Check for updates

A Two-Level Classifier Model for Sentiment Analysis

Haidong Hao[1,2], Li Ruan[1,2(✉)], Limin Xiao[1,2], Shubin Su[1,2], Feng Yuan[3], Haitao Wang[4], and Jianbin Liu[4]

[1] State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China
{haohaidong, ruanli, xiaolm}@buaa.edu.cn
[2] School of Computer Science and Engineering, Beihang University, Beijing 100191, China
[3] Institute of Software Application Technology, Guangzhou and Chinese Academy of Sciences, Guangzhou 511458, China
yf@gz.iscas.ac.cn
[4] Space Star Technology Co., Ltd., Beijing 100086, China
{wanghaitao, liujianbin}@spacestar.com.cn

Abstract. This paper proposes a fast and high performance classifier model for sentiment analysis of textual reviews. The key contribution is three fold. First, a two-level classifier model consists of three base classifiers is proposed, and theory proves that the model could be better than the strongest classifier among the base classifiers in both classification performance and time cost of predict. Second, this paper proposes a lexicon-based classifier as a base classifier using a new part of speech (POS) which is called "weaken words". Finally, we implemented several two-level classifiers by combining the lexicon-based classifier with several machine learning classifiers. Experiments on Chinese reviews dataset show that the two-level classifier model is effective and efficient.

Keywords: Sentiment analysis · POS · Weaken words
Two-level classifier model · Predict time

1 Introduction

With the further popularization of networks and smart phones, publishing reviews online on social events, products and service are increasingly common. These reviews are collectively referred to as IWOM (Internet Word of Mouth), since they contain a lot of useful information to evaluate the evaluation object from various aspects. Based on the understanding of these reviews, the government can understand people's attitude towards a policy, and then make the right decision; the online merchants can find deficiencies in their products and services according to the user's experience and make continuous optimization; consumers can make comparison between kinds of products and service, and ultimately make a reasonable purchase. Therefore, it is very valuable to mining out these reviews' opinion.

I. Romdhani et al. (Eds.): CollaborateCom 2017, LNICST 252, pp. 704–717, 2018.
https://doi.org/10.1007/978-3-030-00916-8_64

Online IWOM monitoring system is a kind of real-time system which aims to classify the semantic orientation (positive or negative) of web reviews (or any other speeches), and conducts further analysis. Sentiment analysis is the basic task of the system, and it has to classify massive amounts of data over a limited period of time, so, the system is not only sensitive to classification performance, but also to time cost. Therefore, a fast and high performance classifier is very essential. However, get better classification performance in less time is often a contradictory problem. High performance classifiers are always complicated and need a long time to complete the calculation. On the other hand, more of the existing work is focused on improving the classification performance or reduce the training time, research on the reduction of predict time is relatively little. So, this paper aims to construct a fast and high performance classifier.

Boosting [1] and bagging [2] are two popular methods to improve the classification performance of classifier. The main idea of them is to train a set of classifiers through multiple rounds of sampling on the training set, and then construct a new classifier by combining these base classifiers. Drawing on this kind of combination idea, this paper takes advantage of the classification performance of strong classifiers and the classification speed of weak classifiers, using them as base classifiers to construct a faster and stronger classifier. Compared with boosting and bagging, the main difference is that the two-level classifier model is not through the linear combination of the base classifiers to predict, but through the stratification to predict, so it can significantly reduce the time cost.

In Sect. 3, this paper derives the conditions should these base classifiers meet to achieve the goal, and one condition is the base classifiers are independent from each other, so both lexicon-based and machine learning classifiers are used as the base classifiers. As for the lexicon-based classifier, this paper takes account of two special cases in expression, and proposes a new POS which is called “weaken words” to construct the classifier. And as for machine learning classifiers, this paper uses Naïve Bayesian, Logistic and SVM (support vector machine) classifiers.

