ChinaRxiv

AI translation · View original & related papers at
chinarxiv.org/items/chinaxiv-202310.00728

Mobile Application Review Mining: A Survey (Postprint)

Authors: Zhang Ji, Kang Lele, Li Bo

Date: 2023-10-08T00:00:00+00:00

Abstract

[Purpose / Significance] User reviews facilitate mobile application innovation for developers. By systematically reviewing and summarizing literature on mobile application review mining, this study provides references for both mobile application development and review mining practices. [Method / Process] Employing text analysis methods, research on mobile application review mining is categorized into three key themes: review classification, review clustering, and review feature extraction. Based on this framework, the development status of this field is elaborated. [Results / Conclusions] The study reveals that: review classification methods have begun evolving from machine learning to deep learning; review clustering primarily utilizes K-Means and DBSCAN; feature extraction still focuses mainly on explicit features of reviews. In the future, three issues in mobile application review mining warrant further investigation: domain dependency, multi-source information fusion, and review value assessment.

Full Text

A Research Review of Mobile Application Review Mining

Zhang Ji[1], Kang Lele[1], Li Bo[2]
[1]School of Information Management, Nanjing University, Nanjing 210023
[2]Business School, Central South University, Changsha 410083

Abstract

[Purpose/Significance] User reviews are instrumental for developers to achieve mobile application innovation. This paper synthesizes literature related to mobile application review mining to provide references for both mobile application development and review mining research. **[Method/Process]** Using text analysis methods, we categorize relevant research into three key

themes: review classification, review clustering, and review feature extraction. Based on this framework, we elaborate on the development status of this field. **[Result/Conclusion]** Our review reveals that review classification methods have begun evolving from machine learning to deep learning; review clustering primarily employs K-Means and DBSCAN algorithms; and feature extraction remains focused on explicit features of reviews. Future research should explore three critical issues: domain dependence, multi-source information fusion, and review value evaluation.

Keywords: mobile application; review mining; review classification; review clustering; feature extraction

1 Introduction

With the development of mobile internet and the proliferation of mobile devices, mobile applications (apps) have become an indispensable part of daily life. Since Apple launched the App Store in July 2008 and Google introduced Android Market (renamed Google Play Store in 2012) in October 2008, mobile applications have emerged in large numbers. After more than a decade of development, Google Play Store now hosts over 3.45 million apps, while Apple App Store features nearly 2.2 million apps [1]. These applications cover numerous life scenarios, ranging from social media and news, business and entertainment, healthcare and education, to online shopping and financial management. In 2020, the COVID-19 pandemic accelerated mobile device usage habits by 2-3 years, with mobile app downloads reaching 218 billion and average daily mobile device usage exceeding 4 hours per user [2].

This massive demand for mobile applications presents both unprecedented opportunities and significant challenges for developers. First, mobile app stores exhibit distinct open characteristics [3]. Within these stores, functional descriptions, user reviews, and update documentation for any application are publicly visible, meaning that once released, an app faces risks of imitation or even plagiarism. Second, requirements analysis has a typical phased characteristic—applications are developed to meet current needs, yet users continuously generate new demands through interaction. Third, market competition is exceptionally fierce. In specific market segments, dozens of functionally similar apps may compete, enabling users to switch easily between them [4].

Innovation has long been recognized as a key source of competitive advantage for mobile applications [5-6]. Based on novelty, innovation can be categorized as either breakthrough or incremental [7]. Breakthrough innovation involves designing entirely new products or proposing novel design methods—a process from 0 to 1. Incremental innovation, conversely, involves continuous iterative optimization of existing products—a process from 1 to N. Mobile application innovation primarily follows the latter path, requiring long-term maintenance and improvement. Unlike physical product innovation, mobile app innovation

cycles are extremely rapid, with Google Play apps averaging updates every 13 days [8]. To achieve satisfactory market performance amid such frequent updates, developers must promptly collect user feedback.

User innovation theory, first discovered and proposed by Eric von Hippel, suggests that in certain industries, users rather than producers often generate creative product or service ideas [9]. Consequently, producers must shift from traditional self-centered innovation to user-centered innovation, providing platforms that stimulate user creativity [10]. Mobile app stores have created an ideal feedback platform for users and an innovation platform for developers to extract knowledge. These stores allow users to submit ratings (1-5 stars) and open-text reviews [11], typically consisting of titles and bodies. When developing new versions, developers utilize approximately 50% of informative reviews [12]. Informative reviews are those potentially helpful for improving app quality or user experience. However, rapidly filtering informative reviews from massive volumes poses challenges: (1) review quantities are enormous and growing rapidly, with popular Google Play Store apps receiving over 500 reviews daily [13], making manual review impractical; (2) informative reviews comprise only about one-third of total reviews [14], with the remainder being spam, irrelevant comments, or non-reviews [15]; (3) review text is noisy, often containing grammatical errors, misspellings, abbreviations, emojis, and inconsistent punctuation [16]; and (4) mobile app reviews exhibit strong timeliness and high value—prompt developer responses to version-specific bug reports or crash complaints significantly enhance user identity and experience.

