The Pennsylvania State University
The Graduate School

MEASUREMENT STUDY OF USER FEEDBACK IN MOBILE APP STORES

A Thesis in
Computer Science and Engineering
by
Ke Zeng

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

December 2014

The thesis of Ke Zeng was reviewed and approved* by the following:

Sencun Zhu
Associate Professor of Computer Science and Engineering
& Information Science and Technology
Thesis Advisor

Kamesh Madduri
Assistant Professor of Computer Science and Engineering

Lee Coraor
Associate Professor of Computer Science and Engineering
Graduate Director of Computer Science and Engineering

*Signatures are on file in the Graduate School.

Abstract

User feedback is an important component for mobile app markets such as Apple App Store, because it is a crucial factor in determining popularity as well as downloads. Since positive ratings and reviews could attract more users and hence more profit, the rating system in mobile app stores has become the target of review promotion attackers. In our thesis, we consider three crucial issues about the user feedback of mobile app stores. First, we analyze characteristics of user feedback and develop statistical results. Second, we perform outlier analysis, especially of the users who give a large number of high ratings. These users would be the potential review promotion attackers. Third, we discover the main causes as to why people love or hate certain mobile apps by topic analysis. We apply our techniques to two user review datasets from Apple App Store and Amazon App Store. The results show that our techniques and analysis would be helpful to discover some characteristics of user feedback in mobile app markets.

Table of Contents

List of Figures

List of Tables

Acknowledgments

I will always feel extremely fortunate to have Professor Sencun Zhu as my advisor who guided me through my graduate studies. Prof. Zhu is knowledgeable, energetic and kind which makes him a source of inspiration and ideas. I am so grateful to Prof. Zhu's encouragement as well as his insightful directions, without which this thesis would not be possible.

I would also like to thank another member in my thesis committee, Professor Kamesh Madduri, for his valuable feedback that helped me to look at aspects of the thesis in a new light. I also want to thank lab member, Ph.D candidate Mr. Zhen Xie, for his kind assistance which helped me overcome various barriers during my thesis research exploration.

Finally, I want to thank all the good friends I made at Penn State for adding fun to my graduate life.

Dedication

I dedicate this thesis to my parents. It is their love and support which enable me to focus on my research and help tide me through several rough patches.

Chapter 1

Introduction of Mobile App Store and Market

1.1 Introduction

The mobile app market is undergoing a dramatic growth as the number of s-martphone users has grown rapidly over the past few years. In fact, smartphone ownership has reached 1.7 billion by 2014, and smartphones account for nearly half of the cellphones' market over the world. In the US market, currently about 80 percent of cellphones are smartphones nowadays. Most smartphone users like to download applications on their phones. These apps could be available from application markets such as Google Play or Apple App Store. There are more than 1 million apps in Google Play Store, while Apple App Store offered nearly 1.3 million apps by Sept. 2014. Since its launch, the average number of apps installed on an Android user's smartphone has significantly increased, from 22 in 2009, to 95 in 2014. The downloads for both markets hit about 75 billion, which makes these markets a crucial source of information for mobile economy trends.

All mobile apps markets provide various kinds of information to help users find and select applications. For Apple App Store, this information includes application description, screenshots, download rate, user rating and comment. Previous work showed that smartphone users usually determine whether a mobile app is good or not by reading the user rating and comments [1]. A potential smartphone user would be aware if the review contained negative information about an application's quality. Since users rely on feedback of apps to determine whether to select an application, it will be very helpful to discover what exactly people's concerns are if we know the characteristics of feedback for an app.

User comments in mobile app store differ largely from the comments in other online markets such as Amazon online market. First, a comment in a mobile app store normally appears to be shorter, because typing is not easy on the cellphone. Thus people do not intend to spend too much time writing a comment before submitting it. Therefore, it is not common to see some long and detailed reviews as in the Amazon online market. Second, there are many words misspelled and emoticons used in the comment, which is very common for mobile users.

Overall the majority of user comments is very informative. Users tend to write about an app's feature or functionality. For instance, the comment would have concern about whether an app works or not and what are the attractiveness or demerits. However, for some positive comments, there are just some emotion expressions such as"good" or "great". These kinds of comments are less informative. At the same time, the rating could reflect the user's overall judgement toward this app. Normally rating systems employ 1 to 5 stars to quantify the user's satisfaction. A 1-star rating means the most negative option which reflects that users like to express their strong disappointment about a certain app. While a 5-star rating means the most optimistic option which implies that users express their high satisfaction toward the app.

A number of previous research work focused on mobile app review analysis. By the time of writing this thesis, there have been a few surveys approaching the mobile app user feedback, but little work has been done to study the characteristics of the user feedback in mobile app markets.

Another rising issue of the mobile app market is security. Intuitively positive ratings and comments would very likely lead to more downloads and thus high ranking in the mobile app store, which would finally bring much more profit. Therefore, some application developers tend to hire certain people called promoters and collusion groups to promote their apps. For instance in order to promote a certain app, the collusion group would give illusive feedback such as high ratings and positive comments to promote the score of an app and thus attract more potential users to download it. Also, there are several online websites which provide app promotion services. Therefore, purifying the feedback of a mobile app store would become the interest of the app store owner. Unfortunately, up to now it is still a difficult problem to precisely capture these promotion and demotion activities or collusion groups [2]. The analysis for outlier in user feedback in mobile app stores would become a potential solution which could provide some clues to detect these review promotion attackers.

The fundamental goal of this thesis is to provide essential knowledge of the user feedback in mobile app markets. In this thesis, we focus on several key research questions. The first one is how we can discover the characteristics of the user feedback in mobile app markets. The next major concern for this thesis is outlier detection. Outlier refers to someone who does not follow the prevailing activity or rule of the members in a large group. For our research, an outlier has a special meaning: it could be some users who give a large number of reviews, an amount much higher than the average, or user comments which are very long in text. The key point for inspecting these outliers is to provide some clues to make a certain suspect list of review promotion attackers. The last key research question is whether we could determine the conveyed information of the user feedback. Then app developers can understand why users love or hate their apps or their competeing apps, and then are able to improve the quality of their apps accordingly. Also, these results could provide developers with some guidance when they design new products.

Specially, this thesis involves analysis of over 10 million user comments of mobile app store. We first identify the characteristics of a user feedback, then we design an outlier detection algorithm for mobile app feedbacks and conducted a case study of outliers of different categories. For the last part, we leverage machine learning techniques to discover the root cause of positive or negative user comments. In this way, we explore whether we can identify the key factors as to why users love or hate an app which information the developer could use to improve the quality of apps in order to attract more customers.

In the next section, we give a brief introduction to each of the three parts of this thesis outlining the key techniques we used.

1.1.1 Three Parts of this Thesis

Based on the objectives discussed above, this thesis consists of three parts. In the first part, we study the characteristics of the user feedback in Apple App Store and display statistical features. In the second part, we present the results of our outliers analysis, which exposes review promotion attackers. In the last part of this thesis, we describe our findings in viewing root causes of positive or negative comments by topic analysis.

1.1.1.1 Statistical Features of User feedback in Mobile App Store

In Chapter 2, we first describe the user feedback dataset from Apple App Store. Then we explore some patterns from this data by analyzing it from 3 different aspects: user, application, and developer. These patterns would help us to discover the trend of user feedback for the mobile app store market. Also, the results would imply that there are many abused apps as well as review promoters in current mobile app rating systems. Hence we would propose outlier detection algorithm in Chapter 3 to unveil the potential promotion attackers.

1.1.1.2 Outliers Analysis - A Way to Expose Review Promoters

In Chapter 3, we focus on the security issue of the mobile app rating system. Since positive ratings and comments would most probably lead to more downloads, by which the developer would make more profit, some app developers tend to hire collusion groups to promote their apps. Hence, we try to find some potential candidates for these promoters via detecting the outliers in the mobile app rating system. More specifically, we focus on the users who give a large number of reviews, many of them with high ratings although a great number of these rated apps are unpopular. Since ordinary users do not have enough time or inclination to write a large number of comments, it is not easy for them to download so many unpopular apps. Thus, these kinds of users would be regarded as suspicious promoters.

1.1.1.3 User Feedback Distillation by Topic Analysis

In Chapter 4, we explore root causes of comments by topic analysis. We intend to use machine learning techniques to discover why people love or hate the app.

We collect the user comment of mobile app, then apply the machine learning tool, Lattent Dirichlet Allocation (LDA) model, to analyze these user reviews. More specifically, we identify the topics which are the root causes for a user's positive or negative review toward a certain mobile app. Also, we try to identify the topic distribution for mobile apps and find the most critical issues which the user praises or criticizes about the mobile app. Further, we show which aspects the user focuses on when judging a mobile application.

1.1.2 Research Contributions

In short, this thesis contributes to measurement study for user feedback of mobile app markets research in the following ways:

- Through an analysis of over 10 million comments for Apple App Store, we discover the characteristics for user comments of mobile app store.
- We propose outlier detection algorithm to unveil outlier of user feedback in mobile app store. In this way, we propose a suspect list of review promoters in mobile app store with which a mobile app store administrator could conduct further investigation.
- We apply topic analysis for user comments to determine the topic distribution of people's concerns toward mobile apps. Further, we conclude that for apps of a certain category, what is the crucial factor that users would focus on.

Collectively, these contributions uncover the characteristics of user feedback in mobile app stores. We also provide the topic analysis for user comments, which would help the mobile app developer to identify the reasons why users complain about a certain app and to improve the quality of their apps. Meanwhile our proposed outlier detection algorithm could help administrators of mobile app stores to purify their rating system.

