Method of Consistency Judgment for App Software's User Comments

Meng Ran[1,2], Ying Jiang[1,2(✉)], Qixin Xiang[1,2], Jiaman Ding[1,2], and Haitao Wang[1,2]

[1] Yunnan Key Lab of Computer Technology Application, Kunming 650500, China
{374085783,346760502}@qq.com, {jy_910,tjom2008,kmwht}@163.com
[2] Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China

Abstract. With the popularity of mobile intelligent terminal, user comments of App software is viewed as one of the research interests of social computing. Faced with the massive App software, most users usually view the other users' comments and marks to selecting the desired App software. Due to the freedom and randomness of the network comments, the inconsistence between the user's comment and mark makes it difficult to choose App software. This paper presents a method by analyzing the relationships among user's comment information, the user's mark and App software information. Firstly, the consistency between user's comment information and App software information is judged. Then, through analyzing the grammar relationships among the feature-words, adverbs and the feature-sentiment-words in App software's feature-sentiment-word-pairs, the user's emotional tendency about App software is quantified combining with the dictionary and the network sentiment words. After calculating the user's comprehensive score of App software, the consistency of App software's user comment is judged by comparing this score and the user's mark. Finally, the experimental results show that the method is effective.

Keywords: Social computing · App software · The consistency of user comment · User's comment information · User's mark · Feature-sentiment-word-pairs · Network sentiment word

1 Introduction

With the development of information technology, social computing is known as one of the data-intensive science which catches the scholars' attentions, especially social networks analysis, generative use and user comments of product [1]. In recent years, In recent years, App software is regarded as a novel experience-based product [2], especially Android and iOS system increased rapidly. According to the data, the number of active mobile devices has reached 899 million up to December 2015 in China.

W. Che et al. (Eds.): ICYCSEE 2016, Part I, CCIS 623, pp. 470–483, 2016.
DOI: 10.1007/978-981-10-2053-7_42

Different from other applications, App software has the following characters: (1) because App's development cycle is usually short, the developer can make the software development strategy after requiring user's requirements through their comments [3]. (2) Before selecting App software, users cannot get the quality information about the software by an advertisement or a brand, which will makes it difficult to choose App software [4]. (3) In order to make the rational choice, users would like to know the quality of App software through user's comment information [4]. User comments of App software include rating marks (usually five marks) scored by user and comment descriptions after using App software. User comments often imply potential information such as the preference and attention to some specific properties of App software. However, as the freedom and random of network comments, there are some inconsistencies among App software information, user's comment information and user's mark, which will bring great difficulties to evaluation of App software's quality for users. So it is an important problem to judge the consistency of user comments of App software.

2 Related Work

Currently, the study of users' comments in social computing has been mature, especially aiming at user comments of online communities, hedonic use, etc. [5]. But user comments of App software are still on study. Gao et al. [6] establish a theme dynamics update model through extracting theme comments and sorting based on different App software user comments. AlQuwayfili et al. [7] divide the user comments into credible comments and incredible comments based on the analysis. The above research shows that the theme and credibility of user comments have influences on user's choice of App software. However, there is lack of comprehensive analysis for user's comment information, user's mark and App software information. Other researches focus on the influences of the sentiment tendency [8] of users to App software on App software's quality in user comments. Islam [9] divides the user comments into different marks by analyzing sentiment and optimizing probability of user comments. Guzman et al. [10] identify the 'coarse feature' of App software by establishing a software mining warehouse and defining feature patterns, then the positive and negative attitudes towards App software in user comments are explored based on the greedy clustering algorithm. In the above researches, the tendencies of user comments are divided into two levels. Usually, users also give five marks when they comment App software. However, the current researches ignore whether the user's comment information and user's mark are consistent when user is selecting App software.

