A BERT and Topic Model Based Approach to reviews Requirements Analysis

Jun Yang
College of Systems Engineering
National University of Defense Technology
Changsha, China
15871438382@163.com

*Yajie Dou**
College of Systems Engineering
National University of Defense Technology
Changsha, China
yajiedou-nudt@163.com

Xiangqian Xu
College of Systems Engineering
National University of Defense Technology
Changsha, China
xuxiangqian@163.com

Yufeng Ma
College of Systems Engineering
National University of Defense Technology
Changsha, China
mashuang9707 @163.com

Yuejin Tan
College of Systems Engineering
National University of Defense Technology
Changsha, China
yjtan@nudt.edu.cn

2021 14th International Symposium on Computational Intelligence and Design (ISCID) | 978-1-6654-3856-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ISCID52796.2021.00094

***Abstract*—With the rise of mobile applications, user reviews are an important avenue of user feedback in which users may mention different issues in using the software. For example, unresponsiveness, low-level privacy, etc. In order to extract effective requirement information and problematic feedback from these huge user reviews, this paper proposes a review non-functional requirement analysis method based on BERT model and topic model (NRABL). Firstly, we use BERT model to classify the reviews with multi-labels classification and then use LDA (latent Dirichlet allocation) to extract topic and review analysis, this method can help developers to quickly understand the user's requirements and the specific usage problems.**

Keywords-user reviews analysis; NFR classification; BERT; topic model

I. INTRODUCTION

With the rise of mobile Internet, mobile users can access their needed apps and share their opinions and feedback about the apps on app distribution platforms such as Apple's App Store and Google's Google Play [1], thus generating a large amount of online review information for mobile apps. Studies have shown that user reviews contain valuable information, such as functional and non-functional requirements, user experience, bug reports, etc. For requirement analysts and developers, these huge user reviews can help them understand user requirements and further improve the software.

Review analysis methods are mainly implemented by classifying user reviews then combining text analysis and sentiment analysis. Maalej et al [2] combined text classification, natural language processing and sentiment analysis techniques to classify application reviews into four types: bug reports, feature requests, user experience and ratings. Yang et al [3] established a product review evaluation index system by using online reviews for opinion mining and sentiment analysis to propose product improvement strategies. Kunaefi et al [4] took another perspective to mine user reviews for actions/decisions and the arguments/motivations behind them (mining user actions/decisions along with their respective arguments/reasons), classifying user reviews according to decision content as Acquiring Decision, Recommending Decision, Requesting Decision, Rating Decision, Relinquishing Decision.

Most of the existing studies use multiclassification methods to achieve user review classification, and only a few studies use multi-label classification, which allows a review data to be categorized into multiple categories at the same time [5].Therefore, this paper proposes a review non-functional requirement analysis method based on BERT model and topic model to investigate user reviews from two aspects of classification and topic analysis to better help developers understand users' non-functional requirements (NFR). First, user review is labeled and classified with multi-labels using the BERT model, and the experimental results show that the classification effect of the BERT model is better. Then a topic model was used to extract topic words for each category of user reviews, so that we were able to understand the specific types of issues in the reviews, such as whether they were battery issues or updates, while determining the user's non-functional requirements. The main contributions of this paper are as follows.

(1) A state-of-the-art natural language processing technique BERT model is proposed to implement a multi-label classification method for user reviews, which greatly

2473-3547/21/$31.00 ©2021 IEEE
DOI 10.1109/ISCID52796.2021.00094

improves the accuracy of classifying the non-functional requirements of user reviews.

(2) Further analysis of user reviews, taking into account classification issues, topic analysis, etc., helps developers understand the real needs of users.

Section 2 of this paper presents the relevant research work, Section 3 describes the overall idea of the methodology and the relevant techniques used; Section 4 describes our experimental and presents and discusses the experimental results; Section 5 concludes the paper.

II. Related work

In recent years, there have been related scholars have conducted relevant research on methods for automatic classification of user reviews. Messaoud et al [5] proposed a multi-label active learning method for a large number of unlabeled and unstructured review dataset .Mcllroy et al [6] studied the review data of more than 20 apps in Google Play and App Store, focusing on the multi-label classification problem of user reviews, and several classifiers, namely, naive Bayes, J48 decision tree and support vector machine was conducted and the results showed that SVM (Support Vector Machine) was the best.

