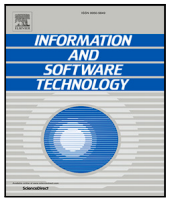
Information and Software Technology 142 (2022) 106741

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof



Understanding in-app advertising issues based on large scale app review analysis

Cuiyun Gao [a], Jichuan Zeng [b,*], David Lo [c], Xin Xia [d], Irwin King [b], Michael R. Lyu [b]

[a] The School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China
[b] The Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, China
[c] The School of Information Systems, Singapore Management University, Singapore
[d] Software Engineering Application Technology Lab, Huawei, China

ARTICLE INFO

Keywords:
Mobile app
User reviews
In-app ads
Ad issues
Cross platform

ABSTRACT

Context: In-app advertising closely relates to app revenue. Reckless ad integration could adversely impact app quality and user experience, leading to loss of income. It is very challenging to balance the ad revenue and user experience for app developers.
Objective: Towards tackling the challenge, we conduct a study on analyzing user concerns about in-app advertisement.
Method: Specifically, we present a large-scale analysis on ad-related user feedback. The large user feedback data from App Store and Google Play allow us to summarize ad-related app issues comprehensively and thus provide practical ad integration strategies for developers. We first define common ad issues by manually labeling a statistically representative sample of ad-related feedback, and then build an automatic classifier to categorize ad-related feedback. We study the relations between different ad issues and user ratings to identify the ad issues poorly scored by users. We also explore the fix durations of ad issues across platforms for extracting insights into prioritizing ad issues for ad maintenance.
Results: (1) We summarize 15 types of ad issues by manually annotating 903 out of 36,309 ad-related user reviews. From a statistical analysis of 36,309 ad-related reviews, we find that users care most about the number of unique ads and ad display frequency during usage. (2) Users tend to give relatively lower ratings when they report the security and notification related issues. (3) Regarding different platforms, we observe that the distributions of ad issues are significantly different between App Store and Google Play. (4) Some ad issue types are addressed more quickly by developers than other ad issues.
Conclusion: We believe the findings we discovered can benefit app developers towards balancing ad revenue and user experience while ensuring app quality.

1. Introduction

In-app advertising is a type of advertisement (ad) within mobile applications (apps). Many organizations have successfully monetized their apps with ads and reaped huge profits. For example, the mobile ad revenue accounted for 76% of Facebook's total sales in the first quarter of 2016 [1], and increased 49% year on year to about $10.14 billion in 2017 [2]. Triggered by such tangible profits, mobile advertising has experienced tremendous growth recently [3]. Many free apps, which occupy more than 68% of the over two million apps in Google Play [4], adopt in-app advertising for monetization. However, the adoption of ads has strong implications for both users and app developers. According to a survey in 2016 [5], almost 50% of users uninstalled apps just because of "intrusive" mobile ads, resulting in a heavy reduction in user volume of the apps. Inappropriate ad integration could also increase the difficulty of ensuring app reliability [6–8]. Moreover, the reduced audiences would generate fewer impressions (*i.e.*, display of ads) and clicks for in-app ads, thereby making developers harder to earn ad profits.

Past studies have conducted surveys to understand users' perceptions of mobile advertising, *e.g.*, perceived interactivity [9], usefulness [10], and credibility [11]. There also exists research devoted to investigating or mitigating the hidden costs of ads, *e.g.*, energy [12], traffic [13], system design [14], and other factors [15,16]. Recent research resorts to user reviews to identify impact of in-app advertising

* Corresponding author.
E-mail addresses: gaocuiyun@hit.edu.cn (C. Gao), jczeng@cse.cuhk.edu.hk (J. Zeng), davidlo@smu.edu.sg (D. Lo), xin.xia@monash.edu (X. Xia), king@cse.cuhk.edu.hk (I. King), lyu@cse.cuhk.edu.hk (M.R. Lyu).

https://doi.org/10.1016/j.infsof.2021.106741
Received 1 April 2021; Received in revised form 23 August 2021; Accepted 28 September 2021
Available online 14 October 2021

on user experience and app reliability. For example, Ruiz et al. [17] analyze the impact of 28 ad libraries on ratings of Android apps. They find that integrating certain specific ad libraries can negatively affect app ratings. Gui et al. [15] also observe that ads can adversely impact user feedback, *i.e.*, over 50% of the studied ad-embedded apps have at least 3.28% of their user complaints dealing with ads. However, few studies have been conducted to identify the common ad issues from app reviews in large scale.

In this paper, we conduct large-scale user review analysis for characterizing common ad issues and providing detailed insights into ad design and maintenance for developers while preserving app reliability. Specifically, we investigate 32 cross-platform apps that rank in the top 100 list of Apple's App Store[1] and Google Play, and examine the following research questions:

RQ1: What are the common types of ad issues in ad-related user feedback?

We answer the question by analyzing a large collection of ad-related user feedback. To determine the common ad issues, we first manually annotate a statistically representative feedback sample. We summarize 15 types of common ad issues based on the manual annotation. Then following a standard keyword-based approach [18,19] for automatic classification, we group the whole ad-related reviews into the 15 manually defined issue types. From the automatic analysis on 36,309 ad reviews, we find that users care most about the number of unique ads and ad display frequency among all the issue types.

RQ2: What are the relationships between ad issue types that users described in their ad-related reviews and the ratings that they gave?

This question aims at helping developers focus on the ad issue types that users tend to be more negative about. In terms of absolute numbers, we discover that nearly half of the low-rated reviews, *i.e.*, with star ratings in the range of one to three, talk about the number of ads and display frequency. Also, users are likely to give poor ratings to ad issues related to notification (*e.g.*, ads notifying users through the status bar) and security (*e.g.*, unauthorized data collection or permission usage) types, despite their lower percentages than other issue types.

RQ3: How different are the distributions of ad issue types in different platforms?

To expand the revenue and reach more users, app developers generally build cross-platform apps, making apps available on multiple platforms. By answering this question, developers can pay attention to the platform differences, and prioritize ad issue types according to platforms. We find that for each ad issue type, its distributions between App Store and Google Play are significantly different.

RQ4: What types of ad issues are addressed more quickly by developers?

App developers would address the important app issues commented by users in the updated versions. Identifying the ad issue types prioritized by many app developers can give us insights for ad maintenance. We find that issue types related to the number of ads and ad contents are solved within relatively longer periods than other types. Moreover, we observe that comparing App Store and Google play, developers solve ad issues at significantly different speeds. Generally, ad issues reflected in Google Play are addressed more quickly than those in App Store.

Our study has implications for both developers and researchers. First, our study indicates the main ad issue types developers should pay attention to. Our study also suggests that developers should pay attention to the platform difference during ad design and maintenance. Additionally, our study shows the existence of platform difference phenomenon (i.e., users respond differently when using the same app in different platforms), and this suggests an interesting direction of future work in platform-aware app design, testing and analysis (e.g., how to automatically customize and test apps for different platforms to improve app reliability). The key contributions of our work are as follows.

(1) We investigate common types of ad issues by analyzing a large user feedback corpus.

(2) We conduct statistical analysis on ad issue types by considering multiple factors, including user ratings, platforms, and the fix durations (*i.e.*, the amount of time that has elapsed before the issue is fixed).

(3) We summarize the implications on better designing and maintaining ads for app developers.

Paper structure. Section 2 presents the methodology we used for cost measurement and user review analysis. Section 3 describes the findings of our study. Section 4 illustrates lessons we learned from review analysis for ad design and maintenance. Section 5 presents threats to validity. Related work and final remarks are discussed in Sections 7 and 8, respectively.

2. Study procedure

In this section, we elaborate on the study procedures we adopted for data collection and categorization of ad issue type.

2.1. Data collection

We manually select 32 popular apps published on both App Store and Google Play from their respective top 100 free app list.[2] The apps are listed in Table 1. The major consideration for the selection is the number of user feedback, *i.e.*, the apps should have more than 100,000 reviews on both app stores. It can also be seen that the subject apps cover a broad range of categories (15 categories in total). After determining the apps, we built a simple web crawler to automatically collect the user reviews of these apps online.

In total, we downloaded 1,840,349 and 3,243,450 of user reviews for the 32 apps from App Store and Google Play, respectively (see Table 2). The reviews on both platforms were collected during September, 2014 to March, 2019. The discrepancy between the number of user reviews across the app stores is due to the different number of users and exposed data for collection on the platforms [21]. We define ad-related reviews (ad reviews, for short) as those containing keywords related to ads, *i.e.*, the words satisfying regex = ad/ads/advert*. In total, we identify 36,309 ad-related reviews. Although such ad review extraction method is not perfect, we hypothesize that the strong selection criterion can reduce false positives.

2.2. Categorizing ad issues

We first introduce the manual process where we define ad issues based on a statistically representative sample of ad reviews, and then present the automated classification method we adopt for automatically classifying the whole ad reviews.

2.2.1. Manual categorization

Users often leave important pieces of information in the feedback while complaining about ads. Such information may relate to the displaying style of ads, and in what way ads affect the functionalities of an app. To determine the ad complaint topics, we perform card sort [22].

Card sorting is a technique that is widely adopted to derive taxonomies from data. We use card sorting here to summarize common ad issues that users complained about. Following the three phases of card sorting [23,24]: In the *preparation* phase, we select 903 out of 36,309 ad reviews to give us a 95% confidence level with 4% confidence interval; in the *execution* phase, reviews are sorted into meaningful categories with a descriptive textual label; finally, in the *analysis* phase, hierarchies are formed in order to deduce general categories. Specifically, our card sort was open, and we let the groups emerge and evolve

[1] In this paper, App Store refers to Apple's App Store.