The remainder of this paper is organized as follows. In Sect. 2, we present a brief description of existing sentiment analysis approaches. In Sect. 3, we present the definition and proof of the two-level classifier model in detail. In Sect. 4, we present in detail the construction of lexicon-based classifier we applied as a base classifier in the model. In Sect. 5, we present the experiments of the model on Chinese reviews dataset and the results show that the model is efficient and effective. Finally, in Sect. 6 we conclude this work.

2 Related Work

There are two main approaches for sentiment analysis of text. One is machine learning and the other is lexicon-based. In terms of machine learning approach, it can be divided into supervised and unsupervised approaches. Naïve Bayesian, maximum entropy and SVM are the most classic supervised classifiers. Pang and Lee [3] had experimented with these classifiers using kinds of features, like n-gram, term frequency and POS etc. and found SVMs tend to do the best. Turney [4] had presented an unsupervised machine learning algorithm. He firstly proposed PMI (point-wise mutual information) to measure the similarity of given words or phrases with positive or negative reference words. And then, calculate the average semantic orientation of the extracted phrases to assign a classification to the review. Zou et al. [5] had considered the words' syntactic properties in basic words-bag to generate a more accurate solution.

In terms of lexicon-based approach, based on WordNet, Kim and Hovy [6] had proposed a method to constructed semantic dictionary. They assembled a small amount of seed words by hand, and sorted them into positive and negative lists by semantic polarity, and then grow these two lists by adding words obtained from WordNet. Liu et al. [7] had proposed three conjunction rules (intra-sentence conjunction rule, pseudo intra-sentence conjunction rule and inter-sentence conjunction rule) and two word rules (synonym and antonym rule) to determine the polarity of the adjectives in a given domain. In another paper [8], they had extracted attributes of the evaluation object, and proposed a method for feature-level opinion mining. Taboada et al. [9] had analyzed the characteristics of various POS in detail. When calculate the semantic orientation of sentiment-bearing words, they took into account valence shifters (intensifiers, down-toners, negation, and irrealis markers). Their method performs well and is robust across domains and texts.

To reduce the time cost of predict, Wang and Zaniolo [10] proposed a classifier using discretization techniques to limit disk I/O at the cost of accuracy, and they remedied the loss of accuracy by using a simplified version of estimation method proposed in CLOUDS [11]. Based on Chinese web page characteristics, Wu et al. [12] proposed a pre-classification method by giving a keywords list to reduce predict time for Chinese web page classification. Wu et al. [13] proposed a normalized feature weighted KNN (k-Nearest Neighbor) text classifier, their method can reduce the feature dimension thus reduce the time cost of predict.

3 Two-Level Classifier

3.1 Model Definition

Let $C = \{C_1, C_2, C_3\}$ be a set of three two-class classifiers, $P = \{P_1, P_2, P_3\}$ be the average accuracy of each classifier and $T = \{T_1, T_2, T_3\}$ be the time cost of predict of each classifier. Assume that C, P and T satisfy the following three conditions:

- C_1, C_2, and C_3 are independent from each other;
- C_3 is a strong classifier, C_1 and C_2 are two weak classifiers ($P_3 > \max\{P_1, P_2\}$);
- C_3 is more complicated than C_1 and C_2 ($T_3 > \max\{T_1, T_2\}$);

Using C_1 and C_2 as low level base classifiers, C_3 as high level final classifier, and utilize multithreading technology we could construct a faster and stronger classifier. The principle is that for easy to be classified texts be classified by low level fast classifiers, and for texts that difficult to be classified be classified by high level complicated classifier. The workflow of the model is as follows:

Algorithm 1. Work flow of the two-level classifier model

for each text in buffer
  extract the features;
  C_1, C_2 and C_3 begin to calculate;
  if (C_1 and C_2 output the same result)
    stop the calculating of C_3;
    take the result as the final classification result of this text;
  else
    wait for C_3 to output its result;
    take this result as the final classification result of this text;

