International Journal of Computing and Digital Systems
2024, VOL. 17, NO. 1, 1–10

http://dx.doi.org/10.12785/ijcds/XXXXXX

A Comprehensive Framework for Prioritizing User Concerns in Mobile App Reviews Using Multi-Metric Scoring

Nimasha Arambepola[1] **and Lankeshwara Munasinghe**[2]

[1]*Software Engineering Teaching Unit, Faculty of Science, University of Kelaniya, Sri Lanka*
[2]*School of Computing, Engineering and Technology, Robert Gordon University, Aberdeen, Scotland, United Kingdom*

Received 12 March 2024, Revised 15 June 2024, Accepted 10 July 2024

Abstract: An increasing number of mobile app user reviews is a vital source on user concerns towards existing apps. These reviews help to optimize and improve the apps. Despite the recent introduction of effective user review analysis methods, analyzing user reviews still poses significant challenges for researchers. One of them is the overwhelming number of informative reviews make it difficulty extract and prioritize user concerns. This research proposes a novel framework to prioritize user concerns in mobile app reviews utilizing Natural Language Processing (NLP) techniques such as sentiment analysis, Latent Dirichlet Allocation (LDA), and word embedding. This comprehensive framework extracts and ranks user concerns and opinions related to user experience (UX) using a weighted scoring mechanism; multi-criteria prioritization formula. This formula includes four key metrics: Entropy score, Topic Prevalence score, Thumbsup count, and Sentiment score for the major topics identified in the reviews. The proposed framework was evaluated using user reviews from eight mobile apps across four popular categories: education, messaging, business, and shopping. A total of 869,731 reviews were scraped from the Play Store for this evaluation. To validate the proposed framework, its prioritization results were compared with a dataset prioritized by expert app developers. Spearman's rank correlation was used to compare the prioritization trends and the average correlation was 0.7569. Additionally, the Mean Absolute Error (MAE) was 0.1724. These results show that the proposed prioritization framework aligns with the expert developers' priorities with a marginal error. Furthermore, this framework is generalizable, as the evaluation included apps from diverse categories. This makes the proposed framework an effective and efficient tool for decision-making in patch, update or version releases in mobile apps, ensuring that critical user concerns are addressed promptly.
Keywords: App user reviews, Opinion prioritization, Information extraction, User experience, Natural language processing

1. INTRODUCTION

Owing to the widespread adoption of smartphones and the increasing dependence on mobile applications, mobile app market has seen a significant expansion throughout the past decade. Especially during and after the covid-19 pandemic, people established and maintain their daily activities via mobile applications [1]. For example, a number of new mobile apps has been introduced to the app market mainly in app categories such as education, social media, Life style, shopping and business [2], [3]. This expansion motivates the continuous and rapid app enhancement to retain and attract users. In this competitive app market, app user reviews are crucial to get insights on users experiences, preferences, expectations and difficulties while using the app. These reviews often highlight issues, suggest features, and express satisfaction or dissatisfaction. Therefore app reviews have become a rich source of information for developers aiming to enhance app quality and user satisfaction [4]. User reviews which are available on platforms like Google Play Store and Apple App Store provide valuable insights into the strengths and weaknesses of apps from the perspective of end-users. App review analysis involves extracting meaningful information from user feedback to identify common issues, desired features, and overall user satisfaction. This process is essential for prioritizing user concerns to modify the app features for the next release plan to optimize the user experience (UX). Large number of reviews, which often consists of varying levels of detail and sentiment, make it difficult to analyze and prioritize this user opinions. The advent of data-driven techniques in Natural Language Processing (NLP) and Machine Learning (ML) has enabled more sophisticated analysis of textual data. In particular, sentiment analysis and topic modeling have become essential tools for extracting actionable insights from user reviews [5]. These methods allow developers to categorize reviews into topics and identify critical issues.

App review analysis is carried out in different categories/types including sentiment analysis, review classification, review summarization, clsutering and reviews pri-

E-mail address: nimasha@kln.ac.lk, l.munasinghe@rgu.ac.uk

oritization [6], [7]. Two main types of prioritization are identified: review prioritization and topic prioritization [8]. Moreover, researchers tend to analyse negative reviews over positive reviews as they reflects user concerns related to app improvements [9], [10]. In there, review prioritization allows developers to quickly sort the reviews which needs to address immediately. In contrast, topic prioritization is crucial for identifying key issues in a specific app version or time period by considering concerns raised across all user reviews. For instance, prioritization serves various objectives, including identifying emerging issues, optimizing release planning, and facilitating prompt feedback by minimizing the time between issue identification and resolution [11], [12], [8], [13], [9]. Widely used prioritization techniques include anomaly detection methods [9], risk matrices combining clustering and graph theory approaches [11], grouping-based ranking methods [12], and regression techniques involving time series matrices and average ratings [14]. However, there is still a need for more refined methodologies that can not only analyze the content of reviews but also prioritize user concerns extracted from those reviews based on factors like user agreement.

This research aims to address this gap by developing a comprehensive framework for prioritizing user concerns in mobile app reviews. A key component of the framework is the proposed multi-criteria prioritization formula which consists of four metrics, entropy score, topic prevalence, the number of thumbs-up votes (thumbsup count) and sentiment scores to provide a holistic view of user feedback. To ensure all relevant aspects of user feedback, this research employed Latent Dirichlet Allocation (LDA) for topic modeling and leveraging Word2Vec embedding to identify related terms. Thus, the primary objective of this research is to propose a comprehensive framework for extracting and prioritizing user concerns from app reviews using a multi-criteria prioritization formula. This will enable developers to systematically analyze and prioritize user concerns, aiding in the planning of future app releases.

The remainder of this paper is organized as follows: the following section discuss the related studies from the literature. Then the prioritization framework and the experimental setup will explained. Then the results of the findings are presented and discussed. The paper is concluded with a discussion of future research directions.

2. RELATED WORK

Over the past decade, user reviews have become more complex due to the increasing diversity of app users and their evolving requirements. Mobile app review analysis primarily aims to enhance the UX by understanding the user feedback. Nevertheless, it is not practically possible for both users and developers to read each and every user review to understand user opinions about an app. Even though app rating system is available to express the overall user opinion, disparities exist between user ratings and review comments [10]. Consequently, a sentiment rating approach has been proposed to provide summarized feedback, as it provides users a clearer understanding of the application beyond the star rating [15], [16]. Additionally, there are common challenges for app review analysis due to the inconsistencies in app user reviews. For example, variability in review length and the language used for writing reviews. Therefore, preprocessing is crucial and must be addressed carefully and appropriately. Researchers employ various preprocessing techniques in addition to common methods such as tokenization, stop word removal, stemming, and lemmatization [17]. For instance, handling non-informative reviews is a critical challenge, as the majority of user reviews lack detailed insights. A review such as ”good app” is useful for sentiment analysis to determine a positive or negative opinion but lacks detail in identifying specific aspects of the app that are appreciated or need improvement. Thus, this is non-informative for extracting meaningful insights. To address this, some researchers have eliminated short reviews to focus on more informative content [9], [10]. Furthermore, custom stop word removal has been widely used, as certain words are meaningless for identifying prominent topics or themes from app user reviews [12], [9].