Consequently, numerous scholars have explored automatic extraction of valuable information from massive, unstructured, informal review texts to incorporate into software development and promote iterative mobile app innovation. Academic research around mobile app review mining has yielded substantial results, with several systematic reviews already conducted. N. Genc-Nayebi and A. Abran [17] examined five aspects: review mining techniques, domain dependence, review usefulness, spam review identification, and software feature extraction, revealing primary research questions. However, their classification system was somewhat fragmented, and limited literature prevented comprehensive evaluation of review usefulness and spam identification. M. Tavakoli et al. [18] surveyed review mining techniques and tools, categorizing them into supervised machine learning, natural language processing, and feature extraction techniques, while listing contemporary tools. Yet their analysis lacked depth and breadth. Given review mining' s significance in mobile app innovation and recent methodological advances, a renewed literature review is warranted.

This paper' s contributions are threefold: (1) we identify literature utilizing user reviews to drive app innovation; (2) using text analysis methods, we categorize relevant research into review classification, review clustering, and feature extraction to clarify the field' s development status; and (3) we propose future research directions from three perspectives: domain dependence, multi-source information fusion, and review value evaluation.

2 Data Sources and Research Framework

2.1 Data Sources

For English literature, we selected the SCI-E, SSCI, and CPCI databases from the Web of Science Core Collection. Based on synonym expansion and preliminary retrieval analysis, our search query was: (TS=("user reviews"*or*"*consumerreview*" or "user feedback" or "user comment")*and*TS = ("*mobileapp*" or "mobile application"*or*"*appstore*" or "app market"))*or*(TS = ("*appreview*" or "application review$")), limited to English-language articles, reviews, and proceedings papers published between 2009-2020. This yielded 54 literature samples related to mobile app innovation and review mining. For Chinese literature, we searched the CNKI full-text database of core journals using: (su=('用户评论' + '用户反馈' + '用户评价') and ('移动应用' + '应用程序' + '应用商店' + '应用市场' + 'app')) or (su= 'app 评论' + '应用评论'), spanning 2009-2020, resulting in 13 relevant papers. Combining these 67 Chinese and English sources, we systematically reviewed research on user review-driven app innovation.

2.2 Research Framework

The frequency distribution of keywords or subject terms that express core content can reveal a field' s development status [19]. Using CiteSpace V [20], we extracted noun phrases from titles, abstracts, keywords, and supplementary keywords of 54 English papers, obtaining 226 noun phrases. We further processed these results by: (1) removing search terms and phrases with identical meanings (e.g., "mobile app reviews"); (2) merging phrases expressing identical themes; and (3) retaining themes with frequency >3, sorted by descending frequency, as shown in Table 1 .

Table 1 High-frequency themes and original noun phrases

Merged Theme	Original Noun Phrases
Informative Reviews	valuable information(4), bug report(4), feature request(4), informative reviews(2), eliciting such critical information(2), app issues(2), user opinions(2), sudden change(2), users needs(2), effective review(1), extracting informative user reviews(1), acquiring knowledge(1), crucial information(1), potential problem(1), important information(1), bug reporting(1), different points(1), major concern(1)

Merged Theme	Original Noun Phrases
App Innovation	actionable software maintenance request(2), evolution work(2), app update(2), release planning(2), recommended software change(2), evolution tasks(2), future maintenance(2), software evolution(2), app development(2), app software maintenance optimization(1), app maintenance(1), app development information(1), changed requirement(1), accurate evolution plan(1), application evolution(1), app software improvement(1), actionable change tasks(1)
App Review Mining	app review mining(3), analyzing reviews(2), text analysis(2), review analysis(2), user review mining(2), mining user reviews(2), app review analysis(2), analysis of online reviews(1), data mining(1), exploiting user feedback(1), analyzing user reviews(1), addressing user reviews(1), analyzing mobile app reviews(1), analyzing informative crowd reviews(1), effective user review analytics tool(1), automatic user review mining(1), analyzing feedback(1)
App Developer	app developer(13), competitive environment developer(2), application developer(2), original developer(2), individual app developer(1), app developers opportunities(1)
Review Classification	classification(4), app review classification(3), automatic classification(2), text classification(2), categorize user reviews(1), app review classification problems(1), app user review classification(1), classifying app reviews(1), accurate review classification process(1), classifying user reviews(1), associative classification(1), defining suitable classification feature(1)
Sentiment Analysis	sentiment analysis(6), fine grained sentiment analysis(1), computing user sentiments(1), analyzing sentiments(1)
Evaluation Metrics	high accuracy(4), average precision(2), average recall(1)
Review Clustering	clustering similar user change request(1), cohesive subgroups(1), cohesive subsets(1), cluster phrases(1), clustering algorithm(1), clustering reviews(1)