In the next section, we present a comprehensive summary of literature related to mobile app review analysis, as well as other relevant domains.

1.2 Related Work

As of the writing of this thesis there has been little work in measurement study of user feedback for the mobile app market.

The majority of previous research work has focused on the apps rather than the user feedback. In [3], Frank et al. collected Android apps from app stores. They intended to look at some patterns in the Android permission requests by applying boolean matrix factorization.

Topic model has been widely used to determine some meaningful topics from text [4, 5, 6]. In [5], Blei et al [5] proposed state space models on the natural parameters of the multinomial distributions that represent the topics. In [7], Hong et al studied problems of using standard topic models in microblogging environments by studying how the models can be trained on the dataset. In [8], Fu et al proposed a system, which could analyze tens of millions of user ratings and comments in Google Play Store. This work is closely related to our work and it inspires us to conduct topic analysis for user reviews in Apple App Store.

Several past works analyzed reviews of other kinds of marketplaces, such as markets of tangible commodity goods and movies in [9, 10, 11, 12, 13]. In [14], Hu et al proposed a set of techniques for mining and summarizing product reviews to provide a feature-based summary of a large number of customer reviews of a product sold online. In [15], Archak et al use text mining to incorporate review text in a consumer choice model by decomposing textual reviews into segments describing different product features. In [9], Joshi et al used the text of film critics reviews from several sources to predict opening weekend revenue. In [10], Chahuneau et al build predictive models of concrete external variables such as restaurant prices.

Other researchers observed reviews for marketing purposes. In [16], authors claim that since the distribution of products have a J-curve feature, we can not make prediction of future sales based on average rating of the product. In [17], authors decompose the reviews into segments and then evaluated the individual characters to predict the features consumers care about. In [18], the authors examine the review from e-commerce sites such as Amazon.com. They found that better book reviews could lead to more sales. In [19], the author found that there are some comments of Google Chrome questioning the developer's abused use for certain permissions.

There are several pieces of work on detecting spam comments [20, 21, 22, 23, 24]. These works examine how to detect collusive actions for a rating system. In [20], Jindal et al studied opinion spam in reviews and conducted outlier analysis in amazon.com. However our work differs from their work by our choosing different features when we analyze the user outlier in mobile app store.

Chapter 2

Statistical Features of User Feedback in Mobile App Store

In this chapter, we first show how the user feedback of the Apple App Store is stored, then explore some characteristics of these user feedback and related discoveries.

2.1 Data Collection

In this section, we discuss what this dataset [2] from Apple App Store looks like. The user feedback data from Apple App Store (U.S. website) and Amazon App Store (U.S. website) is saved in MySQL database. Table 2.1 describes the way these data sets are organized. There are three tables in total, AppData Table records all the key information such as app name, average rating, number of 1-5 star ratings about certain apps. Comment Table records the specific information about every comment, for instance the app name and user id, rating date, and comment. Reviewer Table record the information for every reviewer, such as how many comments they provide as well as the review date.

Table 2.2 shows the size of dataset from Apple App Store in U.S. 23,616 apps in total were randomly picked and 18,925,438 comments made by 10,328,118 reviewers on these apps were collected.

Table 2.1. Tables in MySQL database

AppData Table	**Comment Table**	**Reviewer Table**
app id(primary key)	id(primary key)	reviewer id(primary key)
app name	app id	app ids
developer name	reviwer id	reviewer ratings
average rating	date	review dates
total raters	app version	size
1star num	rating	app versions
2star num	comment	
3star num		
4star num		
5star num		

App store	Apps	Comments	Reviewers
iTunes(U.S)	23,616	18,925,438	10,328,118

Table 2.2. Apple App Store data size.

2.2 User Level Analysis

In this section, we first analyze the dataset from the view of the user level. In this part we also give some statistical analysis based on a single user. In this way, we try to determine the trend of feedback in mobile app store and some information on the outlier analysis.

2.2.1 Average Rating Number Per User

The relationship between the number of users and the number of rating is shown in figure 2.1 and figure 2.2. In figure 2.1, x-axis denotes the number of comments while

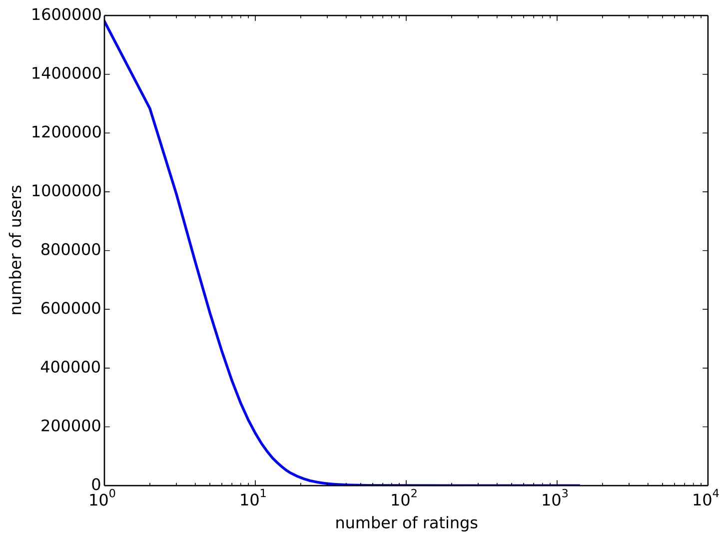

Figure 2.1. Distribution of average rating number per user (PDF).

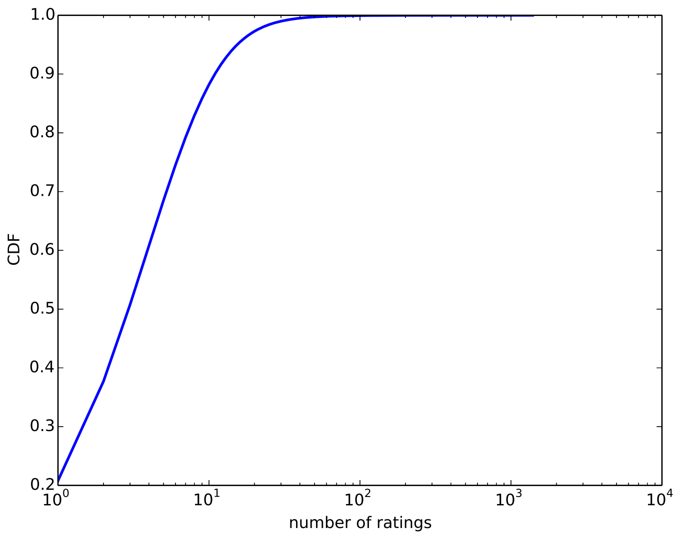

Figure 2.2. Distribution of average rating number per user (CDF).

the y-axis denotes the corresponding user number. In figure 2.2, x-axis denotes the number of comments while the y-axis denotes the corresponding accumulate probability. The average number of rating the user gives is about 5. This shows that normally people do not have much time to give comments. Also, since typing on a cellphone is much more difficult than typing on a computer, a mobile user tends to be less willing to write comments via cellphone. About 20% of users just give one comment and 90% give less than 10 comments. We also observed several users who write a large number of comments such as more than 1000, which implies that the people who write such a large number of comments are outliers. At the same time we know there are some specialist users who intend to spend much time downloading and playing with apps, and also enjoy writing comments in mobile store. However, this group is only part of these outliers. In a later part of the thesis, we try to find out some features of outliers and provide a case study for these outliers that they could also be potential member of a collusion group.

2.2.2 Average Rating Score Per User

In this section, we analyze the average rating score user gives.

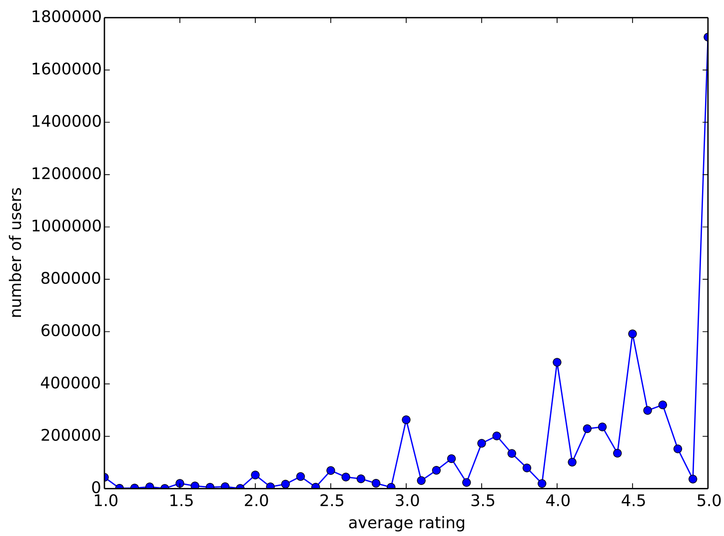

Figure 2.3. Distribution of average rating score per user (PDF).