[2] We referred to the top charts provided by App Annie [20].

Table 1
Cross-platform subject apps.

App category	App name	App category	App name
Casual	Candy Crush	Photography	Camera360
	Minion Rush	Education	Duolingo
	My Talking Tom		TED
Shopping	eBay	Tools	SHAREit
	Amazon	Music	SoundCloud
Entertainment	Netflix	Arcade	Subway Surfers
	YouTube	Travel	TripAdvisor
	Spotify Music	Trivia	Trivia Crack
	VLC	Communication	Line
Social	Facebook		Messenger
	Twitter		Skype
	Pinterest		WeChat
	Snapchat		WhatsApp
	Tango		Viber
	Instagram	Transportation	HERE
Maps	Waze	Productivity	Evernote

Table 2
Statistics for data collected from App Store and Google Play.

All reviews		Ad reviews	
App Store	Google Play	App Store	Google Play
1,840,349	3,243,450	22,343	13,966

during the sorting process. Similar to [23,24], the card sorting process was conducted by the first author and second author separately. Both card sorts led to similar categories of ad issues, with agreement rate at 97.1%, and were finalized based on their joint discussion. Ultimately, this resulted in 15 ad issue types shown in Table 6. The ad issue types are further categorized into two large groups based on whether they are related to the ads (**In-Ad**) or the impact of ads on apps (**App**).

Grouping an ad review into the "Other/Unknown" type is usually based on the following reasons: *a)* Although the review contains the ad-related keywords using regex, it actually does not talk about in-app ads, e.g., the first two pieces of reviews in Table 3; *b)* The in-app ad does not impact user's experience actually. For example, for the third and fourth reviews in Table 3, they state that he/she likes the free music even with ads, or the ad loads fine, respectively; *c)* The review does not clearly state what aspect of the advertisement he/she does not like, or the review does it in a vague way, e.g., the last review in Table 3 describes that the ad is "annoying" but does not describe in what way the ad is annoying. During manual analysis, 39.87% (360/903) ad reviews are labeled as "Other/Unknown" type, which indicates a large proportion of *non-useful* ad reviews.

2.2.2. Automated classification

Each ad review can be categorized into one or more than one issue type. For example, one ad review of a video player app, "*30 s adverts are not skippable and they cannot be loaded properly leading to buffer... So advert is 2 to 3 min long*", was complaining about the non-skippable and timing aspects of the video ads, and also the slow app functionality caused by the ads. We automatically categorize the ad issues of each ad review, taking a similar approach in Ray et al. [18]. This automated classification is performed in two phases: Keyword matching and supervised multi-label classification. The two phases are in a pipeline. In the first phase, we automatically annotated the types of ad reviews based on a restrictive set of keywords and phrases for each type. In the second phase, we train the multi-label classifier using the automatically-annotated data.

Step 1: Keyword matching. We first use a keyword-based search technique to automatically categorize the ad reviews with potential ad complaint types. We choose a restrictive set of keywords and phrases as shown in Table 5. For example, if the ad-related sentences contain any of the keywords: loud, screech, play sound, or volume, we infer the review is related to the *Volume* issue type. Such a restrictive set of keywords and phrases help to reduce false positives. The selection of the keywords and phrases are based on the manually-labeled 903 ad reviews in Section 2.2.1 and the discussion between the first two authors.

Since the ultra imbalanced distribution of categories, which might introduce too much bias for training multi-label classifier [26], we sample up to 280 training instances for each categories. After removing the duplicated multi-labeled reviews, we have 3,630 ad reviews as our training data. The ad reviews vary in length, from several words to hundreds of words. Since a large proportion of review texts may cover a wider range of app issues besides the ad-related ones, we focus on the ad-related sentences, *i.e.*, the sentences containing keywords related to ads (regex=ad/ads/advert*), instead of the whole reviews.

Step 2: Supervised multi-label classification. We use the automatically annotated ad reviews from the previous step as training data for supervised learning of multi-label classification. We then use another manually annotated review set as our validation set for reporting the performance of our multi-label classifier. We first tokenize each ad review into bag-of-words form, remove the common stop words provided by the NLTK toolkit.[3] Then, we lemmatize each word of the review using the popular WordNet Lemmatizer and convert each ad review into tf-idf feature vector. Finally, we utilize Classifier Chains (CC) [27] approach to transform the problem of classifying multi-labeled data into one or more problems of single labeling, and use the well-known Support Vector Machine (SVM)[4] as the basic estimator of CC to build a classifier based on the training data and to classify the remaining ad reviews.

3. Findings

In this section, we try to answer the research questions illustrated in Section 1 and elaborate on our findings.

3.1. RQ1: What are the common types of ad issues?

3.1.1. Motivation

Users play an essential role in the ad-profiting process, since the number of ads viewed or clicked by users determines the ad revenue: User retention and user base are critical for app developers. However, embedding ads inappropriately can ruin user experience. According to a survey in 2016 [28], two in three app users consider mobile ads annoying and tend to uninstall those apps or score them lower to convey their bad experience. Such negative feedback is likely to influence other potential users, which further leads to customer churn and reduced ad revenue. This motivates us to capture the complained ad issues in ad reviews, and draw developers' attention to the problematic aspects of ad usage.

3.1.2. Method

We first evaluate the multi-labeling classifier introduced in Section 2.2, and then use the trained classifier to automatically annotate the whole ad review corpus. For evaluating the classifier, we manually labeled another 280 ad reviews. We compare the result of the automatic classifier with the manual annotation using the label-based precision and recall as evaluation metrics [21]. Given that there are L labels, i.e., issue types, there are L precision and recall. Precision for an issue type refers to the proportion of ad reviews that are correctly assigned to the type, among those that are assigned to the type. Recall for an

[3] https://www.nltk.org/nltk_data/.
[4] https://scikit-learn.org/stable/modules/svm.html.

Table 3
Ad review examples that are labeled as "Other/Unknown" Type.

App name	Title	Review text	Star
Spotify Music	As *advertised*	I've only used it when hooked to WiFi, but so far this app has been awesome.	5
Netflix	Now works	Since the last update, I had problems starting the app, remained to load the splash screen *ad infinitum*[a], then I deleted and reinstalled ...	5
Spotify Music	Love Spotify	... Feels like free music even if I don't have the *free ads version*.	5
YouTube	Unusable	It took me 1 h to watch a 10 min video because it either stops loading or stops playing all together. Whats worse is any other time I try and watch a video it doesn't even load. But *the ads load fine*	1
Spotify Music	Emt	The free version has *annoying ads* and limitations, but certainly a good premium.	4

[a]Ad infinitum is a Latin phrase meaning "to infinity" or "forevermore" [25].

Table 4
Example of evaluation measures.

Two review examples	Labels	L1	L2	L3	L4
Review 1	True	✓	✓		
	Predicted		✓	✓	
Review 2	True	✓	✓		✓
	Predicted	✓			✓
Average Precision by Label	$\frac{1}{4}(\frac{1}{1}+\frac{1}{1}+0+\frac{1}{1})=0.75$				
Average Recall by Label	$\frac{1}{4}(\frac{1}{2}+\frac{1}{2}+0+\frac{1}{1})=0.5$				

issue type refers to the proportion of ad reviews that are correctly assigned to the type, among those that actually belonging to the type. Table 4 illustrates an example of the evaluation measures for two sample reviews.

3.1.3. Findings

Table 5 summarizes the result for each ad issue type. We observe that the precision and recall are acceptable (more than 80%). We then use the built classifier to categorize all the ad reviews. Table 6 summarize the total numbers and percentages of ad reviews classified into each issue type. We remove the 18,007 reviews grouped into "Other/Unknown" category, which leaves us with 18,302/36,309 reviews.

Users complain most about the number of ads and ad display frequency. From the reviews clearly expressing ad issues, we observe that most of the ad reviews complain about the number of ads (45.51%) and display frequency (25.02%). Although in-app advertising is an effective monetization strategy for mobile developers, too many ads and their frequent display can severely degrade user experience. For example, one user complained that "*There are too many ads whenever I try to switch to a different song after an ad. Make it stop. It's really annoying*". Some apps provide reward ads, i.e., offer something to the user in exchange for watching or interacting with an ad. One example is Spotify Music. The users can enjoy 30-min ad free music streaming by watching an ad video in the app. Such reward ads would be less unfavorable to users. For instance, one user commented that "*...You can still listen to music if you're ok with ads every once in awhile. 30 s worth of ads, it isn't that bad*". According to one survey in 2018 [29], reward ads were rated as the most effective for delivering the best user experience by the majority of survey respondents. **Thus, developers could design a reward strategy to alleviate users' dislike for ads.** Additionally, the ads' display frequency should be set at a comfortable rate for app usage. For instance, one YouTube user stated that "*There are too many ads in a video that is 30 min and there are 7 ads*", and gave one-star rating. **Developers could devise an A/B testing experiment to determine an optimal frequency ads should be displayed in an app.**

Table 5
Multi-label classifier precision and recall results.