Merged Theme	Original Noun Phrases
Feature Extraction	app feature extraction(1), fine-grained app feature(1), app feature(1), fine-grained feature(1)

Figure 1 [Figure 1: see original paper] illustrates the technical roadmap for app review mining.

3 Review Mining Techniques

3.1 Review Classification

Review classification aims not only to identify valuable reviews but also to categorize them into finer-grained types. Through manual analysis of 528 reviews from Apple' s App Store, D. Pagano and W. Maalej identified 17 categories [22], with approximately half relevant to mobile app innovation [23-24], such as bug reports, feature requests, and functional defects. H. Khalid focused on negative reviews, manually distinguishing 12 types of user complaints from 6,390 one- or two-star reviews of 20 iOS apps, finding that functional errors, additional feature requests, and program crashes are crucial for developers optimizing apps [25]. Machine learning and deep learning-based classification can rapidly identify useful review types, overcoming the time-consuming and subjective nature of manual classification.

3.1.1 Machine Learning-Based Review Classification The key process for mobile app review classification is shown in Figure 2 [Figure 2: see original paper]. Machine learning requires manual feature construction, where meaningful features significantly improve classification performance. Mobile app review features can be divided into linguistic and external features (see Table 2). External features refer to attributes beyond review text content, while linguistic features primarily include n-grams, part-of-speech tags, and sentiment scores. Classification primarily leverages linguistic features supplemented by review metadata. Common algorithms include Naïve Bayes (NB), K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Decision Tree (DT), and Logistic Regression (LR).

Table 2 Common features and descriptions in machine learning methods

Feature Type	Description
External Features	Attributes beyond review text content, such as star rating, review length, and submission time

Feature Type	Description
n-grams	Assumes word probability depends only on the preceding n-1 words; commonly uses Unigrams, Bigrams, and Trigrams
Part-of-Speech	Different review categories may exhibit different part-of-speech distributions, such as verb tense variations
Sentiment	Calculates review sentiment scores, typically +1 to +5 for positive and -5 to -1 for negative

Combining text analysis, natural language processing, sentiment analysis, and review metadata yields better results than using any single approach [24, 26]. W. Maalej and H. Nabil [27] conducted experiments comparing simple string matching, bag-of-words, natural language processing (stopword removal and lemmatization), review metadata, and sentiment analysis. They found that metadata alone resulted in low accuracy, while combining it with natural language processing achieved 70%-95% accuracy and 80%-90% recall. In all experiments, multiple binary classifiers were more accurate than multi-class classifiers for predicting review types. The following year, W. Maalej et al. [28] further explored combinations of metadata with bag-of-words and natural language processing (particularly bigrams and lemmatization), achieving 88%-92% accuracy and 90%+ recall for all review classifications.

Since supervised methods require extensive manual annotation, active learning and semi-supervised learning have attracted attention for reducing annotation effort without compromising accuracy. While both utilize unlabeled data, they differ in approach: active learning selects the most error-prone samples for expert annotation, minimizing human effort and significantly improving prediction accuracy over random selection in multiple scenarios [29]. Semi-supervised learning, conversely, selects the most confidently classified samples to augment labeled data. Hu Tianyuan et al. [30] comprehensively analyzed user review content and syntactic structures, employing semi-supervised self-learning based on limited review seeds to automatically mine user feedback comments. To control fake reviews that disparage target apps or manipulate rankings, D. J. He et al. [31] proposed a detection method based on PU learning (Positive-unlabeled learning) and behavior density.

Ensemble learning methods aggregate multiple weak supervised models into a strong one, primarily through Bagging and Boosting. Integrating different algorithms (Naïve Bayes, Decision Trees, SVM, Logistic Regression, Neural Networks) via ensemble learning typically outperforms individual models [23, 32].

These studies rely on review text attributes, often producing high-dimensional

models prone to overfitting. N. Jha and A. Mahmoud [33] addressed this using semantic frames to classify reviews into user requirements, bug reports, and others, demonstrating that semantic frames generate lower-dimensional, more accurate models. However, for review summarization, text-based summaries prove more comprehensive than frame-based ones [34].

