

Application of Developers' and Users' Dependent Factors in App Store Optimization

<https://doi.org/10.3991/ijim.v14i13.14143>

Artur Strzelecki

University of Economics in Katowice, Katowice, Poland

artur.strzelecki@ue.katowice.pl

Abstract—This paper presents an application of developers' and users' dependent factors in the app store optimization. The application is based on two main fields: developers' dependent factors and users' dependent factors. Developers' dependent factors are identified as: developer name, app name, subtitle, genre, short description, long description, content rating, system requirements, page url, last update, what's new, and price. Users' dependent factors are identified as: download volume, average rating, rating volume, and reviews. The proposed application in its final form is modeled after mining sample data from two leading app stores: Google Play and Apple App Store. Results from analyzing collected data show that developer dependent elements can be better optimized. Names and descriptions of mobile apps are not fully utilized. In Google Play there is one significant correlation between download volume and number of reviews, whereas in the App Store there is no significant correlation between factors.

Keywords—App store optimization, Google Play, Apple App Store, mobile app store, ASO.

1 Introduction

Together with the increasing use of mobile devices there is an increased supply of applications (apps) used on mobile devices. Software companies and developers create apps that installed on a mobile device can be useful for mobile device owners. Now, in the two most popular app stores (Apple App Store and Google Play) are available millions of apps. Apps are offered in these stores in different categories of software, however, still, competition is very high between apps in one category. Competition means that many apps offer similar functions and users can browse and choose from a variety of selections.

This creates a need to propose an application of developers' and users' dependent factors in app store optimization. App store optimization also can be described as app store marketing or mobile app search engine optimization. The focus is to improve the ranking of mobile app directly within the app store. Recently [1] noticed that developers make efforts to improve mobile app visibility in the app store, but some of such actions can be treated as fraud. Especially when it comes to reviews and downloads, they can be pumped by the developer.

There is limited research on factors which are taken into account when apps ranking is created. Apps are created by developers and used by users; thus ranking is influenced by both of them. App store optimization has its roots in search engine optimization, where different factors are taken into account, when it comes to create a ranking of websites. Earning from achievements in the field of search engine optimization, the author proposes an application for app store optimization. There are identified similarities between these two areas.

The first attempt at the systematic literature review of app store optimization is done recently [2]. Yet, its coverage is not sufficient, because it covers only 9 published works. 6 papers are conference proceedings and 3 papers are published in journals but only one is indexed in Web of Science. In referenced works, there are mentioned different factors, which are key elements for the proposition in this paper. Analyzing how the app store is modeled and configured, some factors are considered as basic elements, like the number of downloads, the name of the app, or the name of the developer. Others were introduced later. Yet, not all visible elements are considered as elements of this application, e.g., app icon, app screenshot, or videos are not considered as direct elements for app store optimization. However, in literature, the icon is treated as a central design element of an app [3] and screenshots can be considered as a representation of the aesthetic and user-friendliness of the mobile app [3].

The research question is which factors are used for creating a ranking of mobile apps directly in the app store. In this paper, the author proposes an application of developers' and users' dependent factors in app store optimization. This paper is organized as follows. Section 2 contains a background for the study. In section 3 the author describes the research method for choosing factors to propose framework. Section 4 contains characteristics of the sample data which are collected from the Apple App Store and Google Play. In section 5 author discuss the results, highlight the contribution of the research and suggest possible implications of results, analyze current limitations of the research, draw conclusions and present ideas for future research on app store optimization.

2 Literature Review

In this section two fields are explored. First is about efforts made by app developers to promote apps in the app store. The developer creates an app and delivers it to the app store. After software is delivered, it can be offered with free app offers, continuous quality updates, investment in less popular (less competitive) categories, and price changes [4]. New releases are found to change user opinion on the app [5]. Number of ratings, number of reviews, number of downloads are always positively changed, since they usually are rising. Some developers update their apps very frequently, even once a week or twice a month. They are not too concerned about detailing the content of new updates and users are not too concerned about such information, whereas users highly rank frequently updated apps instead of being annoyed about the high update frequency [6]. App rating is assigned to an app over its entire lifetime. The app rating is aggregated into one rating that is displayed in the app store. However, many apps do increase their

version-to-version rating, while the store-rating of an app is resilient to fluctuations once an app has gathered a substantial number of raters [7].

There were introduced simulation models of the mobile ecosystem. In the model developed by [8], [9] developers build and upload apps to the app store and the users browse the store. Ranking of apps is built on four algorithms based on the number of downloads. In the model developed by [10], three more additional algorithms are taking into account the number of downloads and time. According to [11] high-rated apps are larger with more complex code, more preconditions, more marketing efforts, more dependence on libraries, and they make use of higher-quality Android libraries. A high review rating of the free version of a mobile app leads to higher sales of its paid version [12]. The proposed approach is to assist developers to select the proper release opportunity based on the purpose of the update and current condition of the app [13]. In a recent study, names, descriptions were analyzed in terms of influencing ranking position in the store [14]. Developers also can do a shady move to fraud app ranking. Another proposition defined two methods of fraud: inflate the app number downloads and ratings in a very short time [1]. Especially, when it comes to download fraud activities, they can be divided into three types according to their intentions: boosting front end downloads, optimizing app search ranking, and enhancing user acquisition and retention rate [15].