To further understand the user requirements in reviews, many scholars have analyzed the review data for functional and non-functional requirements. Rahimi et al [8] used ensemble ML technique to classify FR statements into solution requirements. enablement requirements, action constraint requirements, attribute constraint requirements, definition requirements, and policy requirements into six categories. Baker et al [9] used artificial neural networks (ANN) and convolutional neural networks (CNN) to classify non-functional requirements (NFR) into five categories: maintainability, operability, performance, security and usability. Tiun et al [10] found that fast text had the best classification performance in functional and non-functional requirements classification, especially for binary texts with short text length and small vocabulary.

To better understand users' fine-grained requirements, Bakiu et al [11] proposed to automatically extract and visualize users' satisfaction with UUX (usability and user experience) of specific software features, based on sentiment analysis of user reviews, the keywords that frequently appear in user reviews were used as identifying features. Blei et al [12] proposed the topic mining model LDA in 2003 model, which is the classical model in topic mining. Jiang Wei et al [13] proposed an associative LDA model for the opinion mining problem domain and applied it to user online reviews. Alec et al [14] proposed a pre-trained OpenAI-GPT model using Transformer's encoder. The goal of the model is to learn a generic representation that can be applied on a large number of tasks. Shreda et al [15] presented random and word embedding vectorization methods to feed in different ML classifiers including traditional approaches and deep learning approaches.

In summary, whether it is based on reviews classification for automatic user requirement analysis or keyword-based extraction of user information, the most important thing is the rationality and accuracy of user comment classification, and it is necessary to consider that user reviews are not classified into only one category, so we implement the extraction of topic words under each type for further analysis based on multi-label classification, which can help us to maximize the mining of user comments ,and is the key for developers to be able to understand user needs and implement subsequent steps successfully.

III. Review Analysis Methods Based on BERT and Topic Models

The method in this paper is divided into two phases: classification phase and topic mining phase. First, according to the types of reviews identified in Section 3.2, the reviews are classified by the BERT model with multi-labels, which can yield information about the reviews containing user requirements, and then the classified review data are subjected to LDA thematic analysis separately to further obtain the aspects of the user's non-functional requirements that are of key concern. Fig. 1 illustrates the overall framework processing flow of the approach, which we describe below in turn.

A. Data pre-processing

In this paper, we use NLP-related techniques to achieve pre-processing of APP review data, specifically including filtering out non-English words, removing all irrelevant characters, such as useless emoji, non-alphabetic, numeric characters, converting characters such as "@$&" to "at dollar, and", restoring all abbreviated forms of words, and converting all characters to lowercase. Finally, users on mobile devices may post comments on their phones with typos (i.e. misspelled words), abbreviations and acronyms, such as "usefull" and "excelent", and misspelled words such as "frnds" is an abbreviation for "friends". So we need to reduce the misspelled, abbreviated and abbreviated words[17].

In addition to, this paper performs pre-processing of the data with stopwords to remove useless and colloquial words, which may be present at high frequencies but do not have any practical meaning and are useless for the classification task, and can reduce the noise in the dataset.

B. Multi-labels classification of non-functional requirements

In order to classify user reviews, the types of reviews need to be determined, this paper focuses on classifying user reviews into non-functional requirements, following the set of types of user review non-functional requirements (NFRs) proposed by Kurtanovic and Maalej (2017), as shown in Table 1, they user review non-functional requirements are classified into four categories: Dependability(Dep), Usability(Usa), Performance(Per), and Supportability(Sup), we focus on extracting and classifying NFRs present in app store reviews. These reviews were classified into reviews raising valid user quality concerns (NFRs) and others (miscellaneous), a classification that can be well understood by developers. Table 1 User comments NFR classification categories.