Figure 2.3 shows the distribution of average rating score per user (PDF), in which x-axis denotes the average rating score and y-axis denotes its corresponding user number. While in figure 2.4, x-axis denotes the average rating score and y-axis denotes the corresponding accumulate probability. As shown in figures 2.3 and 2.4, the number of users who gives low average scores is small, and in fact less than 15% of users tend to give an average rating lower than 3 stars. However,

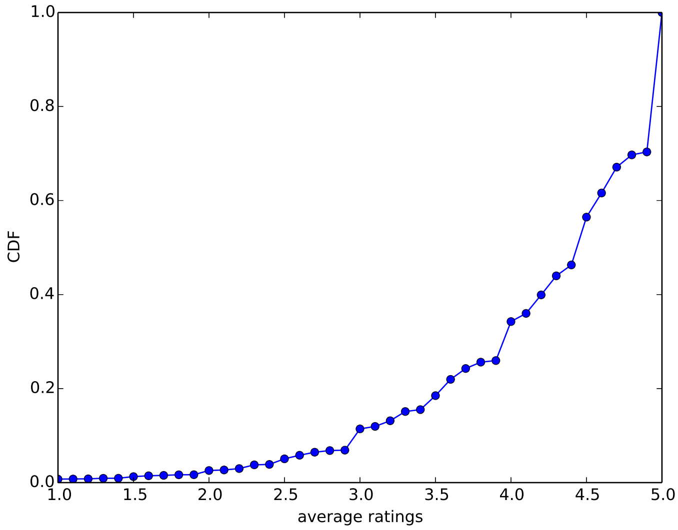

Figure 2.4. Distribution of average rating score per user (CDF).

the number of users who give high scores is incredibly high, since more than 70% of users tend to give an average rating higher than 4 stars. Some research shows that people tend to give ratings when they are satisfied with the product rather than when they are not. However, the result in mobile app store shows that the number of high rating is much larger than the number of low rating. In fact, the number of user who gives 5 stars is 3 times more than users who give 4 stars, and more than 100 times more than users who give a 1 star rating.

2.2.3 Number of Rating Categories Per User

Figure 2.5 shows distribution of number of rating category per user (PDF), in which the x-axis denotes the number of average rating category and the y-axis denotes its corresponding user number. While in figure 2.6, the x-axis denotes the number of average rating category and the y-axis denotes the corresponding accumulate probability. As shown in figures 2.5 and 2.6, the average category a user gives a rating or comment is about 3. About half of the users only give comments for 1 category while the maximum number of category a user give a rating is about 10. About 90% of users just give ratings for less than 5 categories. It shows that normally people do not give comments for apps from multiple categories. However,

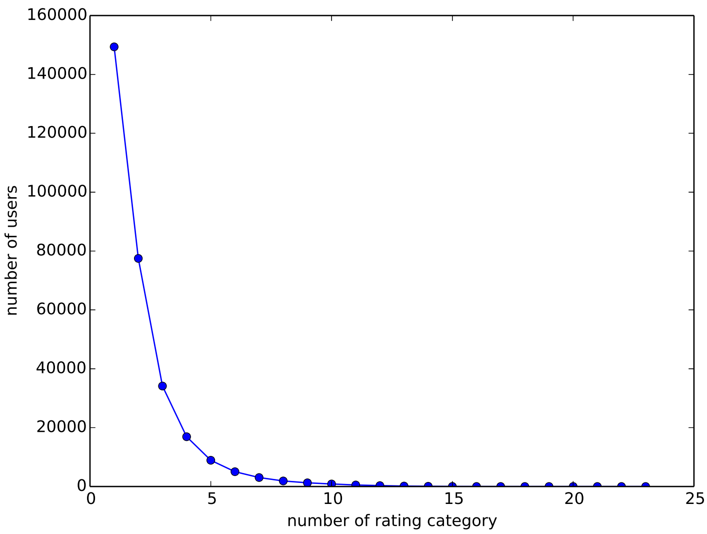

Figure 2.5. Distribution of number of rating category per user (PDF).

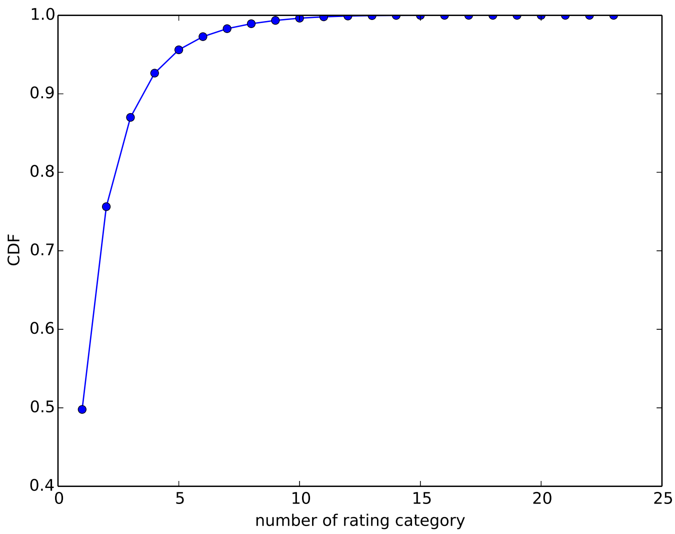

Figure 2.6. Distribution of number of rating category per user (CDF).

there are some people, such as hardcore players, who intend to download many game apps and then provide comments. However, these kinds of users normally focus on certain areas. Normally people do not have the time playing with apps of multiple categories and then giving ratings or comments. Thus, the user who gives ratings or comments for a large number of categories are outliers. And these people would be potential candidates who belong to certain collusion groups.

2.2.4 Interval of Ratings Per User

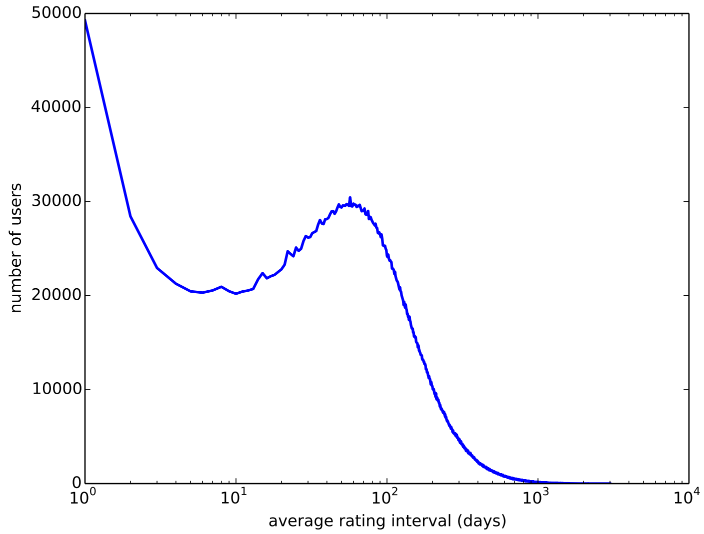

Figure 2.7. Distribution of rating interval per user (PDF).

Figure 2.7 shows the distribution of the number of rating interval (days) per user (PDF), in which x-axis denotes the number of average rating interval (by days) and y-axis denotes its corresponding user number. While in figure 2.8, x-axis denotes the number of average rating interval (by days) and y-axis denotes the corresponding accumulate probability. As shown in figures 2.7 and 2.8, the average interval of rating per user is about 164 days. However, there is a small number of users who provides ratings and comments very frequently. There are 0.8% of people who gives ratings per day. Since most people do not have the time to download apps and write comments, normally people do not give ratings or comments very frequently, thus the users who give ratings and comments frequently are outliers. They may belong to certain collusion groups, in which case they are manipulated to do promotion by a certain company. Especially, if some of them just give a high volume of positive ratings, it would be highly probable that they belong to a certain collusion group.

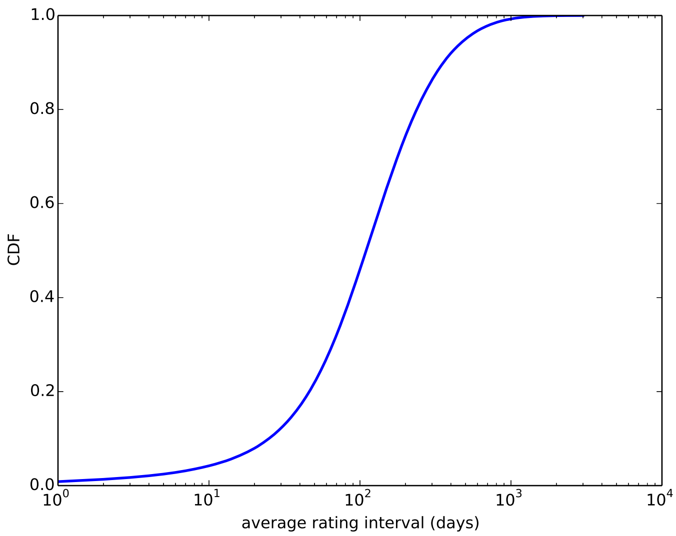

Figure 2.8. Distribution of rating interval per user (CDF).

2.2.5 Promoters Ratio

Figure 2.9 shows the distribution of promoter ratio (PDF), in which the x-axis denotes the number of promote rating ratio (by days) and y-axis denotes its corresponding user number. While in figure 2.10, the x-axis denotes the number of promotion rating ratio (by days) and the y-axis denotes the corresponding accumulative probability. As shown in figures 2.9 and 2.10, the number of promoters who just gives high ratings is very high, about $210,000$. This figure is much higher than users who just give lower ratings, which is only $6,800$. Also, about half of user tend to be a promoter (with value 0.7). Normally, there are many cases when people just give a single 5 star rating for a mobile app store, which would be to promote the app. Also, there are some cases that people just leave one negative rating with 1 star, which could be accounted for a demote action. However, by statistical analysis, the number of people who just do promotion should not be high because one could not have positive attitudes toward all apps they use.