The second field is on users' reviews. Users can publish a review about an app. Reviews have a major influence on the user's purchase decision [3]. Average rating according to the star principle as well as the number of reviews given, determine the buying decision of an app to a very large degree. Reviews area is explored when it comes to see what the sentiment of the review is. Review can have positive or negative sentiment [16]. Reviews analysis is done by text mining [17]. For many apps, the amount of reviews is too large to be processed manually and their quality varies largely. There are proposed automated approach that helps developers filter, aggregate, and analyze user reviews [18]. Most of the feedback is provided shortly after new releases, with a quickly decreasing frequency over time. Reviews typically contain multiple topics, such as user experience, bug reports, and feature requests [19]. Reviews are a source for users' feedback, requests for new features, or reporting bugs. Reviews represent feature requests, i.e. comments through which users either suggest new features for an app or express preferences for the re-design of already existing features of an app [20]. E.g. in the Blackberry app store there is a strong correlation between customer rating and the rank of app downloads yet, surprisingly, there is no correlation either between price and downloads, or between price and rating [21].

Reviews can be specific, as specific is an app [22]. Analyzing feedbacks from a health and fitness-tracking app shows that the users of health and fitness-related apps are concerned about their physical activity records and physiological records. The records include track, distance, time, and calories burned during jogging or walking. App store reviews are used to analyze different aspects of app development and evolution [23]. There are proposed frameworks to acquire reviews in large number, extract informative user reviews by filtering noisy and irrelevant ones, then group the informative reviews automatically using topic modeling [24], [25]. There are recommendation systems aimed at helping users to choose optimal sets of applications belonging to different

categories while minimizing energy consumption, transmitted data, and maximizing application rating [26] or optimize performance [27]. There are also systematic literature reviews of opinion mining studies from mobile app store user reviews, which describes and compares the areas of research that have been explored thus far, drawing out common aspects in app store analysis [28], [29]. Some studies are on both fields, without distinguishing them into developers and user areas [30].

eWOM (Electronic Word of Mouth) is a field that touches the area of the app store. There are works done to measure the impact of eWOM attributes to the success of the app. eWOM plays an important role in mobile application downloads [31]. In the app store, eWOM can be considered as reviews posted by users [32]. Analyzing dimension of the reviews, comments on product quality and service quality affects app downloads in different levels. Users' comments on product quality occupies a larger portion of consumer reviews, but their comments on service quality have a stronger effect on sales rankings [33].

3 Materials and Methods

App stores distribute apps through the app store and have additional information about the app. A set of information is initially set by the developer. The app is delivered by the app developer. The developer sets a name for the app and creates a description of app features. The app is published with the new original url. The developer sets genre, content rating, and define system requirements for an app. Apps can be distributed through the app store for free or user needs to pay for download. Free apps can also offer in-app purchases for additional features. Another set of information is created after the app is released. Users are downloading the app and make the number of downloads growing. Users can also write reviews and rate an app in a range of scale from 1 to 5.

Proposition on figure 1 is based on two areas. The first area depends on the developer. Its content and settings are provided when the app is initially released in the app store. The second area depends on the users. If the app is being popular among users, they are starting to download, rate, and create reviews of the app in the app store.

3.1 Factors dependent on developer

The developer decides on twelve factors that are proposed in figure 1. Clearly, it shows that much of the app store optimization depends on the developer's starting set for an app. Following is a description of each element in the developer's dependent area.

Developer: The name of the developer is a ranking factor, affecting directly the position of the application itself. The positive history of the developer affects the better evaluation of the application in the search ranking. Keywords in the developer's name affects each of its applications.

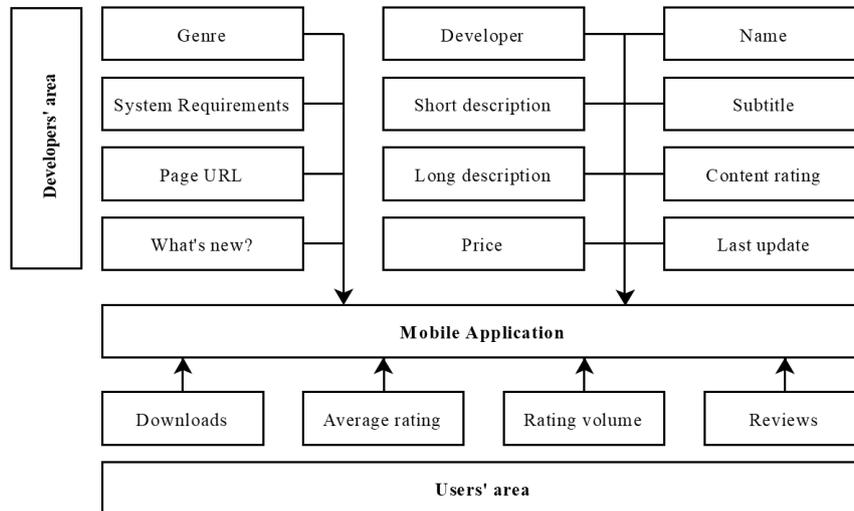


Fig. 1. Categorization of attributes in the app store. Developers' area contains content and settings provided when the app is initially released in the app store. Users' area contains elements revealing users' engagement.