3.2 Model Proof

Based on the basic three conditions, the next task is to find other necessary or sufficient conditions to improve the classification performance while reduce the time cost of predict. The accuracy and time cost formulas of the model are as follows:

$$\mathrm{dis\_}rate = P_2(1 - P_1) + P_1(1 - P_2) \tag{1}$$

$$P = P_1P_2 + P_3 \times \mathrm{dis\_}rate \tag{2}$$

$$T = \max\{T_1, T_2\} + [T_3 - \max\{T_1, T_2\}] \times dis\_rate \tag{3}$$

P is the accuracy of the two-level classifier, T is the time cost, and *dis_rate* is the ratio of "disagree" of C_1 and C_2 to a certain input text ("disagree" means that C_1 and C_2 output different results).

Conditions that P be Greater than P_1 and P_2: Substituting (1) into (2), we get

$$P = P_1P_2 + P_3[P_1 + P_2 - 2P_1P_2)] \tag{4}$$

Without losing generality, assume that $P_2 \geqslant P_1$. If $P > \max\ \{P_1,\ P_2\}$, the inequality below should be true:

$$\frac{P}{P_2} = P_1 + P_3 + (\frac{1}{P_2} - 2)P_1P_3 > 1 \tag{5}$$

It can transform to

$$P_3[1 + (\frac{1}{P_2} - 2)P_1] > 1 - P_1 \tag{6}$$

Since $1 > P_2 > 0$, it can transform to

$$P_3 > \frac{1 - P_1}{1 + (\frac{1}{P_2} - 2)P_1} \tag{7}$$

Regarding P_2 as a constant, and let

$$P_3 > \mathrm{F}(P_1) = \frac{1 - P_1}{1 + (\frac{1}{P_2} - 2)P_1} \tag{8}$$

The derivative of $\mathrm{F}(P_1)$ is

$$\mathrm{F}'(P_1) = \frac{1 - \frac{1}{P_2}}{[1 + (\frac{1}{P_2} - 2)P_1]^2} < 0, (1 > P_2 > 0) \tag{9}$$

Because $\mathrm{F}(P_1)$ is a monotonically decreasing function, if P_1 is more closer to P_2, C_3 would be more easier to improve the accuracy of C_1 and C_2, and

$$P_3 > \mathrm{F}(P_1 = P_2)_{\min} = \frac{1 - P_2}{1 + \left(\frac{1}{P_2} - 2\right)P_2} = 0.5 \tag{10}$$

So, P would be greater than P_1 and P_2 if inequality (7) is true. Furthermore, P_3 must be greater than 0.5.

Conditions that P be Greater than P_3: If $P > P_3$, the inequality below should be true:

$$\frac{P}{P_3} = \frac{P_1 P_2}{P_3} + P_1 + P_2 - 2P_1 P_2 > 1 \tag{11}$$

It can transform to

$$\frac{1}{P_3} > \frac{1 + 2P_1P_2 - (P_1 + P_2)}{P_1 P_2} \tag{12}$$

Regarding P_1 as a constant, and let

$$G(P_2) = 1 + 2P_1P_2 - (P_1 + P_2) \tag{13}$$

The derivative of $\mathrm{G}(P_2)$ is

$$G'(P_2) = 2P_1 - 1 \begin{cases} > 0, & 1 > P_1 > 0.5 \\ = 0, & P_1 = 0.5 \\ < 0, & 0 < P_1 < 0.5 \end{cases} \tag{14}$$

Then, the minimum value of G(P_2) is

$$G(P_2)_{\min} = \begin{cases} \lim\limits_{p_2->0} 1 + 2P_1P_2 - (P_1 + P_2) = 1 - P_1 > 0, & 1 > P_1 > 0.5 \\ 0.5, & P_1 = 0.5 \\ \lim\limits_{p_2->1} 1 + 2P_1P_2 - (P_1 + P_2) = P_1 > 0, & 0 < P_1 < 0.5 \end{cases} \quad (15)$$