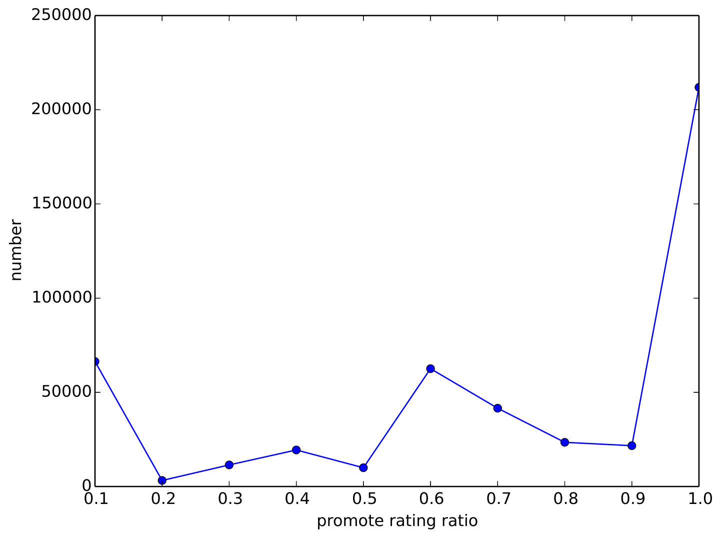

Figure 2.9. Distribution of promoter ratio (PDF).

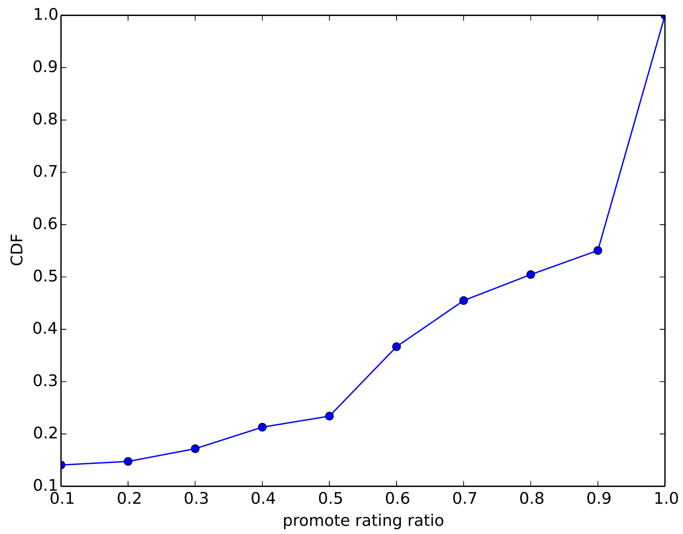

Figure 2.10. Distribution of promoter ratio (CDF).

2.3 App Level Analysis

In this section, we analyze these data from the view of App level. App is the major component of mobile app store. By analyzing the statistics of apps, we can discover the trend of mobile app store. Also we can provide some clues about the promotion activity in the mobile app store.

2.3.1 Average Rating Score Per App

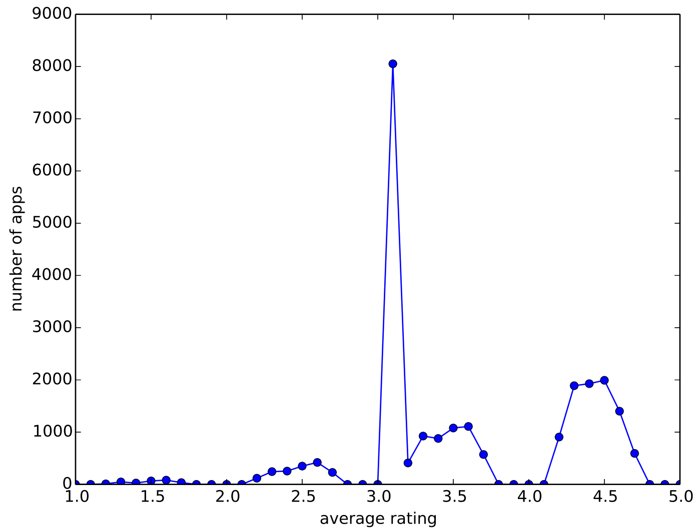

Figure 2.11. Distribution of average rating score per app (PDF)

Figure 2.11 shows the distribution of average rating score per app, in which the x-axis denotes the average rating and the y-axis denotes its corresponding app number. While in figure 2.12, the x-axis denotes the average rating ratio and the y-axis denotes the corresponding accumulative probability. As shown in figures 2.11 and 2.12, the average score per app is about 3.3. However the percent that the apps with a high average score (4.5-5) is about 20%, which is much higher than the percent of apps with a low average score (1.0-1.5), about 3%. The fact that the mobile app market is so profitable implies that there are more high-quality apps than low-quality apps. Besides, for certain apps, the developer would improve the quality of app according to the user comments, especially the negative ones. However, the gap between these two should not be that large.

2.3.2 Rating Score Variance Per App

Figure 2.13 shows the distribution of average rating score per app, in which the x-axis denotes the variance of user rating and the y-axis denotes its corresponding

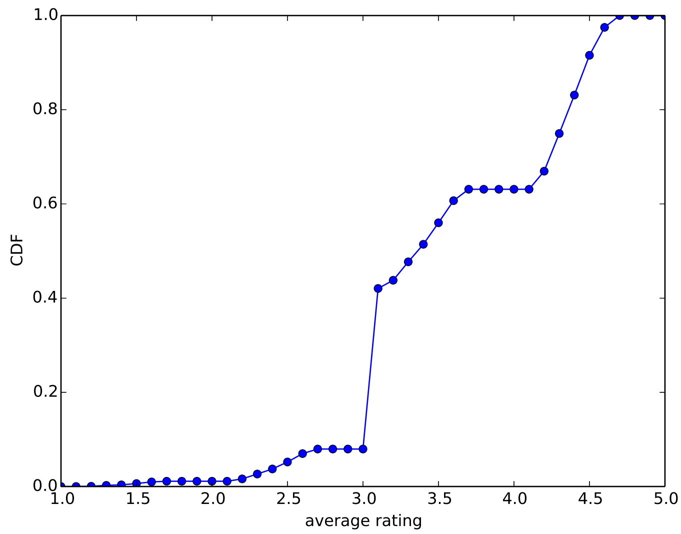

Figure 2.12. Distribution of average rating score per app (CDF)

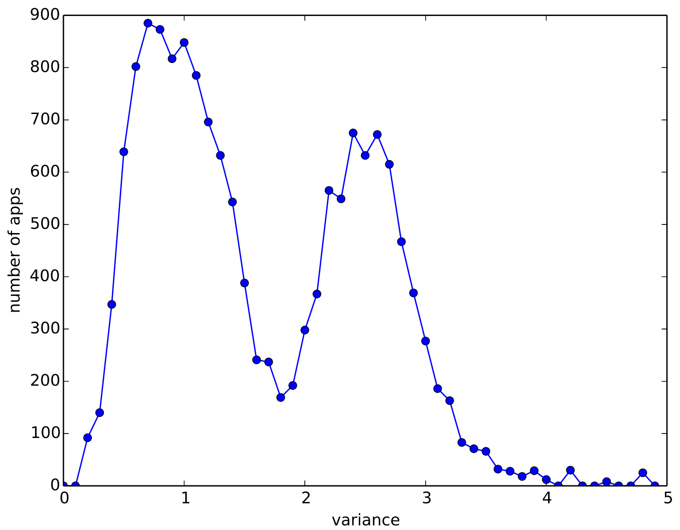

Figure 2.13. Distribution of rating score variance per app (PDF)

app number. While in figure 2.14, the x-axis denotes the variance of user rating and the y-axis denotes the corresponding accumulative probability. Variance of rating scores per app is an important factor, which could reflect the user's attitude toward certain apps. If the variance of rating scores is high for certain apps, it implies that a large number of users have opposite attitude toward this app; some

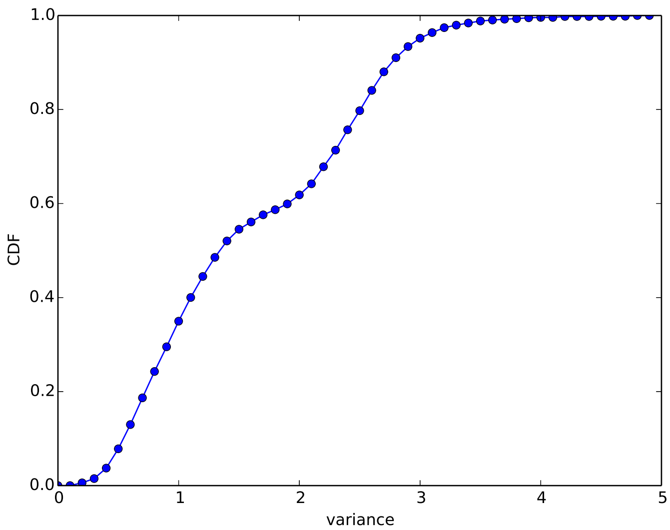

Figure 2.14. Distribution of rating score variance per app (CDF)

users love it while many users hate it. It is quite natural that for one app, some people like it while others hate it. For instance, user A may focus on content, while user B may focus on interface. It is likely that one app is perfect in content while terrible in interface. However, the variance should not be too large. Thus, there is a high probability that the app with high variance of scores have been promoted by a collusion group.