	Ad issue type	#Training data	#Test data	Precision	Recall
In-Ad	Content	280	16	91.45%	77.78%
	Frequency	280	18	89.74%	85.37%
	Popup	280	21	92.44%	100.00%
	Too Many	280	86	91.94%	93.44%
	Non-skippable	280	14	100.00%	72.73%
	Timing	280	21	82.71%	88.44%
	Size	280	17	83.31%	100.00%
	Position	280	12	79.36%	90.48%
	Auto Play	280	16	100.00%	100.00%
	Volume	119	10	100.00%	80.00%
App	Security	280	11	81.58%	89.26%
	Crash	280	15	75.17%	94.73%
	Slow	280	6	100.00%	83.33%
	Notification	280	3	100.00%	100.00%
	Orientation	98	2	100.00%	100.00%
Average		254.46	20.33	91.18%	85.57%

Developers should pay attention to popup ads, ad timing, and ad content. For the in-ad issues, we observe that reviews related to popup (13.52%), timing (12.11%) and content (8.81%) also occupy large proportions among the whole ad reviews. Popup ads can effectively grab the attention of customers, but can also interrupt their interaction with apps. Popping ads in a video is popular among publishers that offer video content within their app [30], and usually display with skippable or close options. For example, one three-star-rating review described that "*... There are times when I'm about to watch a video and an advert pops which I can choose to skip...*". It is worth noting that the popup ads appearing during a call or when music is playing, etc., can lead to extremely unpleasant experience for users. One Tango user commented that "*... right in the middle of a call there was an ear splitting sound. And when I looked at the phone screen there was an ad...*". **Developers should be careful on introducing popup ads, especially ads with audio, such that they do not substantially reduce user experience.**

Too long ad display period and uninteresting content can also interfere with apps' usage for users. A one-star-rating review stated that "*...It's bad enough that I have to sit through a 30-s ad that I'm even not*

Table 6
Categories and distribution of ad issue types for the 18,302 reviews which are not grouped in the "Other/Unknown" category.

Ad issue type		Ad issue description	Search keywords/Phrases	Count	%Count
In-Ad	Content	What is in the ads shown to users	irrelevant, same ad, open install page, target advertisement, random ad, advertise what	1,613	8.81%
	Frequency	How often ads appear in an app	every time, ad rate, continuously, occasional ad, more than once, constantly, every <digit> second	4,580	25.02%
	Popup	The way that ads suddenly appear to users	pop, interruption, the middle of, half way through, onslaught, pop up, during a video, popup, suddenly, keep get in my way, interrupt	2,475	13.52%
	Too Many	How many ads are displayed to users	more ad, increase number, a few ad, ton of advertise, abundance of advertise, fill with ad, more and more, only with advertise, full of ad, much ad, a pile of advertise, more advertise, much advertise, some ad, lot ad, many, lot of ad, block	8,329	45.51%
	Non-skippable	Ads cannot be skipped by users	cant skip, be able to skip, skippable, wont stop, skip available	703	3.84%
	Timing	Time interval of ad displaying	long, permanent, much time, short, never end, brief	2,216	12.11%
	Size	How big of an ad	tiny, space, huge, half of the screen, banner	385	2.10%
	Position	Where an ad is placed	UI, bottom, too high, at the top, front page, button, in browser page, below	1,233	6.74%
	Auto Play	The way that ads start without permission	auto play, automatically play, auto skip	359	1.96%
	Volume	Sound level of video or audio ads	loud, screech, play sound, volume	159	0.87%
App	Security	Unauthorized data collection or permission usage	collect information, scam, private, virus, access your camera, listen through, monitor	341	1.86%
	Crash	Apps not working caused by ads	black screen, doesnt work, doesnt load, turn black, not respond, dont work, freeze, stall, crash	1,846	10.09%
	Slow	Slow app functionalities caused by ads	buffer, laggy, delay, forever to load, for age, slow, for ever to load, take minute to load, lag, try to load, take time to load	614	3.35%
	Notification	Ads notifying users through the status bar	push ad, notification	338	1.85%
	Orientation	The orientation of app screen impacted by ads	portrait, horizontal screen, landscape	105	0.57%

interested in...". Long ad display periods and uninteresting content could try users' patience, and may drive potential users away. **Developers should provide skip option for long ads or consider better personalization to only present long ads with contents highly likely to be of interest to users.**

Developers should notice ads' impact on apps' functionalities. For the app-level issues, we find that crash (10.09%) and slow response (3.35%) issues are non-trivial in number among the whole ad reviews. Ad modules may be poorly implemented and incompatible with app functionality, resulting in app breakdown or slowing performance. For example, one user complained that "*Utterly disappointed with the current ads situation. They mess up AirPlay big time. Try airplaying to your TV. The moment ad starts, everything freezes*". In this case, users cannot use the app properly. **Thus, developers should carefully integrate and test the ad libraries before deployment.** Besides the crash and slow response issues, 1.86% complain about security, 1.85% are related to notification through status bar, and 0.57% are about app orientation being affected by ads. Users are less likely to complain about issues of these types.

Finding 1: Users care most (70.53%) about the number of ads and ad appearing frequency among the ad issues. Other ad issue types such as the design of popup ads, ad timing, ad content, and crash also occupy obvious proportions among the ad reviews.

3.2. RQ2: What are relationships between ad issue types that users described in their ad-related reviews and the ratings that they gave?

3.2.1. Motivation

In RQ1, we identified the ad issue types commonly expressed via user feedback, and analyzed their quantity distributions. Besides review text, each ad review comes with a rating provided by the user on App Store and Google Play. Since user ratings influence how app platforms display apps in response to a user search, and have a great impact on the number of app downloads [31], understanding the users' rating behavior when they complain about ad issues is important. We aim at identifying the ad issue types that are more likely to impact user ratings in this question.

3.2.2. Methods

To answer RQ2, we first divide the ad reviews into three polarities, i.e., positive, neutral, and negative, according to the given ratings, and then compare the quantity distributions of different ad issue types for the three sentiment polarities. We consider the reviews with lower ratings (e.g., one or two) as negative reviews, the ones with higher ratings (e.g., four or five) as positive instances, and the others as neutral reviews.

We determine whether user ratings are independent of ad issue types by using Pearson's Chi-Squared test [32] (or Chi-Squared test for short) at $p\text{-}value = 0.05$ [33].

We use Mann–Whitney U test [34], a non-parametric test, to observe whether two issue types have significantly different rating distributions. We set the confidence level at 0.05 and apply the standard Bonferroni correction (which is the most conservatively cautious of all corrections)

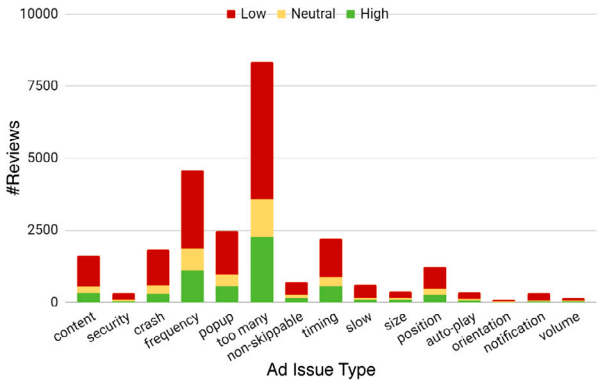


Fig. 1. Count distributions of reviews with high, neutral and low ratings across different ad issue types.

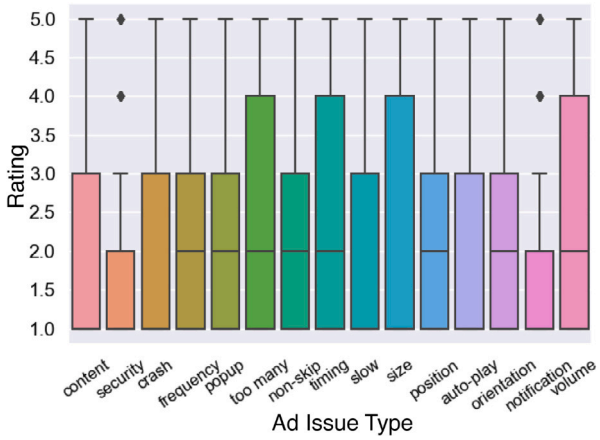


Fig. 2. Rating distribution of different ad issue types.

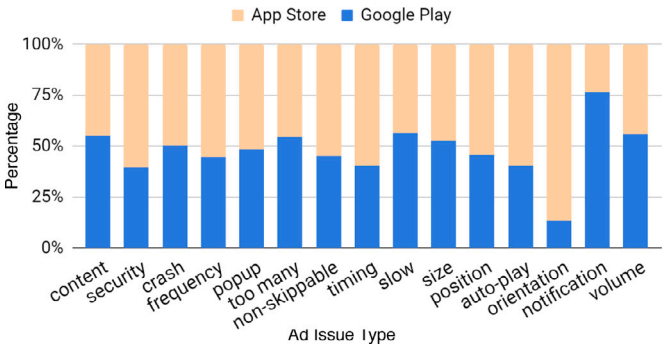


Fig. 3. Percentage distribution of ad issue types across App Store and Google Play.

to account for multiple statistical hypothesis testing. To show the effect size of the difference between the two types, we compute Cliff's Delta (or d), which is a non-parametric effect size measure [35]. Following the guidelines in [35], we interpret the effect size values as small for $0.147 < d < 0.33$, medium for $0.33 < d < 0.474$, and large for $d > 0.474$.

3.2.3. Findings

Fig. 1 illustrates the quantity distributions of the 15 ad issues among feedback with high, neutral, and low ratings. Overall, low- and neutral-rated feedback occupies a major proportion (76.11%) in the ad reviews, implying that **ad reviews tend to have a negative impact on user ratings**. The result of Chi-Squared test ($p\text{-}value = 1.94e-44$) indicates that user ratings and ad issue types are strongly correlated, which means that users tend to rate the severity of different issue types differently.

The *too many* and *frequency* issue types receive the highest number of negative ratings among all the types. Focusing on the negative and neutral ad reviews (as shown in Fig. 1), we find that most of them talk about the number of ads (i.e., the *too many* category) and display frequency. These two issue types account for nearly half of all the low-rated reviews (48.37%). This implies that **users who are averse to ads mostly complain about the number of ads or their display frequency**.