Finally, mobile app review classification frequently faces class distribution imbalance, causing classifier decision boundary shifts and poor practical performance. Current literature addresses this through: (1) cost-sensitive learning methods that assign different misclassification costs to different classes [38-39]; and (2) resampling techniques that under-sample majority classes and over-sample minority classes [40-41].

3.1.2 Deep Learning-Based Review Classification Deep learning eliminates explicit feature construction and has been widely applied to natural language processing, achieving excellent results in text classification tasks. Wang Ying et al. [35] mined software requirements from user reviews across functional and non-functional dimensions using TextCNN, TextRNN, and Transformer, with results significantly outperforming traditional machine learning methods. Similarly, A. Li et al. [36] proposed a large-scale spam review detection model based on Graph Convolutional Networks that integrates homogeneous and heterogeneous graphs to describe local and global contexts, validated through online and offline performance tests showing superiority over baseline models using review information, user features, and product features. While deep learning generally performs better with large training datasets, it may not achieve expected results on small-scale data. For instance, C. Stanik et al. [37] obtained results comparable to Convolutional Neural Networks using traditional machine learning methods. More complex models also entail higher time costs.

3.2 Review Clustering

Review classification assigns predefined category labels, whereas review clustering groups similar, unlabeled reviews together. Typical clustering algorithms include K-Means and DBSCAN, where K-Means is centroid-based and DBSCAN is density-based. Zhang Liman et al. [42] combined Word2vec with Canopy and K-Means clustering, using Canopy to determine cluster numbers before applying K-Means, effectively identifying and aggregating user requirements. Unlike K-Means, DBSCAN automatically determines cluster numbers without pre-specification, attracting scholarly attention. L. Villarroel et al. [4] employed DBSCAN to cluster bug reports and new feature suggestions, then prioritized these clusters. S. Scalabrino et al. [43] further refined classification by adding four non-functional requirement categories: security issues, performance issues, excessive energy consumption, and usability improvement requests. However, comparative studies of K-Means versus DBSCAN performance on mobile app review datasets remain needed.

3.3 Feature Extraction

While review classification and clustering can mine high-value reviews from large volumes, developers still require manual analysis to identify which specific features users like or dislike. To address this, scholars have proposed various methods to efficiently extract app features and analyze user sentiments toward them. Synthesizing current research and following B. Liu' s classification of aspect extraction methods [44], we categorize relevant literature into four types: frequency-based, syntax-based, supervised learning-based, and topic model-based feature extraction.

3.3.1 Frequency-Based Feature Extraction Frequency-based methods typically use natural language processing tools (ICTCLAS, jieba, Stanford Parser) for part-of-speech tagging, extract nouns and verbs from annotated corpora, and retain words exceeding a threshold as candidate features [44]. P. M. Vu et al. [45] extracted all nouns and verbs as keywords, ranking them by review star rating and frequency to help developers find relevant reviews. However, single words only superficially and sporadically express user opinions, while phrases provide more complete information. Therefore, P. M. Vu et al. [46] used part-of-speech patterns to extract phrases, grouping them by similarity, sorting, and monitoring dynamic changes to help developers capture main user viewpoints.

Many scholars employ association analysis to mine frequently mentioned features, based on the assumption that users use consistent terminology when evaluating app features [47]. Thus, frequently occurring nouns or verbs likely represent app features. To improve mining effectiveness, Lü Hongyu et al. [48] first identified feature request reviews through pattern matching and sentiment analysis, then extracted software features using Apriori association rule mining. Wen Tao et al. [49] similarly used Apriori but further identified <feature word, opinion word> pairs in each review sentence. Given traditional frequent itemset mining algorithms' (e.g., Apriori) computational intensity and limited scalability, C. Gao et al. [50] adopted the Eclat algorithm to rapidly obtain candidate phrases exceeding support thresholds.

3.3.2 Syntax-Based Feature Extraction Syntactic relationships often characterize the evaluation or modification relationships between opinion words and their targets, enabling feature extraction through syntactic parsing [44]. Syntax analysis examines grammatical relationships between words, including constituent structure and dependency analysis. Z. Peng et al. [51] used Stanford Parser to extract verb-noun and noun phrases from dependency analyses, determining phrases as feature requests based on their relevance to topics. Considering that app reviews are always context-related, D. Sun et al. [52] extracted kernel concerns using phrase structure trees and dependency relations, constructing aggregated scenario models for each concern to help requirements analysts more completely and accurately understand user intentions.