2.3.3 Number of Ratings by Week

Figure 2.15 shows the distribution of number of ratings by week for an app, in which the x-axis denotes the week series and the y-axis denotes its corresponding new comments number. While in figure 2.16, the x-axis denotes the number of new comments and the y-axis denotes the corresponding accumulate probability. As shown in figure 2.15, we list the number of ratings per week for one app. There are peaks and valleys for all time slots, the peak period implies that more users are giving comments for this app during the time. The average number of new comment is 29, while the peek is 81 and the valley is 8. However, if most of the ratings for this period are incredibly high, it may imply that there are some collusion groups, which promote this app. In other cases, if most of ratings are

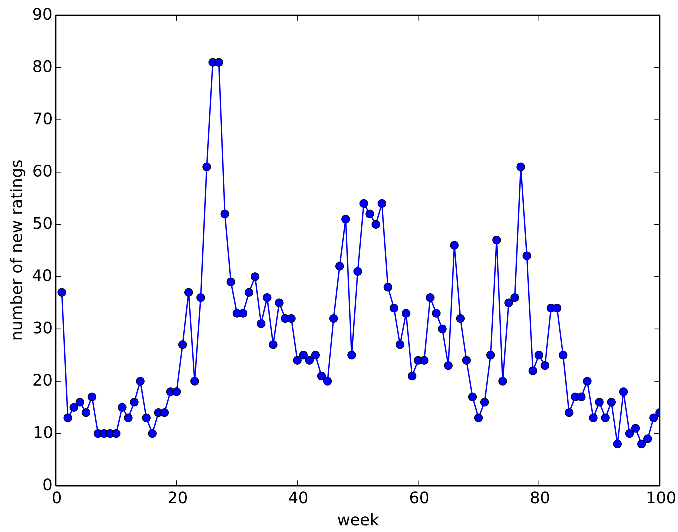

Figure 2.15. Distribution of number of ratings by week (PDF)

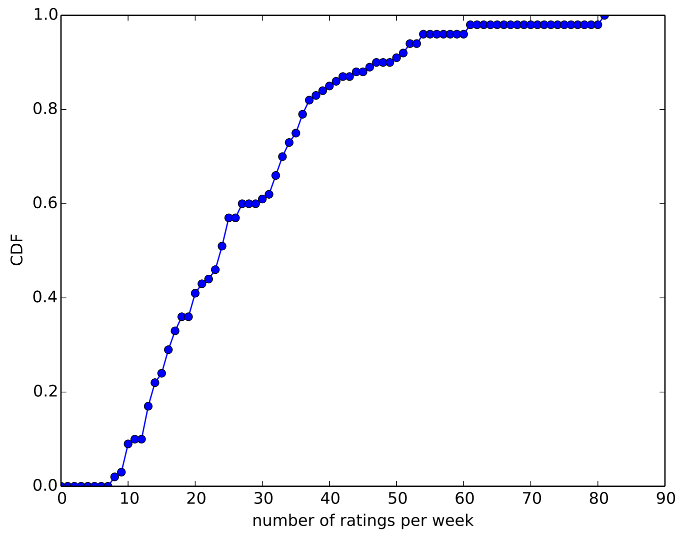

Figure 2.16. Distribution of number of ratings by week (CDF)

very low, it would imply that perhaps there are some collusion groups demoting this app. However, it is not always the case. There may be some cases that the developer has provided some new attractive features, which would cause a burst of new high ratings. However, the analysis can provide some guidance for us to find a suspect list of promoted apps.

2.3.4 Average Ratings by Week

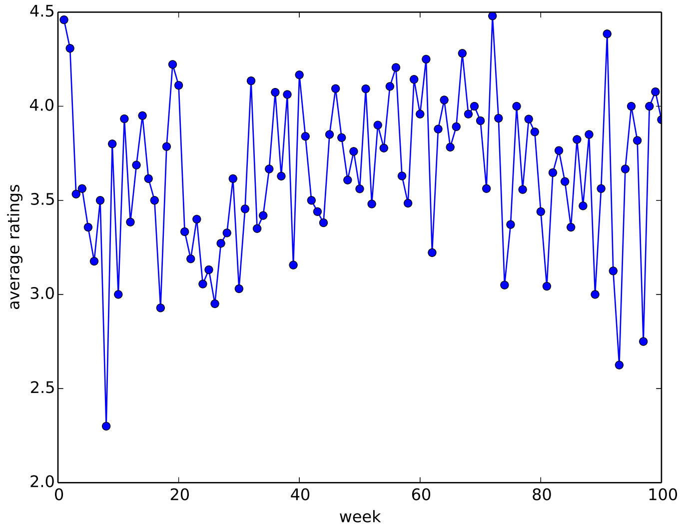

Figure 2.17. Distribution of average ratings by week (PDF)

Figure 2.17 shows distribution of average ratings by week for an app, in which the x-axis denotes the week series and the y-axis denotes its corresponding new average ratings. While in figure 2.18, the x-axis denotes the week series and the y-axis denotes the corresponding accumulative probability. As shown in figure 2.17, we list the average ratings per week for one app.

For an average rating by week, there are also peaks and valleys during different time periods. In most cases, this fluctuation is reasonable because when a new version comes out, there may be some bugs in the application, which leads to low ratings. However, later when the problem is corrected, the new ratings would be higher. Another reason may be due to promotions when certain apps have been recommended by some celebrity in a newspaper or TV show. Such promotions could attract a large number of users causing a high average rating burst in a short time period. However, if for a certain period most of the new ratings are very high, but there are few new user comments, these new ratings are suspicious because they may belong to certain collusion groups intending to promote this app.

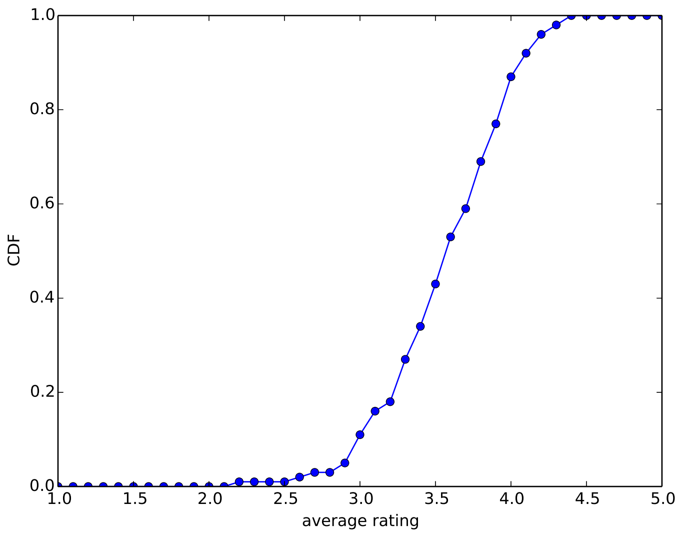

Figure 2.18. Distribution of average ratings by week (CDF)

2.4 Developer Level Analysis

In this section, we analyze the mobile store market and the user feedback from the point of view of developers.

2.4.1 Number of Apps Per Developer

Figure 2.19 shows the relationship between the average ratings and developers, in which the x-axis denotes the number of apps per developer and the y-axis denotes the corresponding accumulative probability. More than 70% of all developers only produce one app. Only about 1% of developers have produced more than 10 apps.

2.4.2 Average Rating Per Developer

Figure 2.20 shows the average rating per developer, in which the x-axis denotes the average rating and the y-axis denotes the corresponding accumulative probability. About 30% of developers have a high average score (higher than 4), which could be referred to as excellent developers, while about 10% of developers have a low average score (less than 2), which could be referred to as bad developers.

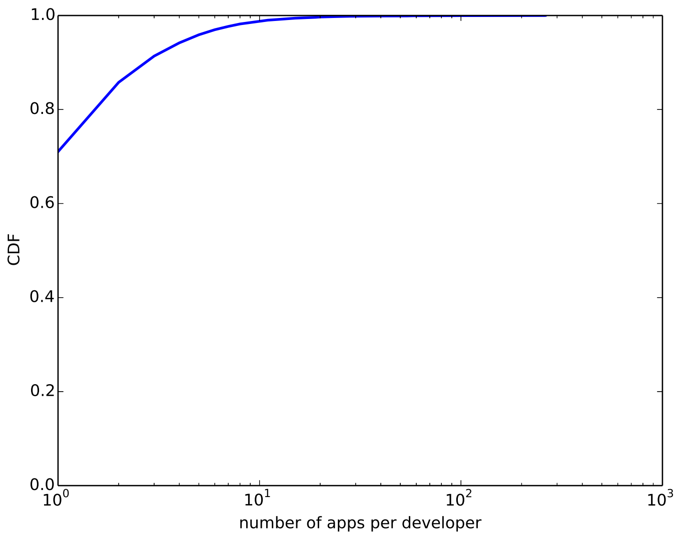

Figure 2.19. Number of apps per developer (PDF)