App review analyses have been conducted for various purposes, with some studies specifically focusing on particular app categories, such as health and fitness [18], [19]. Furthermore, the results of app review analyses are useful at different stages of the software development life cycle, from requirement gathering to app maintenance [20], [21], [22]. Consequently, researchers have conducted app review analysis to identify the supporting software engineering activities as well as to investigate user reviews related to specific aspects of apps. For example, usability and UX identification through app reviews is widely adopted [5], [23], [24], [25], with particular emphasis on user interface improvements [26]. Moreover, apps that satisfy users in some countries may not meet the expectations of users in other countries due to economic disparities and different user expectations [27]. Thus, country-specific feature requests are vital for customizing mobile apps based on user groups and their preferences. Furthermore, app review analysis is crucial for market research for app development, as it allows for the comparison of competitive mobile apps in app stores [28]. Different tools and frameworks have been proposed to analyse app reviews. For example, SURminer permits sentiment analysis together with topic modeling [29] while MARK [30] classify the user reviews. In addition, specific tools were developed to assist specifically in prioritizing app reviews. For instance, AR-Miner [14], PAID [12], IDEA [13], and MApp-IDEA [9] are some of the tools developed to prioritize user reviews. These advancements show significant progress in app review prioritization. Despite this, it is yet to be considered how user agreements (such as thumbs-up counts) can be used as weighting factors for prioritizing user reviews or the topics extracted from app user reviews them.

3. PRIORITIZATION FRAMEWORK

This research aims to develop a comprehensive framework for prioritizing user concerns derived from mobile app reviews to enhance the UX. The framework encompasses several stages as shown in figure 1. The main stages are data extraction, topic identification, topic analysis (topic scoring and prioritization using a multi-criteria prioritization formula), and visualization. This proposed framework was evaluated based on the priority scores assigned by external app developers.

A. Data extraction

Data collection is the initial step. In there the reviews which needs to be considered for user concern prioritization are scraped from the relevant app. Subsequently, basic preprocessing techniques are applied. These steps include converting text to lowercase, tokenization, removal of puctuation and stopwords. Then Lemmatization is used to convert the words in the processed reviews into their root forms. Unlike stemming, it preserves the semantic meaning of words, making it more suitable for topic modeling [31]. Moreover, Non-English reviews are excluded, and short reviews; reviews with three words or fewer are filtered out due to their limited contribution for topic identification. Then the advanced preprocessing involves removing custom stopwords and emojis, which are considered to be no contribution to topic modeling [10]. These custom stopwords can be selected through two approaches: (1) from literature, (2) through a manual review of a sample reviews from each app.

B. Topic identification, scoring and prioritization

The Latent Dirichlet Allocation (LDA) algorithm is then employed to identify topics within the processed dataset. There are numerous informative reviews in app user reviews, and among them choosing the issues/concerns which need immediate action is crucial. Thus, only the negative and neutral reviews was considered for this study as those consists of user concerns which are helpful for improvement of the app. This proposed framework use a topic scoring method to identify the high priority topics which needs immediate actions. In this study, we proposed a prioritization formula based on four key metrics: Entropy, Topic Prevalence, Thumbsup count, and Sentiment score. Each of these metrics captures a different aspect of user reviews to help prioritize issues that are of most importance to developers, particularly when optimizing mobile app UX.

1) Entropy

Entropy is a concept borrowed from information theory that measures the unpredictability or diversity of information in a dataset. In the context of app reviews, entropy (E) quantifies the diversity of information conveyed by the reviews [32]. This metric allows differentiation between reviews that provide novel or diverse insights and those that contain frequently repeated feedback. Reviews with higher entropy values indicate greater variability in the keywords used. This variability can highlight critical user concerns

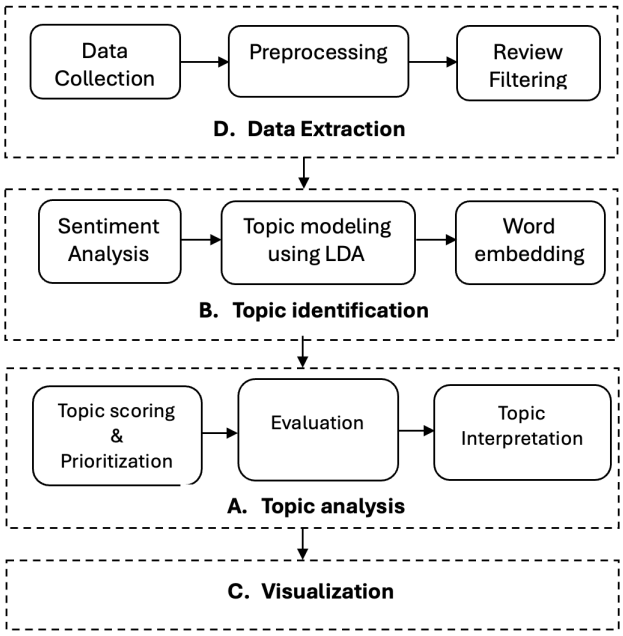


Figure 1. Prioritization framework

that are mentioned less frequently. The entropy score is calculated as follows:

$$Entropy \quad E(T_i) = -\sum_{j=1}^{m} P(k_j) \log_2 P(k_j) \tag{1}$$

Where $P(k_j)$ represents the probability of occurrence of a keyword k_j within the review compared to its occurrence in the entire review corpus. Higher entropy values indicate that the keywords are more evenly distributed, reflecting a greater diversity of information in the reviews. Conversely, lower entropy values indicate that a topic is more focused and well-defined, making it more actionable for developers. To reflect this in the prioritization process, we invert the entropy scores so that topics with lower entropy (i.e., more concentrated user concerns) are assigned higher priority. Additionally, these entropy scores are normalized to a common scale to facilitate comparison with other metrics in our analysis, as explained in Section 3-B5. This normalization enhances the interpretability and utility of the scores in the multi-criteria prioritization formula.

2) Topic Prevalence

Topics generated through topic modelling using LDA are used to identify the main user concerns expressed in user reviews. In this context, Topic Prevalence (TP) measures how often a specific topic is discussed across all reviews [33]. A review is considered to mention a particular topic if it contains at least three keywords associated with that topic. This appraoch aims to prioritize the user concerns that are frequently raised by users. It aids to understand their

needs. The Topic prevalence score for a topic is calculated as follows:

$$Topic\ Prevalence\ \ TP(Ti) = \frac{Number\ of\ reviews\ mentioning\ topic(Ti)}{Total\ number\ of\ reviews} \quad (2)$$

Higher TP scores indicate issues that are raised more often by users, suggesting that these issues may have a greater impact on UX and app functionality.

3) Thumbsup Count

Thumbs Up (TU) counts represent the number of users who found a particular review helpful or relevant [34]. This metric serves as a proxy for user validation, indicating user agreement or endorsement on a particular review [10]. Therefore, in the context of our study, utilizing thumbs up counts is essential for prioritizing topics derived from user reviews. Higher thumbs-up counts reflect experiences or issues that users find crucial. This indicates general agreement among the user community. This helps in effectively directing improvement efforts toward the most impactful areas. The formula for calculating the thumbs up count as below.

$$Thumbsup\ Count\ TC(T_i) = \sum_{\substack{r \in R \\ |\{\text{keywords}(T_i) \cap r\}| \geq 3}} TU(r) \quad (3)$$

These values were normalized (as mentioned in section 3-B5) to enhance the comparability across topics by standardizing values within a conman range (0-1) and mitigating bias from outliers.