As we can see, G(P_2) > 0 is always true, hence, inequality (12) can transform to

$$\frac{P_1P_2}{1 + 2P_1P_2 - (P_1 + P_2)} > P_3 \quad (16)$$

Without losing generality, assume that $P_2 \geqslant P_1$ and take P_1 as a constant, let

$$L(P_2) = \frac{P_1P_2}{1 + 2P_1P_2 - (P_1 + P_2)} > P_3 \quad (17)$$

The derivative of L(P_2) is

$$L'(P_2) = \frac{P_1(1 - P_1)}{[1 + 2P_1P_2 - (P_1 + P_2)]^2} > 0, (1 > P_1 > 0) \quad (18)$$

Because L(P_2) is a monotonically increasing function, if P_3 is more greater, P_1 and P_2 should be more greater either to improve P_3, and

$$L(P_2 = P_1)_{\min} = \frac{P_1^2}{1 + 2P_1^2 - 2P_1} > P_3 \quad (19)$$

Then, let

$$K(P_1) = \frac{P_1^2}{1 + 2P_1^2 - 2P_1} > P_3 \quad (20)$$

The derivative of K(P_1) is

$$K'(P_1) = \frac{2P_1(1 - P_1)}{[1 + 2P_1^2 - 2P_1]^2} > 0, (1 > P_1 > 0) \quad (21)$$

K(P_1) is a monotonically increasing function, P_3 is greater than 0.5, and K(P_1 = 0.5) = 0.5, so that P_1 should be greater than 0.5, it means P_2 should be greater than 0.5 too.

So, P would be greater than P_3 if inequality (16) is true. Furthermore, P_1 and P_2 must be greater than 0.5.

Conditions that *T* be Less than T_3: Without losing generality, assume that $T_2 \geqslant T_1$. If $T < T_3$, the inequality below should be true:

$$T = T_2 + (T_3 - T_2) \times dis\_rate < T_3 \tag{22}$$

It can transform to

$$(T_3 - T_2) \times dis\_rate < T_3 - T_2 \tag{23}$$

Because *dis_rate* is less than 1, if $T_3 > T_2$, inequality (23) would be true and *T* would be less than T_3.

3.3 Model Conclusions

Based on the basic 3 conditions and model proof, we can get conclusions as below:

- If inequality (16) is true, the classification performance of the model would be better than P_3, and the necessary condition is P_1, P_2 and P_3 are all greater than 0.5;
- The predict time of the model is less than T_3.

In order to ensure the independence of these three base classifiers, this paper used both lexicon-based and machine learning classifiers to construct the two-level classifier.

4 Lexicon-Based Classifier

4.1 Sentence Structure

Lexicon-based classifier needs to extract the words related to sentiment analysis. Except for sentiment-bearing words (we marked the POS as "emo"), this paper takes account negative words (they can reverse the orientation of sentiment-bearing words, and we marked the POS as "non"), degree adverb (they can strengthen or weaken the orientation degree of sentiment-bearing words, and we marked the POS as "dg"), and a new type of words which is called "weaken words" (we marked the POS as "wk"). Why we need weaken words, we would like to show the necessity throng a special case in expression:

"这种新药减轻了他的痛苦/This kind of new drug has eased his pain".

In this example, if we ignore the word "减轻/eased", the extracted word of this sentence is "痛苦/pain", which expresses negative orientation. However, if we take account the word "减轻/eased", we find it weakened the negativity. It shows that we should consider the role of this new kind of words in sentiment analysis.

4.2 Semantic Dictionary

In Chinese field, HowNet [14] and NTUSD (National Taiwan University Semantic Dictionary) [15] provides the basic semantic dictionary. There are 4,566 positive emotional words and 4,370 negative emotional words in HowNet; 2,810 positive emotional words and 8,276 negative emotional words in NTUSD. HowNet also provides 219 degree adverbs, and these words are subdivided into six types according to the words' tone strength. Because neither HowNet nor NTUSD provides negative words, we concluded 22 negative words from web corpus. By the way, these words should be merged, because each kind of words may be repeated with other kinds of words. The final results are shown in Table 1.