Name: The name of the application is important both for the app store optimization and for the experience. Often observed solution made by developers is to create a name by combining the brand name and the most important keywords for the app. Google Play limits the name to 50 characters, whereas Apple App Store limits the name to 30 characters.

Subtitle: (Only in Apple App Store) Subtitle is placed right below the title and brings additional information on the mobile app. It complements the app name by communicating the purpose and value of the app in detail and is limited to 30 characters.

Genre: Genre is a category for software. Currently Google Play offers 31 categories for mobile apps and Apple App Store offers 28 genres. In Apple App Store, apps sometimes are assigned to more than one genre. The app can have a primary genre and secondary genre. In Google Play app has a primary category and secondary family filter.

Short description: App stores allows to prepare two types of description, short and long version. The short version is only visible in the mobile app store and its maximum length is 80 characters for Google Play and 170 characters for Apple App Store.

Long description: The long version is also visible in a desktop version and the maximum size is 4000 characters. However, only around 250 characters are visible after the description is displayed, the rest is hidden. It can be shown after clicking the button "Read more".

Content rating: Content rating rates the suitability of mobile applications for its audience. It tells what age group is suitable to use a mobile app. Content rating differs in different geographic.

System requirements: App store providers require that new apps target operating system version. The system version is set in requirements. For new apps requirement

in Google Play is at least Android 8.0 and in Apple App Store is at least iOS 12. Older apps can have lower operating system requirements.

Page url: The url can only be defined when the application is published, hence it is an element worth refining. Google Play url uses the following format: *https://play.google.com/store/apps/details?id=<package_name>*, Apple App Store url uses the following format: *http://itunes.apple.com/<country>/app/<app-name>/id<app-ID>?mt=8*.

Last update: (Only in Google Play) It is the date of the last update. For the application store, security is important, therefore the factor that has a ranking importance, but also affects the opinions about the application, is the frequency of its updating. It is necessary to update the application not only at the time of major changes to the functionality of the app, but also at every subsequent update of the operating system.

What's new: This element is intended to describe the updates introduced into the app. Developers can change the contents of the section only after the new app version is submitted to the store. Since April 2019, in Apple App Store developers are only able to edit the "What's new" text when submitting a new version of their app. Any changes to the text are subject to the standard review process. It's possible that previously a few developers were taking advantage of the flexibility for bad reasons, such as pointing support or marketing URLs to malicious websites.

Price: The developer sets a price tier, and then the app store automatically defines the price in local currency. The developer has no control over the price in each country. For Apple App Store there is a price matrix with 87 tiers. Tier 0 is priced equal to 0, so it means the app is for free. Maximum tier 87 is the price set to 1199.99 USD. For Google Play the maximum price for an app is 400 USD. Developers often use freemium app strategy [12], where the app download is for free, but the app offers in-app purchases.

3.2 Factors dependent on users

Users influence four factors in the figure 1. It is fewer elements than are depending on the developer, however, high ratings and a high volume of downloads, together with good reviews make the app popular in the app store. According to Google [34], apps in Google Play are ranked based on a combination of ratings, reviews, downloads, and other factors, however, details of these weights and values are a proprietary part of the Google search algorithm. Below is a brief description of each element in the user's dependent area.

Downloads volume: (Only in Google Play) Downloads is the number of app installation on devices worldwide. Google Play publishes the only threshold number that has been reached. Threshold always has 1 or 5 in the beginning and is rising by adding zeros. Currently, the upper threshold of 5 billion downloads is reached by only one app: YouTube.

Average rating: Average rating is a number with one decimal place in the scale from 1 to 5 and it is an aggregated rating from all ratings given by users. Total rating is only available for apps that have received five or more ratings. The way app's rating in Google Play is calculated has been updated in May 2019 to place greater significance

on the app's most recent ratings. In Google Play summary of app's ratings is being shown to users on starting in August 2019.

Ratings volume: Number of all ratings given by users. Individual ratings influence the app's summary rating, which is displayed on the product page and in search results. This summary rating is specific to each territory on the Apple App Store and it can be reset when the developer releases a new version of the app. Resetting the summary rating can ensure that it reflects the most current version of the app, however, having few ratings may discourage potential users from downloading the app. Resetting summary rating does not reset the app's written reviews. Past reviews will continue to display on the product page.

Reviews: Apart from ratings users can write a review about an app. Reviews are a source for users' feedback, requests for new features, or reporting bugs [35]. Developers can ask users to rate and review the app at appropriate times throughout the user experience. In Apple App Store developer can prompt for ratings up to three times in 365 days. Users will submit a rating through the standardized prompt, and can write and submit a review without leaving the app.

Data for analysis and further usage in application for app store optimization were collected from Google Play and Apple App Store. The author used two different techniques for retrieving data. The first technique applied regular expressions to match elements like the number of downloads, last date of update, content rating, range of pricing, and software requirements. The second technique used CSSPath to match the next elements like developer, number of reviews, category, name, average rating. The url was also retrieved for each application during the crawling process.