4) Sentiment Score

Sentiment analysis is applied to measure the polarity (positive, negative, or neutral) of the language in user reviews. For this, the Valence Aware Dictionary and sEntiment Reasoner (VADER) tool is utilized, which performs particularly well with text data from online reviews [35]. The sentiment score ranges from -1 (most negative) to 1 (most positive). Negative reviews often indicate issues requiring urgent attention [8]. In this study only the neutral and negative reviews were considered. To prioritize topics effectively, we invert the sentiment scores so that topics with lower sentiment scores (more negative sentiment) receive higher priority in our analysis. The formula for calculating the sentiment score(S) as below.

$$S(T_i) = - \sum_{\substack{r \in R \\ |\text{keywords}(T_i) \cap r| \geq 3}} \text{VADER sentiment score}(r) \quad (4)$$

5) Normalization

After calculating the each metric, all metrics were normalized to a range of [0, 1] to ensure the consistency in their contribution to the combined score. The raw values of each metric varied. For instance, the thumbs-up count can span a wide range of values, while entropy scores are typically bounded between 0 and 1. Therefore, normalization was required to avoid any single metric from disproportionately influencing the final combined score. This step ensured that each metric contributed appropriately according to its assigned weight. The following formula was used for normalization.

$$\text{Normalized value} = \frac{\text{Raw value} - \text{Minimum value}}{\text{Maximum value} - \text{Minimum value}} \quad (5)$$

By applying this formula, all values were scaled so that the lowest value in each dataset corresponded to 0, and the highest value corresponded to 1. This standardization process allowed for a fair and consistent combination of the metrics.

The final prioritization score is calculated as a weighted combination of these four metrics as equation 6. The weights are assigned based on the significance of each factor in reflecting user concerns.

$$CombinedScore(Ti) = w1.(E(Ti)) + w2.TP(Ti) + w3.TC(Ti) + w4.(S(Ti)) \quad (6)$$

where,

- $CombinedScore(Ti)$: The overall priority score for topic Ti, used to rank topics.
- $(E(Ti))$: Normalized entropy measures how focused a topic is.
- $TP(Ti)$: Normalized topic prevalence reflects how often the topic appears across reviews, giving higher priority to frequently mentioned issues.
- $TC(Ti)$: Normalized ThumbsUp counts indicate user agreement or endorsement, assigning higher scores to topics supported by more users.
- $(S(Ti))$: Normalized sentiment scores prioritize topics where user feedback is more critical, highlighting areas of dissatisfaction.

Each weight $(w1, w2, w3, w4)$ can be adjusted to emphasize different factors in the prioritization process.

4. EXPERIMENTAL SETUP

This section discusses the experimental setup under four sections, Data extraction, Topic identification, Topic

analysis and prioritization, and Evaluation.

A. Data Extraction

The first step involves collecting reviews from popular app categories, including education, messaging, business, and shopping. A total of 896,649 reviews were scraped from eight different apps as shown in the column, 'Initial number of reviews' in Table I. These app categories were selected based on their popularity. They are in the ten most popular app categories according to statista [36]. For this study, two popular apps from the each of the above app categories were selected [37]. This ensures a broad representation of UX and generalization of the proposed frameworks. Then the reviews were preprocessed by follwoing the preprocessing steps mentioned in the section 3-A. Non-English reviews and short reviews were removed as they do not contribute for topic identification. For example, a review like, "app is good" does not reflect any semantic meaning for identifying user opinion for a particular app's improvement. Then the custom stopwords were selected from both the approaches mentioned in previously. Initially custome stopwords were selected from the literature. Then, 100 app user reviews were studied from each app category to identify custom stopwords through manual review. Custom stopwords are in different types as shown in table II. The number of reviews considered for the experiments from each app after preprocessing is shown in table I. Sentiment analysis is conducted using VADER, a tool that excels at understanding sentiment in text, especially in social media contexts [35]. This helps measure the polarity value of user opinions associated with each review. Then, only the neutral and negative reviews were selected for topic identification in this experiment as they are the most crucial reviews for immediate addressing by identifying the issues experienced by users. The number of reviews considered for the prioritization is shown in the last column of the table I.

B. Topic identification

The LDA algorithm is employed to identify topics within the processed dataset. To evaluate LDA's performance, Perplexity and Coherence Score metrics are utilized. Perplexity measures how well the model predicts unseen data, while Coherence Score assesses the semantic relevance of topics [38]. Hyperparameters, including alpha, beta values, and the number of topics, are fine-tuned to optimize the performance of the LDA model. The experiment continued with the generated optimal number of topics along with their keywords for each app. In there, 8-10 keywords were considered for each topic.

C. Topic analysis and prioritization

Word embedding is utilized to enhance the keyword list in the generated topic list. Subsequently, a Word2Vec model is trained on the tokenized reviews to identify synonyms for each keyword within the generated topics, enhancing the precision of keyword relevance [39]. For each keyword, three synonyms were generated. Then above mentioned (section 3-B), topic prioritizing metrics were calculated for each topic. In there, if at least three keywords from the expanded keyword set (either the original keyword or one of its three synonyms) appears in the text, that text is considered to mention the topic for calculating each metric. Finally the combined score was calculated for each topic for each selected app to identify the priority of each topic.

TABLE I. Summary of the review dataset

App category	App	Number of reviews		
		Initial	After preprocessing	Neutral & negative
Business	LinkedIn	166500	87208	35665
	msTeams	135000	133823	41162
Education	Coursera	26984	18687	5626
	Udemy	73247	52265	1665
Messenger	Messenger	90000	89808	52976
	Whatsapp	153000	152369	66266
Shopping	eBay	126000	123738	30187
	Amazon	99000	97763	44024

TABLE II. Types of Custom stopwords

Type	Example custom stopwords
Domain-specific terms	Business (kpi, invoice), Education ('education', 'learn', 'learning', 'student', 'teach', 'teaching'), Messenger (sync, management), Shopping (promotion, discount, delivery, return)
App-specific terms	'udemy', 'coursera', 'ebay', 'linkedin', 'whatsapp', 'messenger', 'amazon', 'msteams', 'app', 'application'
Common adjectives	'amazing', 'good', 'bad', 'really', 'awesome', 'great', 'enjoy', 'wonderful', 'love', 'best', 'excellent', 'nice', 'easy', 'difficult', 'worst'
Common verbs	'use', 'try', 'like', 'could', 'get', 'through'
Greetings	'thank', 'thanks', 'may', 'dear'

D. Evaluation

Proposed review prioritization framework was evaluated by using the app reviews prioritized by app developers. For that, a stratified random sample of informative user reviews was selected from each app. The calculated sample sizes for each app is shown in table III. According to the literature, a sample size of 10 participants (human raters) is reliable enough to evaluate the outcomes of software engineering research [40]. Therefore, these samples were then sent to a group of 16 external evaluators for review prioritization. External evaluators are chosen based on their

TABLE III. Reviews sent for external reviewing

App	Number of reviews
LinkedIn	381
msTeams	381
Coursera	360
Udemy	313
Messenger	382
Whatsapp	382
eBay	380
Amazon	381

educational and professional qualifications. They are app developers with a Software Engineering related degree and with 3-6 years of experience in app development. Moreover, they are users of the respective apps. The external app developers were given guidelines for prioritizing the reviews (Table IV).