In this paper, we aim at how to judge the consistency of user comments on App software through analyzing the relationship among user's comment information, user's mark and App software information comprehensively. The consistency between the user's comment information and the App software information is judged. When the user's comment information is consistent with the App software information, the consistency between the user's comment information and the user's mark is judged by the following steps: (1) the grammatical relations

among feature words, adverbs and sentiment words in feature-sentiment-word-pairs of App software are analyzed. (2) The sentiment tendency of users to App software is quantified with dictionaries and network sentiment words. (3) The comprehensive score of App software in each user comment is calculated.

The main contributions in this paper are as follows:

(1) The internal relations among user's comment information, user's mark and App software information are mined through comprehensively analysis. In order to identify whether the user's comment information is aim at certain App software, the App software information is used as the basis to judge the consistency between the user's comment information and the App software information.
(2) In order to improve the coarse-grained qualitative evaluation method which divided the sentiment into satisfaction and dissatisfaction, the comprehensive score in the user's comment information of App software is calculated. The comprehensive score is divided into five marks, which are compared with five marks to judge whether the user's comment information conform to the user's marks.

The method in this paper will identify whether user comments are consistent with the App software information and whether the degree of sentiment tendency of user comments is consistent with user's marks. It is helpful for users to select App software.

3 The Consistency Judgment Between the User's Comment Information and the App Software Information

The App software information is published by the developers. Because there are the most features of App software, the App software information can be used as the basis to judge whether the user comments are consistent with the App software information. Most users would like to make comments containing one or more features because the user comments for App software is relatively free on the network. However, some user's comment information is not concerned with the App software's features, which leads to the inconsistency between user's comment information and the App software information and the inconsistency between user's comment information and user's mark. So whether the user's comment information aims at App software must be judged at first.

Currently, App software information and user's comment information usually are described by natural language. We cannot identify the consistency of their information directly. Hence the software features of the App software information and the feature-sentiment-word-pairs in the user comments are extracted to judge the consistency.

3.1 Feature-Sentiment-Word-Pairs Extraction of App Software

Currently, most researches acquire the description or comment data about product characteristics through extracting feature and sentiment words from user comments. In this paper, user name, user ID, comment software type, user's mark and user's comment information are extracted from the massive user comments data. After analyzing the above information, users usually use 'sentiment words' or 'adverbs + sentiment words' to modify the software features. For example, a user gives comment on App software '微信' as follows: 下载很麻烦'. In this comment, the sentiment word '麻烦' modifies the feature word '下载' and the degree adverb '很' modifies the sentiment word '麻烦'. Therefore, there are corresponding relations among feature word, sentiment word and adverb. Features words and sentiment words usually occur in pairs. We name them as feature-sentiment-word-pairs of App software, which is defined as follows.

Definition 1. Feature-sentiment-word-pairs of App software $f = (Wh, Wd, Wa)$. Wh is App software feature focused by user, such as '下载', '画面' and so on. Wa is sentiment word modifying the feature Wh, such as '麻烦', etc. Wa includes network sentiment words which express user's objective impression on App software feature, such as '神器', etc. As adverb modifying the sentiment word Wa, Wd expresses degree of user's sentiment tendency to feature words, such as '很', '非常' and so on.

The App software information includes software name, software type, software ID, software introduction, and software source. It is released by the developer, and can describe most features of App software. In order to judge the consistency of user comments on App software, we extract nouns, verbs and noun phrases as feature set of App software information. Based on feature words and sentiment words extraction as in [11], we extract feature-sentiment-word-pairs f at the same time.

3.2 The Consistency Judgment Between the User's Comment Information and the App Software Information

In order to judge the consistency between user's comment information and App software information, we compare Wh in f with software feature in App software information and calculate their similarity. Based on the current user comments on App software, it is can be shown that most users give comprehensive comments. Wh is usually omitted in the user's comment information, such as '很好，很满意，很不错'. We regard this type of comments as for the whole App software, which means it is consistent between the user's comment information and the App software information.

Additionally, a user comment may be involved in many feature-sentiment-word-pairs of App software, which every feature is different. The degrees of user's sentiment tendency are varied as well. This sentiment tendency can influence the user's mark on App software. Therefore, the set of App software feature-sentiment-word-pairs is defined as follows.