Developers should notice the issues related to security and notification. Fig. 2 shows the rating distributions of different ad issues. We can observe that most of the ad reviews discussing about specific ad issues are scored with ratings lower than or equal to three, with median star ratings at two. By computing the average scores, we discover that both the *security* and *notification* issues have the lowest ratings (1.8) on average. For example, one one-star-rating review from WeChat says that "*...Every time I try to watch or do something, the ad notifications always pop out, and it always directly opens App Store by itself...*". We further use Mann–Whitney U test to examine whether these two issues receive significantly lower ratings than other issue types respectively. The results of Mann–Whitney U test ($p\text{-}value < 0.05$) and $d > 0.147$ show that both issues have significantly different rating distributions from other issues with at least a small effect size. Thus, developers need to notice the two issue types and try to fix them quickly (more details can be found in Section 3.4).

Developer should be cautious about popup and crash-related ad issues. Focusing on the issue types with median values at 1.0 (as shown in Fig. 2), we find that the *auto-play*, *popup*, *crash*, *size*, and *slow* issues also correspond to low star ratings besides the *security* and *notification* issues. Considering the percentage distributions obtained in RQ1, we suggest that developers should pay attention to the reviews complaining about *popup* and *crash*, as both constitute of more than 10% of the ad reviews.

> **Finding 2:** Nearly half (48.37%) of the negative and neutral ad feedback relates to the number of ads and ad display frequency. Besides, developers should pay attention to the ad issues related to popup and crash which tend to receive poorer user ratings and account for more than 10% of ad reviews. Also, the *security* and *notification*-related ad reviews generally receive lower scores than other types of ad reviews.

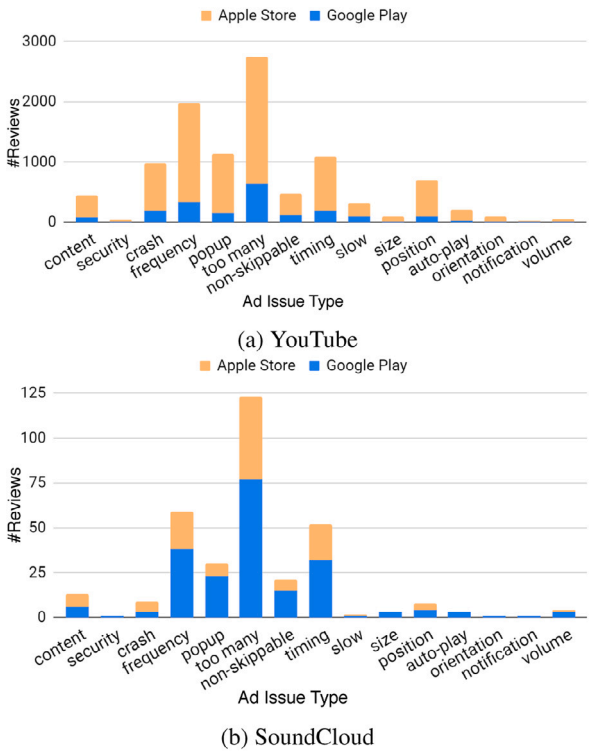
3.3. RQ3: How different are the distributions of ad issue types in different platforms?

3.3.1. Motivation

Popular apps generally publish their products on multiple systems, such as Android, iOS, and Windows. A report in 2018 [36] showed that the cross-platform app market was expected to hit $7.5 billion by 2018, and the amount was still on the rise. Users of different platforms may have different preference. Also, the two operating systems are different in many aspects, such as Android is more customizable. For maximizing mobile revenue, many popular apps choose to publish their app versions on multiple platforms, especially App Store and Google Play [37]. Thus, studying the difference of ad issue distributions on the two platforms can help developers weight ad issue types according to the platforms during ad design.

3.3.2. Methods

Based on the quantity distributions of ad issue types across platforms per app, we also use Pearson's Chi-Squared test [32] to determine whether two datasets have the same distribution. As a null hypothesis, we make the assumption that for the same app, the frequency distributions of issue types are similar on different platforms.



(a) YouTube

(b) SoundCloud

Fig. 4. Review quantity distributions among ad issue types for YouTube (a) and SoundCloud (b).

3.3.3. Findings

Fig. 3 shows the percentage distribution of ad issue types on the two platforms. **We can observe that generally some issues such as *security*, *timing*, *auto-play*, and *orientation* are more complained by iOS users, while other issues including *notification*, *volume*, and *slow* are more concerned by Android users.** By applying Chi-Squared test to the count distributions of ad issue types among the subject apps, we find that all the issue types show significant differences cross platforms, all with $p\text{-}value < 0.001$. This indicates that the distributions of ad issue types are significantly different on different platforms. **Developers should design ad maintenance strategies differently for the two platforms.**

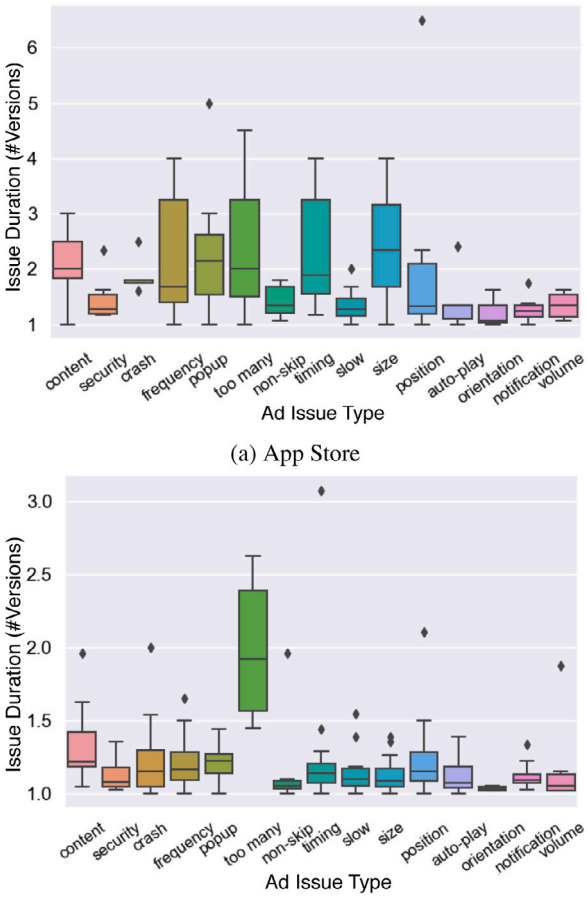
We further analyze the quantity distributions of ad issue types for each subject app. Fig. 4(a) and Fig. 4(b) illustrate the review quantity distributions on the issue types for YouTube and Soundcloud, respectively. We can see that the two apps present obviously opposite issue distributions across platforms, e.g., SoundCloud receives more reviews related to the *too many* issue from Google Play than those from App Store, while it is the opposite for YouTube. For SoundCloud, although the difference between issue distributions on both platforms is not statistically significant, its Android app has clearly more ad complaints than its iOS app. This observation further suggests that developers should design ad maintenance strategies according to the deployed platforms.

Finding 3: The quantity distributions of the ad issues show significant differences between App Store and Google Play. For an app, its issue distributions may also behave differently for different platforms.

3.4. RQ4: What types of ad issues are addressed more quickly by developers?

3.4.1. Motivation

App developers would address the important app issues feedbacked by users in the updated versions. Similarly, if one ad issue is solved by developers in a timely manner, we can infer that the ad issue is crucial from developers' perspective. We suppose that the developers of popular apps are experienced, and can prioritize issues during maintenance professionally. Thus, the duration of an ad issue can reflect whether the issue type is valued by developers, and provide us additional insights into ad maintenance.



(a) App Store

(b) Google Play

Fig. 5. Durations of ad issue types on App Store (a) and Google Play (b). The duration is measured in the number of versions.

3.4.2. Methods

We first determine the subject apps for this question by removing those with the number of consecutive versions fewer than three in our collection, since more versions provide us more accurate information about issue changes. As it is challenging to manually check whether an ad issue is indeed fixed in one app version practically, we follow the common strategy used in defect warning analysis [38]. In this paper, we define an ad issue is addressed by developers if its percentage is significantly reduced in a version, and not increased in the next version.

We suppose that the percentages of one ad issue over versions $P = \{p_1, \ldots, p_v, \ldots, p_V\}$, where V denotes the total version number, follow a Gaussian distribution $P \sim G(\mu, \sigma)$. An ad issue in one version can be considered addressed if $\frac{p_v - \mu}{\sigma} > \epsilon$, where ϵ indicates how far the actual value differs from the expected value relative to the typical difference. In statistics, a relative deviation of 2 (i.e., $\epsilon = 2$) is often considered as significant [39]. Thus, if $\epsilon > 2$ and a decreased trend appears in version v (i.e., $p_v < p_{v-1}$), we can assume that the ad issue is addressed for that version. The issue duration d is calculated as the version span between the nearest version v_n with at least one user-review regarding the issue and the current version v, i.e., $d = v - v_n$.

Table 7
Number of identified versions for the selected 20 apps.