Each set of reviews was independently prioritized by recruited external app developers based on the given guidelines and with their experiences. The remaining unlabeled reviews belongs to each app were then labeled using a semi-supervised learning approach. Among the several approaches, Support Vector Regression (SVR) [41] was utilized to predict continuous priority values, treating these values as labels for the reviews. For evaluating the performance of the SVR model, Mean Squared Error (MSE) and Mean Absolute Error (MAE) were employed as the evaluation metrics. These explains how accurate the model is overall and how close its predictions are to the actual priority values, making them suitable for evaluating performance in this context [42]. Hyperparameter tuning was conducted using GridSearchCV to determine the optimal values for parameters such as C, gamma, and kernel. The optimal values obtained during this process along with the corresponding MSE and MAE results of each app's predictive model is presented in table V. These parameters were fine-tuned to achieve the best possible performance for each app. Then priority scores were assigned to remaining reviews. Subsequently, a python-based algorithm was developed to determine the most relevant topic(s) for each review based on the presence of at least three keywords or the generated synonyms from the topic keywords in the review. If a review was associated with multiple topics, its priority score assigned by the external app developers was allocated to each of those topics. This approach allows for the calculation of priority scores for each topic based on the developers' assigned priority values.

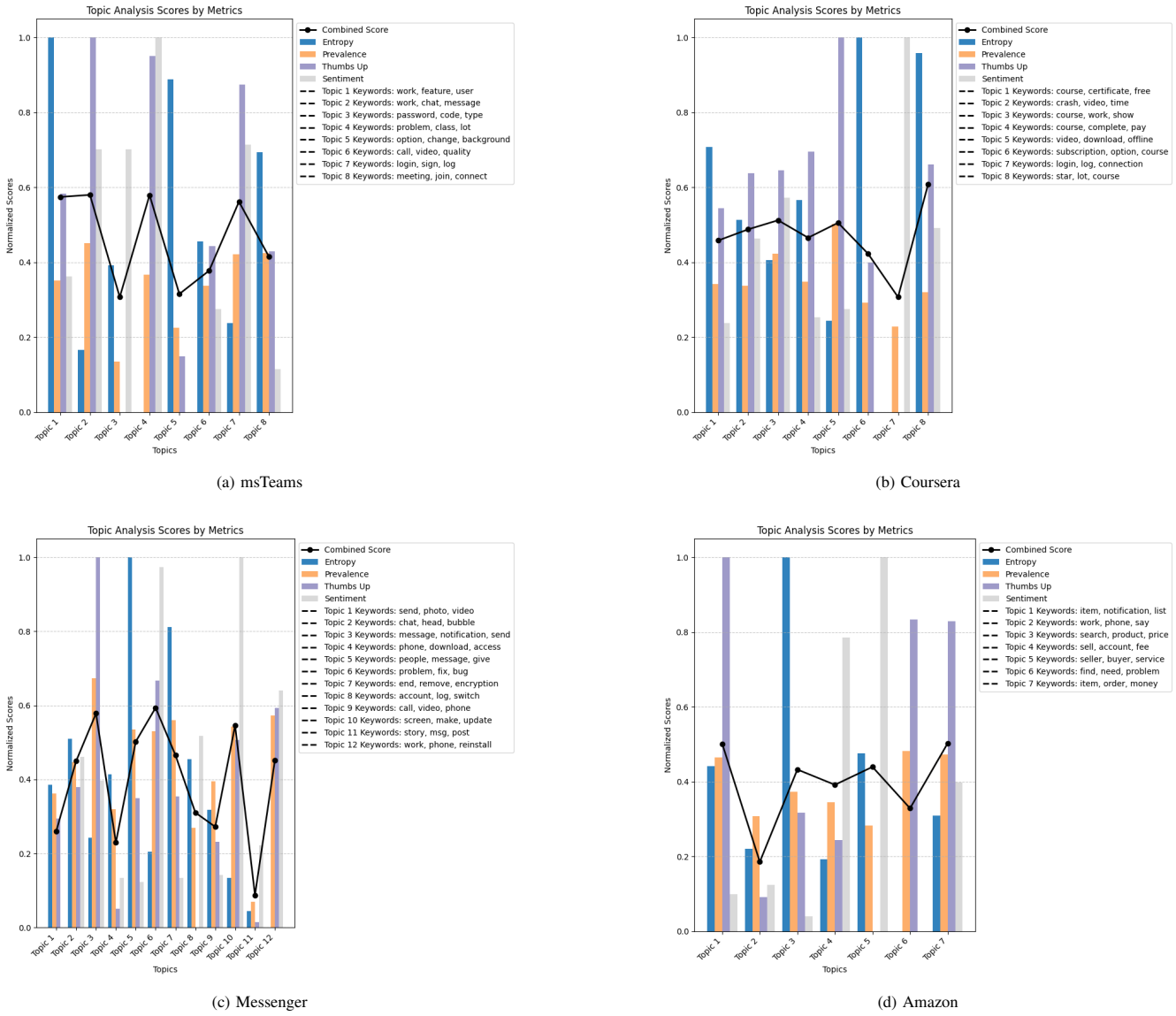
Finally, two prioritization results, one from the proposed prioritization formula and other one from the external app developers' prioritization were compared to validate the proposed framework. The validity of the proposed framework was evaluated utilizing multiple metrics and approaches. In here the topic priority pattern is more crucial than the priority values in two approaches as we need to identify the user concerns which need to address early.

TABLE IV. Priority assignment guidelines

Priority value	Description	Examples
0–0.3 Low	Reviews that provide insight into bugs, enhancements or feature requests related to the app that seem optional and not essential for the app's core functionalities or performance.	"App is good but is not provide some features. Like this zooming and play and paush button on the center of video. So, I humble request for developer team. Please add the video zooming feature in this app, and push and play button add on the centre of video.", "Some functionality not working properly and course contents are also not updated more often", "I completed a whole session, but it says I'm only 94% complete. Need help"
0.4–0.6 Medium	Reviews that provide insight into bugs, enhancements or feature requests related to the app that seem mandatory for the app's core functionalities or performance.	"The app has so many bugs, sometime i can't log in, then i cant see my enrolled courses. Please have good app development to fix these issues. Btw the courses are good", "It is not working in Samsung tab what is solution .it not opening again can u do something!
0.7–1.0 High	Reviews that provide insight into bugs, enhancements or feature requests related to the app that seem critical for the app's core functionalities or performance.	"Irritating when play pause button is not auto disappearing one one screen click.", "Unable to login to my existing account in the app or to create a new account using Google account."

TABLE V. App Hyperparameters and Evaluation Metrics

App	Hyperparameters			Evaluation Metrics	
	C	Gamma	Kernel	MSE	MAE
Udemy	0.1	Scale	Linear	0.0529	0.1908
Amazon	0.1	Scale	Linear	0.03897	0.1348
Coursera	1	Scale	RBF	0.0759	0.2083
Messenger	10	Scale	RBF	0.0659	0.21
eBay	10	Auto	RBF	0.0366	0.155
msTeams	10	Auto	RBF	0.0193	0.0973
WhatsApp	1	Scale	Linear	0.0292	0.1343
LinkedIn	1	Scale	RBF	0.0371	0.0162



(a) msTeams

(b) Coursera

(c) Messenger

(d) Amazon

Figure 2. Topic visualization along with the entropy, prevalence score, thumbsup count and sentiment score

Therefore, the results were visualized using a line graph to provide a clearer understanding of the prioritization framework's effectiveness by comparing the normalized topic scores with the combined priority scores. This visualization highlights how the two scoring methods align or diverge. Moreover, it reveals the relative importance of each topic as perceived by users. In addition, quantitative evaluation is vital to validate this framework. Therefore, both MAE and Spearman's rank correlation used for the proposed framework evaluation. MAE measures the average absolute differences between predicted priority scores (from the proposed formula) and actual priority scores (from external app developers). Spearman's correlation provides insights into how well the prioritization trends align between prioritization framework and the expert judgments of the developers [43]. Results of the evaluation discussed in the following section.