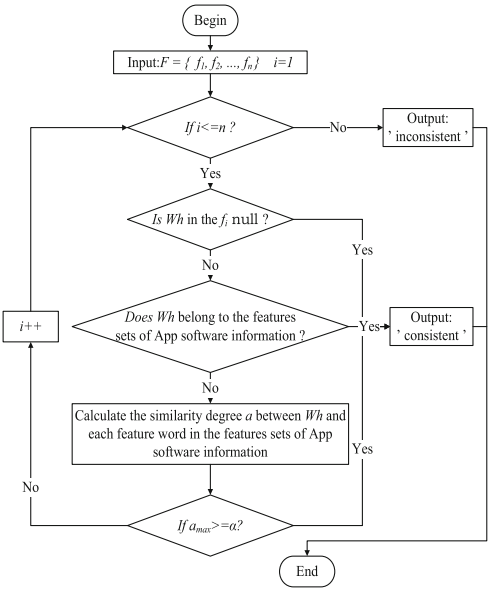


Fig. 1. Flow diagram of the consistency Judgment between the user's comment information and the App software information.

Definition 2. The set of App software feature-sentiment-word-pairs $F = \{f_1, f_2, ..., f_n\}$. $f_i = (Wh, Wd, Wa)$ $(i = 1, 2, ..., n)$ indicates feature-sentiment-word-pairs of App software.

Based on the set of App software feature-sentiment-word-pairs, the consistency between user's comment information and App software information is judged. The flow diagram is as shown in Fig. 1.

The similarity calculation method in Fig. 1 uses the method described in [12]. And a_{max} indicates the maximum value in similarity a, threshold value $\alpha = 0.1$.

4 The Consistency Judgment Between the User's Comment Information and the User's Mark

When the user's comment information is consistent with the App software information, the users would still be confused about selecting App software if the user's comment information is inconsistent with user's mark. The users cannot judge whether the user's comment information or user's mark are the focus. So the consistency between the user's comment information and user's mark should be judged further. Because the set of App software feature-sentiment-word-pairs F is natural language, its consistency with user's mark cannot be judged directly. Therefore, the emotional tendency of each f in App software feature-sentiment-word-pairs set F is quantified at first. Then the composite score of App software feature-sentiment-word-pairs set F is calculated. Finally, the score will be divided

into five level marks corresponding with the user's mark. Accordingly, the consistency between the emotional tendency in the user's comment information and the uses' mark of this App software can be judged.

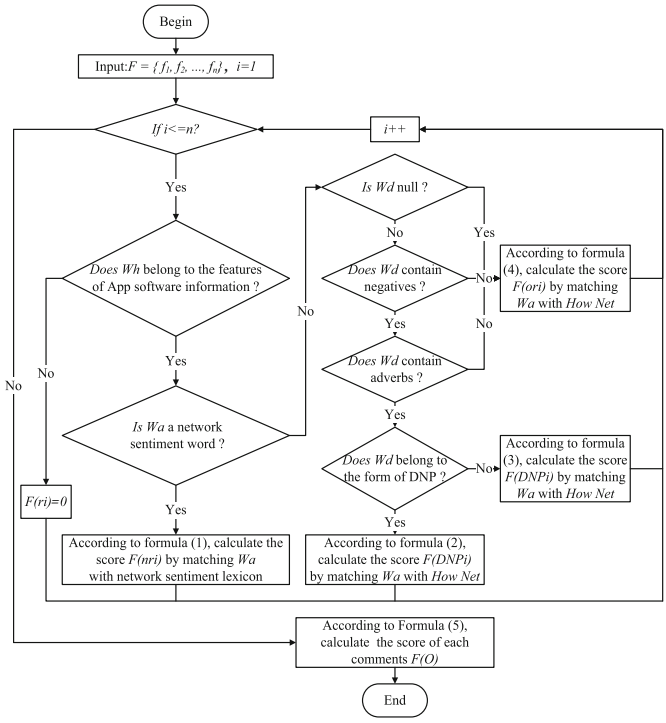
4.1 Sentiment Tendency Degree Quantification