These two different techniques were needed because some of the elements are written down firmly into a website structure and are always placed in the same context. These were retrieved by the CSSPath. The rest of the elements can change their position in website structure, due to the incomplete data provided by the developer. Some of the apps in Google Play and Apple App Store do not have all of the information usually displayed on the app store. Regular expressions helped to collect the data in different parts of websites.

4 Results

The mining of apps was done in May 2019. The samples contain 49,990 apps from Google Play and 6,040 apps from Apple App Store. The method for collecting the data was crawling web pages from the starting point. For Google Play, it was the start page for the app store. The crawler was limited only to crawl web pages with a detailed app description. The starting point for Google Play was <https://play.google.com/store/apps> and detailed web pages for the app have common part <https://play.google.com/store/apps/details?id=> where after = character is a unique part of url for each app. Web pages with detailed app descriptions were discovered by crawling app urls from the start page and then each detailed page url has a section with similar apps named Similar and section More by developer name. After reaching 50,000 downloaded apps with this method, the author decided that this sample is representative. Now, the dataset contains

apps for every category in Google Play, most of the apps have ratings, all download threshold, and all types of content ratings are in the dataset. In table 1 is a comparison of both app stores.

Table 1. Comparison of Google Play and Apple App Store

App Store	Google Play	Apple App Store
Maximum length of name	50	30
Number of developers	724,000	494,000
Maximum length of subtitle	none	30
Number of categories/genres	31	28
Maximum length of short description	80	170
Maximum length of long description	4000	4000
Content rating types	ESRB, PEGI, USK, ACB, Brazil, GRAC	4+, 9+, 12+, 17+, GRAC
System requirements	Android	iOS
Page url	Structured	Structured
Last update	Date	None
What's new	After new release	After new release
Maximum price	1199.99USD	400USD
Downloads volume	Thresholds	none
Average rating	1 to 5	1 to 5
Ratings volume	Yes	Yes
Reviews	Yes	Yes

Source: App stores mining and [36]

Apple App Store does not list any link to detailed web pages for apps through the web browser on the start page. However, previously mentioned structured, that detailed web page for app contains sections with links to other apps, also appear in Apple App Store as sections: *More by this developer* and second section *You may also like*. For Apple App Store the starting page for crawler was <https://apps.apple.com/pl/app/wikipedia/id324715238>. The crawler was only crawling web pages that matched this criterion: https://itunes.apple.com/pl/app/*mt=8. * refers to regular expression and allows to be any string of any characters replacing “.*”. In Apple App Store, the number of downloaded apps is significantly lower. Unfortunately, app store blocks connection after reaching around 200 downloaded apps and bans the IP address from which crawler was connecting. After around 30 attempts of switching IP, the author was able to download 6,040 apps, which were in the distance of 6 nodes (crawl depth) from the starting point.

In the process of app store mining, the author has crawled from Google Play 50 K apps, which belong to 31 different categories. Additionally, there is the main category Games which includes 17 subcategories (action, adventure, arcade, board, card, casino, casual, educational, music, puzzle, racing, role-playing, simulation, sports, strategy, trivia, word). In the crawled sample games take 16,66% of the total number of applications, and other apps belong to 31 different categories. Figure 2 represents the ratio of all categories in this sample.

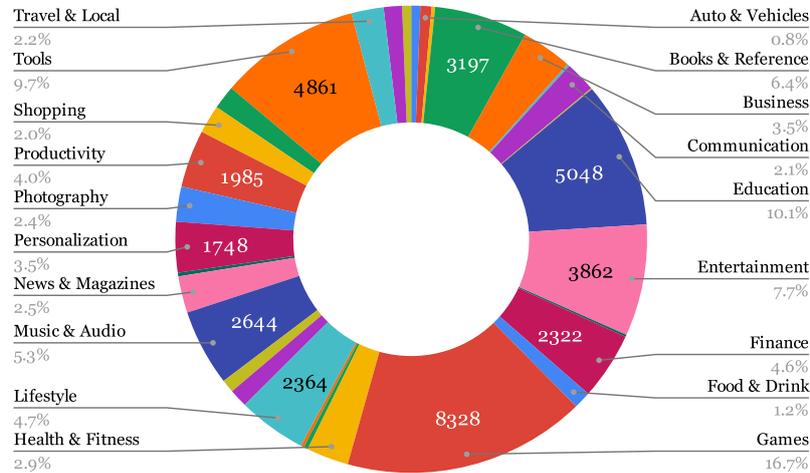


Fig. 2. App categories in the Google Play store. The sample shows 31 different categories and merged 17 categories into the Games. The sample contains 49990 crawled apps.

Table 2 shows the descriptive statistics of 49 990 sample apps crawled from Google Play and 6040 sample apps crawled from Apple App Store. Based on extracted data Table 2 contains the main characteristics of factors dependent on developers based on Google Play and Apple App Store.