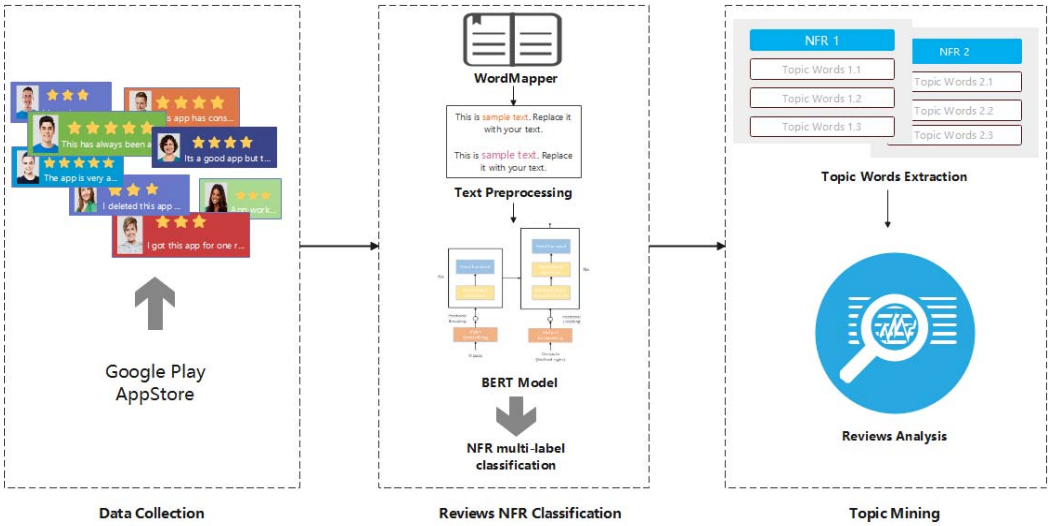

Figure 1. Article Methodology Architecture

In this paper, the classification of user reviews is done by multi-label classification, as NFRs are by nature vague, usually more than one NFR is described in a user review, it can be classified in more than one category, for example, the review " Delayed and inaccurate in my I paid for this version that's the disappointing part. please make it more accurate" is classified under Dependability and Performance.

The BERT model is based on a bidirectional Transformer encoder implementation. Where Transformer is made up of 6 Encoders stacked together to form his encoding layer and 6 Decoders stacked together to form its decoding layer. The model needs to perform Word Embedding on the input data, then perform self-attentive processing and feedforward neural network computation, and the output obtained will go to the next Encoder layer. “Fig. 2” show the general structure of BERT.

TABLE I. USER COMMENTS NFR CLASSIFICATION CATEGORIES

Type of NFR	*Description*	*Example*
Usability	An NFR attribute to which an app or software achieves user satisfaction or goals effectively and efficiently	I love the app because it connects to my smartwatch seamlessly.
Dependability	A set of NFR attribute related to the ability of software to maintain its level of performance over a specified period of time and under specified conditions conditions	App previously had a lot of bugs, before every time I would go to open the app it crashed immediately.
Performance	NFR attribute related to software run time, speed and scalability	It's a great app to run on your phone and it doesn't slow
Supportability	NFR attribute related to aspects of maintainability and interoperability with other apps or devices	It says the version isn't supported. i am very disappointed. please update the app.

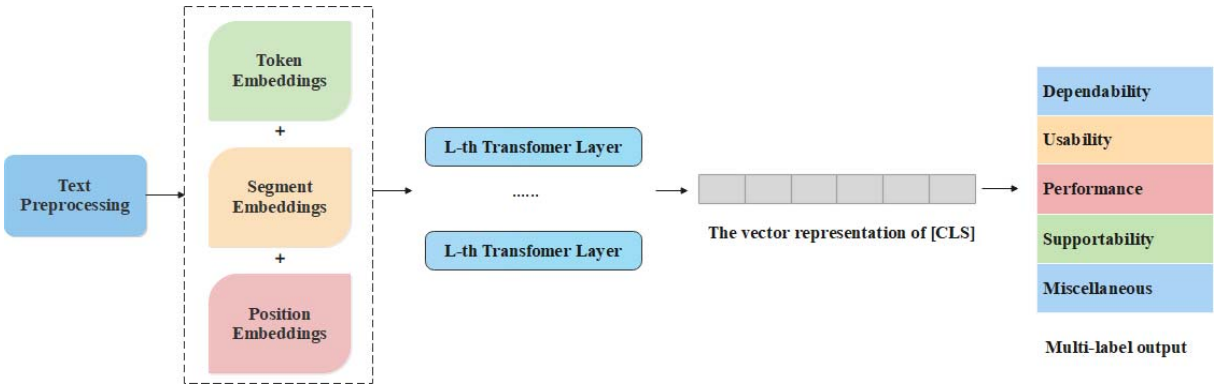

Figure 2. The general structure of BERT

C. NFR THEME MINING

User non-functional requirement classification can only represent the general requirements of users at the macro level, and developers cannot directly perform requirement analysis and subsequent improvement based on NFR classification. Therefore, it is necessary to further continue to extract the comment information under each NFR classification to achieve more fine-grained user comment mining, which in this paper mainly includes topic extraction and requirement identification and analysis.