5. RESULTS AND DISCUSSION

This section provides the results of the experimental analysis and evaluation of the proposed topic prioritization framework.

After identifying optimal number of topics, each prioritization metric were calculated and the priority score (combined score) was generated for each topic using the multi-criteria prioritization formula (equation 6). The prioritized topics for each app were then visualized using bar charts. In there, each bar represents the values of the individual metrics and the combined score for each topic. Figures 2 illustrates topic visulatization bar charts for one

selected app from each app category. The corresponding entropy, prevalence score, thumbsup count and sentiment score is shown for each topic. The optimal number of topics identified for each app, along with the metrics used for prioritization (entropy, prevalence, thumbs-up count, and sentiment score), and the combined prioritization score, are presented in Table VI. In this experiment, equal weights were assigned to each metric to calculate the overall priority score. Additionally, the last column shows the priority values assigned to each topic after external evaluation by the app developers. According to the priority scores assigned by external app developers, topic priorities were calculated as explained in the section 4-D. Then multiple approaches were utilized to evaluate the proposed framework through the proposed multi-criteria prioritization formula. In this topic prioritization, pattern comparison is more important than the priority value comparison as the need is to identify user concerns which need high priority. Therefore, initially visualized the comparison using line graphs as show in figure 3. The line graph displayed two lines: one for the normalized scores (blue), which is from the external app developers priorities, and another for the combined scores (orange), calculated using the proposed multi-criteria prioritization formula. These graphs effectively demonstrate the validity of our prioritization formula which use to prioritize the user concerns raised through app user reviews. Even though certain disparities exist, such as Topic 0 in LinkedIn and Topics 0 and 2 in msTeams, both the normalized and combined scores show alignment in the majority of cases. For example, topic priority distribution of apps such as Messenger, eBay, Whatsapp, Udemy and Amazon are closely matched.

In order to further quantify the alignment between the two prioritization methods, MAE and Spearman's rank correlation were calculated as shown in Table VIII. The MAE values ranged from 0.0855 for Amazon to 0.2845 for Coursera, with an average of 0.1724 across all applications. These results indicate a generally low level of deviation between the priority scores assigned by the proposed framework and those determined by the external developers. Average Spearman's rank correlation is 0.7569 across all eight apps. These results indicate a strong alignment between the two prioritization methods. For instance, LinkedIn and Udemy demonstrated high Spearman's rank correlations of 0.9000, indicating near-identical ranking patterns between the two approaches. Even though apps such as Coursera and Amazon exhibited lower rank correlations, overall results demonstrate generally high Spearman's correlations across most apps. Therefore, the comparative performance assessment demonstrates that the proposed multi-criteria prioritization formula effectively aligns with expert evaluations, providing a reliable method for prioritizing user concerns derived from app user reviews. This consistency across multiple apps from different app categories demonstrates that the proposed multi-criteria prioritization formula is both applicable and generalizable for prioritizing user concerns specifically in neutral and negative reviews. In this context, it is difficult to interpret the topic using only the topic keywords. To facilitate a clearer and more accurate interpretation, this framework presents three sample user reviews for each topic, as illustrated in Table VII. This table provides sample reviews for two topics from two selected apps as an example. It allows for easy and effective interpretation of the topics beyond relying on keywords alone.

This research utilized multi-criteria prioritizing formula for topic prioritization instead review prioritization by considering a novel metric which is thumbsup count. This metric emphasizes user agreement on each topic in the prioritization process. Some existing studies have explored the prioritization of user reviews utilizing approaches such as anomaly detection methods [9] and risk matrices that integrate clustering and graph theory techniques [11]. However, these studies did not recognize user agreement as a significant metric for prioritization. In contrast, thumbsup counts alone cannot prioritize user concerns. Thus, this research utilizes several metrics for prioritization. Therefore, findings and methodologies proposed in this research are expected to contribute significantly to the field of software engineering, particularly in the areas of UX optimization and quality assurance in mobile apps. However, it is recommended to use this framework for analyzing user reviews over specified time periods. Each of the metrics used in the prioritization formula measures distinct and critical aspects relevant to the generated topics. However, assigning equal weights to each metric in the evaluation in this study could be seen as a limitation. Nevertheless, developers need to adjust the weights to better align with specific requirements or app goals in their next release plan.

While the multi-criteria prioritization formula used in this study assigns equal weights to all metrics (entropy, topic prevalence, thumbs up count, and sentiment score), the experiment of this study assumes that all metrics contribute equally to the prioritization process. This decision was made to ensure generalizability across different app categories. However, it is worth noting that, certain metrics may carry more significance depending on specific app contexts or developer objectives. Therefore, future research could explore adaptive weighting strategies, where the importance of each metric can be dynamically adjusted based on the app's focus on the particular update or version release or past performance data. This flexibility would allow the prioritization framework to better align with varying development goals and app specific requirements. In addition this research can be extended in two directions to enhance the proposed framework in future. First, the framework can be expand to identify persistence issues over different app versions by analysing the prevalence of generated topics across consecutive app versions. It provides vital information on long term UX. Moreover, topic interpretation is a crucial task and this research can be expand by considering bigrams which enable identifying more effective phases from reviews.

TABLE VI. Summary of the detailed topic analysis and prioritization

App	Topic	Keywords	Entropy	Prevalence	Thumbs Up count	Sentiment score	Combined Score	Priority score (by developers)
LinkedIn	Topic 0	account, login, log, sign, password, even, use, security, check, email	0.3175	0.2249	0.2712	0.3122	0.2814	0.8948
	Topic 1	premium, money, mode, dark, charge, cancel, subscription	0.5311	0.05845	0	0	0.1474	0
	Topic 2	notification, want, phone, contact, need, turn, way, download, many, force	0.1427	0.2141	0.2886	0.0824	0.1820	0.2766
	Topic 3	job, year, professional, people, experience, work, company, search, business, platform	1	0.2981	0.2981	0.1201	0.4117	0.3701
	Topic 4	work, message, post, open, time, profile, show, load, fix, crash	0	0.3257	1	1	0.5814	1
msTeams	Topic 0	login, sign, log, account, password, meeting, say, even, ask, able	0.06531	0.3433	0.31188	0.3835	0.2760	0.763186
	Topic 1	chat, message, notification, send, show, desktop, see, delete, mobile, receive	0.0901	0.2909	0.6793	0.2566	0.3292	0.757437
	Topic 2	call, phone, laptop, notification, computer, handy, disconnect, turn, group, confuse	1	0.2144	0.3823	0.1141	0.4277	0.238592
	Topic 3	time, take, every, waste, download, load, upload, long, open, picture	0.5131	0.2536	0.2584	0.4617	0.3717	0.502548
	Topic 4	work, properly, phone, day, stop, mobile, Works, Doesnt, last, issue	0.7946	0.3620	0.8321	0.6790	0.6669	0.831042
	Topic 5	meeting, option, background, join, give, change, remove, see, problem, show	0.2297	0.3611	1	0.3269	0.4794	0.618735
	Topic 6	connect, device, Unable, sound, Bluetooth, android, screen, meeting, Cant, audio	0.7154	0.2438	0	0.2080	0.2918	0.528463
	Topic 7	open, crash, bug, slow, keep, get, sometimes, opening, hang, many	0	0.3044	0.4145	0.8240	0.3857	0.572952
	Topic 8	problem, class, lot, quality, data, video, network, school, lag, issue	0.2204	0.3149	0.2113	1	0.4367	0.724784
	Topic 9	feature, work, Android, convenient, team, communication, use, well, user, way	0.9773	0.3239	0.3151	0	0.4041	0.115266
Coursera	Topic 0	course, certificate, free, quality, give, complete, change, test, screen, make	0.7083	0.3422	0.5451	0.2370	0.4582	0.4136
	Topic 1	crash, video, time, phone, open, Android, anything, Chromecast, try, start	0.5135	0.3368	0.6377	0.4638	0.4880	0.8780