App Name	Platform		App name	Platform	
	App Store	Google Play		App Store	Google Play
Candy Crush	6	14	Duolingo	17	47
eBay	17	24	SHAREit	3	50
Amazon	5	17	SoundCloud	8	24
Minion Rush	4	5	Subway Surfers	7	20
Netflix	9	55	TripAdvisor	8	11
YouTube	14	99	Trivia Crack	20	47
My Talking Tom	3	15	Skype	9	40
Twitter	35	68	Pinterest	12	30
Snapchat	10	23	Viber	5	35
Waze	8	20	Instagram	8	70

3.4.3. Findings

We first remove the apps with fewer versions (i.e., < 3) or no version information, where the apps will also be removed if we only have their version information on one platform. The version information of each app is identified from App Annie [20]. After this step, we have left 20 apps and 922 versions in total. Table 7 lists the version number for each app, and Fig. 5 presents the computed duration distributions among the ad issue types on App Store (upper) and Google Play (below). The result of Mann–Whitney U test [34] ($p\text{-}value = 6.53e-5$) on the average issue durations across platforms shows that developers solve ad issues in significantly different paces for different platforms. As can be observed in Fig. 5, issues on Google Play, with average version duration at 1.23 and median duration at 1.19, are generally addressed more quickly than the ones on App Store (avg. 1.78 and med. 1.47).

We also find that **some ad issues would be more quickly addressed by developers than other issue types.** For instance, iOS developers tend to solve *orientation*, *auto-play*, and *notification* issue types more quickly. For Android developers, they would solve the *orientation*, *volume*, and *non-skippable* issue types in the next updated version, with median issue durations at 1.03, 1.05, and 1.05 versions, respectively, as shown in Fig. 5(b). Taking an example of YouTube, the app used to receive several user feedback complaining about the non-skippable ads in version 12.01.55. One user commented that "*I don't have the option to enable non-skippable in-stream ads on my videos, what can I do?*". The issue also aroused an intense discussion on YouTube Community [40], and was finally solved by the developers [41]. In our collected reviews, the number of ad reviews related to the non-skippable issue dropped to zero in the next updated version. For the SHAREit app, the version 4.6.88 receives many user feedback complaining about the *notification* issue. For example, one user described that "*It is very useful and everyone I know use this app but the problem is there are just too many ads especially on my notification tab and sometimes the ads have explicit images. I have to disable the notification on the app.*", and the issue was also intensively discussed on the Reddit forum [42]. We also confirmed that all the ad reviews in the next updated version were not associated with the notification issue, which may reflect that the issue was fixed in the update.

For some ad issues such as *too many* and *content*, both iOS and Android developers may take a longer period to address them. One possible reason is that the ad module is built on specified ad provider and the ad content may be difficult for developers to modify. Overall, some ad issue types are more likely to be solved in the next immediate version while other issue types may exist in several consecutive versions.

Finding 4: Some ad issue types are prone to be quickly addressed by developers than other ad issues. Also, developers of different platforms for the same app may solve ad issues in different paces.

4. Implications

In this section, we describe the implications of our findings on ad design, ad maintenance, and future research.

4.1. Implication on in-app ad design

Developers should optimize the ad display settings such as the number of ads, display frequency, and display style. From our analysis, we find that the complaints about the display settings occupy a substantial percentage of studied ad reviews and the display setting related reviews tend to be accompanied with poor ratings. Developers are suggested to conduct A/B testing to determine an optimal setting for in-app ads. Moreover, strategies such as integrating rewards for watching ads can alleviate users' dislike for ads. It is also worth noting that popup ads appearing during a call or when music is playing, can lead to an unpleasant experience for users, and should be avoided.

Developers should carefully design effective strategies to manage ads with long display periods. We observe that the content and timing-related issues also account for a substantial percentage of ad reviews. Watching long video ads that are not of interest to users would try their patience. Developers should design effective personalization strategy to recommend the right ads of interest to different users. Providing a skip option is another strategy to relieve users' negative emotions.

4.2. Implication on in-app ad maintenance

Developers should ensure app stability as ads are displayed in apps. Our findings indicate that the crash-related issue appears in a large number of reviews, and corresponds to low user ratings. If the ad libraries are configured incorrectly, the apps' functionalities could be corrupted or slowed down. So we recommend developers to carefully integrate and test the ad libraries before deployment, and to fix the related issue in a timely manner.

Developers should prioritize ad issues on different platforms differently. Our findings demonstrate that the quantity distributions of the ad issue types across different platforms are significantly different. For example, iOS developers tend to solve *orientation*, *auto-play*, and *notification* issue types more quickly than Android developers; while Android developers care more about the *orientation*, *volume*, and *non-skippable* issue types. These results suggest that app developers for a specific platform (Android or iOS) need to put more focus on a subset of ad issues during ad maintenance instead of treating them equally. Besides, some of the issues are determined by the ad platform and hard to be controlled by app developers. So developers of different platforms should identify and try to address the corresponding controllable issues, and report the uncontrollable issues to the ad platform.

4.3. Implication on future research

More empirical research on balancing user experience and ad revenue is needed. Although anecdotal evidence exists on the adverse impact of in-app ads, unfortunately, few research work has empirically explored how to properly design mobile ads while preserving ad benefits (*e.g.*, click-through rate and ad revenue). We encourage future researchers to perform such studies so that impact of detailed ad design strategies (*e.g.*, choice of ad format and content, ad display frequency, etc.) to ad revenue can be measured and estimated. Developers can then pick ad design strategies in a more informed way by considering the trade-offs of ad revenue and its negative impact to user experience.

More research on studying the strategies to deal with different types of ad issues is needed. Our findings show that developers should pay attention to some types of ad issues, e.g., the design of popup ads, ad timing, ad content, and crash. The ad issues are commonly faced by app developers, but how to well address the issues has rarely been explored. The developers may consume much time on fixing the issues without relevant guidance. Future research is encouraged to investigate and provide strategies to mitigate the ad issues separately.

5. Threats to validity

5.1. External validity

Threats to external validity concern the possibility to generalize the findings [43]. In this work, we consider two platforms, App Store and Google Play, as these two platforms are the two largest global app markets [37]. We select 32 apps that exist in the top 100 app charts in both Google Play and App Store as subjects. Hence our results may not generalize to all mobile applications. To mitigate this threat, the apps are selected to cover a broad range of categories and have a significant number of user reviews on both platforms for ensuring their popularity and representativeness. Besides, the two platforms may not provide access to all the user reviews. Martin et al. [44] observed that using incomplete data in app stores may bias the findings. To reduce such a bias on the findings, we collect all the user reviews (i.e., 1,840,349 and 3,243,450 reviews for App Store and Google Play respectively) gradually from September 2014 to March 2019.

5.2. Internal validity

First, we identify the ad reviews if they contain keywords related to ads, *i.e.*, regex = ad/ads/advert*. Such strong criterion could lead to significant numbers of true negatives, and might affect the soundness of our findings. To explore the influence caused by the retrieval method, we randomly label a statistically representative sample of 1000 reviews (out of the whole 5,083,799 reviews), providing us with a confidence level of 95% and a confidence interval of 3%. The labeling process was conducted by the first author and the second author separately, and reached 100% agreement rate from both authors. Among the 1000 reviews, five reviews are labeled as related to in-app ads, and our retrieval method can achieve 83.3% (5/6) and 100% for precision and recall, respectively. This indicates that the regex-based retrieval method can identify ad-related reviews completely.

Second, our manual categorization of the ad reviews is subjected to annotators' bias. We alleviate such threat by following standard card sorting process and making sure that the two annotators agree on the final decision.

Third, our effort to automatically categorize numbers of ad reviews could potentially raise some questions. Especially, the categorization can be tainted by the initial choice of keywords. Also, users express the same issues in various ways. To mitigate the threat, we evaluate our classification against the manual annotation of 280 ad reviews, as discussed in Section 2.2.2. Regarding the classification methods, we compare the adopted algorithm, *i.e.*, combining Classifier Chains approach with Support Vector Machine (CC+SVM), with other typical multi-label classifiers, including random weighted classifier [45], K nearest neighbors (KNN) algorithm[5] [46], and also CC jointly trained with Logistic Regression (CC+LR) [47]. Table 8 presents the comparison results. The results demonstrate that Classifier Chains (CC) algorithms have a better performance when compared to Random Weight and KNN algorithms. For the basic estimator of CC, SVM shows a better performance than LR. Therefore we choose CC with SVM as our multi-label classifier in our study.

Fourth, during answering RQ2, we suppose that the given ratings are relative to the ad issue types, and then analyze the relationships. However, the given ratings are actually caused by various factors besides the ad issues, and users from different platforms may perceive differently for the same ad issues [48]. To mitigate such threat, we involve a large number of reviews, i.e., 18,302 reviews, for analysis. In future work, we will consider using sentiment analysis tools to infer users' sentiment about the ad issues.

[5] We set $K = 5$ via cross-validation.

Table 8
Comparison results of different multi-label classifiers.

Method	Precision	Recall
Random weight	78.43%	75.18%
KNN	68.92%	66.01%
CC+LR	87.13%	79.54%
CC+SVM	**91.18%**	**85.57%**

Table 9
Multi-label classifier precision and recall results.

	Ad issue type	Precision	Recall
In-ad	Content	80.00%	62.50%
	Frequency	80.00%	76.92%
	Popup	92.00%	83.33%
	Too Many	90.00%	86.36%
	Non-skippable	98.00%	71.43%
	Timing	78.00%	86.67%
	Size	68.00%	100.00%
	Position	76.00%	90.48%
	Auto Play	76.00%	100.00%
	Volume	94.00%	83.33%
App	Security	88.00%	100.00%
	Crash	80.00%	100.00%
	Slow	82.00%	100.00%
	Notification	86.00%	100.00%
	Orientation	88.00%	100.00%
Average		83.73%	89.40%

Finally, during answering RQ4, we determine the fixing durations of an ad issue according to the percentages of the issue over versions by following the strategy in defect warning analysis [38]. However, it is difficult to manually evaluate whether an ad issue is indeed fixed in one app version, due to the inaccessibility of the full changelogs for the subject apps. We alleviate such threat by providing more case analysis. In the future, we will try to collaborate with industrial companies for collecting practical data and solve the question with more comprehensive data.