In machine learning and natural language processing, a topic model is a statistical model used to discover abstract "topics" in a collection of documents, and is often used to discover hidden semantic structures in the body of a text, where each user comment expresses some topic, so that specific words appear in comments on different topics. In this paper, we use the model of LDA (latent Dirichlet allocation) [17] to generate topic words, which is a typical bag-of-words model, i.e., a comment is composed of a set of words without considering the order of the words, which simplifies the complexity of the semantic association problem.

1) Topic extraction : In this paper, the four non-functional requirement classification comments obtained based on the BERT model are input into the LDA model, and the output is 10 topics and the corresponding topic words.The basic idea of LAD topic word generation, based on the topic distribution corresponding to the document from the Dirichlet distribution β The topics are generated by sampling from the z ,the topics z corresponding word distribution φ_z φ_z , the word distribution φ_z is generated from the Dirichlet distribution with parameters β of the Dirichlet distribution is generated, and then the polynomial distribution of words φ_z and then sampling from the polynomial distribution of words to generate the final words ω. Table 2 shows the results of topic generation for the "Dep" type. In order to visualize the topic content, a word cloud map is used to visualize the topic words under the "Dep" category. (Fig. 3)

2) Requirements identification and analysis: In this stage, the input is the topic words obtained by the LDA model and the output is the user comments with high relevance to the topic. The basic idea is that the t-vector is assumed to be $(t_1, t_2, \ldots, t_n)$, each t_i represents the probability of a user comment being assigned to each topic, and assuming the number of topics is 8, the review i vector of comments t_i is $(t_{i-1}, t_{i-2}, \ldots, t_{i-8})$, and so on, for topic 1, if the probability value of a particular comment at t_{i-1} position has the highest probability value among all comments, then that comment is the sentence with the highest relevance to the topic[17]. In this way, we select the top 5 comments with the highest relevance to the topic as a reference for non-functional requirement analysis.

TABLE II. "DEP" TOPIC WORDS

Topic 1	contact using crashes zillow home much time crash back
Topic 2	work week update crash even time keyboard money nothing seriously
Topic 3	email file account error program number website experiencing need
Topic 4	message time inaccurate work sound working problem location load
Topic 5	waste money work heartbeat alone stated simply dont time please
Topic 6	login name area google info password value wrong contour special
Topic 7	map google station data simple story customer message delete along
Topic 8	week loved spot stopped house last please deleted listed reinstalled
Topic 9	sure house home mile right location please photo update zillow
Topic 10	step fixed needs many direction movement frustrating showing wrong

Figure 3. "Dep" Topic words cloud map

IV. EXPERIMENTAL PROCEDURE AND RESULTS

A. Research questions

We use natural language processing techniques to classify user reviews with NFR multi-labels, topic extraction and comment sentence analysis, our aim is to provide user requirements to app developers so that they can find the key information in the large amount of review information and reduce the time consumption of developers in obtaining user feedback. Our experimental design was guided by the following two main questions.

RQ1: Is our multi-label classification method based on the BERT model more advantageous compared to the traditional NFR-based multi-label classification?

RQ2: Can the NRABL approach proposed in this paper help developers capture user requirements and inform subsequent maintenance and updates?

B. Data preparation

To answer our research questions, this paper collects user review data from Apple's App Store and Google's Google Play. In selecting apps, we follow several principles: (1) choosing popular apps in the market, which have a large

volume of users and are regularly updated and maintained by the developers; (2) including different categories of apps that ensures the diversity of review data content, which includes Music, Videos, Games, Education, Entertainment, etc.

To answer question 1 we need to manually label the 6000 data in the dataset. The results of the classification are then compared with the manually labeled true set. Table 3 shows the overall results of our labeled dataset, this true set was used to generate a training set and a test set for our NFR multi-label classification algorithm, we randomly selected 80% as the training set and the remaining 20% formed the test set.

Table 3 shows the number of sentences and the proportion of comments under the NFR type in the above dataset. “Miscellaneous” types are the most frequent, accounting for 60.3% of the dataset, which indicates that a large proportion of users' comments are not related to NFR, which is in line with common sense, and we often see random comments from users, such as "good!", which cannot be of practical use to developers. The categories "Usability" and "Dependability" are also relatively high, at 18.5% and 17.2% respectively, while "Performance " and "Supportability" appear in only 8.4% and 8.3% of sentences, respectively, which indicates that user reviews are less about app performance and supportability.