Table 2. Descriptive statistics of samples from Google Play and Apple App Store

App Store	Google Play	Apple App Store
Number of downloaded apps	49990	6040
Number of developers	16912	2786
Number of genres	31	25
Avg length of name	23	23
Avg length of subtitle	na	26
Avg length of description	1338	1694
Median length of description	1052	1489
Apps with reviews	97%	79%
Average number of avg rating	4,29	4,35

Autor crawled a smaller sample from Apple App Store because this app store prevents against the massive extraction of data and after a few hundreds of crawled apps, it turns on the HTTP response code 403 on every web page with an app. Table 3 contains factors dependent on users based on Google Play like the number of downloads, the average number of reviews in download threshold, and average from average rating in download threshold.

Table 3. Framework elements dependent on users based on Google Play

Download threshold	Number of apps	Percentage of dataset	Avg of reviews	Avg from avg rating
0	67	0,13%	21	4,35
1	217	0,43%	2	4,90
5	158	0,32%	61	4,76
10	767	1,53%	4	4,69
50	542	1,08%	4	4,57
100	2324	4,65%	8	4,43
500	1517	3,03%	15	4,36
1000	5025	10,05%	37	4,25
5000	2933	5,87%	86	4,22
10000	8502	17,01%	283	4,26
50000	4392	8,79%	785	4,27
100000	10016	20,04%	2757	4,27
500000	3753	7,51%	9020	4,29
1000000	5833	11,67%	31539	4,31
5000000	1520	3,04%	102576	4,32
10000000	1813	3,63%	343902	4,34
50000000	322	0,64%	1113916	4,37
100000000	244	0,49%	3541721	4,39
500000000	27	0,05%	9739412	4,41
1000000000	17	0,03%	26792154	4,27
5000000000	1	0,00%	39797335	4,40

5 Discussion

Data in the Google Play store adapts to language settings of the user's browser and operating system. During the initial screening, the author has seen different settings coming from different languages. First, the price of the app itself or in-app purchases were displayed in the currency related to the language set in the web browser. Second, the types of content ratings were different for different localization settings. The author checked three options of language & country setting: US, Poland, and Ukraine. For the US the rating is displayed in the Entertainment Software Rating Board (ESRB) standard, whereas for Poland and Ukraine the rating format comes from Pan European Game Information (PEGI) standard. [37]. These ratings were encountered during sample crawling. Other countries have their ratings. Ratings in Germany are maintained by Unterhaltungssoftware Selbstkontrolle (USK). Ratings in Australia are maintained by the Australian Classification Board (ACB). Ratings in Brazil are maintained by Class-Ind. The content rating system in South Korea is approved by GRAC (Game Rating and Administration Committee). Third, except for recommending applications based on language and country, Google Play provides different search results for logged-in and anonymous users. If the user is not authorized into Google account, it is suggested a set of applications that differs from the one an authorized user will get (yet, the query in both situations could be the same).

Crawled data on factors depending on developers reveal, that they do not use fully all of the framework elements. The name of the app can be a maximum of 30 characters long, however average name length is 23 characters. This suggests, that there is still space that can be used by developers in the name element. Subtitle is an optional element in Apple App Store. Not every app has set subtitle. In the collected sample, 41% of apps did not have a subtitle. Subtitle can be a maximum of 30 characters long and the average length is 26 characters. This suggests, that if the subtitle is used, developers use it in its almost maximum capacity. Long description can be a maximum of 4000 characters long, however, the average length is 1338 characters and the median 1056 characters long for Google Play and the average length of 1755 characters and median 1534 characters for Apple App Store. This suggests, that there is still space that can be used by developers in the description element.

Limits of name length have been changing in recent years. The previous limit in Google Play was 30 characters. In 2017 app name length in Google Play increased to 50 characters. A different approach was in Apple App Store. Introducing the App Store, the limit was 255 characters. In 2016 Apple introduced the limit of 50 characters and one year later decreased it to 30 characters. It can be noticed that in the sample are apps from Apple App Store with name lengths from previous limits.

Factors depending on users reveal that, if the larger download threshold is achieved, the larger number of average reviews apps have. It is a linear relationship, which is expected, since more downloads can result in more reviews. However, the average rating for apps with the lowest download threshold is the highest and then for the next thresholds is decreasing. The average rating is decreasing to 4,22 with a threshold of 5000 downloads and then is increasing to a threshold of 500 million downloads. Users' depending factors are built on data divided according to the download threshold. It shows, that if the app is more and more downloaded, the overall rating from the user is growing. Usually developers are taking into account requests made in reviews and update apps with new and requested features. That is why the users are rating the app higher.

Average ratings for apps differ between both app stores. The collected sample shows that higher averages are in Apple App Store. In Apple App Store there is no significant Spearman's correlation coefficient between Length of name, Length of Description, Number of Reviews, and Average Rating. Table 4 shows the Spearman's correlation coefficient matrix.

In Google Play there is only one significant Spearman's correlation coefficient between Downloads volume and Number of reviews, $p < 0,001$. Other metrics are having a weak correlation. Table 5 shows the correlation matrix.

