An Implicit Emotion Mining Method of User Consultants-oriented

Yan-ping ZHAO¹,a, Fang WANG²,b and Wen-xing FAN³,c

¹,²,³Information Center, Beijing Jiao Tong University, Beijing, China

zaoh@bjtu.edu.cn, bfangw@bjtu.edu.cn, cwxfan@bjtu.edu.cn

Keywords: Text categorization, Semi-supervised, Feature selection.

Abstract. Aiming at the characteristics of short texts which are sparse, nonstandard and ambiguous in subject, we present an effective classification method. This paper analyzes this kind of data, and utilize semi-supervised to select significant syntactic features (substring/subsequence), and then uses SVM for text categorization. A machine-learning based emotion analysis method is implemented to mine such implicit emotion. The average accuracy rate and recall rate can reach 84.19%, At last, the method is proven effective by applying in the veritable data of telecommunication field.

Introduction

Related Concepts

With the rise of the Internet, more and more businesses are concerned about users' consultation and comments on their products, so as to conduct user satisfaction analysis, orientation analysis, and user correlation analysis and give better decision support to businesses. It is a kind of text classification method to analyze what kind of products or problems the user's inquiry is aiming at. In the field of telecommunications, users' inquiries are all within 160 words, which are called short texts. This paper classifies these short texts on topics, measures the features of substrings and sub-sequences, and USES SVM classifier to realize classification.

There are two methods for the classification of common short texts. One is to use search engines to extend the short texts [1]. This classification method only extends the text, but it does not deal well with invalid information and noise, and it takes a long time. Second, on the basis of the features of the short text, the semantic relationship between words can be found by using the knowledge base [2]. For example, Wikipedia uses semantic features for classification. Another classification method is to use the topics hidden in the short text as an additional feature set, which can be used for training and classification together with the original features [3]. All the above studies use external relevant data for feature expansion or similarity calculation, and have achieved good results. However, there is a high demand for the correlation of external data, which is usually collected according to the domain correlation under the condition of human intervention. It is difficult to obtain external knowledge of relevant fields in practical application. This paper gets rid of such external dependence and proposes PLWAP algorithm based on web pattern sequence [4] to obtain frequent substrings and sub-sequence features for classification in specific fields. After experiments, good results are obtained.

PLWAP Algorithm based on Web Pattern Sequence

Sequence pattern mining is an important category in data mining, which refers to the frequent occurrence of sequences or events. PLWAP [5] algorithm is based on WAP algorithm [6] to mine the suffix Tree of a project, while wap-tree algorithm is to mine the prefix Tree of a project. Each node of plwap-tree is an independent code. Therefore, the construction condition wap-tree is avoided. The algorithm is described as follows:

Algorithm: PLWAP - Tree
Input: web access sequence database WASD;
Support threshold $\lambda$ ($0 < \lambda \leq 1$);
Frequent event set L;
Output: PLWAP - Tree
Methods:
1) Scan the original data to get frequent items FE set
2) Scan each event in FE again
   if current tree node has children
      if event is one of the child:
         the count of current node = count + 1
      else:
         insert into tree as a new node
         the count of new node = 1
         position code = rightmost child node position code + '0'
   else:
      insert into tree as a new node;
      the count of new node = 1
      position code = parent node position code + '1'

After that, plwap-tree is mined to output all frequent patterns that meet the minimum support threshold. It uses the Header table of the isomorphic node queue position of isomorphic node in the tree with the binary code to find all the suffix tree root node of the mining. If the root node set is greater than $\lambda$, the total of the isomorphic nodes are frequent events. And then add these frequent events to the previous step to obtain frequent events which are a set of new, length add 1 frequent events. We keep recursively mining until we find all the frequent sequences. This method is used to obtain all frequent substrings and subsequences.

**Classification Method of Consulting Articles**

**Research Background and Significance**

This paper relies on the project "data mining in the field of telecommunications", and the data set is based on the real data of a customer service platform of the mobile company. The platform receives real-time consultation from users on various services provided by operators through mobile phone/Web every day, and automatically responds to all questions. We collected data for four months, and there were millions of texts. We divided all texts into four categories: introduction, opening, usage and cancellation. The introduction class refers to the user's understanding of the business, the opening class refers to the user's consultation on how to open or in the process of opening, and the usage class refers to the user's consultation after opening. In addition, discount consulting and comparison consulting are unified into the usage class. The cancel class is a consultation on how to cancel. The examples of ringtone service are as follows:

| Consulting | Subject category |
|------------|-----------------|
| How much do you charge for the ring tones? | introduction |
| How to open the ring bells? | opening |
| I cannot download ring tones. | usage |
| Is there any discount for ring tones? | usage |
| What is the difference between ring and ringtone gagarin? | usage |
| How to cancel ring tones? | cancellation |

The objective is to analyze each user’s no-structured text. Structured information is used to support enterprise further statistical analysis of the different business. It is divided into four tasks: to identify user’s emotion tendency, to analyze the service, to analyze the theme of the information and extract a summary. For example:
For the first advisory statement in Table 1, the information extracted by the system is as follows:

- Emotional tendency: "non-negative emotion"
- Related business: "ring bells"
- Related topic: "open"
- Text abstract: "charge for ringtones"

This research mainly completes the third subtask.