3.3.3 Supervised Learning-Based Feature Extraction Feature extraction can be transformed into a sequence labeling task, with primary algorithms including Hidden Markov Model (HMM) and Conditional Random Field (CRF). CRF improves upon HMM by breaking its two unrealistic assumptions—homogeneous Markov property and observation independence—making it more effective and commonly used for feature extraction [53]. Cui Jianling et al. [54] proposed a feature extraction method integrating ontology and CRF, applying deep learning Recursive Autoencoder for sentiment analysis to form a five-tuple <feature, topic, sentiment word, sentence, polarity>, demonstrating that RERM (Requirement Elicitation method based on Review Mining) effectively classifies potential software requirement types, providing more valuable information than ASUM (Aspect and Sentiment Unification Model) [55].

3.3.4 Topic Model-Based Feature Extraction Topic models are generative probabilistic models that aim to discover latent topics in document collections [56]. The most widely used topic model in app review mining is Latent Dirichlet Allocation (LDA) proposed by D. M. Blei, which infers document topic distributions and topic word distributions from observed words across documents [57]. Wang Xinyan et al. [58] used LDA to extract user review topic words and employed Glove vector similarity to obtain topic semantic associations, constructing a semantic association topic graph to provide developers with new methods for efficiently acquiring user requirements. Recent studies have proposed various LDA variants for review mining, including dynamic LDA [59], adaptive online LDA [60], and E-LDA [61].

Beyond LDA and its variants, other topic models like ASUM [55] and non-negative matrix factorization [62] are also employed. Some scholars have compared different models: E. Suprayogi et al. [63] found non-negative matrix factorization performed better than LDA in topic coherence; C. Gao et al. [64] compared latent semantic indexing, LDA, random projection, non-negative matrix factorization, and Gibbs sampling-based LDA, with the latter achieving comparable hit rates to AR-Miner [14] while enabling dynamic tracking of top-ranked review themes.

Most existing topic models are based on LDA and probabilistic latent semantic analysis, but they perform poorly on short texts due to data sparsity and difficulty in disambiguating word meanings [65]. To address this, M. A. Hadi and F. H. Frad [66] proposed an adaptive online Biterm topic model that effectively alleviates sparse word co-occurrence patterns, extracting more coherent and discriminative topics from app reviews.

---

4 Summary and Outlook

Mobile app stores aggregate vast amounts of user experiences and suggestions, providing developers with a crucial competitive advantage. User reviews contain

valuable information such as functional defects and feature requests that help developers optimize apps and enhance user experience. This paper systematically reviews literature from three perspectives: review classification, review clustering, and feature extraction. First, supervised learning remains mainstream for review classification, but methods are evolving from machine learning to deep learning, with deep learning often outperforming traditional methods. Second, review clustering typically follows classification, as specific categories may contain hundreds of reviews, and clustering further reduces the effort required for developers to extract information. While many clustering algorithms exist, comparative studies of different algorithms or parameter settings on mobile app review datasets are lacking. Third, feature extraction literature has primarily focused on explicit features, with topic models partially addressing implicit feature extraction, though dedicated research on implicit feature extraction for app reviews is still needed.

Future research should address three critical issues:

(1) Domain Dependence. Words exhibit different meanings and language patterns across app categories, making most studies applicable only to specific experimental contexts. For example, T. Johann et al. [67] proposed SAFE (a Simple Approach for Feature Extraction), which identified 18 part-of-speech patterns and 5 sentence patterns through manual analysis of app pages and reviews. SAFE achieved 87% precision for well-maintained Google Drive but only 56% average precision across 10 evaluated apps. However, F. A. Shah et al. [68] applied SAFE to 8 different datasets (6 app review datasets, 1 laptop review dataset, and 1 restaurant review dataset), obtaining average precision far below reported performance. Thus, domain adaptation in app review mining represents a challenging research direction.

(2) Multi-Source Information Fusion. Different app stores have distinct management strategies and user communities, resulting in varying feedback for the same app across platforms [69]. Developers must understand not only their own app' s strengths and weaknesses but also competitors' advantages and shortcomings. Therefore, integrating user feedback from different app stores with competitor reviews, product descriptions, and update documentation is essential. Beyond app store mining, runtime app data can also be collected. Fusing app store data with runtime data provides a more comprehensive reflection of app status and more accurate user understanding.

(3) Review Value Evaluation. Mobile app review quality varies significantly, with few useful reviews and many low-value ones. Efficient review value assessment holds practical significance for app development. Most current studies treat review value evaluation as a technical rather than theoretical problem. An appropriate evaluation system must be constructed to analyze mobile app reviews from multiple perspectives. Specifically, reviews should be evaluated across dimensions including information value, temporal value, and innovation value to maximize review value extraction and better drive the evolution of app review mining.

References

[1] Statista Research Department. Number of apps available in leading app stores 2021[EB/OL].[2021-08-02].https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores.