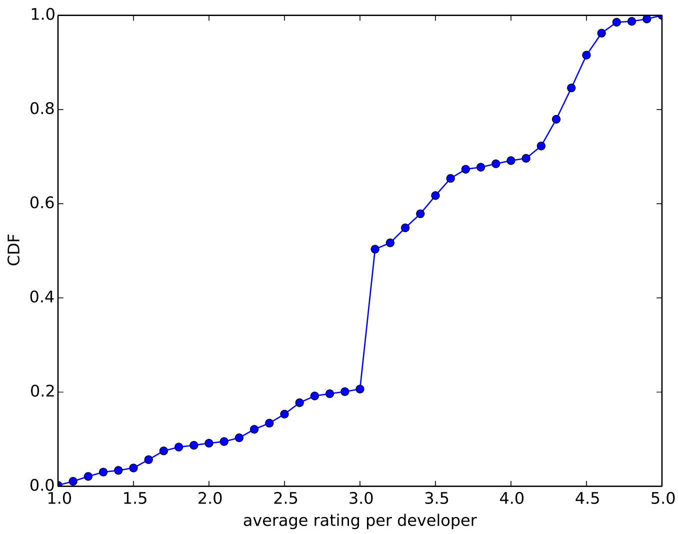

Figure 2.20. Average rating per developer (CDF)

2.4.3 Number of Categories Per Developer

Figure 2.21 shows that about 60% of developers just develop apps in one category. The x-axis denotes the number of categories and the y-axis denotes the corresponding accumulative probability. Only about 2% of developers develop apps in more

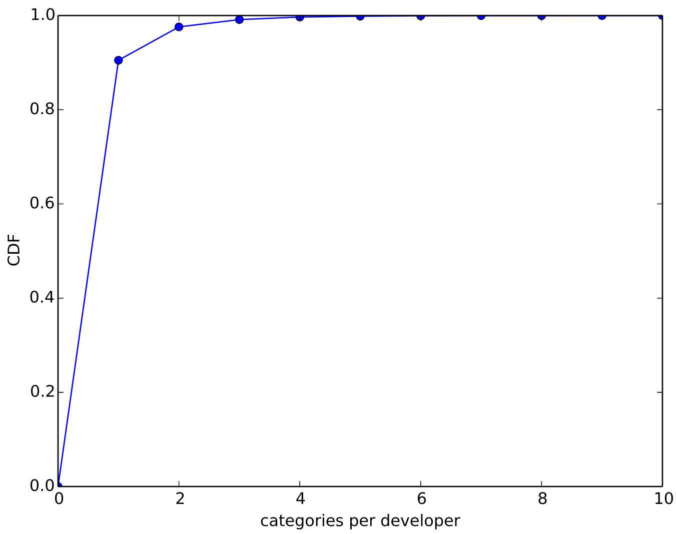

Figure 2.21. Distribution of number of categories per developer (PDF)

than two categories; about 60% of developer just develop one app.

2.4.4 Variance of Ratings Per Developer

Figure 2.22 shows the distribution of variance of apps per developer. The x-axis denotes the variance of ratings per developer and the y-axis denotes the corresponding accumulative probability. About 99% of developers have low variance (less than 2), which means their quality of apps is balanced.

2.5 Conclusion

In this Chapter, we first describe the user feedback data obtained from the Apple App Store. Then we explore some features from this data by analyzing from 3 different aspects: user, application, and developer. These patterns will help us to discover the trend of user feedback for the mobile app store market. Also, these patterns would imply that there are many outliers in the current mobile app rating system. Hence, we would propose an outlier detection algorithm to reveal the potential promotion attackers in Chapter 3.

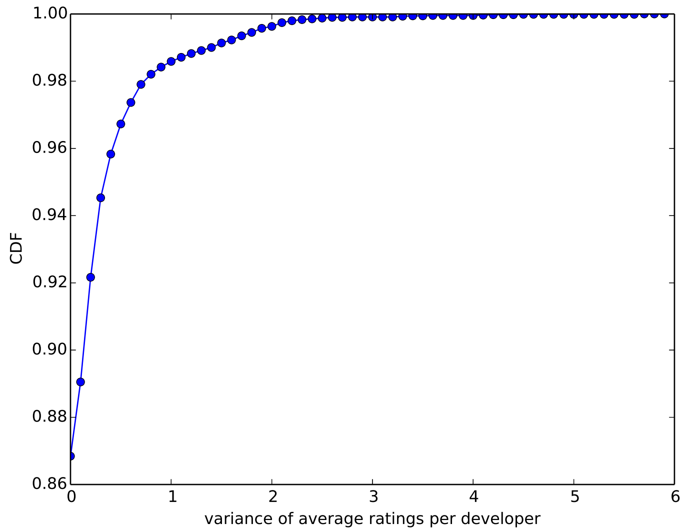

Figure 2.22. Distribution of variance of ratings per developer (CDF)

Chapter 3

Outliers Analysis - A Way to Expose Review Promoters

3.1 Introduction

In statistics, an outlier is an observation point that is distant from other observations. In our research, we consider an outlier to be a certain user who does not follow the normal activity of most users or certain comments which differ from the normal ones. For example, in chapter 2, we list the distribution for the number of comments a user makes. Thus, a user who provides some special comments when giving ratings or provides a large number of comments would be considered an outlier. Also, most comments are not long and focus on one or two topics, so the comments that are lengthy and comprehensive would be considered to be outliers. In [20], Jindal et al studied opinion spams in reviews and conducted an outlier analysis in amazon.com. However, our work differs from this work because we choose different features when we analyzed the user outlier in mobile app stores.

In this part, we use user feedback data from the Amazon app store, and identified the features of outliers. In this thesis, we mainly focus on the user outlier.

3.2 User Outlier Detection

In this section, we focus on how to identify the user outlier and what kind of information we could obtain from these results. As in Chapter 1, there are many promotion activities in the mobile app store. Thus, it is an urgent task to identify the promoters as well as the abused apps. Outlier analysis would serve this purpose. Since intuitively, if one user's behavior is much different from normal users, by definition, it is an outlier. It is not necessarily true that an outlier is indeed a promoter; however, through outlier analysis, we can identify a suspect list for administrators of mobile app stores to conduct further investigations. Consequently, we could identify the abused apps if one app has been rated and commented by many promoters.

3.2.1 Outlier Detection Algorithm

In this part, first we try and identify the features of user outlier and the potential suspicious promoters. This would be very useful because it would allow the administrator of a mobile app store to identify the user in the suspect list case by case based on other useful information such as user ip address.

As in Chapter 2, normal users do not provide a large number of ratings and comments. Also, normal users do not give ratings for apps in a large number of categories. In order to identify the promoters in the mobile rating system, we focus on the outliers with the following features.

- Feature 1: user who gives large amount of comments.
- Feature 2: most of the ratings are high (4-star or 5-star).
- Feature 3: most of rated apps are unpopular app.
- Feature 4: there are some patterns for the content of user comments.

Features 1 and 2 are intuitive, since it is opposite to the activity of normal users. Feature 3 means that we try to identify the user who gives most of the ratings and comments for unpopular app. Here, we define the popularity of app based on the number of total user ratings. It is more natural if we number the downloads for the app to be metric. However, we could not obtain the number of downloads from the mobile app store. Instead, we use the number of user ratings. Intuitively if an app is very popular, it would attract more user ratings and comments. For feature 4, we try to detect if there are some special patterns for all comments of one user. For instance, for all the comments made by one user, the content patterns are highly similar which may be derived from a certain template. Therefore the outlier detection algorithm would work as follows.

In this section, considering the above characteristics, we introduce an algorithm to detect promoters by outlier analysis. First, a user set S_1 is created for users which has feature 1 by setting the number of comments to be a threshold n. In the second step, from the set S_1 obtained from step 1, another set S_2 is derived by adding the condition that the high ratings ratio (high ratings over total ratings) is larger than a value t. Then new set S_3 are obtained by adding another condition that the rated unpopular apps ratio (unpopular apps over total apps) would be larger than r. Finally the users in the set S_3 would be grouped into 4 categories according to contents of the comments. Users in category one write spam comments which contain certain website links. For the rest of users, we employ the duplicate comments ratio θ (duplicate comments over total comments) to classify. Users in category two write many duplicate comments (with $\theta > \beta$). Users in category three write some duplicate comments as well as some individual comments (with $\alpha \leq \theta \leq \beta$). Users in category four write almost individual comments (with $\theta < \alpha$). The detailed algorithm is shown in Algorithm 1. In Algorithm 1, N_i, T_i, R_i and θ_i denote the number of comments, high ratings ratio, the rated unpopular apps ratio, and duplicate comments ratio by user i, respectively. For duplicate comments detection, we employ Levenshtein distance [25] to measure similarity between strings.

3.2.2 Case Study for Outlier

In this part, we employ the above algorithm to identify the outliers in the Amazon app rating system and propose a list of suspect promoters. Also, we show several cases of outliers to identify the features of an outlier, and provide an analysis of how to identify if it is a suspect promoter or not.

Algorithm 1 User Outlier Detection Algorithm

Input: The set of users in a mobile app store dataset S_0; Empty sets S_i ($i = 1, 2, 3$); n, t, r, α, β;

Output: The set of user outliers in ith category C_i ($i = 1, 2, 3, 4$);

```
for each i ∈ S_0 do
  if N_i ≥ n then
    S_1 = S_1 ⋃ {i}
  end if
end for
for each i ∈ S_1 do
  if T_i ≥ t then
    S_2 = S_2 ⋃ {i}
  end if
end for
for each i ∈ S_2 do
  if R_i ≥ r then
    S_3 = S_3 ⋃ {i}
  end if
end for
for each i ∈ S_3 do
  if comments by user i contain website links then
    i ⟶ C_1
  else if θ_i > β then
    i ⟶ C_2
  else if θ_i ≤ β and θ_i ≥ α then
    i ⟶ C_3
  else if θ_i < α then
    i ⟶ C_4
  end if
end for
```

Table 3.1 shows the data size we have for the Amazon App Store in U.S. In total 128,979 apps were picked up and 1,817,808 comments made by 736,666 reviewers were collected.