Table 1. The semantic dictionary.

POS	Value	Quantity	
emo	positive emotional word is 1, negative emotional word is –1 "高兴/happy"—1, "难过/sad"— –1 etc.	positive emotional: 6506 negative emotional: 11185	
dg	different "dg" word has different value "极其/extreme"—2, "很/very" — 1.7, "较/more"— 1.4, "稍/-ish"— 1.1, "欠/insufficiently"— 0,8, "超/over"— 0.5	extreme: 61 more: 35 insufficiently: 11	very: 37 -ish: 29 over: 24
non	All are –1 "不/not"— –1, "没有/none"— –1 etc.	22	
wk	All are –0.2 "减轻/ease"— –0.2, '减少/reduce" – –0.2etc.	87	

HowNet is a common-sense knowledge base unveiling inter-conceptual relations and inter-attribute relations of concepts as connoting in lexicons of the Chinese and their English equivalents. The "concept" and "primitive" are two basic concepts in HowNet. In HowNet, one word can be described by several concepts and every concept represents the word's one meaning in different context. Concept is described by primitive, and primitive is the smallest unit in HowNet. We can construct a tree-like semantic hierarchy according to the upper and lower relationship between primitives.

Liu and Li [16] had proposed a method to compute the word similarity based on HowNet. For two Chinese words W_1 and W_2, W_1 has n concepts S_{11}, S_{12}, …, S_{1n}, W_2 has m concepts S_{21}, S_{22}, …, S_{2m}, and the similarity of W_1 and W_2 is the max value of each pair of concepts, namely

$$Sim(W_1, W_2) = \max_{i=1\ldots n, j=1\ldots m} Sim(S_{1i}, S_{2j}) \tag{24}$$

Because concept is described by primitive, Sim (S_{1i}, S_{2j}) in (24) can be computed by

$$Sim(P_1, P_2) = \frac{\alpha}{\mathrm{Dis}(P_1, P_2) + \alpha} \tag{25}$$

P_1 and P_2 are two primitives, *Dis* (P_1, P_2) is the distance of P_1 and P_2 in the semantic hierarchy, and α is an adjustable parameter.

Based on the word similarity formula, we chose 20 typical “wk” words as seed words, and took the similarity lower threshold of 0.9 to extend the “wk” words. We got 156 words at the beginning, and after manual selection, 87 words were left (included the 20 seed words).

In the semantic dictionary, every word has two attributes, one is the word’s POS tag, and the other is its value. The dictionary looks like below (the last column is the number of this kind of words):

4.3 DFA (Deterministic Finite Automaton) Model of the Algorithm

After extracted the sentiment analysis related words, a sentence’s semantic orientation can be analyzed according to the words sequence. The simplest solution is scanning the words sequence from back to front, and for every word update the current emotional value according to its POS tag and value. For example, when scanning to a “dg” word, multiply the current emotional value by the word’s value. But there are two special cases should be considered:

- *“non-dg” order words* [17]. *When “non” word’s next word is “dg” word. For example, “不是很高兴/not very happy” and “很不高兴/very unhappy”, although they all express the feeling unhappy, the strength are not the same;*
- *“dg-wk” order words. When “dg” word’s next word is “wk” word. For example, “极大得减轻他的痛苦/greatly eased his pain”, the semantic orientation reversed from very negative to very positive;*

Considering these two special cases, we find “non-dg” order words behaves like a “wk” word as a whole, and the “wk” word behaves like a “non” word in “dg-wk” order words. So, “non-dg” order words should be transformed to a “wk” word, and the “wk” word in “dg-wk” words should be transformed to a “non” word (Fig. 1). Add these two transform rules we concluded the DFA model of the algorithm like below:

DFA = {Q, Σ, q0, δ, {F}}
Q = {q0, q1, q2, F}, Σ = {emo, dg, wk, non}
δ(q0, dg) = q1, δ(q0, wk) = q2, δ(q0, {emo, non}) = q0, δ(q1, dg) = q1,
δ(q1, non) = q2, δ(q1,{emo, wk}) = q0, δ(q2, wk) = q2,
δ(q2,{emo, non, dg}) = q0, δ({q0, q1, q2}, ε) = F

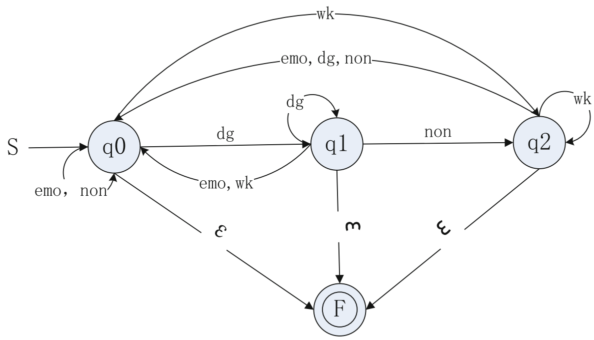

Fig. 1. DFA model of the algorithm

5 Experiments

5.1 Experiments on Base Classifiers

We crawled five different products and service reviews from ctrip.com (http://www.ctrip.com) and jd.com (https://www.jd.com) to test the model. These products and service are clothing, fruit, hotel, PDA (personal digital assistant) and shampoo. For each product or service, we crawled 5,000 positive and negative reviews. The test indicators are average accuracy and time cost of predict (milliseconds per 10,000 pieces of reviews). We used 5 fold cross validation (for lexicon-based classifier, we divided the reviews into 5 groups randomly) and took the average value as the final result. The base classifiers experiments results are as follows (Table 2):

Table 2. The base classifiers experiments results.

Products & service	Test indicators (average accuracy, time cost)			
	Lexicon-based	Naïve Bayesian	Logistic	SVM (Gamma = 0.25, C = 2)
Clothing	(0.87,1032)	(0.89,2520)	(0.90,3180)	(0.94,7280)
Fruit	(0.83,1351)	(0.88,2995)	(0.88,3245)	(0.90,7930)
Hotel	(0.81,3191)	(0.85,9030)	(0.88,9025)	(0.90,22070)
PDA	(0.81,1446)	(0.87,3650)	(0.88,4245)	(0.93,9735)
Shampoo	(0.81,1245)	(0.89,3220)	(0.89,3360)	(0.92,8235)
All	(0.83,1600)	(0.86,4275)	(0.88,4486)	(0.91,11105)

As we can see, in terms of classification performance, the lexicon-based classifier is weaker than other classifiers, Naïve Bayesian and Logistic classifiers are similar, and SVM classifier is the best. And in terms of time cost of predict, SVM classifier takes the longest time, Naïve Bayesian and Logistic classifiers followed, and the lexicon-based classifier takes the shortest time.

5.2 Experiments on Two-Level Classifiers

According to Sect. 3, if inequality (16) and (23) are true, the two-level classifier would be faster and stronger than the strongest classifier. After verified, we used the lexicon-based classifier as C_1, Naïve Bayesian and Logistic classifier as C_2, SVM classifier as C_3 and constructed 2 two-level classifiers. The experiments results are as follows (Tables 3 and 4):

Table 3. Two-level classifier 1 experiments result. "the" stands for "theoretical", "exp" stands for "experimental".