Currently, the user's emotional tendency analysis includes dictionaries-based method and corpus-based method. The dictionaries-based method identify the vocabulary similarity degree between the undetermined polarity words and the fiducially polarity words based on dictionaries. For example, after giving a seed polarity word, the expanded semantic lexicon of synonyms and synonym are found using *Word Net* [13]. The corpus-based method makes the identification through the co-occurrence model or modified model of the undetermined words and the fiducially words, such as word frequency mutual information, syntactic dependencies and association rules [14].

In user's comment information of App software, the intensity difference between network sentiment words and adverbs will influence user's sentiment tendency degree. And the semantic difference result from the co-occurrence words order of adverbs and negatives also will influence user's sentiment tendency degree. So, we further analyze the polarity words of sentiment words. Combining the grammatical relations among Wh, Wa and Wd in f, the sentiment corpus in *How Net* and network sentiment words, the degree of the sentiment tendency is quantified gradually. Finally, the comprehensive score of each user comment is calculated. Wh include nouns, adverbs and noun phrases. Wa include adjectives, verbs and network sentiment words. Wd include adverbs and negatives. The flow diagram of sentiment tendency degree quantification for user's comment information is as shown in Fig. 2.

(1) Processing of Network Sentiment Words. As user's comment information on App software is a typical network comment, most users would like to use network sentiment words (including adjectives, nouns and verbs) to express sentiment tendency to App software. For example, '这就是个SB软件', where 'SB' is a noun expressing a strong negative sentiment to the current App software. Another example, '这款软件超赞', where the adverbs '超' and the network sentiment word '赞' express a strong positive sentiment to App software. Currently, these network sentiment words cannot be found in *How Net*. Due to they can express user's sentiment tendency to the features of App software, the consistency judgment of user comments on App software will be influenced. Therefore, we build a network sentiment lexicon including 137 high-frequency network sentiment words, such as 'TMD', 'SB', '神器' and so on. We define the weight and polarity of these network sentiment words, which express user's strong attitude to features of App software. We define the quantitative formula of sentiment tendency degree as follows.

$$F(nr) = F(nd) * F(na) \tag{1}$$



,

Fig. 2. The flow diagram of sentiment tendency degree quantification for user's comment information.

$F(nr)$ represents the score of emotional tendency degree that made by the current network word on certain feature comment. $F(nd)$ is the polar parameter of adverbs, and $F(na)$ indicates the original polarity of emotional words in network sentiment lexicon.

(2) Processing of Degree Adverb and Co-occurrence of Degree Adverb and Negative Adverb. As the adverbs and negatives in user comments on App software have important impacts on the quantitative result of the sentiment tendency degree and on the accuracy of consistency judgment for user comments, the adverbs and negatives must be processed. The existing researches show that adverbs have different intensity grades. For example, Xu et al. [13] divide adverbs into two categories and set four intensity grades. Lin [14] divide adverbs into six categories based on *How Net* dictionary, where the polarity parameters of '最' and '超' are '1.6'. So we combine the '最' and '超' categories into the '超' category. Finally we divide adverbs in user comments on App software into five different polarity parameter categories, which as shown in Table 1.

When processing negatives, Yao et al. [15] adopt a sentiment polarity processing method for negatives prefix, which negated and then divided by 2. According

Table 1. Degree adverbs categories and polarity parameters

Category	Polarity Parameter	Word(some examples)	Amount
1.最	1.6	非常、特别	99
2.很	1.4	很、太	39
3.较	1.2	蛮、挺	36
4.稍	0.8	有点、略	18
5.欠	0.6	相对、大概	8

to this algorithm, the sentiment tendency degrees of '不很满意' or '很不满意' are the same. In fact, they are different because this algorithm does not consider the semantic difference resulted from the order when negatives and adverbs co-occur. Aimed at the user comments on App software, we set the polarity parameter of negatives as -0.5 based on [16]. The co-occurrence order of negatives and adverbs can be divided into two categories.