Table 4. Spearman's correlation coefficient values for Apple App Store elements

	Name	Description	Reviews	Average rating
Name				
Description	0,056			
Reviews	0,002	0,032		
Average rating	-0,039	0,095	0,029	

Table 5. Spearman's correlation coefficient values for Google Play elements

	Name	Downloads	Reviews	Average rating
Name				
Downloads	-0,007			
Reviews	0,001	0,600		
Average rating	0,153	0,009	0,018	

Using this study developers could encourage to use the possibility to ask the user for the ratings and reviews. Both app stores limit this function to a few times a year, but results show, that there is a coefficient of 0,600, $p < 0,001$ between the number of reviews and downloads. On the other side, users using reviews can request new features or do bug reports. This finding is confirmed along with other studies [19], [20].

In two downloaded samples, 400 apps are the same in both app stores. The metric comparison of the same apps in both stores shows the correlation for average rating as 0,437 and description length correlation is 0,725. There is a moderate relationship between Average ratings in both app stores and a strong relationship of description lengths. Google Play users tend to rate apps more than Apple App Store users which is in line with [38] results.

In this paper an application of developers' and users' dependent factors in app store optimization is proposed. The application is based on two main fields. The first field is dependent on the developer. The developer sets the initial setting for each app, which is distributed to the app store. After the app is being distributed, users' engagement in app distribution is the second field. Users cause number of downloads, write reviews and give ratings. An application is proposed for both leading app stores: Google Play and Apple App Store. Answering the research question stated in the paper, sixteen different factors were identified as these, which are influencing app store ranking and are key elements in app store optimization.

Results show that developers are not fully using elements that are dependent on them. The lowest use is with description, in Google Play, developers on average use 33% of its capacity, and in Apple App Store developers on average use 41% of its capacity. The second element which still has some space to use is the name of an app. In both app stores, on average, developers use 76% of its full capacity. The last observed changes in adapting the length of name show that app stores would like to display only the brand name or generic name and just one keyword. Even though, the limit in Apple App Store is 30 characters, a guide for developers suggests using a maximum of 23 characters. Yet, some developers use all of the characters space to use more keywords, which could be used to find the app in the app store. In terms of description, users usually read few first lines, not paying attention to all the text, that's why, developers prepare just the length of text, that has a chance to be read.

Recent changes in Apple App Store are on extending control over apps description. Developers are only able to edit the "What's New" text when submitting a new version of their app. The app store is tightening the pricing policy, it means that the user has to know if the app is offering in-app purchases to have a full working app. There are also published constant updates to review guidelines. Recent changes in Google Play store concern updating the app by users. Apps can have features installed in the background

depending on if they are needed. This will decrease install time and also download size, together with features that can be delivered conditionally to users depending on the country, device features. Users are allowed to update an app without ever closing it. This comes with two options, a mandatory app update, which locks the app until the user updates it, and an optional app update, which allows the user to continue to use the app while updating in the background. Google Play offers more customization options to allow app owners to have more control over their app listings. App owners can customize their app listing depending on if a user has not yet installed the app, if they have the app installed, or if they have uninstalled the app. Personalized content helps with the acquisition, retention, and re-engagement of users.

Mining the app stores with a crawling technique allowed us to check if the elements in the application are consistent. It happens, that some elements that are depending on the developer, are not set, thus not visible on the web page. App store mining also allowed to compare if apps belonging to different categories, are having different requirements to be set for the launch. After crawling, all the apps fit the same set of factors. This proved, that all apps, regardless of belonging to different categories, have the same set of factors. The final concept of the application was formed after data mining was done. Crawling also allowed to conclude some results, based on downloaded data.

6 Conclusion

This paper is one of the first attempts to create an application, which will explain, what elements are taken into account, when the ranking is created in app stores. Framework is created on data downloaded from two leading app stores: Google Play and Apple App Store. The framework assumes that this ranking is depending on sixteen elements factors. All of these factors are identified in this paper. The author divided them into two groups, where one is depending on app developers and the second is depending on the users' engagement.

The practical implication of this paper is that there is a lack of broad knowledge for developers on how to prepare marketing actions in the app store. Developers know how to build an app, but the marketing strategy is not used sufficiently. Developers now can pay attention to all elements, that influence ranking position in the app store.

The proposed application has some limitations. First is that the application is proposed only on data that is publicly visible and accessible. Perhaps there are some other elements, not visible for users. This could be the number of app uninstalls from the device, the number of app removals from the app store. Second, it takes only data from two app stores. There are other app stores like Windows Phone or BlackBerry World, which were not taken into building this application. This could be the future direction of research, to take also data from these stores and enhance the proposed framework.