TABLE III. DATA SET

Type	*number*	*radio*
Usability	1110	18.5%
Dependability	1034	17.2%
Performance	504	8.4%
Supportability	495	8.3%
Miscellaneous	3616	60.3%

C. *valuation indicators*

RQ1 metrics: we use the Precision, Recall and F1 metrics commonly used in machine learning to evaluate the results, as defined in the following way.

Considering that the NFR classification in this paper is a multi-label classification problem, we use the average of the individual classification results as a measure, i.e., Macro Pre, Macro Rec, MacroF1. Also hamming loss is an important measure. where n denotes the number of categories in the classification, where p is the number of examples, h(-) is a multi-label classification. Δ stands for the symmetric difference between two sets.

$$
\begin{aligned}
MacroPre &= \frac{1}{n}\sum_{i=1}^{n} Precision \\
MacroRec &= \frac{1}{n}\sum_{i=1}^{n} Recall \\
MacroF1 &= \frac{1}{n}\sum_{i=1}^{n} F-measure \\
HammingLoss &= \frac{1}{p}\sum_{i=1}^{n}\frac{1}{n}|h(x_i)\Delta Y_i
\end{aligned}
\qquad (1)
$$

D. *Analysis of results*

RQ1: Is our multi-label classification method based on the BERT model more advantageous compared to the traditional NFR-based multi-label classification?

Efficient classification of NFR is the basis for achieving accurate user feedback for developers, and to express the efficiency of our classification algorithm, we designed different multi-label classification algorithms for comparison, using the classical Naïve Bayes, Support Vector Machine, Random Forest, ML-KNN, and the BERT model in this paper to compare the classification results.

Table 4 shows the results for Hamming Loss, Macro Pre, Macro Rec, and MacroF1 under each classification algorithm. We see that the BERT model achieves better classification results for all the different measures.

TABLE IV. RESULTS OF DIFFERENT CLASSIFICATION METHODS

Classifier	*Hamming Loss*	*Macro Pre*	*Macro Rec*	*MacroF1*
NB	0.22	0.48	0.23	0.26
SVM	0.20	0.58	0.49	0.51
RF	0.14	0.68	0.50	0.56
ML-KNN	0.18	0.54	0.45	0.50
BERT	**0.06**	**0.70**	**0.65**	**0.66**

TABLE V. CLASSIFICATION RESULTS OF REVIEWS NFR BASED ON BERT MODEL

Categories	*Precision*	*Recall*	*F1-score*	*Auc*
Usability	0.606	0.615	0.636	0.819
Dependability	0.523	0.821	0.650	0.873
Performance	0.723	0.556	0.618	0.870
Supportability	0.750	0.519	0.622	0.860
Mis	0.908	0.758	0.780	0.891
average	0.702	0.653	0.6612	0.863

RQ2: Can the NRABL approach proposed in this paper help developers capture user requirements and inform subsequent maintenance and updates?

User reviews of mobile applications are important source of user requirement feedback, with a large amount of review data and rich product experience information, and there is an important value in mining the requirements of these feedback information through mobile phones, which can help developers understand user requirements quickly and intuitively. In this paper, we propose a review requirement analysis method NRABL based on BERT model and topic model. Firstly, we perform multi-label classification of user reviews for NFR, and then perform topic mining on the review data under each category to generate topic words, and analyze the reviews with high relevance to the topic words to help developers quickly understand the problem type of each NFR. multi-label classification can help developers automatically extract categorized non-functional requirements from a large amount of comment data, and LDA topic model can help developers understand the user feedback problem situation under each non-functional requirement.

In order to visualize the topic content, 8 topic words were extracted for each non-functional requirement based on the most representative words in each topic and combined

with expert knowledge. For example, Table 6 shows that the common words under the "Performance" type are "stability" "slow" "bug", etc. Based on this information, the developer can further get the information about what problems the user is experiencing.

TABLE VI. THEME EXTRACTION RESULTS

Dependability	*Usability*	*Performance*	*Supportability*
Contact	limit	stability	support
crash	useless	slow	update
update	feature	bug	fitbit
fix	easy	response	sync
problem	help	battery	access
login	save	speed	connection
location	option	hard	start
data	make	wait	work

The analysis of the sentences with high relevance to the topic words can help developers to quickly grasp the user feedback and provide help to improve the maintenance work in the next step. For example, the subject line "dependability" includes contact, crash, update, and the sentences with high relevance to the topic word "contact" are found to be mainly related to the user information and location.

- Besides mining **contact and location data**, and pointing the user to other Avira apps, this app doesn't appear to be useful for, well, Anything.