(1) When adverbs are prior to negatives, it affirms negatives degree of adverb, and the negative degree is increasing. For example, '画面不好' and '画面很不好'. So we calculate the degree of the sentiment tendency described in [14], which formula is as follows.

$$F(DNP) = -0.5 * F(d) * F(a). \tag{2}$$

$F(DNP)$ indicates the score of emotional tendency degree that made by adverbs prior to negatives. -0.5 is the polar parameter of negatives. $F(d)$ is the polar parameter of adverbs. $F(a)$ is the original polarity of emotional words.

(2) When negatives are prior to adverbs, it negates the degree of adverbs, and the degree of adverbs is weakening. For example, '画面不很好' and '画面相对好'. The sentiment degrees of the two sentences are identical in principle. In the case, they are capable of inferring with each other semantically [16]. So we set the polarity parameter of this type of negative adverb as '欠' category in Table 1, which formula is as follows.

$$F(NDP) = 0.6 * F(d) * F(a). \tag{3}$$

$F(NDP)$ shows that the score of emotional tendency degree made by negatives prior to adverbs. 0.6 is polar parameter of negatives.

(3) Sentiment Tendency Degree Quantification. For App software, the degree of user's sentiment tendency on features depends on adverbs, negatives and sentiment words. We calculate the degree of user's sentiment tendency in user's comment information of the App software using the following formula.

$$F(or) = F(oa) * F(od) * F(on) \tag{4}$$

$F(oa)$ is the original polarity of emotional word Wh. $F(od)$ is the polar parameter of adverbs. $F(on)$ is the polar parameter of negatives. When adverbs or negatives are null, the polar parameter is set to 1. When the adverbs and negatives are co-occurrence, then the score should be calculated according formulas (2) and (3).

(4) Comprehensive Score Calculation of Sentiment Tendency Degree. In F, there are differences in user's sentiment tendency degree of each f. In order to judge the consistency between user's comment information and user's mark more exactly, we calculate the sentiment tendency degrees of features in F comprehensively. The formula is as follows.

$$F(O) = \sum_{i=1}^{m} F(ri)/m \tag{5}$$

Wherein, $F(ri)$ means the score of emotional tendency degree for the ith features comment in the users' comments. $F(ri)$ includes $F(nr)$, $F(or)$ and other conditions. m represents the number of f which is consistent with the feature in the App software information, that is the number of $f(ri)! = 0$.

4.2 Range Division of Comprehensive Score

Based on Sect. 4.1, the user's mark of App software depends on the comprehensive score $F(O)$. Accordingly, we divide $F(O)$ into five marks, which respectively corresponding to user's mark of App software. User's mark include five-mark (very good), four-mark (good), three-mark (general), two-mark (bad) and one-mark (very bad). Then we judge the consistency of user's comments on App software by comparing the grades of $F(O)$ and user's mark.

The range of $F(O)$ can be calculated in $[-1.6, +1.6]$ based on Sect. 4.1. In Table 1, the maximum value of polarity parameters is 1.6, the polarities of positive and negative sentiment words are respectively $+1$ and -1, and the polarity of negative words is -0.5. Therefore, the maximum value of five-mark is $1.6 * 1 = 1.6$. The maximum value of four-mark is 1 and the minimum is $-1 * (-0.5) = 0.5$. So the value range of five-mark is $(1, 1.6]$, the range of four-mark is $[0.5, 1]$. The range is as shown in Table 2.

Table 2. Range division of comprehensive score

User's mark	5	4	3	2	1
$F(O)$ Range	$(1, 1.6]$	$[0.5, 1]$	$(-0.5, 0.5)$	$[-1, -0.5]$	$[-1.6, -1)$

5 Experimental Result and Analysis

5.1 Experimental Data Source and Processing

In order to verify the effectiveness of the method in this paper, 20296 pieces of user's comment information of 7 categories and 27 types of App software are picked up randomly from Android electronic market. The comment time is from March 2014 to November 2015. We establish App software information database and user's comment database. Some information is shown in Tables 3 and 4.