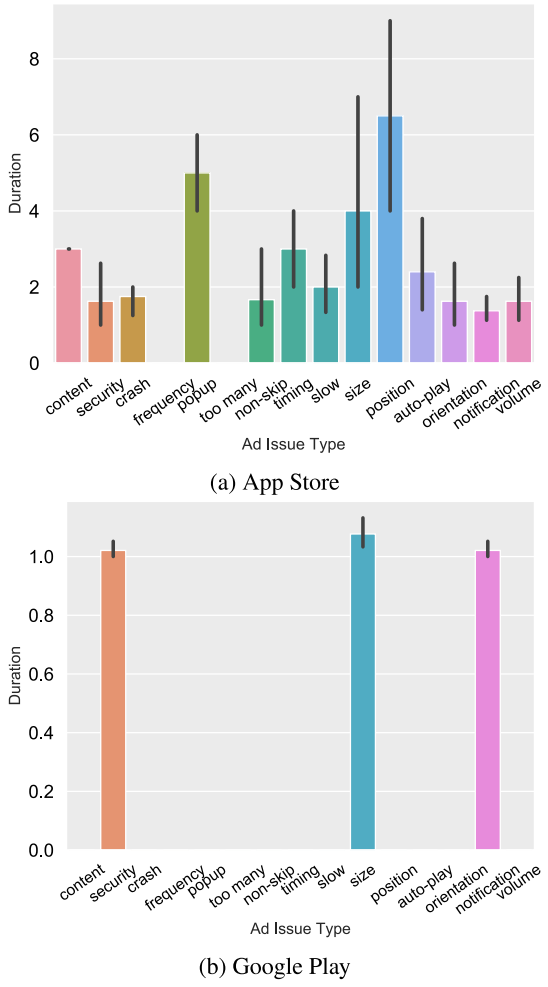
5.3. Construct validity

There are several approaches to understand what aspects users are complaining about mobile in-app ads. For example, interviewing and surveying mobile users might be one way. In this paper, we chose to instead look at the actual user feedback. Both approaches have their benefits and limitations. For example, with surveys, users might miss reporting on some ad issue types since we are depending on their collection. Nevertheless, a mining approach might be limited since the collected data cannot represent all issue types related to ads. Thus, we suggest that future studies are needed to triangulate our findings through user surveys.

6. Discussion

6.1. More evaluation of the multi-label classifier

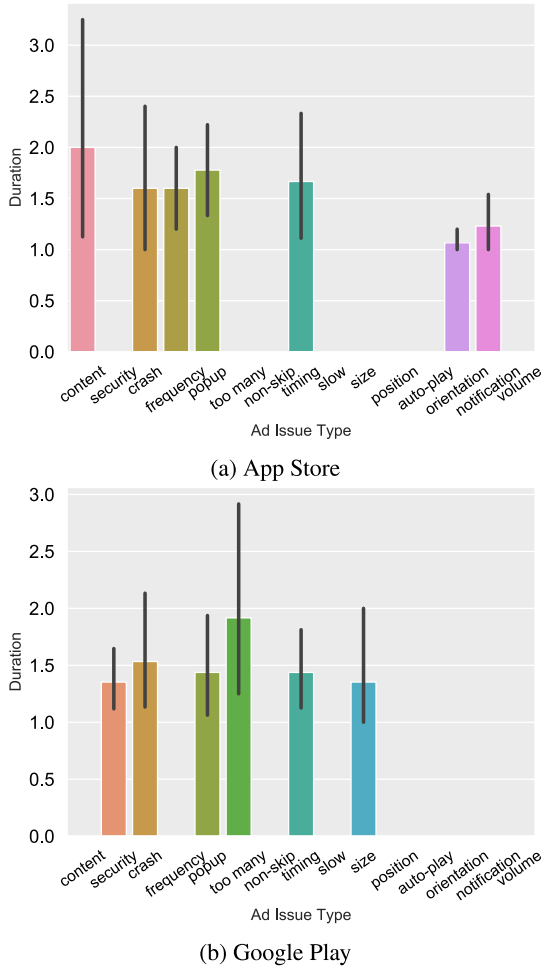
In this work, we evaluate the performance of different categories on a small test set, which may not well reflect the performance of the multi-label classifier. To ensure a sufficient number of test reviews for each type, we randomly select 50 reviews associated with each ad issue type from the 18,302 reviews in Table 5, which results in 738 unique reviews for further manual annotation. The first two authors discusses together about the ad issue types associated with each review, and then evaluate the multi-label classifier on the new annotated dataset. The results are illustrated Table 9. We can observe that the multi-label classifier also achieves high performance on the larger dataset, which further demonstrates the classifier's effectiveness.



(a) App Store

(b) Google Play

Fig. 6. Durations of ad issue types on App Store (a) and Google Play (b) for the YouTube app. The zero durations for some ad issues mean that the percentages of the issues do not present a significantly reduced trend over versions.



(a) App Store

(b) Google Play

Fig. 7. Durations of ad issue types on App Store (a) and Google Play (b) for the eBay app.

6.2. Manual inspection of the app-related ad reviews

In Table 5, the ad issue types are further categorized into two large groups based on whether they are related to the ads (In-Ad) or the impact of ads on apps (App). The "App" categories include five ad issue types, such as security, crash, slow, notification, and orientation. However, the mentioned ad issues in reviews, e.g., the crash issue, may not be indeed related to the ads. To inspect the relationship between the app issues and ads, we conduct a manual analysis on the reviews classified as the "App" category.

For the 2822 unique reviews belonging to the "App" category, we randomly selected 251 ad reviews to give us a 95% confidence level with 6% confidence interval. The first two authors then annotated the reviews separately. The Cohen's kappa score is 0.81, which indicates the two annotators achieve high agreement. Finally, the annotators discussed together to reach a consensus. The results show that 90.8% of the app issues are related to the ads. For example, one review described that "*Every time I click a video and an ad play, then it crashes. Keep crash.*". For the remained 9.2% of the reviews, most of them do not actually complain about the ads, e.g., the review "*The video opens instantly. No issue of advertisement as well as it play with out any buffer.*". The results imply that our keyword-based automatic classifier may not well learn the impact of negative words on the classification. Future work can consider involving semantic dependency parsing for more accurate issue type classification.

6.3. Further discussion of RQ4

In Section 3.4, we analyze the fixing durations of different types of ad issues in terms of app versions. However, the version durations would be significantly different from app to app. We choose two apps, i.e., YouTube and eBay, for more detailed analysis. The durations of different ad issue types for the two apps are illustrated in Figs. 6 and 7, respectively. We can find that the fixing durations are obviously different for different platforms for the two apps. For example, the *size* issue type of YouTube on the Android platform is fixed more quickly than the issue type on the iOS platform. The results of Mann–Whitney U test [34] ($p\text{-}value < 0.001$) on the average issue durations across platforms also show that developers of both apps solve ad issues in significantly different paces for different platforms. We further observe that some issue types would be faster addressed than other issue types, e.g., the *security* and *orientation* issue types with average issue durations fewer than two versions. The observation is consistent with our findings in Section 3.4.

6.4. Analysis of the security-related ad reviews

Adware is one kind of malware that is designed to display ads, redirect users' search requests to advertising websites and collect marketing-type data about users without permission [49]. Apps providing multiple functionalities (*e.g.*, photographing and messaging) usually require sensitive permissions (*e.g.*, contact access) to be granted

Table 10
Analysis on the topics that users complained about in the ad reviews related to the *security* issue type. "Private Data Collection" refers to reviews that describe concerns about collection of private data; some reviews specifically mention the type of the private data (contact, message, etc.), while some others do not specifically this. "Private Data Transaction" refers to reviews that describe concerns about private data being sold to many other parties.

Security-related topic		Number	Percentage (%)
Private data collection	Private Information	11	37.9%
	Camera	2	6.9%
	Microphone	5	17.2%
	Contact	4	13.8%
	Message	2	6.9%
	Phone History	1	3.4%
	Location	1	3.4%
Privacy data transaction		9	31.0%

by users. These granted permissions could be maliciously employed by the in-app ads for targeted advertising. We manually analyze the ad reviews related to the *security* issue type. As shown in Table 10, some users are concerned that their privacy data collected for ads might be improperly used or further transacted. As adware can acquire private data in various ways, it would be challenging to perform static analysis to determine if an app is an adware. However, user feedback can effectively be used by store owners and administrators to detect whether there are potentially malicious behaviors in in-app ads after app release.

7. Related work

7.1. App review analysis

App review analysis explores the rich interplay between app customers and their developers [50]. The analysis has been proven helpful and significant in various aspects of app development.

Iacob et al. [51] manually label 3,278 reviews of 161 apps, and discover the most recurring issues users report through reviews. Since mining app reviews manually is labor-intensive due to the large volume, more attempts on automatically extracting app features are conducted in prior studies. For example, Iacob and Harrison [52] design MARA for retrieving app feature requests based on linguistic rules. Man et al. [48] propose a word2vec-based approach for collecting descriptive words for specific features, where word2vec [53] is utilized to compute semantic similarity between two words. Vu et al. [39,54] have investigated how to facilitate keyword retrieval and anomaly keyword identification by clustering semantically similar words or phrases. Another line of work focuses on condensing feature information from reviews and captures user needs to assist developers in performing app maintenance [55,56]. Maalej and Nabil [57] adopt probabilistic techniques to classify reviews into four types such as bug reports and feature requests. Di Sorbo et al. [55] build a two-dimension classifier to summarize user intentions and topics delivering in app reviews. [58,59], and [60] concentrate on specific app features and propose methods to identify corresponding user sentiment or opinions. There are also review-based explorations aiming at supporting the evolution of mobile apps [61–63]. Specifically, Palomba et al. [63] trace informative crowd reviews onto source code changes to monitor what developers accommodate crowd requests and users' follow-up reactions as reflected in user ratings. They observe that developers implementing user reviews are rewarded in terms of significantly increased user ratings. Other research considers device- or platform-specific app issues [21,48,64]. There are review classification methods such as Naive Bayes classifier and J48 in the literature [65,66]. However, they are not designed for multi-label classification and not applicable for our scenario. We refer to Martin et al.'s survey for an extensive overview of mobile app store analysis research [67].

