Algorithm

The 2.2 Section used the KNN algorithm to extract and classify the features according to the similarity between words. Though the above method can quickly extract product information, the similarity of some words and words in the training set is very high and these words do not belong to product features. In the Example 1, the similarity between the word “chi cun” and the word “shen cai” is 0.99999, which exceeds the threshold of 0.9999, so the word “shen ti” belong to level 2 indicator “size”. In fact, the word “shen ti” does not belong to level 2 indicator “size”. So to improve the accuracy of product features, this paper uses PMI algorithm to filter \( I_3 \).

Popescu, AnaMaria, Etzioni and Oren are based on the following hypothesis: the more frequent co-occurrence of two words, the higher the correlation between them, so Popescu proposed a point mutual information PMI method to extract product features [11]. Wang et al use the PMI algorithm based on Web search engine to filter product features. PMI values are calculated as follows.

$$PMI(\text{level 1 indicator, level 2 indicator}, I_3) = \log_2 \frac{\text{hit (“level 1 indicator” and “I_3” and “level 2 indicator”)}}{\sqrt{\text{hit (“level 1 indicator”)hit (“level 2 indicator”)hit (“I_3”)}}}$$

where hit (x) is the number of pages returned by the search engine using the “x” as keywords, and Hit (x and y) is the number of pages returned at the same time using “x” and “y” as keywords.

The subject of this paper is the product features in the review; we first consider the covariance of the two words based on a large number of comment sets. This paper calculates co-occurrence degree between the second indicators and the initial feature set \( I_3 \), in which hit (x) is the number of comments returned when x is the keywords, and hit (x and y) is the number of comments returned for x and y at the same time as keywords. However, the result is not satisfactory. We illustrate the reason by giving an example, the data set in this article is comments set of one clothing brand, therefore, critics will rarely comment on level 2 indicator “chi cun” and the feature “chi ma” at the same time. Finally, this paper uses the PMI algorithm based on Web search engine to filter product features, and then gets the final product feature set \( I_4 \).
This paper computes the co-occurrence degree -PMI values between level 1 indicators, level 2 indicators and the initial feature set $I_3$ to filter the initial product feature set $I_3$ based on Web search engine, and sorts PMI value. We set the corresponding threshold to filter the initial product feature set $I_3$, and get the final product feature set $I_4$.

2.4. Research Process

The specific process is as follows:

1) Establishing the product characteristic classification system $M$ based on manufacturer defined product metrics.

2) The extraction of product feature sets based on KNN algorithm mainly includes the following steps:

   a) Use word segmentation tool for segment comments and part of speech tagging;

   b) Filter the stop words and punctuation marks, extract the noun and gerund in the comment and get the candidate feature set $I_0$;

   c) Establish and classify training set $I_1$ manually according to product feature classification system and $I_0$;

   d) Build the test set $I_2$, and manually classify $I_2$;

   e) Use the KNN classification algorithm, calculate the distance between words by the synonyms forest and classify $I_2$ by setting the threshold;

   f) Classify the candidate product feature set $I_0$ using KNN algorithm after getting the highest accuracy of the test set $I_2$ by constantly modifying the training set.

3) Filtering the product feature sets of each classification based on PMI algorithm

Compute the co-occurrence degree -PMI values between level 1 indicators, level 2 indicators and the initial feature set $I_3$ using the PMI algorithm based on Web search engine and sorts PMI value. We set the corresponding threshold by to filter the initial product feature set $I_3$, and get the final product feature set $I_4$.

3. Experimental Results and Performance Assessment

3.1. Data Set

This paper uses the python web crawler from Tian Mao website (https://www.tmall.com) to crawl review information of one clothing brand from December 2016 to February 2017. The review information includes the user’s Name, date of comment, comment content. After establishing the product characteristic classification system $M$, we select part of the candidate feature set $I_8$ as $I_1$. We manually classify the training set $I_1$ according $M$. The training set accounts for 30% of the candidate feature set and the test set accounts for 30% of the set of candidate product features. Training set and test set contains the word information as shown in Table 3. Artificially classified information of the training set is shown in Table 4.
Table 3. The lexical information contained in the training set and the test set.

<table>
<thead>
<tr>
<th>Set</th>
<th>The number of words</th>
</tr>
</thead>
<tbody>
<tr>
<td>the training set</td>
<td>365</td>
</tr>
<tr>
<td>the test set</td>
<td>365</td>
</tr>
<tr>
<td>the initial candidate feature set</td>
<td>1219</td>
</tr>
</tbody>
</table>

Table 4. Artificially classified information of the training set.