[2] App Annie. State of mobile 2021[EB/OL].[2021-08-02]. https://www.appannie.com/cn/go/state-of-mobile-2021.

[3] BENLIAN A, HILKERT D, HESS T. How open is this platform? The meaning and measurement of platform openness from the complementers' perspective[J]. Journal of information technology, 2015, 30(3): 209-228.

[4] VILLARROEL L, BAVOTA G, RUSSO B, et al. Release planning of mobile apps based on user reviews[C]//Proceedings of the 38th International Conference on Software Engineering. Piscataway: IEEE, 2016: 14-24.

[5] WEN W, ZHU F. Threat of platform-owner entry and complementor responses: evidence from the mobile app market[J]. Strategic management journal, 2019, 40(9): 1336-1367.

[6] MIRIC M, JEPPESEN L B. Does piracy lead to product abandonment or stimulate new product development? evidence from mobile platform-based developer firms[J]. Strategic management journal, 2020, 41(12): 2155-2184.

[7] KAPLAN S, VAKILI K. The double-edged sword of recombination in breakthrough innovation [J]. Strategic management journal, 2015, 36(10): 1435-1457.

[8] COMINO S, MANENTI F M, MARIUZZO F. Updates management in mobile applications: iTunes versus Google Play[J]. Journal of economics & management strategy, 2019, 28(3): 392-419.

[9] HUANG J. Let users create value for you—an interview with Eric von Hippel, founder of user innovation theory[J]. Tsinghua Management Review, 2016(10): 6-11.

[10] YE H J, KANKANHALLI A. User service innovation on mobile phone platforms: investigating impacts of lead userness, toolkit support, and design autonomy[J]. MIS quarterly, 2018, 42(1): 165-188.

[11] SUSAN M M, DAVID S. What makes a helpful online review? a study of customer reviews on amazon.com [J]. MIS Quarterly, 2010, 34(1): 185-200.

[12] PALOMBA F, LINARES-VASQUEZ M, BAVOTA G, et al. Crowdsourcing user reviews to support the evolution of mobile apps[J]. Journal of systems and software, 2018, 137: 143-162.

[13] MCILROY S, SHANG W, ALI N, et al. User reviews of top mobile apps in Apple and Google app stores[J]. Communications of the ACM, 2017, 60(11): 62-67.

[14] CHEN N, LIN J, HOI S C H, et al. AR-miner: mining informative reviews for developers from mobile app marketplace[C]//Proceedings of the 36th international conference on software engineering. New York: ACM, 2014: 767-778.

[15] JINDAL N, LIU B. Opinion spam and analysis[C]//Proceedings of the 2008 international conference on web search and data mining. New York: ACM, 2008: 219-230.

[16] MCILROY S, ALI N, KHALID H, et al. Analyzing and automatically labelling the types of user issues that are raised in mobile app reviews[J]. Empirical software engineering, 2016, 21(3): 1067-1106.

[17] GENC-NAYEBI N, ABRAN A. A systematic literature review: opinion mining studies from mobile app store user reviews[J]. Journal of systems and software, 2017, 125: 207-219.

[18] TAVAKOLI M, ZHAO L, HEYDARI A, et al. Extracting useful software development information from mobile application reviews: a survey of intelligent mining techniques and tools[J]. Expert systems with applications, 2018, 113: 186-199.

[19] TIAN D, LIU Y, WANG Y. Bibliometric analysis of hotspot analysis articles —taking word frequency analysis method as an example[J]. Information Science, 2017, 35(8): 156-160.

[20] CHEN C. CiteSpace II: Detecting and visualizing emerging trends and transient patterns in scientific literature [J]. Journal of the American Society for Information Science and Technology, 2006, 57(3): 359-377.

[21] THELWALL M, BUCKLEY K, PALTOGLOU G, et al. Sentiment strength detection in short informal text [J]. Journal of the American Society for Information Science and Technology, 2010, 61(12): 2544-2558.

[22] PAGANO D, MAALEJ W. User feedback in the appstore: an empirical study[C]//21st IEEE international requirements engineering conference. Piscataway: IEEE, 2013: 125-134.

[23] GUZMAN E, EL-HALIBY M, BRUEGGE B. Ensemble methods for app review classification: an approach for software evolution[C]//2015 30th IEEE/ACM international conference on automated software engineering. Piscataway: IEEE, 2015: 771-776.

[24] PANICHELLA S, SORBO A D, GUZMAN E, et al. How can I improve my app? classifying user reviews for software maintenance and evolution[C]//2015 IEEE international conference on software maintenance and evolution. Piscataway: IEEE, 2015: 281-290.

[25] KHALID H. On identifying user complaints of iOS apps[C]//2013 35th international conference on software engineering. Piscataway: IEEE, 2013: 1474-1476.