App store	Apps	Comments	Reviewers
Amazon(U.S)	128,979	1,817,808	736,666

Table 3.1. Amazon App Store data size.

First, we have the original data set to be the top 3000 users with a large number of ratings in the Amazon dataset. The range for the number of comments is 14 to

434, with an average number of 32. Then we get the statistical information about ratio of 5 star ratings over all ratings per user.

Table 3.2. Ratio of 5-star ratings over all ratings per user

range	**0-0.2**	**0.2-0.4**	**0.4-0.6**	**0.6-0.8**	**0.8-1**
frequency	0.155	0.282	0.287	0.178	0.098

Table 3.3. Ratio of 4 and 5-star ratings over all ratings per user

range	**0-0.2**	**0.2-0.4**	**0.4-0.6**	**0.6-0.8**	**0.8-1**
frequency	0.026	0.110	0.245	0.352	0.267

As shown in Table 3.2, there are about 10% of users in the set that tend to give 5 star ratings (80% − 100%). Also, if we consider high ratings as 4 star and 5 star, the result is shown in Table 3.3. It shows that about 26.7% of users just give high ratings.

Then we set the threshold for high ratings ratio t to be 0.7 and the one for unpopular app ratio r to be 0.6 in the above algorithm to conduct the experiment. Also, we set α and β to be 0.05 and 0.8 respectively. In the next part, we present several cases of user outlier and further analyze the features of these outliers to discover if they are suspect promoters. In our experiment, we define an app to be popular if the total rating number is larger than 100, and vice versa.

In the next part, we conduct classification for the outliers and display some typical cases for each category.

3.2.2.1 Category 1

The first outlier example we would present here is User ID A2OXP1IJ3UGTZ. This user gives 45 ratings with comments in total. All these ratings are 5 stars and only focus on apps of game category. 35 of these apps are unpopular. Also, the content of these comments are very similar.

As shown in Table 3.4, many of the comments are duplicated. There are a total of 5 different comments templates and all the comments contain a website

Index	Time	Comment
1	25	FOR AFFORDABLE MARKETING THAT GETS RESULTS BUDGETMARCOM.COM Looks like wife gets one these things day!
2	11	Looks like wife gets one these things day! For affordable marketing that gets results check out www.budgetmarcom.com
3	6	For affordable marketing that gets results check out www.budgetmarcom.com Looks like wife gets one these things day!
4	2	wife seems get one these day. think she really likes them! For affordable marketing that gets results check out http://www.budgetmarcom.com
5	1	wife loved and seems buy one each day For affordable marketing that gets results check out http://www.budgetmarcom.com

Table 3.4. User outlier: case 1

link "www.budgetmarcom.com". This link is an advertising company. Hence, we could conclude that these comments are spams. The user poses these comments in order to increase the hit rate of their own website. We classify this type of outlier to be category 1.

3.2.2.2 Category 2

The second case is user ID A2RGJK0EG8XDDQ. This user made 43 5-star comments in total and these comments covered apps of three categories: music, games, and entertainment. 32 of these apps are unpopular. There are many similarities among the contents of the comments, as shown in Table 3.5.

Index	Time	Comment
1	39	this game for tablet was good son enjoyed lot this app when downloaded this app accomplish expactations
2	3	good app recommend for all you also free and working without problems all plataforms and equipments....
3	1	good material recommend for all you buy this app. The price utility and game are very good and very nice

Table 3.5. User outlier: case 2

We notice that this user uses several templates for comments, which is suspicious because normal users tend to give specific comments for different apps. Generally when a user gives comments for an app, the user expresses their feelings about good or bad points of the app. Thus, for different apps, the comments made by one user should not be exactly the same. For this case, the user seems to have just wanted to give quick comments thus copying previous comments.

Therefore, for this user outlier, the user provides a large number of comments which cover multiple categories, all with 5 star ratings. A large number of the apps the user rated are unpopular ones. Also there are many duplicate comments. All these features show that this user is a highly suspicious promoter. We classify this type of users as category 2.

3.2.2.3 Category 3

The third case is similar to the second one, but there are some differences. The type of comments given by the users in this group are mixed. Some comments are based on templates, others are individual comments. The user id is A3B3OYIME142VO. There are 68 comments in total and 58 of the rated apps are unpopular.

As shown in Table 3.6, we just list several comments here. The items 1 and 2 are duplicate comments, the items 3 to 7 are individual comments. For the first two items, similar to the user in case 2, this user makes many duplicate comments, which is suspicious because the normal user tends to give a different comment for each app. When normal users give comments for an app, they would like to express their feeling about their like or dislike about the app. Thus for different apps, there are some differences among the comments made by one user.

While there are many identical comments for this one user, other comments are individual. This implies that the user intends to disguise his promotion activity in a certain way. By doing so, this user tries to act more like a normal user, rather than an outlier. However, we still classify this type of user as a suspect promoter. We classify this type of user to be category 3.

3.2.2.4 Category 4

The last category is a special one. For this category, the user seems to be a promoter at first sight. However after careful analysis about the content of comments,

Index	Time	Comment
1	21	another fantastic purchase from amazon love the site and the offers find great availability and always great apps for your tablet and smartphone
2	9	enjoy app buys from amazon never disappointed with purchases the test drive feature always help before purchase wish was available all apps though
3	1	lovin the evil genius look the assassin. took few minutes get used how aim and shoot but once you figure that out taking out the shotgun guy piece cake
4	1	love this app very cool worth the purchase wish there were more apps like this one huge fan logic thinking games but the little guy always looks the same direction
5	1	well thought out game test your logic skills and makes you wish you hadn't blew off that physics class lol
6	1	while i'm not craftswoman trade find this app incredibly useful. tend tackle jobs myself and this app comes handy especially you need visita hardware store. i'll consult handy gadget before asking male clerk since they always seem talk woman she doesn't know sheetmetal from foil and hammer from philips screwdriver. not bad for the app
7	1	useful when wanting analyze your own dreams and what they might mean always good take the definitions with grain salt though

Table 3.6. User outlier: case 3

it is most likely a normal user rather than a suspicious promoter. The user (id A3ATAIGN3826YQ) gives 72 5-star ratings. The user concentrates on apps of two categories: novelty and games. And 70 of these apps are unpopular ones. Therefore, this user seems to be a promoter candidate. However, most of the rated apps are games, and the content of comments show that this user is a father who downloads many game apps for his daughter.

As shown in Table 3.7, we just list several comments here. All these comments express the user's personal feelings about the app and the content is detailed. Thus the user is less likely to be a candidate for a suspicious promoter. Because intuitively promoters would not just focus on apps of a single subcategory. Also when writing comments, promoters would not specify "daughter" in the comments.

Index	Time	Comment
1	1	daughter loves playing this dress game. The celebrities selection great lots dress items choose from plenty different backgrounds and musical accessories keep her playing the game. Highly recommend!
2	1	daughter loves these animal dentist games. love them because they teach her how important take care her teeth her animals. She loves using all the tools fix the animals teeth adores playing the most with Grumpy cat. Provides hours fun highly recommend!
3	1	The kids loved the first game the series and this one even more fun play educational well. Plus you have new characters perform dental procedures which great! Highly recommend!
4	1	Really wonderful game for the kids yourself (I'm early 30's) for Halloween anytime. really enjoyed it. Plus you don't have the mess having clean real pumpkin.
5	1	The kids love dressing the dragons various outfits. The graphics are wonderful and the game also enjoyable for grown ups too!
6	1	Super fun game! Who wouldn't want make their own happy meal? Lots different food items choose from and hours enjoyment for the kids too!
7	1	daughter loved Christmas Dentist Office Santa much bought her this one too she could fix the animals teeth bling them out too! Provides her with hours fun educational value! highly recommend it!
8	1	daughter had blast playing this game. She knew all celebrities and enjoyed popping the pimples all them! Highly recommend for the kids!

Table 3.7. User outlier: case 4

Even though there are some phrases which appear frequently in these comments, like "Highly recommend!", these comments appear to be more like a personal expression custom rather than template or duplicate. However, this type of user needs more investigation to further validate whether or not it is a promoter. We categorize this type of users to be category 4.

Category	1	2	3	4
User Number	1	39	30	807

Table 3.8. Suspect list discovered by outlier detection algorithm

3.2.3 Result Analysis

As shown in Table 3.8, we find a suspect list of outliers of four categories, which account for about 29% of the top 3000 users. The users in category 2 and 3 are the ones with high-level suspicions. While the ones in category 1 and 4 are less suspicious promoters. Therefore, by applying our outlier detection algorithm, we successfully propose a suspect list of highly likely review promoters, with which mobile app store administrators could conduct further investigations.

Chapter 4

User Feedback Distillation by Topic Analysis

4.1 Introduction

In [8], Fu et al proposed a system which could analyze tens of millions of user ratings and comments in Google Play Store. This work inspires us to conduct topic analysis for user reviews in Apple App Store. However previous work only considers negative reviews, thus missing topics with positive ratings. In this thesis, we also discover topics of positive comments.

Overall, the majority of user comments are informative. We found that people often focus on features or the functionality of mobile apps. At the same time there are some comments which are just expressions of emotion. For instance, simple words such as “good”, “great work” would be used to express strong feeling of happiness, which means a high satisfaction with the application.