Table 3. App software information (some examples)

App Types	App Softw-are Name	Introduction	Version No.	Comment Amount
影音	酷狗音乐	【核心功能】1.更人性化的界面...	7.3.9	980
	百度视频	【活动主题】：登录就送好...	7.12.0	960
社交	QQ	-主要功能-聊天消息：随时随地聊天...	5.6.1	970
	同城蜜恋	同城蜜恋是一个专业的婚恋相亲...	2.8	1000

Table 4. User's comment information of App software (some examples)

User Name	User ID	App Softw-are Name	User's Mark	User's Comment Information
周隆	601014127921598465	QQ	4	SB软件
圆音	599514748127678479	QQ	4	您透婆婆十块钱

As shown in Table 4, some users give high user's mark on App software, the sentiment tendency of user's comment information expresses negative attitude, or the comment is not aimed at features of App software. For example, user '周隆' gives user's mark '4' on App software 'QQ', but the sentiment tendency of its user's comment information '软件' expresses negative attitude. And user '圆音' gives user's mark '4' on software 'QQ', but the sentiment tendency of its user's comment information '您透婆婆十块钱' is not aimed at features of 'QQ'. Therefore, there is inconsistency in the user comments on App software.

Table 5. Data processing results (some examples)

Before word segmentation	After word segmentation	The set of Feature-sentiment-word-pairs
不太好	不/d 太/d 好/a	{ (null, (不/d 太/d), 好/a) }
软件很赞	软件/n 很/d 赞/v	{ (软件/n, 很/d, 赞/v) }

ICTCLAS 2015 is used as data preprocessing tool in this paper, which completes comment word segmentation and part-of-speech marking. Then feature-sentiment-word-pairs of App software are extracted. Some data processing results are shown in Table 5.

5.2 Experimental Result and Analysis

Aiming at the above experimental data, the consistency of user comments is judged. In order to verify the effectiveness of our methods, some App software users are invited to manual mark the consistency of comment information. Some judgment results of consistency are shown in Table 6 ('Consistency between UCI and UM(manual marking)' means the result that artificial marking is used to judge whether the 'User's Comment Information' is consistent with the 'User's Mark'. 'Consistency between UCI and ASI' means the result judged by the method set out in Sect. 3. 'UM after Quantification' means the corresponding mark converted from F (O) in Sect. 4. 'Consistency between UCI and UM' means the comparison result between 'User's Mark after quantization' and 'User's Mark'.).

Table 6. Conformity Judgment Results (some examples)

User ID	App Software Name	User's Mark	User's Comment Information	Consistency between UCI and UM(manual marking)	Consistency between UCI and ASI	UM after Quantification	Consistency between UCI and UM
589651758 389792771	美图秀秀	5	很好，很满意,很不错	Yes	Yes	5	Yes
601724559 133118478	QQ	4	劳资日穿墙	No	No	-	No
543218047 436066818	畅读	4	TMD，SB软件	No	Yes	1	No

From the above experiment, the average consistency between user's comment information and user's mark only cover 40.06 %. The result of user comments consistency on various App software is shown in Fig. 3, in which '社交类' only cover 32.98 %. It is obvious that most of user's comment information in the current App software comments is inconsistent with user's mark.

In comparison with the manual marking, the accuracy of consistency judgment using our methods is averagely up to 78.78 %, in which the accuracy of '浏览器类' is the highest, up to 81.67 %, while that of '资讯类' is the lowest, only 76.84 %.