[26] PANICHELLA S, SORBO A D, GUZMAN E, et al. Ardoc: App reviews development oriented classifier[C]//International requirements engineering conference workshops. Piscataway: IEEE, 2019: 220-226.

[27] MAALEJ W, NABIL H. Bug report, feature request, or simply praise? on automatically classifying app reviews[C]//2015 IEEE 23rd international requirements engineering conference. Piscataway: IEEE, 2015: 116-125.

[28] MAALEJ W, KURTANOVIC Z, NABIL H, et al. On the automatic classification of app reviews [J]. Requirements engineering, 2016, 21(3): 311-331.

[29] DHINAKARAN V T, PULLE R, AJMERI N, et al. App review analysis via active learning: reducing supervision effort without compromising classification accuracy[C]//2018 IEEE 26th international requirements engineering conference. Piscataway: IEEE, 2018: 170-179.

[30] HU T, JIANG Y. Mining user reviews of APP software reflecting usage feedback[J]. Journal of Software, 2019, 30(10): 3168-3385.

[31] HE D J, PAN M H, HONG K, et al. Fake review detection based on PU learning and behavior density [J]. IEEE network, 2020, 34(4): 298-303.

[32] PHETRUNGNAPHA K, SENIVONGSE T. Classification of mobile application user reviews for generating tickets on issue tracking system[C]//2019 12th international conference on information & communication technology and system. Piscataway: IEEE, 2019: 229-234.

[33] JHA N, MAHMOUD A. Mining user requirements from application store reviews using frame semantics[C]//International working conference on requirements engineering: foundation for software quality. Berlin: Springer, 2017: 273-287.

[34] JHA N, MAHMOUD A. Using frame semantics for classifying and summarizing application store reviews[J]. Empirical software engineering, 2018, 23(6): 3734-3767.

[35] WANG Y, ZHENG L, ZHANG Y, et al. A software requirements mining method for Chinese APP user review data[J]. Computer Science, 2020, 47(12): 170-176.

[36] LI A, QIN Z, LIU R, et al. Spam review detection with graph convolutional networks[C]//Proceedings of the 28th ACM international conference on information and knowledge management. New York: ACM, 2019: 2703-2706.

[37] STANIK C, HAERING M, MAALEJ W. Classifying multilingual user feedback using traditional machine learning and deep learning[C]//2019 IEEE 27th international symposium on foundations of software engineering. New York: ACM, 2019: 1023-1027.

[38] ZHANG L, HUANG X Y, JIANG J, et al. CSLabel: an approach for labelling mobile app reviews[J]. Journal of computer science and technology, 2017, 32(6): 1076-1089.

[39] CHEN Q, ZHANG L, JIANG J, et al. A review analysis method based on support vector machine and topic model[J]. Journal of Software, 2019, 30(5): 1547-1564.

[40] GOMAA A, EL-SHORBAGY S, EL-GAMMAL W, et al. Using resampling techniques with heterogeneous stacking ensemble for mobile app stores reviews analytics[C]//International conference on advanced intelligent systems and informatics. Berlin: Springer, 2019: 831-841.

[41] NI Y, PENG R, SUN D, et al. A method for discovering potential evolutionary requirements based on user reviews[J]. Journal of Wuhan University (Natural Science Edition), 2015, 61(4): 387-392.

[42] ZHANG L, ZHANG X, TAO X, et al. Research on aggregation of academic APP service requirements oriented to review semantic relations[J]. Information Studies: Theory & Application, 2020, 43(1): 155-162.

[43] SCALABRINO S, BAVOTA G, RUSSO B, et al. Listening to the crowd for the release planning of mobile apps[J]. IEEE Transactions on Software Engineering, 2017, 45(1): 68-86.

[44] LIU B. Sentiment analysis: mining opinions, sentiments, and emotions [M]. Cambridge: Cambridge University Press, 2020: 168-171.

[45] VU P M, PHAM H V, NGUYEN T T, et al. Tool support for analyzing mobile app reviews[C]//2015 30th IEEE/ACM international conference on automated software engineering. Piscataway: IEEE, 2015: 789-794.

[46] VU P M, PHAM H V, NGUYEN T T, et al. Phrase-based extraction of user opinions in mobile app reviews[C]//Proceedings of the 31st IEEE/ACM international conference on automated software engineering. Piscataway: IEEE, 2016: 726-731.

[47] HU M, LIU B. Mining and summarizing customer reviews[C]//Proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining. New York: ACM, 2004: 168-177.

[48] LÜ H, FAN K, YANG J. Research on software feature mining for App user reviews[J]. Library Theory and Practice, 2019(7): 106-112.