Products & service	Two-level classifier 1 (Naïve Bayesian as C_2)						
	dis_rate (C_1 & C_2)		*dis_rate* (C_2 & C_3)		*P*		*T*
	the	exp	the	exp	the	exp	
Clothing	0.21	0.17	0.16	0.09	0.97	0.94	4840
Fruit	0.25	0.19	0.20	0.07	0.95	0.91	5475
Hotel	0.28	0.22	0.22	0.12	0.94	0.90	11736
PDA	0.27	0.21	0.18	0.09	0.95	0.92	6180
Shampoo	0.26	0.23	0.17	0.08	0.96	0.92	5699
All	0.26	0.21	0.20	0.09	0.95	0.91	6875

Table 4. Two-level classifier 2 experiments result.

Products & service	Two-level classifier 2 (Logistic as C_2)						
	dis_rate (C_1 & C_2)		*dis_rate* (C_2 & C_3)		*P*		*T*
	the	exp	the	exp	the	exp	
Clothing	0.20	0.15	0.15	0.06	0.97	0.94	4691
Fruit	0.25	0.18	0.20	0.07	0.95	0.91	5531
Hotel	0.26	0.20	0.20	0.07	0.94	0.89	11503
PDA	0.26	0.21	0.17	0.07	0.95	0.92	6217
Shampoo	0.27	0.23	0.17	0.07	0.97	0.92	5540
All	0.25	0.20	0.19	0.07	0.96	0.91	6990

As the results show, the two-level classifiers' time cost of predict reduced to about 63% of C_3, but in terms of accuracy, the experimental value had a large gap with the theoretical value. The main reason is C_1, C_2 and C_3 are not independent. In fact, the average *dis_rate* deviation of C_1 and C_2 is about 21%, C_2 and C_3 is about 51% and 62%, so the independence of C_1, C_2 and C_3 are poor. Thus, as the final classifier, C_3 is the bottleneck of classification performance. Although the accuracy hadn't been improved, at least the same as C_3, it's in line with our needs too.

To maximize have to guarantee the independence of C_1, C_2 and C_3, we divided the "emo" words into two parts and constructed 2 lexicon-based classifiers. Since the semantic dictionaries are almost completely different (other kinds of words are the same), the 2 lexicon-based classifiers are surly almost completely independent. We can take them as C_1 and C_2. In the meanwhile, to meet inequality (16), we constructed a new SVM classifier as C_3. The new base classifiers and two-level classifier experimental results are as follows (Tables 5 and 6):

Table 5. The new base classifiers experiments result.

Products & service	Test indicators (average accuracy, time cost)		
	Lexicon-based (1)	Lexicon-based (2)	SVM(Gamma = 1.5, C = 1)
Clothing	(0.76,1012)	(0.72,952)	(0.83,13495)
Fruit	(0.71,1210)	(0.70,1201)	(0.78,14378)
Hotel	(0.72,3026)	(0.72,3166)	(0.74,30855)
PDA	(0.67,1276)	(0.68,1325)	(0.75,17385)
Shampoo	(0.68,1178)	(0.68,1152)	(0.77,14878)
All	(0.71,1586)	(0.70,1562)	(0.81,18280)

Table 6. Two-level classifier 3 experiments result.

Products & service	Two-level classifier 3						
	dis_rate (C_1 & C_2)		*dis_rate* (C_1 & C_3)		P		T
	the	exp	the	exp	the	exp	
Clothing	0.38	0.36	0.33	0.28	0.86	0.84	7565
Fruit	0.42	0.40	0.38	0.35	0.82	0.81	8212
Hotel	0.40	0.37	0.39	0.38	0.81	0.80	19258
PDA	0.44	0.40	0.42	0.36	0.78	0.76	11109
Shampoo	0.44	0.37	0.40	0.39	0.80	0.80	8781
All	0.42	0.40	0.37	0.36	0.83	0.83	10360

For this two-level classifier, the average *dis_rate* deviation of C_1 and C_2 was reduced to about 8%, C_1 and C_3 was reduced to about 8% too, so, C_1, C_2 and C_3 are almost independent from each other. As we can see, the new two-level classifier's time cost reduced to about 59% of C_3, and in terms of accuracy, the experimental value was close to the theoretical value, and higher than C_3. It proved that the model is really effective and efficient.