When the method in Sect. 3 is used to judge whether user's comment information is consistent with App software information, Wh of some user's comment information is mistaken as the App software feature. The reason is that Wh is not

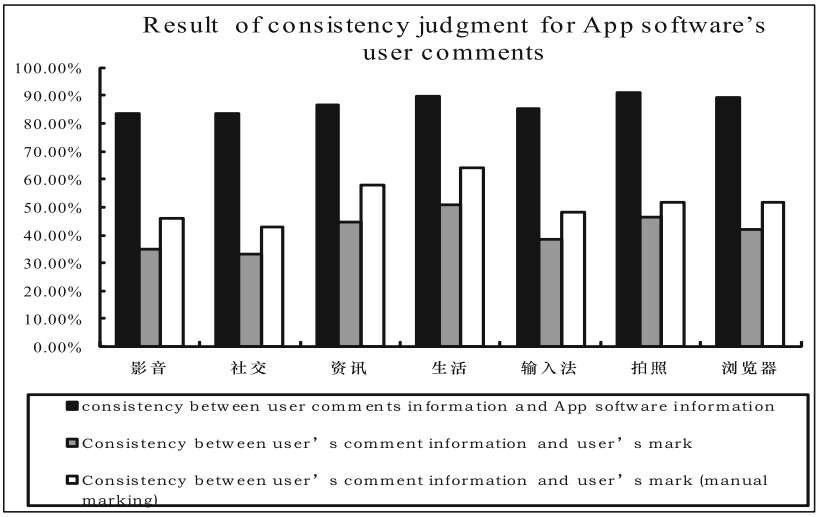
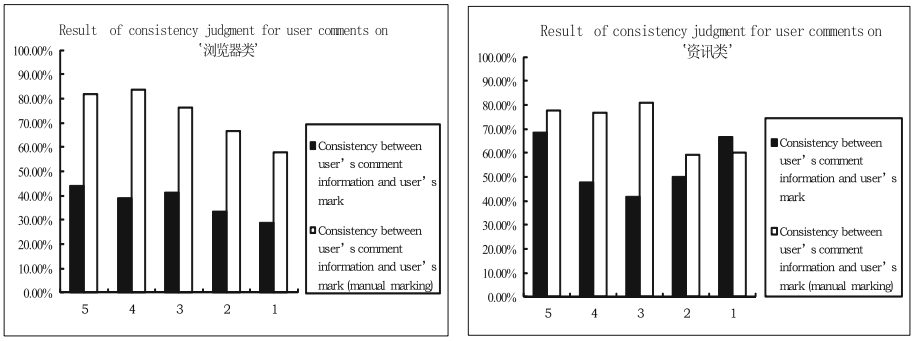


Fig. 3. Result of consistency judgment for App software's user comments.

precise enough to calculate the similarity between Wh and feature words in App software information, which results in a higher consistency between user's comment information and App software information. Then the judgment accuracy of consistency between user's comment information and user marks is influenced.

The consistency of user marks for '浏览器类' and '资讯类' are shown in Fig. 4. The accuracy of two-mark and one-mark are a little low, respectively 66.67 % and 57.89 %, 59.26 % and 60.00 %. This is the common in 7 types of App software.



(a) (b)

Fig. 4. (a) Result of consistency judgment for user comments on '浏览器类'; (b) Result of consistency judgment for user comments on '资讯类'.

There are three reasons that cause the above problem: (1) while extracting the feature-sentiment-word-pairs of App software, we consider little about the general or specific features of the same type App software. (2) Users would use wrongly written or mispronounced characters when they are not satisfied with App software. When we are segmenting words and quantifying user's sentiment tendency to App software, ICTCLAS 2015 and our methods cannot recognize these characters, which can affect the accuracy of consistency judgment. (3) The network sentiment words we collected are not enough, which make the quantification of some network sentiment words inaccurately. So the judgment accuracy of consistency between user's comment information and user's mark is affected. Otherwise, the errors caused by manually marking also affect the experimental results.