Continued on next page

TABLE VI – continued from previous page

App	Topic	Keywords	Entropy	Prevalence	Thumbs Up count	Sentiment score	Combined Score	Priority score (by developers)
	Topic 2	course, work, show, assignment, even, try, review, submit, error, class	0.4066	0.4237	0.6458	0.5729	0.5123	0.8841
	Topic 3	course, complete, pay, load, take, money, page, time, certificate, material	0.5663	0.3478	0.6948	0.2536	0.4656	0.5330
	Topic 4	video, download, offline, watch, course, play, work, problem, fix, need	0.2437	0.5027	1	0.2757	0.5055	1
	Topic 5	subscription, option, course, payment, service, card, contact, platform, cancel, charge	1	0.2917	0.4	0	0.4229	0.0600
	Topic 6	login, log, connection, error, sign, account, even, internet, say, network	0	0.2284	0	1	0.3071	0
	Topic 7	star, lot, course, study, u, dark, mode, would, stop, give	0.9582	0.3198	0.6618	0.4913	0.6078	0.3703
Udemy	Topic 0	download, video, course, load, offline, play, work, watch, content, even	0.4313	0.5706	1	0.0891	0.5228	0.773552
	Topic 1	course, purchase, money, show, price, payment, buy, paid, go, pay	1	0.3981	0.3122	0.0631	0.4434	0.439295
	Topic 2	account, login, sign, log, able, problem, email, password, try, error	0	0.2896	0.1467	1	0.3591	0.521574
	Topic 3	notification, need, lot, thing, way, knowledge, program, learn, discount, sale	0.4014	0.1385	0	0	0.1350	0.098763
	Topic 4	work, screen, crash, fix, time, problem, phone, cast, video, open	0.4031	0.4086	0.8288	0.8618	0.6256	1
Messenger	Topic 0	send, photo, video, picture, cant, sent, see, share, view, edit	0.3857	0.3622	0.2944	0	0.2606	0.248825
	Topic 1	chat, head, bubble, back, open, pop, Chat, game, note, work	0.5106	0.4471	0.3795	0.4618	0.4498	0.25648
	Topic 2	message, notification, send, receive, open, show, see, even, get, time	0.2430	0.6740	1	0.3963	0.5783	1
	Topic 3	phone, download, access, need, want, apps, force, FB, ad, space	0.4137	0.3201	0.0513	0.1338	0.2297	0.046146
	Topic 4	people, message, give, would, want, option, u, contact, delete, group	1	0.5347	0.3503	0.1239	0.5022	0.302882
	Topic 5	problem, fix, bug, connect, always, work, connection, even, show, internet	0.2048	0.5300	0.6674	0.9728	0.5937	0.549086
	Topic 6	end, remove, encryption, back, delete, feature, conversation, bring, suck, message	0.8121	0.5605	0.3546	0.1342	0.4653	0.326408

Continued on next page

TABLE VI – continued from previous page

App	Topic	Keywords	Entropy	Prevalence	Thumbs Up count	Sentiment score	Combined Score	Priority score (by developers)
	Topic 7	account, log, switch, page, say, sign, password, login, ask, back	0.4545	0.2699	0	0.5180	0.3106	0.08903
	Topic 8	call, video, phone, sound, notification, play, voice, make, turn, audio	0.3189	0.3955	0.2319	0.1424	0.2722	0.186395
	Topic 9	screen, make, update, work, change, time, go, back, bad, type	0.1340	0.5436	0.5060	1	0.5459	0.332772
	Topic 10	story, msg, post, select, trash, upload, today, separate, choose, folder	0.0445	0.0706	0.0143	0.2217	0.0878	0
	Topic 11	work, phone, reinstall, open, try, keep, still, fix, stop, say	0	0.5732	0.5930	0.6396	0.4514	0.629999
WhatsApp	Topic 0	status, video, send, picture, photo, quality, problem, download, image, upload	0.6800	0.4472	0.4157	0.0132	0.3890	0.384213
	Topic 1	message, chat, option, group, delete, feature, add, see, want, person	0.4832	0.4565	0.5274	0.0460	0.3783	0.236521
	Topic 2	call, message, notification, problem, video, voice, work, fix, send, show	0.0845	0.5994	1	0.1173	0.4503	0.432291
	Topic 3	data, phone, chat, backup, lose, back, delete, message, restore, take	1	0.4479	0.2831	0.2036	0.4836	0.510961
	Topic 4	work, problem, try, download, day, phone, say, time, update, still	0	0.5632	0.5986	0.2710	0.3582	0.243746
	Topic 5	update, status, feature, channel, change, option, make, remove, want, see	0.1568	0.4531	0.4523	0	0.2655	0.128342
	Topic 6	account, number, ban, block, reason, get, problem, use, service, support	0.8598	0.3913	0	1	0.5628	0.313862
	Topic 7	contact, people, give, use, privacy, u, would, user, make, star	0.3743	0.3498	0.1613	0.2222	0.2769	0.107438
eBay	Topic 0	item, notification, list, see, load, message, go, work, purchase, show	0.4417	0.4643	1	0.0995	0.5014	0.511717
	Topic 1	work, phone, say, keep, sign, download, log, open, let, try	0.2204	0.3073	0.0907	0.1241	0.1857	0.089828
	Topic 2	search, product, price, item, reliable, fast, Works, option, shipping, convenient	1	0.3740	0.3167	0.0400	0.4327	0.342877
	Topic 3	sell, account, fee, customer, make, suspend, reason, service, charge, get	0.1923	0.3449	0.2438	0.7863	0.3918	0.291345
	Topic 4	seller, buyer, service, customer, review, experience, feedback, fake, scammer, scam	0.4766	0.2822	0	1	0.4397	0.261232

Continued on next page

TABLE VI – continued from previous page

App	Topic	Keywords	Entropy	Prevalence	Thumbs Up count	Sentiment score	Combined Score	Priority score (by developers)
	Topic 5	find, need, problem, time, look, use, year, buy, thing, go	0	0.4821	0.8334	0	0.3289	0.383812
	Topic 6	item, order, money, refund, seller, back, receive, never, say, day	0.3097	0.4730	0.8296	0.3979	0.5025	0.62754
Amazon	Topic 0	review, prime, shop, product, post, allow, negative, way, consumer, trust	0	0.4315	0.4534	0.1909	0.2690	0.371341
	Topic 1	dark, mode, still, issue, UI, white, get, update, background	0.2618	0.2107	0.6421	0.5321	0.4116	0.464518
	Topic 2	search, make, find, back, want, user, add, experience, hate, item	0.2132	0.3831	0.9954	0.559	0.5376	0.47443
	Topic 3	price, product, shopping, fast, delivery, need, want, always, change, real	0.1586	0.3396	0.6794	0	0.2944	0.320628
	Topic 4	open, work, crash, keep, phone, go, load, Cant, time, fix	0.0142	0.4773	0.5085	0.5287	0.3822	0.565068