7 References

- [1] H. Zhu, H. Xiong, Y. Ge, and E. Chen, "Discovery of Ranking Fraud for Mo-bile Apps," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 1, pp. 74–87, Jan. 2015, <https://doi.org/10.1109/tkde.2014.2320733>
- [2] M. Karagkiozidou, C. Ziakis, M. Vlachopoulou, and T. Kyrkoudis, "App Store Optimization Factors for Effective Mobile App Ranking," in *Strategic Innovative Marketing and Tourism*, A. Kavoura, E. Kefallonitis, and A. Giovanis, Eds. Springer, Cham, 2019, pp. 479–486. https://doi.org/10.1007/978-3-030-12453-3_54
- [3] S. Böhm and S. Schreiber, "Mobile App Marketing: A Conjoint-based Analysis on the Importance of App Store Elements," in *CENTRIC 2014: The 7th International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services*, 2014, pp. 7–14.
- [4] G. Lee and T. S. Raghu, "Determinants of Mobile Apps' Success: Evidence from the App Store Market," *J. Manag. Inf. Syst.*, vol. 31, no. 2, pp. 133–170, Oct. 2014, <https://doi.org/10.2753/mis0742-1222310206>.
- [5] W. Martin, F. Sarro, and M. Harman, "Causal impact analysis for app re-releases in google play," in *2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering - FSE 2016*, 2016, pp. 435–446, <https://doi.org/10.1145/2950290.2950320>
- [6] S. McIlroy, N. Ali, and A. E. Hassan, "Fresh apps: an empirical study of frequently-updated mobile apps in the Google play store," *Empir. Softw. Eng.*, vol. 21, no. 3, pp. 1346–1370, Jun. 2016, <https://doi.org/10.1007/s10664-015-9388-2>.
- [7] I. J. Mojica Ruiz, M. Nagappan, B. Adams, T. Berger, S. Dienst, and A. E. Hassan, "Examining the Rating System Used in Mobile-App Stores," *IEEE Softw.*, vol. 33, no. 6, pp. 86–92, Nov. 2016, <https://doi.org/10.1109/ms.2015.56>
- [8] S. L. Lim and P. J. Bentley, "How to be a successful app developer: Lessons from the simulation of an app ecosystem," in *GECCO'12 - 14th International Conference on Genetic and Evolutionary Computation*, 2012, pp. 129–136, <https://doi.org/10.1145/2330163.2330182>
- [9] S. L. Lim and P. J. Bentley, "Investigating app store ranking algorithms using a simulation of mobile app ecosystems," in *2013 IEEE Congress on Evolutionary Computation*, 2013, pp. 2672–2679, <https://doi.org/10.1109/cec.2013.6557892>
- [10] L. Cocco, K. Mannaro, G. Concas, and M. Marchesi, "Simulation of the Best Ranking Algorithms for an App Store," in *Lecture Notes in Computer Science*, vol. 8640, I. Awan, M. Younas, X. Franch, and C. Quer, Eds. Springer, Cham, 2014, pp. 233–247. https://doi.org/10.1007/978-3-319-10359-4_19
- [11] Y. Tian, M. Nagappan, D. Lo, and A. E. Hassan, "What are the characteristics of high-rated apps? A case study on free Android Applications," in *31st International Conference on Software Maintenance and Evolution*, 2015, pp. 301–310, <https://doi.org/10.1109/icse.2015.7332476>
- [12] C. Z. Liu, Y. A. Au, and H. S. Choi, "Effects of Freemium Strategy in the Mobile App Market: An Empirical Study of Google Play," *J. Manag. Inf. Syst.*, vol. 31, no. 3, pp. 326–354, Jul. 2014, <https://doi.org/10.1080/07421222.2014.995564>
- [13] S. Shen, X. Lu, Z. Hu, and X. Liu, "Towards Release Strategy Optimization for Apps in Google Play," in *9th Asia-Pacific Symposium on Internetware Internetware'17*, 2017, pp. 1–10, <https://doi.org/10.1145/3131704.3131710>.
- [14] J. M. Padilla-Piernas, M. C. Parra-Meroño, and M. Á. Beltrán-Bueno, "The Importance of App Store Optimization (ASO) for Hospitality Applications," in *Digital and Social Media Marketing*, 2020, pp. 151–161. https://doi.org/10.1007/978-3-030-24374-6_11
- [15] Y. Dou, W. Li, Z. Liu, Z. Dong, J. Luo, and P. S. Yu, "Uncovering Download Fraud Activities in Mobile App Markets," in *2019 IEEE/ACM International Conference on*