6 Conclusion

In this paper, we extracted the features of App software information and feature-sentiment-word-pairs in user's comment information. Firstly, the consistency between user's comment information and App software information is judged. Secondly, user's sentiment tendency to App software is quantified by analyzing the grammatical relations in feature-sentiment-word-pairs of App software combining with dictionaries and network sentiment words. Thirdly, the comprehensive score of App software in each user's comment information is calculated. In order to judge the consistency between user's comment information and user's mark, the comprehensive score is compared with user's mark.

The experimental result shows that our method is practicable. To improve the judgment accuracy of the consistency, the follow issues will be researched further: the feature-sentiment-word-pairs extraction on the general and specific characters of the same type App software, feature similarities of the App software calculation, the network sentiment words collection and the emotional tendency quantification.

Acknowledgments. This research is sponsored by the National Science Foundation of China No. 60703116, 61063006 and 61462049, and the Application Basic Research Plan in Yunnan Province of China No. 2013FZ020.

References

1. Meng, X., Li, Y., Zhu, J.: SO information. Social computing in the era of big data: opportunities and challenges. J. Comput. Res. Dev. **50**(12), 2483–2491 (2013)
2. Kim, H.J., Kim, I., Lee, H.G.: The success factors for app store-like platform businesses from the perspective of third-party developers: an empirical study based on a dual model framework. In: Pacific Asia Conference on Information Systems, PACIS 2010, Taipei, Taiwan (2010)
3. Hao, L., Li, X., Tan, Y.: The economic role of rating behavior in third-party application market. In: Proceedings of the International Conference on Information Systems, ICIS, Shanghai, China (2011)

4. Liu, Y., Liao, X., Liu, Y.: The impact of online comment on software and platforms pricing strategies. J. Syst. Eng. **29**(4), 560–570 (2014)
5. Zhou, J., Sun, J., Kumaripaba, A., Dinesh, W., Mika, Y.: Pervasive social computing: augmenting five facets of human intelligence. J. Ambient Intell. Hum. Comput. **3**(2), 153–166 (2012)
6. Gao, C., Xu, H., Hu, J., Zhou, Y.F.: AR-tracker: track the dynamics of mobile apps via user comment mining. In: 2015 IEEE Symposium on Service-Oriented System Engineering (SOSE), pp. 284–290. IEEE (2015)
7. AlQuwayfili, N., AlRomi, N., AlZakari, N., Al-Khalifa, H.S.: Towards classifying applications in mobile phone markets: the case of religious apps. In: 2013 International Conference on Current Trends in Information Technology (CTIT), pp. 177–180. IEEE (2013)
8. Khalid, H.: On identifying user complaints of iOS apps. In: 35th International Conference on Software Engineering (ICSE), pp. 1474–1476. IEEE (2013)
9. Islam, M.R.: Numeric rating of apps on google play store by sentiment analysis on user reviews. In: 1st International Conference on Electrical Engineering and Information Communication Technology, pp. 1–4 (2014)
10. Guzman, E., Maalej, W.: How do users like this feature? a fine grained sentiment analysis of app reviews. In: 22nd International Requirements Engineering Conference (RE), pp. 153–162. IEEE (2014)
11. Hu, Z., Zheng, X.: Product recommendation algorithm based on users' comments mining. J. Zhejiang Univ. (Eng. Sci.) **47**(8), 1475–1485 (2013)
12. Zhu, Z., Sun, J.: Improved vocabulary semantic similarity calculation based on *HowNet*. J. Comput. Appl. **33**(8), 2276–2279, 2288 (2013)
13. Xu, L., Lin, H., Yang, Z.: Text orientation identification based on semantic comprehension. J. Chin. Inform. Procession **21**(1), 96–101 (2007)
14. Lin, Q.: Design and Implementation of the Product Comments Analysis System Based on Affective Computing. Fudan University (2013)
15. Yao, T., Lou, D.: Research on semantic orientation analysis for topics in Chinese sentences. J. Chin. Inform. Procession **21**(5), 73–79 (2007)
16. Yin, H.: Research on Syntax and Semantics for Co-occurring of Negatives and Adverbs. Graduate School of Chinese Academy of Social Sciences, Beijing (2008)