(a) LinkedIn (b) msTeams (c) Udemy (d) Coursera

(e) Messenger (f) WhatsApp (g) eBay (h) Amazon

Figure 3. Overall evaluation of selected apps

TABLE VII. Sample reviews for generated Topics

App	Topic	Topic keywords	Sample reviews	Topic interpretation
Coursera	1	crash, video, time, phone, open, Android, anything, Chromecast, try, start	Very slow, videos cannot play properly, and courses are not at all optimized for mobile, so they are full of bugs that crash videos and prevent tasks from being completed. Excellent on a computer. Complete trash on a phone as no effort was made to adapt anything to mobile. Portrait and landscape flips crash the app. Searching for classes crashes the app. Looking sideways at your phone crashes the app The courses are great. The app is terrible. The lesson never starts without a ”whoops, restart the player” message, with no explanation why the player will not work. Then if I do get lucky and the player works, the video ends up freezing 30 seconds into it. I like courseras programs, but I cannot do them away from my computer because of these problems.	Video player crashes in mobile app
	6	subscription, option, course, payment, service, card, contact, platform, cancel, charge	Worst experience ever. Wanted a course , subscribed for a 7 day free trial , did not have the tome to cancel so practically 36 euros were out of my account for no reason . Then tried to cancel subscription but it was impossible. There were instructions but nothing like the described helped . I am still trying to find this gear to find ”manage subscription ” settings. So far nothing. I wish I would never have used this app Terrible. Registered for a certificate with a 7-day free trial, but instead I got charged right away. The certificate I tried to enroll in does not show up in my purchases, instead it is some random ”specialization”. I cannot access the other courses in the certificate, keep getting the ”enroll” option. There is no customer service email or phone , live chat is nowhere to be found. When you go to the ”help center” and you click on log in, it just refreshes the page, with no option to sign in.	Issue in subscription feature
Amazon	2	dark, mode, still, issue, UI, white, get, update, background	No Dark Mode UI option. Totally unacceptable. You’ve got the technology ... the Prime Video portion of the app is Dark Mode. How about making it possible for the entire UI to have a Dark Mode option (and default to the system setting). And it does not work at all with the talkback screen reader. Once again this is totally unacceptable and pathetic. No dark mode support is a big no for me.	Requesting dark mode option
	1	search, make, find, back, want, user, add, experience, feature, item	I really hate the new AI feature (”Rufus”). The fact that there’s no setting to even just disable it is absolutely ridiculous, and dismissing it is largely ineffective because it pops back up if you want to look at a different version of the same product (switching colors of an article of clothing, for instance). I don’t understand the reasoning behind why this feature is essentially mandatory for people who don’t want it/find it annoying. The AI feature is completely useless. We used to be able to search the reviews for keywords and now it’s all AI nonsense unrelated to the item I want to buy!	Issue with the newly added AI feature

TABLE VIII. Evaluation results of the prioritization

App	MAE	Spearman's rank correlation
LinkedIn	0.2232	0.9000
msTeams	0.2540	0.667
Coursera	0.2845	0.6190
Udemy	0.1656	0.9000
Messenger	0.1650	0.8811
Whatsapp	0.1077	0.8095
eBay	0.0935	0.6786
Amazon	0.0855	0.6000
Average	**0.1724**	**0.7569**

6. CONCLUSION

This study proposed a novel framework for topic prioritization aimed at optimal app improvement. The key component of the framework is the multi-criteria topic prioritization formula. It consist of four criteria: entropy, topic prevalence, thumbsup count and sentiment score. In this study, all these components were considered as equally important and calculated the priority scores by assigning equal weights. In order to generalize the approach, eight apps from four different app categories were used to validate the framework. Here, LDA was used to topic modeling while Word2Vec was utilized for synonyms identification. The framework's effectiveness was demonstrated through a comparative evaluation against priority scores assigned to user reviews by external app developers. The results indicated that the proposed framework achieved an average MAE of 0.1724 and a Spearman's rank correlation of 0.7569. These evaluation results highlight a strong alignment between the predicted and actual priorities. Additionally, the visualization of results facilitated a clear comparison of pattern of topic priorities. Overall, this proposed framework addresses a crucial aspect of app improvement by emphasizing UX and user concerns. It allows app developers to identify and prioritize the most critical user concerns within time period to address in the next patch or update release of the app.

REFERENCES

[1] K. Chemnad, S. Alshakhsi, M. B. Almourad, M. Altuwairiqi, K. Phalp, and R. Ali, "Smartphone usage before and during covid-19: A comparative study based on objective recording of usage data," *Informatics*, vol. 9, no. 4, 2022. [Online]. Available: https://www.mdpi.com/2227-9709/9/4/98

[2] L. Y. Li T, Zhang M, L. E, T. S, and H. P., "The impact of covid-19 on smartphone usage," *IEEE Internet Things J.*, vol. 8(23), pp. 16 723–16 733, 04 2021.

[3] M. Wiścicka Fernando, "The use of mobile technologies in online shopping during the covid-19 pandemic - an empirical study," *Procedia Computer Science*, vol. 192, pp. 3413–3422, 10 2021.

[4] J. Dabrowski, E. Letier, A. Perini, and A. Sussi, "Finding and analyzing app reviews related to specific features: A research preview," in *25th International Conference on Requirements Engineering: Foundation for Software Quality*, 03 2019.

[5] W. T. Nakamura, E. C. de Oliveira, E. H. de Oliveira, D. Redmiles, and T. Conte, "What factors affect the ux in mobile apps? a systematic mapping study on the analysis of app store reviews," *J. Syst. Softw.*, vol. 193, no. C, nov 2022. [Online]. Available: https://doi.org/10.1016/j.jss.2022.111462

[6] S. S. Rabeya Sultana, "App review mining and summarization," *International Journal of Computer Applications*, vol. 179, no. 38, pp. 45–52, Apr 2018. [Online]. Available: https://ijcaonline.org/archives/volume179/number38/29329-2018916918/

[7] L. Villarroel, G. Bavota, B. Russo, R. Oliveto, and M. Di Penta, "Release planning of mobile apps based on user reviews," in *ICSE '16: Proceedings of the 38th International Conference on Software Engineering*, 05 2016, pp. 14–24.

[8] S. Malgaonkar, S. A. Licorish, and B. T. R. Savarimuthu, "Prioritizing user concerns in app reviews – a study of requests for new features, enhancements and bug fixes," *Information and Software Technology*, vol. 144, p. 106798, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950584921002366

[9] C. Gao, J. Zeng, Z. Wen, D. Lo, X. Xia, I. King, and M. R. Lyu, "Emerging app issue identification via online joint sentiment-topic tracing," *IEEE Transactions on Software Engineering*, vol. 48, no. 8, pp. 3025–3043, 2022.

[10] N. Arambepola, L. Munasinghe, and N. Warnajith, "Factors influencing mobile app user experience: An analysis of education app user reviews," in *2024 4th International Conference on Advanced Research in Computing (ICARC)*, 2024, pp. 223–228.

[11] V. M. A. d. Lima, J. C. Barbosa, and R. M. Marcacini, "Issue detection and prioritization based on app reviews," 2023.

[12] C. Gao, B. Wang, P. He, J. Zhu, Y. Zhou, and M. R. Lyu, "Paid: Prioritizing app issues for developers by tracking user reviews over versions," in *2015 IEEE 26th International Symposium on Software Reliability Engineering (ISSRE)*. Los Alamitos, CA, USA: IEEE Computer Society, nov 2015, pp. 35–45.

[13] C. Gao, J. Zeng, M. R. Lyu, and I. King, "Online app review analysis for identifying emerging issues," in *Proceedings of the 40th International Conference on Software Engineering*, ser. ICSE '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 48–58.