[49] WEN T, YANG D, LI J. Design and implementation of a Chinese software review mining system[J]. Computer Engineering and Design, 2013, 34(1): 163-167.

[50] GAO C, ZHENG W, DENG Y, et al. Emerging app issue identification from user feedback: experience on Wechat[C]//2019 IEEE/ACM 41st international conference on software engineering: software engineering in practice. Piscataway: IEEE, 2019: 279-288.

[51] PENG Z, WANG J, HE K, et al. An approach of extracting feature requests from app reviews[C]//International conference on collaborative computing: networking, applications and worksharing. Berlin: Springer, 2016: 102-113.

[52] SUN D, PENG R. A scenario model aggregation approach for mobile app requirements evolution based on user comments[M]//Requirements engineering in the big data era. Berlin: Springer, 2015: 75-91.

[53] LI Z, WANG M, ZHAO P. Research on extraction of "evaluation feature-evaluation word" pairs based on conditional random field model[J]. Journal of the China Society for Scientific and Technical Information, 2017, 36(4): 99-108.

[54] CUI J, YANG D, LI J. RERM: A requirements elicitation method based on review mining[J]. Computer Applications and Software, 2015, 32(8): 28-33.

[55] CARRENO L V G, WINBLADH K. Analysis of user comments: an approach for software requirements evolution[C]//2013 35th international conference on software engineering. Piscataway: IEEE, 2013: 582-591.

[56] HUANG J, LI P, PENG M, et al. Research on topic models based on deep learning[J]. Chinese Journal of Computers, 2020, 43(5): 827-855.

[57] BLEI D M. Probabilistic topic models[J]. Communications of the ACM, 2012, 55(4): 77-84.

[58] WANG X, ZHANG X, ZHANG L. Research on semantic association of online review topics for academic APP users[J]. Information Science, 2020, 38(6): 25-31.

[59] GAO C, WANG B, HE P, et al. Paid: prioritizing app issues for developers by tracking user reviews over versions[C]//2015 IEEE 26th international symposium on software reliability engineering. Piscataway: IEEE, 2015: 171-180.

[60] GAO C, ZENG J, LYU M R, et al. Online app review analysis for identifying emerging issues[C]//Proceedings of the 40th international conference on software engineering. New York: ACM, 2018: 48-58.

[61] LIU Y, LI Y, GUO Y, et al. Stratify mobile app reviews: E-LDA model based on hot "Entity"discovery[C]//2016 12th international conference on signal-image technology & internet-based systems. Piscataway: IEEE, 2016: 581-588.

[62] LUIZ W, VIEGAS F, ALENCAR R, et al. A feature-oriented sentiment rating for mobile app reviews[C]//Proceedings of the 2018 world wide Web conference. New York: ACM, 2018: 1909-1918.

[63] SUPRAYOGI E, BUDI I, MAHENENDRA R. Information extraction for mobile application user review[C]//2018 International conference on advanced computer science and information systems. Piscataway: IEEE, 2018: 343-348.

[64] GAO C, XU H, HU J, et al. Ar-tracker: track the dynamics of mobile apps via user review mining[C]//2015 IEEE symposium on service-oriented system engineering. Piscataway: IEEE, 2015: 284-290.

[65] CHENG X, YAN X, LAN Y, et al. Btm: topic modeling over short texts[J]. IEEE transactions on knowledge and data engineering, 2014, 26(12): 2928-2941.

[66] HADI M A, FARD F H. AOBTM: adaptive online biterm topic modeling for version sensitive short-texts analysis[C]//2020 IEEE international conference on software maintenance and evolution. Piscataway: IEEE, 2020: 593-604.

[67] JOHANN T, STANIK C, MAALEJ W. SAFE: a simple approach for feature extraction from app descriptions and app reviews[C]//2017 IEEE 25th international requirements engineering conference. Piscataway: IEEE, 2017: 21-30.

[68] SHAH F A, SIRTS K, PFAHL D. Is the SAFE approach too simple for app feature extraction? a replication study[C]//International working conference on requirements engineering: foundation for software quality. Berlin: Springer, 2019: 21-36.

[69] HU H, WANG S, BEZEMER C P, et al. Studying the consistency of star ratings and reviews of popular free hybrid Android and iOS apps[J]. Empirical software engineering, 2019, 24(1): 7-32.

Author Contributions:
Zhang Ji: Drafted the initial manuscript;
Kang Lele: Proposed the research topic, adjusted the paper structure, and revised the manuscript;
Li Bo: Revised the manuscript.

Note: Figure translations are in progress. See original paper for figures.

Source: ChinaXiv —Machine translation. Verify with original.