<table>
<thead>
<tr>
<th>Level 1 index</th>
<th>Level 2 index</th>
<th>Artificially classified product features</th>
<th>The number of Artificially classified product features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cutting</td>
<td></td>
<td>XianTou (线头)</td>
<td>1</td>
</tr>
<tr>
<td>Color</td>
<td></td>
<td>Yan Se (颜色), Se Chan (色差), Lan Se (蓝色), Shen Se (深色), Jun Ly Se (军绿色), Hui Se (灰色)</td>
<td>6</td>
</tr>
<tr>
<td>Designing</td>
<td></td>
<td>Kuan Shi (款式), Ban Xing (版式), Ku Xing (裤型), She Ji (设计), Yang Shi (样式), Feng Ge (风格), Yang Zi (样子), Zhi Tong (直筒), Ha Lun (哈伦)</td>
<td>9</td>
</tr>
<tr>
<td>Dress feeling</td>
<td></td>
<td>Shen Shi Du (舒适度), Fang Feng (防风)</td>
<td>2</td>
</tr>
<tr>
<td>Size</td>
<td></td>
<td>Chi Ma (尺码), Ma (码), Ma Shu (码数), Chi cun (尺寸), Chang Du (长度), Pang (胖)</td>
<td>6</td>
</tr>
<tr>
<td>Quality</td>
<td></td>
<td>Zhi Liang (质量), Pin Zhi (品质)</td>
<td>2</td>
</tr>
<tr>
<td>True and false</td>
<td></td>
<td>Zheng Pin (正品), Jia (假)</td>
<td>2</td>
</tr>
<tr>
<td>Elasticity</td>
<td></td>
<td>Tan Xing (弹性)</td>
<td>1</td>
</tr>
<tr>
<td>Thickness</td>
<td></td>
<td>Bao Kuan (薄款), Hou Du (厚度)</td>
<td>2</td>
</tr>
<tr>
<td>The feeling of touch</td>
<td></td>
<td>Shou Gan (手感)</td>
<td>1</td>
</tr>
<tr>
<td>Fabric</td>
<td></td>
<td>Qian Liao (面料), Liao Zi (料子), Chun Mian (纯棉), Quan Mian (全棉), Mian Zhi (棉质), Cai Zhi (材质), Jie Shi (结实)</td>
<td>7</td>
</tr>
<tr>
<td>Prize</td>
<td></td>
<td>Jia Ge (价格), Shi Hui (实惠), Xing Jia Bi (性价比)</td>
<td>3</td>
</tr>
<tr>
<td>Smell</td>
<td></td>
<td>Wei Dao (味道), Ma Xiang (木香)</td>
<td>2</td>
</tr>
<tr>
<td>Packing</td>
<td></td>
<td>Yin Z (印子), Bao Zhuang Zhi (包装纸)</td>
<td>2</td>
</tr>
<tr>
<td>Total number</td>
<td></td>
<td></td>
<td>46</td>
</tr>
</tbody>
</table>

3.2. Performance Evaluation

The goal of this paper is to improve the ratio of the coverage information in the reviews, so the performance evaluation indexes selected in this paper are information coverage rate C and accuracy rate Z.

Each comment information coverage $C_i$ is calculated as:

$$C_i = \frac{X}{Y}$$
where $X$ represents the number of feature words extracted from each comment and $Y$ indicates the number of candidate feature words $I_0$ contained in each comment.

The weight of each comment $A_i$ is the ratio of the number of words in each comment to all words of the total comments and $n$ is the number of reviews.

The total coverage rate $C$ can be expressed as:

$$C = \sum_{i=1}^{n} (A_i \times P_i)$$

The accuracy rate $Z$ is calculated as follows:

$$Z = \frac{A}{A + B}$$

where $A$ is the number of the correct product feature of identification, and $B$ is the number of the wrong product feature of identification [1] [8] [9] [10].

### 3.3. Experimental Results

Taking the comment data of the 3.1 Section as the research object, the FP-Growth algorithm proposed in [10] is used as the contrast object. We compare the difference of the accurate rate and information coverage ratio between the proposed method and the FP-Growth algorithm. According to the indexes of performance evaluation in the 3.2 Section, we calculate the performance indicators of the two methods as shown in Figure 2. We can see in the Figure 2 that the method proposed in this paper has improved the accuracy and information coverage indicators compared with the FP-Growth method.

As can be seen from the Figure 2, the information coverage by the proposed method and the method is no more than 20%. We analyze the reason why the information coverage rate is relatively low is that: 1) in the real comment, the customers are involved in the dimensions such as: services, purchase in addition to the description of the product itself; 2) the part of speech (POS) selected product features is adjectives besides nouns and gerund. For example, this dress is very cheap. Although the word “cheap” is adjective, but it is a description of the price of the product. It belongs to the product feature. This paper further
analyzes the reasons for the differences in the product feature extraction process as follows:

1) Due to the imperfection of the word segmentation tools, the result of the word segmentation is biased;

2) With the rapid development of the Internet, a lot of new words have been produced in a short time, and the synonyms lexicon proposed by HIT may not update the thesaurus in time.

3) Because of the complexity of the Chinese language, the same language in different contexts may have different meanings, and the method of calculating the similarity of words also needs to be further improved.

4. Conclusion

Product feature extraction is an important research area in mining reviews. In this paper, a method based on KNN product feature is proposed from the point of view of improving information coverage. Experiments based on real comments show that this method is effective and can improve information coverage in some degree. The focus of the following work is to analyze the sensitivity of the ratio of training set to the sample set. Under the premise of guaranteeing the correct rate and extracting the product characteristic information coverage rate, the workload of manual participation in the training set is minimized.

References