6 Conclusions

Online IOWM monitoring system is a powerful tool to help people making decisions. A fast and high performance sentiment analyzer is essential for this kind of systems. In order to meet this demand, this paper proposes a two-level classifier model consists of three base classifiers, and derives a theoretical proof, that the classifier could be better than the strongest classifier among the base classifiers in both classification performance and time cost of predict. To ensure the independence of the base classifiers, this paper used both lexicon-based and machine learning classifiers. In addition, this paper constructs a lexicon-based classifiers, and proposes a new POS which is called "weaken word" to improve its accuracy. At last, several two-level classifiers were constructed using the lexicon-based classifier, Naïve Bayesian, Logistic and SVM classifiers, and as the experiments results show, the time cost could be reduced greatly and the accuracy is at least similar to that of the strongest classifier. In short, the two-level classifier model is efficient and effective.

Acknowledgements. This work was supported by the National Science Foundation of China (Grant Nos. 61370059, 61772053 and 61232009 (key project)), Beijing Natural Science Foundation (Grant No. 4152030), the fund of the State Key Laboratory of Computer Architecture (CARCH201507), the fund of the State Key Laboratory of Software Development Environment (SKLSDE-2016ZX-15), the State Administration of Science Technology and Industry for National Defense, the major projects of high resolution earth observation system under Grant No. Y20A-E03, and the National Key R&D Program of China under Grant No. 2017YFB0202004.

References

1. Breiman, L.: Bagging predictors. Mach. Learn. **24**(2), 123–140 (1996)
2. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. J. Comput. Syst. Sci. **55**(1), 119–139 (1997)
3. Pang, B., Lee, L.: Thumbs up? Sentiment classification using machine learning. In: Proceedings of EMNLP, pp. 79–86 (2002)
4. Turney, P.D.: Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: Meeting on Association for Computational Linguistics Association for Computational Linguistics, pp. 417–424 (2002)
5. Zou, H., et al.: Sentiment classification using machine learning techniques with syntax features. In: International Conference on Computational Science and Computational Intelligence, pp. 175–179. IEEE (2016)
6. Kim, S.M., Hovy, E.: Determining the sentiment of opinions (2004)
7. Ding, X., Liu, B.: The utility of linguistic rules in opinion mining. In: International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 811–812. ACM (2007)
8. Ding, X., Liu, B., Yu, P.S.: A holistic lexicon-based approach to opinion mining. In: International Conference on Web Search and Data Mining, pp. 231–240. ACM (2008)
9. Taboada, M., et al.: Lexicon-based methods for sentiment analysis. Comput. Linguist. **37**(2), 267–307 (2011)
10. Wang, H., Zaniolo, C.: CMP: a fast decision tree classifier using multivariate predictions. In: International Conference on Data Engineering, p. 449. IEEE Computer Society (2000)
11. Alsabti, K., Ranka, S., Singh, V.: CLOUDS: a decision tree classifier for large datasets. In: Knowledge Discovery and Data Mining (1998)
12. Shi-Ming, X.U., et al.: Efficient SVM Chinese Web page classifier based on pre-classification. In: Computer Engineering and Applications (2006)

13. Wu, F., Zheng, Y., Cheng, W.: Adaptive normalized weighted KNN text classification based on PSO. Scientific Bulletin of National Mining University (2016)
14. Dong, Z., Dong, Q.: HowNet—a hybrid language and knowledge resource. In: International Conference on Natural Language Processing and Knowledge Engineering Proceedings, pp. 820–824 (2003)
15. http://www.datatang.com/data/44317
16. Liu, Q., Li, S.: Word similarity computing based on how-net. In: Computational Linguistics and Chinese Language Processing (2002)
17. Yang, K., Duan, Q.: Analysis of emotional tendency based on emotional dictionary. Comput. Era **3**, 10–13 (2017)