- Advances in Social Networks Analysis and Mining (ASONAM), 2019. <https://doi.org/10.1145/3341161.3345306>
- [16] B. Fu, J. Lin, L. Li, C. Faloutsos, J. Hong, and N. Sadeh, "Why people hate your app," in 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '13, 2013, pp. 1276–1284, <https://doi.org/10.1145/2487575.2488202>
- [17] W. Martin, M. Harman, Y. Jia, F. Sarro, and Y. Zhang, "The App Sampling Problem for App Store Mining," in 2015 IEEE/ACM 12th Working Conference on Mining Software Repositories, 2015, pp. 123–133, <https://doi.org/10.1109/msr.2015.19>
- [18] E. Guzman and W. Maalej, "How Do Users Like This Feature? A Fine Grained Sentiment Analysis of App Reviews," in 2014 IEEE 22nd International Requirements Engineering Conference (RE), 2014, pp. 153–162, <https://doi.org/10.1109/re.2014.6912257>
- [19] D. Pagano and W. Maalej, "User feedback in the appstore: An empirical study," in 2013 21st IEEE International Requirements Engineering Conference (RE), 2013, pp. 125–134, <https://doi.org/10.1109/re.2013.6636712>
- [20] C. Iacob and R. Harrison, "Retrieving and analyzing mobile apps feature re-quests from online reviews," in 2013 10th Working Conference on Mining Software Repositories (MSR), 2013, pp. 41–44, <https://doi.org/10.1109/msr.2013.6624001>
- [21] M. Harman, Yue Jia, and Yuanyuan Zhang, "App store mining and analysis: MSR for app stores," in 2012 9th IEEE Working Conference on Mining Software Repositories (MSR), 2012, pp. 108–111, <https://doi.org/10.1109/msr.2012.6224306>
- [22] R. Paiva-Godinho and R. S. Contreras-Espinosa, "Online user reviews as a design strategy for global communities," *Int. J. Interact. Mob. Technol.*, vol. 13, no. 11, pp. 70–84, 2019, <https://doi.org/10.3991/ijim.v13i11.11042>
- [23] Y.-H. Lai, F.-F. Huang, and P.-Y. Chiou, "Analysis of User Feedback in The Mobile App Store Using Text Mining: A Case Study of Google Fit," in 2017 IEEE 8th International Conference on Awareness Science and Technology (ICAST), 2017, pp. 50–54. <https://doi.org/10.1109/icawst.2017.8256508>
- [24] N. Chen et al., "AR-miner: mining informative reviews for developers from mobile app marketplace," in 36th International Conference on Software Engineering - ICSE 2014, 2014, pp. 767–778, <https://doi.org/10.1145/2568225.2568263>
- [25] M. Nayebi, H. Cho, and G. Ruhe, "App store mining is not enough for app improvement," *Empir. Softw. Eng.*, vol. 23, no. 5, pp. 2764–2794, Oct. 2018, <https://doi.org/10.1007/s10664-018-9601-1>
- [26] R. Saborido, G. Beltrame, F. Khomh, E. Alba, and G. Antoniol, "Optimizing User Experience in Choosing Android Applications," in 2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER), 2016, pp. 438–448, <https://doi.org/10.1109/saner.2016.64>
- [27] R. Saborido, F. Khomh, A. Hindle, and E. Alba, "An app performance optimization advisor for mobile device app marketplaces," *Sustain. Comput. Informatics Syst.*, vol. 19, no. September 2017, pp. 29–42, 2018, <https://doi.org/10.1016/j.suscom.2018.05.008>
- [28] N. Genc-Nayebi and A. Abran, "A systematic literature review: Opinion mining studies from mobile app store user reviews," *J. Syst. Softw.*, vol. 125, pp. 207–219, Mar. 2017, <https://doi.org/10.1016/j.jss.2016.11.027>
- [29] W. Martin, F. Sarro, Y. Jia, Y. Zhang, and M. Harman, "A Survey of App Store Analysis for Software Engineering," *IEEE Trans. Softw. Eng.*, vol. 43, no. 9, pp. 817–847, Sep. 2017, <https://doi.org/10.1109/tse.2016.2630689>.
- [30] A. Mahmood, "Identifying the influence of various factor of apps on google play apps ratings," *J. Data, Inf. Manag.*, Nov. 2019, <https://doi.org/10.1007/s42488-019-00015-w>
- [31] S. Oh, H. Baek, and J. Ahn, "The effect of electronic word-of-mouth (eWOM) on mobile application downloads: an empirical investigation," *Int. J. Mob. Commun.*, vol. 13, no. 2, p. 136, 2015, <https://doi.org/10.1504/ijmc.2015.067960>

- [32] J. E. Timmerman and I. Shepherd, “Does eWOM Affect Demand for Mobile Device Applications?” *J. Mark. Dev. Compet.*, vol. 10, no. 3, pp. 9–16, Dec. 2016, doi: 10.33423/JMDC.V10I3.1829.
- [33] T.-P. Liang, X. Li, C.-T. Yang, and M. Wang, “What in Consumer Reviews Affects the Sales of Mobile Apps: A Multifacet Sentiment Analysis Approach,” *Int. J. Electron. Commer.*, vol. 20, no. 2, pp. 236–260, Dec. 2015, <https://doi.org/10.1080/10864415.2016.1087823>
- [34] “Get discovered on Google Play search,” 2019. [Online]. Available: <https://support.google.com/googleplay/android-developer/answer/4448378>. [Accessed: 06-Sep-2019].
- [35] M. Rizun and A. Strzelecki, “Knowledge Graph Development for App Store Data Modeling,” in *Information Systems Development: Information Systems Beyond 2020 (ISD2019 Proceedings)*, 2019.
- [36] “Global developers per app store 2017.” [Online]. Available: <https://www.statista.com/statistics/276437/developers-per-appstore/>. [Accessed: 03-Oct-2019].
- [37] D. Felini, “Beyond Today’s Video Game Rating Systems,” *Games Cult.*, vol. 10, no. 1, pp. 106–122, Jan. 2015, <https://doi.org/10.1177/1555412014560192>
- [38] M. Ali, M. E. Joorabchi, and A. Mesbah, “Same App, Different App Stores: A Comparative Study,” in *2017 IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILESoft)*, 2017, pp. 79–90, <https://doi.org/10.1109/mobilesoft.2017.3>

8 Author

Artur Strzelecki is with University of Economics in Katowice, Katowice, Poland (artur.strzelecki@ue.katowice.pl).

Article submitted 2020-03-09. Resubmitted 2020-04-19. Final acceptance 2020-04-20. Final version published as submitted by the authors.