[14] N. Chen, J. Lin, S. C. H. Hoi, X. Xiao, and B. Zhang, "Ar-miner: mining informative reviews for developers from mobile app marketplace," in *Proceedings of the 36th International Conference on Software Engineering*, ser. ICSE 2014. New York, NY, USA: Association for Computing Machinery, 2014, p. 767–778.

[15] P. Rodrigues, I. S. Silva, G. A. R. Barbosa, F. R. d. S. Coutinho, and F. Mourão, "Beyond the stars: Towards a novel sentiment rating to evaluate applications in web stores of mobile apps," in *Proceedings of the 26th International Conference on World Wide Web Companion*, ser. WWW '17 Companion. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee, 2017, p. 109–117.

[16] D. Yu, Y. Mu, and Y. Jin, "Rating prediction using review texts with underlying sentiments," *Information Processing Letters*, vol. 117, pp. 10–18, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020019016301107

[17] W. Luiz, F. Viegas, R. Alencar, F. Mourão, T. Salles, D. Carvalho, M. A. Gonçalves, and L. Rocha, "A feature-oriented sentiment rating for mobile app reviews," in *Proceedings of the 2018 World Wide Web Conference*, ser. WWW '18. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee, 2018, p. 1909–1918.

[18] O. Haggag, J. Grundy, M. Abdelrazek, and S. Haggag, "A large scale analysis of mhealth app user reviews," *Empirical Softw. Engg.*, vol. 27, no. 7, dec 2022. [Online]. Available: https://doi.org/10.1007/s10664-022-10222-6

[19] H. Ahn and E. Park, "Motivations for user satisfaction of mobile fitness applications: An analysis of user experience based on online review comments," *Humanities and Social Sciences Communications*, vol. 10, p. 3, 01 2023.

[20] J. Dabrowski, E. Letier, A. Perini, and A. Susi, "Analysing app reviews for software engineering: systematic literature review," *Empirical Software Engineering*, vol. 27, pp. 1–63, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:247027209

[21] A. A. Al-Subaihin, F. Sarro, S. Black, L. Capra, and M. Harman, "App store effects on software engineering practices," *IEEE Transactions on Software Engineering*, vol. 47, no. 2, pp. 300–319, 2021.

[22] A. P. Jacek Dabrowski, Emmanuel Letier and A. Susi, "Mining and searching app reviews for requirements engineering: Evaluation and replication studies," *Information Systems*, vol. 114, p. 102181, 2023.

[23] W. T. Nakamura, E. C. C. de Oliveira, E. H. T. de Oliveira, and T. Conte, "Ux-mapper: A user experience method to analyze app store reviews," in *Proceedings of the XXII Brazilian Symposium on Human Factors in Computing Systems*, ser. IHC '23. New York, NY, USA: Association for Computing Machinery, 2024.

[24] E. Bakiu and E. Guzman, "Which feature is unusable? detecting usability and user experience issues from user reviews," in *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)*, 2017, pp. 182–187.

[25] Z.-Y. Lim, L.-Y. Ong, and M.-C. Leow, "A review on clustering techniques: Creating better user experience for online roadshow," *Future Internet*, vol. 13, no. 9, 2021. [Online]. Available: https://www.mdpi.com/1999-5903/13/9/233

[26] Q. Chen, C. Chen, S. Hassan, Z. Xing, X. Xia, and A. E. Hassan, "How should i improve the ui of my app? a study of user reviews of popular apps in the google play," *ACM Trans. Softw. Eng. Methodol.*, vol. 30, no. 3, apr 2021. [Online]. Available: https://doi.org/10.1145/3447808

[27] K. Srisopha, C. Phonsom, M. Li, D. Link, and B. Boehm, "On building an automatic identification of country-specific feature requests in mobile app reviews: Possibilities and challenges," in *Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops*, ser. ICSEW'20. New York, NY, USA: Association for Computing Machinery, 2020, p. 494–498.

[28] Y. Li, B. Jia, Y. Guo, and X. Chen, "Mining user reviews for mobile app comparisons," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 1, no. 3, sep 2017. [Online]. Available: https://doi.org/10.1145/3130935

[29] A. Di Sorbo, S. Panichella, C. V. Alexandru, J. Shimagaki, C. A. Visaggio, G. Canfora, and H. C. Gall, "What would users change in my app? summarizing app reviews for recommending software changes," in *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, ser. FSE 2016. New York, NY, USA: Association for Computing Machinery, 2016, p. 499–510.

[30] M. V. Phong, T. T. Nguyen, H. V. Pham, and T. T. Nguyen, "Mining user opinions in mobile app reviews: A keyword-based approach (t)," in *2015 30th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 2015, pp. 749–759.

[31] D. Khyani and S. B S, "An interpretation of lemmatization and stemming in natural language processing," *Shanghai Ligong Daxue Xuebao/Journal of University of Shanghai for Science and Technology*, vol. 22, pp. 350–357, 01 2021.

[32] R. Takahira, K. Tanaka-Ishii, and Debowski, "Entropy rate estimates for natural language—a new extrapolation of compressed large-scale corpora," *Entropy*, vol. 18, no. 10, 2016. [Online]. Available: https://www.mdpi.com/1099-4300/18/10/364

[33] L. Kontoghiorghes and A. Colubi, "New metrics and tests for subject prevalence in documents based on topic modeling," *International Journal of Approximate Reasoning*, vol. 157, pp. 49–69, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0888613X2300021X

[34] A. W. de Jong, "Making sense of app reviews: Efficient analysis of user reviews for mobile apps with stm," Ph.D. dissertation. [Online]. Available: https://diglib.uibk.ac.at/download/pdf/7617150.pdf

[35] C. Hutto and E. Gilbert, "Vader: A parsimonious rule-based model for sentiment analysis of social media text," *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 8, no. 1, pp. 216–225, May 2014. [Online]. Available: https://ojs.aaai.org/index.php/ICWSM/article/view/14550

[36] L. Ceci, "Google Play most popular app categories 2024 — Statista — statista.com," https://www.statista.com/statistics/279286/google-play-android-app-categories/#:~:text=As%20of%20the%20second%20quarter,with%20a%2011.5%20percent%20share., [Accessed 26-09-2024].

[37] ——, "Most downloaded Android apps worldwide 2024 — Statista — statista.com," https://www.statista.com/statistics/1448018/global-leading-android-downloaded-mobile-apps/, [Accessed 26-09-2024].

[38] A. Fan, F. Doshi-Velez, and L. Miratrix, "Assessing topic model relevance: Evaluation and informative priors," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 12, no. 3, pp. 210–222, 2019.

[39] M. Fauzi, "Word2vec model for sentiment analysis of product reviews in indonesian language," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, p. 525, 07 2018.

[40] C. Katsanos, N. Tselios, and N. Avouris, "Are ten participants enough for evaluating information scent of web page hyperlinks?" in *Human-Computer Interaction – INTERACT 2009*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 419–422.

[41] G. Kostopoulos, S. Karlos, S. Kotsiantis, and O. Ragos, "Semi-supervised regression: A recent review," *Journal of Intelligent Fuzzy Systems*, vol. 35, pp. 1–18, 06 2018.

[42] V. Plevris, G. Solorzano, N. Bakas, and M. Ben Seghier, "Investi-

gation of performance metrics in regression analysis and machine learning-based prediction models," 06 2022.

[43] F. Franceschini, D. A. Maisano, and L. Mastrogiacomo, *Ranking Association Measures*. Cham: Springer International Publishing, 2022, pp. 49–83. [Online]. Available: https://doi.org/10.1007/978-3-030-89865-6_4