**Classification Characteristics**

By observing characteristics of each type, we found that each of these categories will often appear some word or string, such as opening consultation appears "open", "how to open" frequently, cancel consultation often appear "cancel ", "how to cancel", fault consultation often appears "how to use", "why can't download". Similarly, other themes have the corresponding patterns. The patterns are text strings and subsequences, which are visually useful categorization features.

Because the accuracy and time efficiency of Chinese dependency grammar analysis are not high, this research does not analyze the sentence dependency grammar. We mainly use the first feature, named the syntactic feature. It can be divided into two categories: substrings which occur continuously in a statement and subsequences which occur sequentially in a statement.

1. **Substring**: a substring is a string formed by continuously occurring elements in a string. It can be divided into the following two types:
   a) n-gram: an n-gram substring (n=1, 2, 3) is a string of n words that appear continuously in a statement
   b) Frequent common substring: Frequent or consecutive strings of words that are formed by dividing statements into words. For example: my mobile phone 7610, how can I use the network? How can I open the Internet?. If the substring "how can I open" in other statements often appear, then the substring is a frequent common string.

2. **Frequent common subsequence**: Sequence consisting of one or more words orderly appear in the statement which is divided into words. The purpose of statement segmentation is to get frequent subsequence with clear meaning, so as to avoid generating subsequence composed of single words. For example, the sentence “Do you charge for opening the mobile network?” after word processing, the possible calculated frequent subsequence is: “open... Charge...”

**Feature Extraction**

For frequent substring and sub-sequence features, when the given support threshold is small, there may be a large number of redundant features. When the given support threshold is large, the features may not be suitable for the text class. Therefore, we need to make feature selection. The most commonly used feature measurement methods include selective Information Gain (IG), Chi-square statistics (CHI), Mutual Information (MI) and Document Frequency (DF) etc. Literature [7] selected highly indicative features for classification. Literature [8] used CHI for feature selection, and then predicted the short text with the LDA theme model to obtain the corresponding theme distribution, and then extended the theme words into the features of the short text for classification. Literature [9] established a rule base for classification as features. Literature [10] compared various methods of measuring features in Chinese text classification, and IG and CHI were the most effective methods for feature selection in their experiments. Several feature selection methods are also compared in this research. Limited by space, we will only introduce the definition of Information Gain:

Suppose text D has m categories $C_i$, $C_2... C_m$, $P(C_i)$ represents the probability of the occurrence of category $C_i$ without any given feature. $P(C_i|f)$ represents the probability of the occurrence of category $C_i$ with a given feature f, and then the information gain of a feature f is the entropy of the text with a given feature f and the entropy gain of the text without any given feature f, which is

\[
Gain(f) = Entropy(D) - Entropy(D|f) = -\sum_{i=1}^{m} P(c_i) \log P(c_i) + \sum_{i=1}^{m} P(C_i|f) \log P(C_i|f) + \sum_{i=1}^{m} P(C_i|\overline{f}) \log P(C_i|\overline{f})
\]

formalized as follows:

However, it is well known that some measures (such as CHI) are not accurate enough for
low-frequency words. On the other hand, we find that in a given field, the subsequence or substring frequently appeared in users’ consultation has strong semantics and reflects certain emotions. In order to extract these features, we proposed a semi-supervised method for feature selection (Chart 1), which includes the following two steps:

1) For marked and unmarked text, frequent mining algorithm is used to obtain frequent subsequence/substring to form the candidate feature set.

2) Calculate IG/MI/CHI on the marked text, select the feature with high value, and get the effective feature set.

Some typical features are shown in Table 2.

This feature selection method effectively uses a large amount of information in unmarked text, avoids the ignored low-frequency features caused by insufficient training corpus, and ensures the selection of features that up to a certain frequency level.