- The first thing this app wants to do is get your **contact and location**.

V. CONCLUSIONS

In this paper, we propose a review analysis method based on BERT and topic model, which first classifies user reviews into non-functional requirements, and then performs topic mining on the reviews under each category. The experimental results show that the BERT model has good results for the task of classifying the non-functional requirements of user reviews, and that the topics under each type can be mined better using LDA topic analysis, and the method in this paper can help developers understand user requirements and further improve software quality.

REFERENCES

[1] Inukollu V N, Keshamoni D D, Kang T, et al. Factors influencing quality of mobile apps: Role of mobile app development life cycle. international Journal of Software Engineering&Applications, 2014,5(5):15-34

[2] W. Maalej and H. Nabil, "Bug report, feature request, or simply praise? On automatically classifying app reviews," 2015 IEEE 23rd International Requirements Engineering Conference (RE), 2015, pp. 116-125.

[3] Yang C, Wu L, Tan K, Yu C, Zhou Y, Tao Y, Song Y. Online User Review Analysis for Product Evaluation and Improvement. journal of Theoretical and Applied Electronic Commerce Research. 2021; 16(5):1598-1611.

[4] A. Kunaefi and M. Aritsugi, "Extracting Arguments Based on User Decisions in App Reviews," in IEEE Access, vol. 9, pp. 45078-45094,2021, doi: 10.1109/ access.2021.3067000.

[5] Messaoud M B, Jenhani J, Jemaa N B, et al. A multi-label active learning approach for mobile app user review classification//Proceedings of the 2019 International Conference on Knowledge Science, Engineering and Management. athens, Greece, 2019:805-816.

[6] McIlroy, S., Ali, N., Khalid, H. Hassan AE. analyzing and automatically labelling the types of user issues that are raised in mobile app reviews. Empirical Software Engineering, 2016,21(3):1067-1106

[7] V. T. Dhinakaran, R. Pulle, N. Ajmeri and P. K. Murukannaiah, "App Review Analysis Via Active Learning: Reducing Supervision Effort without Compromising Classification Accuracy," 2018 IEEE 26th International Requirements Engineering Conference (RE), 2018, pp. 170-181, doi: 10.1109/ RE.2018.00026.

[8] Rahimi, N.; Eassa, F.; Elrefaei, L. An Ensemble Machine Learning Technique for Functional Requirement Classification. symmetry 2020, 12, 1601. https: //doi.org/10.3390/sym12101601

[9] C. Baker, L. Deng, S. Chakraborty and J. Dehlinger, "Automatic Multi-class Non-Functional Software Requirements Classification Using Neural Networks," 2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC), 2019, pp. 610-615, doi: 10.1109/COMPSAC.2019.10275.

[10] S Tiun, U A Mokhtar, S H Bakar and S Saad4. Classification of functional and non-functional requirements in software requirements using Word2vec and fast Text. journal of Physics: conference series, 2020,25-27.

[11] E. Bakiu and E. Guzman, "Which Feature is Unusable? Detecting Usability and User Experience Issues from User Reviews," 2017 IEEE 25th International Requirements Engineering Conference Workshops (REW), 2017, pp. 182-187, doi: 10.1109/REW.2017.76.

[12] Blei DM, Ng AY, Jordan MI. latent dirichlet allocation. journal of Machine Learning Research, 2003,3:993-1022.

[13] Li Z, Huang X Y, Jing J, et al. CS Label: an Approach for Labelling Mobile App Reviews[J]. Journal of Computer Science and Technology, 2017, 32(006):1076-1089.

[14] ALEC R,KARTHIK N, TIM S, et al. Improving Language Understanding by Generative Pre-Training [EB/OL]. [2020-07-01].

[15] Q. A. Shreda and A. A. Hanani, "Identifying Non-functional Requirements from Unconstrained Documents using Natural Language Processing and Machine Learning Approaches," in IEEE Access, doi: 10.1109/ACCESS.2021.3052921.

[16] XIAO Jian-Mao; CHEN Shi-Zhan; FENG Zhi-Yong; LIU Peng-Li; XUE-Xiao. An Automatic Analysis of User Reviews Method for APP Evolution and Maintenance [J]. Chinese Journal of Computers,2020,43(11):2184-2202.

[17] Chen Q, Zhang L, Jiang J, Huang XY. review analysis method based on support vector machine and latent dirichlet allocation. ruan Jian Xue Bao/Journal of Software, 2019,30(5):1547-1560 (in Chinese).