7.2. User perceptions of in-app ads

According to the research [68], privacy & ethics and hidden cost are the two most negatively perceived complaints (and are mostly in one-star reviews) among all studied complaint types. An interesting empirical study by Gui et al. [15] exhibits obvious hidden costs caused by ads from both developers' perspective (*i.e.*, app release frequencies) and users' perspective (*e.g.*, user ratings). [13,69] discover that the "free" nature of apps comes with a noticeable cost by monitoring the traffic usage and system calls related to mobile ads. Ullah et al. [70] also find that although user's information is collected, the subsequent usage of such information for ads is still low. To alleviate these threats, [12,71] develop a system to enable energy-efficient ad delivery. Gui et al. [72] propose several lightweight statistical approaches for measuring and predicting ad related energy consumption, without requiring expensive infrastructure or developer effort. Gao et al. [73] investigates the performance costs raised by different advertisement schemes, and demonstrates that some ad schemes that produce less performance cost and provide suggestions to developers on ad scheme design. Ruiz et al. [17] also find that integrating certain ad libraries can negatively impact an app's rating. In Gui et al. [74]'s work, ad-related complaints are extracted from manually annotating 400 user reviews. Different from the prior work, we focus on analyzing ad issues based on a large-scale user review corpus and considering multiple factors such as fix durations and app platforms.

8. Conclusion

Inappropriate ad design could adversely impact app quality and ad revenue. Understanding common in-app advertising issues can provide developers practical guidance on ad incorporation.

In this paper, we have presented a large-scale analysis on ad reviews to summarize common issues of in-app advertising. We discover the common ad issue types by manual annotation. Based on the automatic categorization results of a large-scale ad reviews, we observe the general distributions of the ad issue types, the relations between ad issue types and user ratings, the distributions of ad issues across platforms, and fix durations. We summarize our findings and their implications to app developers for more effective and reliable design and maintenance of in-app ads. In the future, we will consider other aspects such as mobile device types to gain further insights about the impact of in-app advertising on app quality.

CRediT authorship contribution statement

Cuiyun Gao: Ideal proposal, Experiments, Writing – original draft. **Jichuan Zeng:** Experiments, Writing, Discussion. **David Lo:** Ideal Refinement, Editing. **Xin Xia:** Editing, Supervision. **Irwin King:** Editing, Reviewing. **Michael R. Lyu:** Writing – review & editing.

Acknowledgement

This work was supported by National Natural Science Foundation of China under project No. 62002084, Stable support plan for colleges and universities in Shenzhen under project No. GXWD20201230155427003-20200730101839009, and the Research Grants Council of the Hong Kong Special Administrative Region, China (CUHK 14210717).

References

[1] Facebook reports, Facebook reports first quarter 2016 results, 2016, https://bit.ly/2p8EB9m.
[2] Facebook ads, Facebook ad revenue growth, 2017, https://bit.ly/2UlQRa4.
[3] Ad report, A hand-held world: the future of mobile advertising, 2017, http://www.business.com/mobile-marketing/the-future-of-mobile-advertising/.
[4] App market, Distribution of free and paid Android apps in the Google Play, 2018, https://bit.ly/2POJPbu.

[5] Ad survey, Top 7 reasons why people uninstall mobile apps, 2016, https://bit.ly/2YXbc43.

[6] M. Backes, S. Bugiel, E. Derr, Reliable third-party library detection in android and its security applications, in: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016, 2016, pp. 356–367.

[7] X. Chen, Y. Zhao, Z. Cui, G. Meng, Y. Liu, Z. Wang, Large-scale empirical studies on effort-aware security vulnerability prediction methods, IEEE Trans. Reliab. 69 (1) (2020) 70–87.

[8] G. Tao, Z. Zheng, Z. Guo, M.R. Lyu, Malpat: Mining patterns of malicious and benign android apps via permission-related APIs, IEEE Trans. Reliab. 67 (1) (2018) 355–369.

[9] J.H. Yu, You've got mobile ads! Young customers' responses to mobile ads with different types of interactivity., Int. J. Mobile Mark. 8 (1) (2013).

[10] S. Soroa-Koury, K.C. Yang, Factors affecting consumers' responses to mobile advertising from a social norm theoretical perspective, Telemat. Inform. 27 (1) (2010) 103–113.

[11] H.K. Chowdhury, N. Parvin, C. Weitenberner, M. Becker, Consumer attitude toward mobile advertising in an emerging market: An empirical study, Int. J. Mobile Mark. 1 (2) (2006).

[12] P. Mohan, S. Nath, O. Riva, Prefetching mobile ads: Can advertising systems afford it? in: Proceedings of the 8th European Conference on Computer Systems, EuroSys, ACM, 2013, pp. 267–280.

[13] S. Nath, Madscope: Characterizing mobile in-app targeted ads, in: Proceedings of the 13th International Conference on Mobile Systems, Applications, and Services, MobiSys, ACM, 2015, pp. 59–73.

[14] M.C. Grace, W. Zhou, X. Jiang, A.-R. Sadeghi, Unsafe exposure analysis of mobile in-app advertisements, in: Proceedings of the Fifth Conference on Security and Privacy in Wireless and Mobile Networks, WISEC, ACM, 2012, pp. 101–112.

[15] J. Gui, S. Mcilroy, M. Nagappan, W.G. Halfond, Truth in advertising: The hidden cost of mobile ads for software developers, in: Proceedings of the 37th International Conference on Software Engineering, ICSE, IEEE, 2015, pp. 100–110.

[16] S. Son, D. Kim, V. Shmatikov, What mobile ads know about mobile users, in: 23rd Annual Network and Distributed System Security Symposium, NDSS 2016, San Diego, California, USA, February 21-24, 2016, 2016.

[17] I.J.M. Ruiz, M. Nagappan, B. Adams, T. Berger, S. Dienst, A.E. Hassan, Impact of ad libraries on ratings of android mobile apps, IEEE Softw. 31 (6) (2014) 86–92.

[18] B. Ray, D. Posnett, V. Filkov, P.T. Devanbu, A large scale study of programming languages and code quality in github, in: Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering, (FSE-22), Hong Kong, China, November 16 - 22, 2014, 2014, pp. 155–165.

[19] P.S. Kochhar, D. Wijedasa, D. Lo, A large scale study of multiple programming languages and code quality, in: IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering, SANER 2016, Suita, Osaka, Japan, March 14-18, 2016 - Vol. 1, 2016, pp. 563–573.

[20] Appannie, App Annie, https://www.appannie.com/.

[21] S. McIlroy, N. Ali, H. Khalid, A.E. Hassan, Analyzing and automatically labelling the types of user issues that are raised in mobile app reviews, Empir. Softw. Eng. 21 (3) (2016) 1067–1106.

[22] D. Spencer, Card Sorting: Designing Usable Categories, Rosenfeld Media, 2009.

[23] A. Begel, T. Zimmermann, Analyze this! 145 questions for data scientists in software engineering, in: 36th International Conference on Software Engineering, ICSE '14, Hyderabad, India - May 31 - June 07, 2014, 2014, pp. 12–23.

[24] M. Kim, T. Zimmermann, R. DeLine, A. Begel, The emerging role of data scientists on software development teams, in: Proceedings of the 38th International Conference on Software Engineering, ICSE 2016, Austin, TX, USA, May 14-22, 2016, 2016, pp. 96–107.

[25] Ad infinitum, https://en.wikipedia.org/wiki/Ad_infinitum.

[26] A. Ali, S.M. Shamsuddin, A.L. Ralescu, et al., Classification with class imbalance problem: a review, Int. J. Advance Soft Compu. Appl 7 (3) (2015) 176–204.

[27] J. Read, B. Pfahringer, G. Holmes, E. Frank, Classifier chains for multi-label classification, Mach. Learn. 85 (3) (2011) 333–359.

[28] Annoying ad, Which ads do Internet users dislike the most?, 2016, https://bit.ly/2ahFPct.

[29] Why marketers should consider in-app advertising, https://bit.ly/2uTKWtu.

[30] In-App Advertising: Trends, tools, and tips for maximizing campaign performance, https://bit.ly/2OU9iwE.

[31] M. Harman, Y. Jia, Y. Zhang, App store mining and analysis: MSR for app stores, in: 9th IEEE Working Conference of Mining Software Repositories, MSR 2012, June 2-3, 2012, Zurich, Switzerland, 2012, pp. 108–111.

[32] K. Pearson, On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling, London, Edinb. Dublin Phil. Mag. J. Sci. 50 (302) (1900) 157–175.

[33] M.L. McHugh, The chi-square test of independence, Biochem. Med.: Biochem. Med. 23 (2) (2013) 143–149.

[34] H.B. Mann, D.R. Whitney, On a test of whether one of two random variables is stochastically larger than the other, Ann. Math. Stat. (1947) 50–60.

[35] S.E. Ahmed, Effect sizes for research: A broad application approach, Technometrics 48 (4) (2006) 573, http://dx.doi.org/10.1198/tech.2006.s437, URL https://doi.org/10.1198/tech.2006.s437.