**Processing of Some Redundant Features**

Because for any frequent sub-sequence, any sub-sequence is still frequent. One of the problems with using frequent substring and sub-sequence features is the potential for redundancy. Similarly, every substring of frequent substring \( f_2 \) is still a frequent substring. Feature selection algorithms have been used to pick out features that are capable of categorizing text, so that some redundancy may have been eliminated. We use maximum matching to analyze the role of the redundant feature in this part:

When transforming from text to vector representation only the maximum feature is used. It is called maximum matching which is formalized as follows:

Sentence \( s \), denoted by \( k \) features \( f_1, f_2, \ldots, f_k \), then, \( \forall i, j \leq k, i \neq j, f_i \) is not a substring or subsequence of \( f_j \).

For example, the sentence: “The special offer for August is 150 yuan for 300 yuan free. I didn't receive complimentary benefits” which has already matched a feature “not receive complimentary benefits”. And then it matches a sub-sequence feature “special offer... for... not receive complimentary benefits”. It only matches the current maximum characteristics, that is, “special offer... for... not receive complimentary benefits”. When the text is transformed into an eigenvector, all the features are maximal and there is no inclusion relationship between them.

![Chart 1. Based on semi-supervised feature selection.](image)
Table 2. Examples of typical features.

| Numbers | Features                      | Types         |
|---------|-------------------------------|---------------|
| f1      | open                          | 1-gram        |
| f2      | surf Internet                 | 2-gram        |
| f3      | data flow charge              | 3-gram        |
| f4      | How to open                   | frequent sub-string |
| f6      | 12593 17951                   | frequent sub-sequence |
| f7      | Cancel…charge…still deducted  | frequent sub-sequence |

Algorithm Experimental

Experimental Results

In this research, 150,000 short messages are used as experimental corpus, each of which contains part of speech tagging information and business classification information. 10,000 pieces were selected manually as training corpus, 5,000 pieces as closed test corpus, and the rest as open test corpus. We use two existing open source algorithms: the word segmentation algorithm is called ICTCLAS which was developed by the institute of computing technology of the Chinese academy of sciences. The frequent subsequence mining algorithm is called the WAP tree which is proposed by C.I. Ezeife et al.

We use the SVM classifier (libsvm developed by C.-J. Lin\[11\]), which is widely used in text classification. The soft margin parameter is set to the default value of 1. The experimental steps are as follows:

a) Preprocessing: remove stop words (such as "I", "and", "is", etc.) and divide each user text into sentences;
b) Generate 1-gram, 2-gram and 3-gram for the marked text;
c) Word segmentation is carried out based on unmarked text and marked text, and then frequent subsequence and frequent subsequence are calculated, in which the minimum support threshold is set to 20/10000;
d) Calculate IG and make feature selection with all the features obtained by b and c;
e) Transform the statement into a feature vector: each dimension of the vector corresponds to a feature. When the statement has a feature, the corresponding value of the feature on the vector is 1; otherwise, it is 0;
f) Training of SVM classifier.

In the above experiments, the support threshold of frequent subsequence and frequent subsequence can be determined by cross validation. We conducted three groups of comparative experiments: The first is the performance comparison of maximum matching and non-maximum matching using tenfold cross validation on the training corpus (Table 3); The second is the performance comparison between using substring features only and adding sub-sequence features (Table 4); The last is the performance comparison of classifiers with IG, MI and CHI algorithm (Table 5). As the proportion of each category in the experiment is different, the recall rate and overall accuracy rate (recall rate) of each topic category are listed in our experiment. Performance indicators are defined as follows:

Recall rate of theme A = correctly identify the number of subjects A/ total number of theme A texts

Overall accuracy/recall rate= correctly identify the number of texts/ total number of texts

Table 3. The performance comparison of two matching methods using enfold cross validation(%).

| matching methods | opening | canceling | introduction | usage | average |
|------------------|---------|-----------|--------------|-------|---------|
| maximum matching | 89.52   | 91.25     | 78.60        | 74.50 | 83.47   |
| non-maximum matching | 89.92   | 91.40     | 78.95        | 75.10 | 83.84   |
Table 4. The performance comparison between using substring features only and adding sub-sequence features(%).

| matching methods                  | opening | canceling | introduction | usage | average |
|-----------------------------------|---------|-----------|--------------|-------|---------|
| substring features only           | 89.60   | 92.10     | 70.69        | 72.34 | 81.18   |
| substring and sub-sequence features | 89.96   | 92.73     | 72.56        | 74.30 | 82.39   |

Table 5. The performance comparison of classifiers with IG, MI and CHI algorithm(%).

| matching methods | opening | canceling | introduction | usage | average |
|------------------|---------|-----------|--------------|-------|---------|
| IG               | 90.12   | 92.35     | 78.95        | 75.33 | 84.19   |
| CHI              | 89.58   | 91.62     | 78.41        | 74.65 | 83.57   |
| MI               | 89.20   | 91.34     | 78.10        | 74.26 | 83.23   |

Result Analyses

As can be seen from Table 3, the performance of maximum matching and non-maximum matching is almost equal. It can be seen that the effect of partial redundancy caused by the use of subsequence features on classifier performance can be ignored. Table 4 shows the performance after only using substring features and sub-sequence features. It can be seen that high performance can be achieved by only using substring features, and the performance is slightly improved after introducing frequent sub-sequence. In literature [8], various methods of measuring characteristics in Chinese text classification were compared. In their experiments, IG and CHI had the highest performance. Table 5 shows the classification results of the classifier trained by three measures in this paper. It can be seen that the performance differences among the three are small and the performance of IG is slightly higher.