[36] Cross platform mobile app development guide, https://bit.ly/2MiVt9a.

[37] Number of apps available in leading app stores as of 3rd quarter 2018, https://bit.ly/2dycQpS.

[38] N. Ayewah, W. Pugh, J.D. Morgenthaler, J. Penix, Y. Zhou, Evaluating static analysis defect warnings on production software, in: Proceedings of the 7th ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering, PASTE'07, San Diego, California, USA, June 13-14, 2007, 2007, pp. 1–8.

[39] P.M. Vu, T.T. Nguyen, H.V. Pham, T.T. Nguyen, Mining user opinions in mobile app reviews: A keyword-based approach (t), in: Proceedings of the 30th International Conference on Automated Software Engineering, ASE, IEEE, 2015, pp. 749–759.

[40] Non-skippable YouTube Ads, https://bit.ly/2VuuOKW.

[41] YouTube Will Stop Placing Unskippable 30-Second Ads in Front of Videos, https://bit.ly/2WQGWWO.

[42] SHAREit, SHAREit Giving ads in notifications, 2019, https://www.reddit.com/r/assholedesign/comments/bxs0n4/shareit_giving_ads_in_notifications/.

[43] E. Noei, F. Zhang, Y. Zou, Too many user-reviews, what should app developers look at first? IEEE Trans. Softw. Eng. (2019).

[44] W. Martin, M. Harman, Y. Jia, F. Sarro, Y. Zhang, The app sampling problem for app store mining, in: 12th IEEE/ACM Working Conference on Mining Software Repositories, MSR'15, 2015, pp. 123–133.

[45] L. Bao, Z. Xing, X. Xia, D. Lo, A.E. Hassan, Inference of development activities from interaction with uninstrumented applications, Empir. Softw. Eng. 23 (3) (2018) 1313–1351.

[46] K.Q. Weinberger, J. Blitzer, L.K. Saul, Distance metric learning for large margin nearest neighbor classification, in: Advances in Neural Information Processing Systems 18 Neural Information Processing Systems, NIPS 2005, December 5-8, 2005, Vancouver, British Columbia, Canada, 2005, pp. 1473–1480.

[47] D.W. Hosmer, S. Lemeshow, Applied Logistic Regression, second ed., Wiley, 2000.

[48] Y. Man, C. Gao, M.R. Lyu, J. Jiang, Experience report: Understanding cross-platform app issues from user reviews, in: 27th IEEE International Symposium on Software Reliability Engineering, ISSRE 2016, Ottawa, on, Canada, October 23-27, 2016, 2016, pp. 138–149.

[49] E. Chien, Techniques of adware and spyware, in: The Proceedings of the Fifteenth Virus Bulletin Conference, Dublin Ireland, Vol. 47, 2005.

[50] A. Finkelstein, M. Harman, Y. Jia, W. Martin, F. Sarro, Y. Zhang, Investigating the relationship between price, rating, and popularity in the Blackberry World App Store, Inf. Softw. Technol. 87 (2017) 119–139, http://dx.doi.org/10.1016/j.infsof.2017.03.002, URL http://www.sciencedirect.com/science/article/pii/S095058491730215X.

[51] C. Iacob, V. Veerappa, R. Harrison, What are you complaining about?: a study of online reviews of mobile applications, in: BCS-HCI '13 Proceedings of the 27th International BCS Human Computer Interaction Conference, Brunel University, London, UK, 9-13 September 2013, 2013, p. 29.

[52] C. Iacob, R. Harrison, Retrieving and analyzing mobile apps feature requests from online reviews, in: Proceedings of the 10th Working Conference on Mining Software Repositories, MSR '13, San Francisco, CA, USA, May 18-19, 2013, 2013, pp. 41–44.

[53] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a Meeting Held December 5-8, 2013, Lake Tahoe, Nevada, United States, 2013, pp. 3111–3119.

[54] P.M. Vu, H.V. Pham, T.T. Nguyen, et al., Phrase-based extraction of user opinions in mobile app reviews, in: Proceedings of the 31st International Conference on Automated Software Engineering, ASE, ACM, 2016, pp. 726–731.

[55] A. Di Sorbo, S. Panichella, C.V. Alexandru, J. Shimagaki, C.A. Visaggio, G. Canfora, H.C. Gall, What would users change in my app? summarizing app reviews for recommending software changes, in: Proceedings of the 24th SIGSOFT International Symposium on Foundations of Software Engineering, FSE, ACM, 2016, pp. 499–510.

[56] L. Villarroel, G. Bavota, B. Russo, R. Oliveto, M.D. Penta, Release planning of mobile apps based on user reviews, in: Proceedings of the 38th International Conference on Software Engineering, ICSE 2016, Austin, TX, USA, May 14-22, 2016, 2016, pp. 14–24.

[57] W. Maalej, H. Nabil, Bug report, feature request, or simply praise? On automatically classifying app reviews, in: 23rd IEEE International Requirements Engineering Conference, RE 2015, Ottawa, on, Canada, August 24-28, 2015, 2015, pp. 116–125.

[58] E. Guzman, W. Maalej, How do users like this feature? a fine grained sentiment analysis of app reviews, in: Proceedings of the 22nd International Conference on Requirements Engineering, RE, IEEE, 2014, pp. 153–162.

[59] X. Gu, S. Kim, "What parts of your apps are loved by users?" (T), in: 30th IEEE/ACM International Conference on Automated Software Engineering, ASE 2015, Lincoln, NE, USA, November 9–13, 2015, 2015, pp. 760–770.

[60] W. Luiz, F. Viegas, R.O. de Alencar, F.M. ao, T. Salles, D.B. Carvalho, M.A. Gonçalves, L.C. da Rocha, A feature-oriented sentiment rating for mobile app reviews, in: Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018, 2018, pp. 1909–1918.

[61] C. Gao, B. Wang, P. He, J. Zhu, Y. Zhou, M.R. Lyu, PAID: Prioritizing app issues for developers by tracking user reviews over versions, in: Proceedings of the 26th International Symposium on Software Reliability Engineering, ISSRE, IEEE, 2015.

[62] C. Gao, J. Zeng, I. King, M. Lyu, Online app review analysis for identifying emerging issues, in: Proceedings of the 40th International Conference on Software Engineering, ICSE, ACM, 2018.

[63] F. Palomba, M.L. Vásquez, G. Bavota, R. Oliveto, M.D. Penta, D. Poshyvanyk, A.D. Lucia, Crowdsourcing user reviews to support the evolution of mobile apps, J. Syst. Softw. 137 (2018) 143–162.

[64] X. Lu, X. Liu, H. Li, T. Xie, Q. Mei, D. Hao, G. Huang, F. Feng, PRADA: prioritizing android devices for apps by mining large-scale usage data, in: Proceedings of the 38th International Conference on Software Engineering, ICSE 2016, Austin, TX, USA, May 14-22, 2016, 2016, pp. 3–13.

[65] S. Panichella, A.D. Sorbo, E. Guzman, C.A. Visaggio, G. Canfora, H.C. Gall, How can i improve my app? Classifying user reviews for software maintenance and evolution, in: R. Koschke, J. Krinke, M.P. Robillard (Eds.), 2015 IEEE International Conference on Software Maintenance and Evolution, ICSME 2015, Bremen, Germany, September 29 - October 1, 2015, IEEE Computer Society, 2015, pp. 281–290.

[66] M. Lu, P. Liang, Automatic classification of non-functional requirements from augmented app user reviews, in: E. Mendes, S. Counsell, K. Petersen (Eds.), Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering, EASE 2017, Karlskrona, Sweden, June 15-16, 2017, ACM, 2017, pp. 344–353.

[67] W. Martin, F. Sarro, Y. Jia, Y. Zhang, M. Harman, A survey of app store analysis for software engineering, IEEE Trans. Software Eng. 43 (9) (2017) 817–847.

[68] H. Khalid, E. Shihab, M. Nagappan, A.E. Hassan, What do mobile app users complain about? IEEE Softw. 32 (3) (2015) 70–77.

[69] X. Wei, L. Gomez, I. Neamtiu, M. Faloutsos, ProfileDroid: multi-layer profiling of android applications, in: Proceedings of the 18th International Conference on Mobile Computing and Networking, MobiCom, ACM, 2012, pp. 137–148.

[70] I. Ullah, R. Boreli, M.A. Kâafar, S.S. Kanhere, Characterising user targeting for in-App Mobile Ads, in: 2014 Proceedings IEEE INFOCOM Workshops, Toronto, on, Canada, April 27 - May 2, 2014, 2014, pp. 547–552.

[71] N. Vallina-Rodriguez, J. Shah, A. Finamore, Y. Grunenberger, K. Papagiannaki, H. Haddadi, J. Crowcroft, Breaking for commercials: characterizing mobile advertising, in: Proceedings of Conference on Internet Measurement Conference, IMC, ACM, 2012, pp. 343–356.

[72] J. Gui, D. Li, M. Wan, W.G. Halfond, Lightweight measurement and estimation of mobile ad energy consumption, in: Proceedings of the 5th International Workshop on Green and Sustainable Software, GREENS@ICSE 2016, Austin, Texas, USA, May 16, 2016, 2016, pp. 1–7.

[73] C. Gao, J. Zeng, F. Sarro, M. Lyu, I. King, Exploring the effects of ad schemes on the performance cost of mobile phones, in: Proc. of the International Workshop on Advances in Mobile App Analysis, a-Mobile, 2018.

[74] J. Gui, M. Nagappan, W.G. Halfond, What aspects of mobile ads do users care about? an empirical study of mobile in-app ad reviews, 2017, arXiv:1702.07681.