From the experimental results, it can be seen that the classifier has a good overall classification ability, and the features of opening and canceling classes are obvious, so the recall rate is relatively high. However, the introduction and use classes have the characteristics of mutual confusion, so the recall rate of the two is relatively low. In addition, frequent sub-sequences or frequent substrings with high IG values indicate that they occur frequently in a certain category. Meanwhile, they can be found to embody strong semantics, which can further support the construction of business ontology and abstract extraction of user problems.

Discussion

We have summarized some misclassified sentences, and the main ones that are difficult to categorize are as follows:

1. Short text: resulting in the "feature sparse", so the classifier is easy to misclassify, For example: "Check password" (features "none", correct classification: use, experimental results: others)
2. Contrast: the sentence states the facts by contrast, but the classification is wrong because the characteristics are not obvious. For example, the following consultation is correctly classified to usage, and the experimental result is the others:
   a) "Just now I made a phone call and talked for 55 seconds. Why did it become 1.08 minutes?"
3. Other reasons are the inclusion of wrong words in the text, unclear expression, lack of punctuation etc. In addition, there are some problems that can be further improved, such as establishing a conditional or sequential correlation model for the characteristics of different clauses. We can judge the classification more accurately and generate a summary of the text more easily by studying the relationship between several clauses in the text.

Conclusion

In this paper, we try to classify the short text. We use two syntactic features, substring and sub-sequence and effectively utilize the information of unmarked text. At the same time, the feature redundancy caused by frequent substrings/subsequences is discussed, and the experiment shows that these features can achieve a better classification effect.
However, in the process of text classification, the following questions need further study: how to extract more effective syntactic features [12], how to classify the unbalanced data, How to improve the accuracy of classification by utilizing the dependency of evaluators in sentences and introducing semantic resources.

Reference

[1] Bollegala D, Matsuo Y, Ishizuka M. Measuring semantic similarity between words using web search engines. In: Proceedings of the 16th International Conference on World Wide Web. New York, USA: ACM, 2007. 757–766

[2] Gabrilovich E, Markovitch S. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence. San Francisco, USA: Morgan Kaufmann, 2007. 1606–1611

[3] Phan X H, Nguyen L M, Horiguchi S. Learning to classify short and sparse text & web with hidden topics from largescale data collections. In: Proceedings of the 17th International Conference on World Wide Web. New York, USA: ACM, 2008. 91–100

[4] Ezeife, C., Y. Lu, et al. PLWAP sequential mining: open source code. In: proceedings of the Open Source Data Mining workshop on Frequent Pattern Mining Implementations, in conjunction with ACM SIGKDD. Chicago, 2005.26-29.

[5] Ezeife C I, Lu Y. Mining web log sequential patterns with position coded pre-order linked WAP-tree[J]. Data Mining and Knowledge Discovery, 2005, 10(1):5-38.

[6] Jian P, Han J W, Behzadmortazavi—as 1B, et a1. Mining access patterns efficiently from web logs[C]//Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining. London: Springer, 2000:396-407.

[7] Zhao Yan-ping, Cao Cun-gen , Xie Li-cong. Service Name Recognition Based on CRFs and Domain Rules [J].Computer Engineering, 2011, 37(11):200-202.

[8] Xu Xue-gang, Yang Chao-qun, Zhang Yu-hong. Short text classification based on extension with its own features [J].Application Research of Computer, 2017, 34(4):1008-1010.

[9] LV Chaozhen, JI Donghong, WU Feifei. Short text classification based on expanding feature of LDA. Computer Engineering and Applications, 2015, 51 (4):123-127.

[10]Wu Wei. Research on filteration and classification methods of large-scale short text [D]. Beijing university of posts and telecommunications, 2007:1-41.

[11]Qiu Liqing, Zhao Ruyi, Zhou Gang, et al. An extensive empirical study of feature selection for text categorization[C]//Proceedings of the 7th IEEE/ACIS International Conference on Computer and Information Science. Washington, DC: IEEE Computer Society, 2008:312-315.

[12]http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html