In this section, we try to identify the root causes of user comments by topic model analysis. In this way, we will find out why people love or dislike the app. This kind of information is highly valuable for mobile app developers, because app developers could improve certain features of the app in later versions. Also the app developer could focus more on certain functionalities when developing new apps. We refer to this task as user feedback distillation.

No	Score	Comment
1	5	simply the best new tech device have ever purchased. the folks who designed the phone nailed it!
2	5	Worth the download most definitely. Great work! Thanks!
3	5	Love the app awesome!. The stories are exciting. must have endless reading interesting stories.
4	5	love it! highly recommend it! Easy use and has excellent tools use. well worth every penny!
5	5	This app beautifully designed and full great catalogs. love browse catalogs and this even better than flipping through the paper
6	5	Exciting Game! I never get anything done because I can't put this game down! You really feel like you're running for your life from those big black apes!
7	5	I noticed that when the cops block the road they also float in the air but the cones don't. Also you should make it where you can choose that when you die your in first person view. I luv it.
8	5	The Square is working fine again after the recent update. Have not tried the new features yet that queue when there is no service in certain areas but I will assume that it will work.

Table 4.1. Examples of positive user comment

4.2 Topic Analysis Introduction

User comments in mobile app store differs largely from the comments in other online markets, such as the Amazon online market. First, the comments in mobile app store are shorter, because typing is not easy on the cellphone, thus people do not intend to spend much time writing comments before submitting them. Therefore, it is not common to see a long and detailed review as found in the Amazon online market. Second, there are many misspellings and emoticons in the comments, which is very common for mobile users.

In order to focus on the root causes of user comments, we divide the user comments into two categories: positive or negative. We define user comment to be positive if the rating is 4-stars or 5-stars, while negative comments are those with

No	Score	Comment
1	1	needs waaay too many permissions, was constantly running in the background on my kindle doing who knows what, memory hog etc. not worth it!
2	1	this game is boring and pointless you get the same. moods more than once and it just chooses the mood at random
3	1	this game has poor graphics and does not even fill the whole screen. uninstalled within 13 minutes time.
4	1	This game boring...why does everyone like much?
5	1	Very disappointed that these puzzles are sooooo easy solve.
6	1	Crashes open. Every time without exception.
7	1	App keeps freezing and have power off reset ipad2.
8	1	The features are great and was much needed update but the messages still have sending and receiving issues have iPhone 3G

Table 4.2. Examples of negative user comment

1-stars or 2-stars. Several positive comment examples are listed in Table 4.1. We also show several negative comment examples in Table 4.2. We list ten typical user comments for both positive comments and negative ones.

For these comments, there are some which are not very informative. They are just some kinds of emotional expressions or too general expressions about failures, such as item 2 in Table 4.1. Most user comments are informative, some praise or criticize certain features or functionalities. For instance, for items 3 and 5 in Table 4.1, the users liked the app because its content was attractive, or the design was excellent. For items 6 and 8 in Table 4.2, the users complained about the app because of the stability issue or compatibility.

4.3 Market-level Topic Analysis

User comments for mobile apps are relatively short because typing in a mobile phone is not very convenient. Sometimes certain user comments may contain just a sentiment, rather than an informative review. Thus, first we try to concatenate comments of the same app together as a new document. In this section, we try to first discover what the most important aspects are that users would focus on when they rate an app. According to the Apple app store, apps are divided into 22 categories, such as Games, Books, Educations, etc. Then we identify whether people have certain bias over the expectation for apps of different categories. These results are important because the developers would have a good understanding of users' specific demands or expectations when designing products.

Latent Dirichlet Allocation (LDA) model [4] in machine learning is a popular tool for topic modeling. We apply the LDA model to perform a precise topic analysis for user reviews. In this way, we identified topics which represent the user's main concerns toward apps. We analyzed the topic distribution for apps of different categories and identified the most concerned factors users have.

4.3.1 Topic Analysis Experiments

User comments are normally shorter in mobile app stores compared to comments from other types of stores, like amazon. The average length of the user comments for an app store is only 61 characters. As a result, we concatenated the comments from the same app together as a new review document. We used the Stanford Topic Modeling Toolbox to train an LDA model [4].

cause	cost	fun	communication	experience	version
words	money	play	messages	stupid	version
	pay	fun	email	waste	new
	free	more	phone	more	news
	paid	games	message	really	has
	more	playing	send	hate	read
	want	every	people	make	more
	buy	really	iphone	how	old
	waste	because	call	better	great
	should	very	back	could	used
	month	make	text	said	too
ratio	7%	6%	6%	15%	11%

Table 4.3. List of words for negative feedback I

cause	location	stability	connectivity	compatibility	media
words	location weather very find iphone map off gps more does	open won every crash crashing back after keeps then used	messages log check mobile card login make every after had	ipad iphone very sync version has great then list back	watch video music play videos shows then great songs download
ratio	13%	14%	13%	8%	6%

Table 4.4. List of words for negative feedback II

4.3.2 Topic Analysis for Negative Comments

To determine the reasons why people dislike certain apps, we selected only negative comments with 1 star and 2 star rating. Further, we obtained all the review documents for $2,200$ popular apps for LDA model training, with 100 apps from each category. We did not use all the user comments from the dataset because the proportion of apps in game category is very large, which would skew the topic analysis results. Tables 4.3 and 4.4 show the results of a 10-topic LDA model analysis. We list the top-10 weighted words for its vocabulary set. We also added certain words to give a summary of each topic to show clear reasons why people like or dislike certain apps. As the tables show, most topics exhibit clear reasons why people dislike an app. There are some factors about function, such as communication failure, stability, compatibility, or other issues such as the cost of the app.

4.3.3 Top Complaints in Different Category

In the previous section, we showed the top-10 complaints included in negative feedback. In this section, we extend our analysis for negative feedback to discover if users have the similar complaints for apps in different categories. The result proves that this is not the case. For each category, we determined the most common complaints. The top-3 complaints in each category are shown in table 4.5. The percentages shown in the brackets in the table indicate the weights of apps for each complaint. For instance, for apps in the books category, 19% are criticized for connectivity issues, 15% suffered from compatibility problems, while 12% complained about the version.

The results show that user complaints are concentrated on different aspects for apps in different categories. For instance, complaints on connectivity are dominant in the Business, Finance and News categories, while for apps in the Food, Medical, Reference and Lifestyle categories, the most common complaints concern the experience issue. It seems logical for apps in the Business, Finance and News categories that connection is a very important factor for user experience and thus would be a major source of complaints. Results also show that communication is the most important factor for apps of Social networking. This is expected because communication is the basic function apps in social networking would provide. The focus of complaints for different categories could provide useful information for app developers to improve the quality of their products.

In [8], Fu et al presented the result of topic analysis for apps in Google Play. Compared with their findings, our results show that there are some differences about the complaint factors. In Google Play, the topics such as accuracy and attractiveness stand out; however, for the Apple App Store, topics such as experience would be considered more important. The difference implies that users in Google Play and Apple App Store may focus on different aspects when evaluating a mobile app.

4.3.4 Topic Analysis for Positive Comments

A follow up question one may ask is whether users focus on the same aspects when writing positive comments. To determine the reasons why people like certain apps, we selected only positive comments with 5 stars rating. Further, similar to the setting in previous experiment for negative comments, we obtained all the review documents for $2,200$ popular apps for LDA model training. Tables 4.6 and 4.7 show the results of a 10-topic LDA model analysis. We list the top-10 weighted words for its vocabulary set.

Unfortunately, the results are not as good as in the previous experiments. As the tables show, most topics do not exhibit clear reasons why people like an app.

	Category	1st Complaints	2nd Complaints	3rd Complaints
Application	Books	connectivity (19%)	compatibility (15%)	version (12%)
	Business	connectivity (17%)	compatibility (12%)	experience (12%)
	Catalogs	stability (18%)	experience (13%)	cost (12%)
	Education	fun (20%)	media (12%)	communication (10%)
	Entertainment	fun (18%)	stability (15%)	experience (11%)
	Finance	connectivity (17%)	experience (12%)	stability (12%)
	Food	experience (22%)	compatibility (14%)	media (11%)
	Games	fun (27%)	connectivity (16%)	media (12%)
	Health & Fitness	experience (23%)	connectivity (18%)	version (12%)
	Lifestyle	experience (30%)	stability (17%)	fun (12%)
	Medical	experience (32%)	cost (17%)	connectivity (11%)
	Music	media (37%)	connectivity (21%)	compatibility (12%)
	Navigation	location (39%)	stability (13%)	compatibility (10%)
	News	connectivity (35%)	compatibility (13%)	version (11%)
	Photo	media (28%)	stability (15%)	version (14%)
	Productivity	experience (28%)	compatibility (17%)	connectivity (10%)
	Reference	experience (32%)	version (18%)	cost (13%)
	Social Networking	communication (37%)	connectivity (16%)	stability (12%)
	Sports & Fitness	experience (17%)	connectivity (13%)	version (12%)
	Travel	location (47%)	experience (12%)	media (10%)
	Utilities	experience (17%)	stability (11%)	cost (10%)
	Weather	location (17%)	experience (12%)	version (11%)

Table 4.5. Top 3 complained aspects of each app category

Maybe in the positive comments, a large proportion of contents are just simple words such as "awesome" or "great", which are used to express strong feeling of happiness. Thus, it makes the positive comments less informative. Consequently, the topic analysis results by LDA model is worse than expected.

words	iphone ipad browser work only google safari fast apps need	pictures read books reading day book favorite articles keep way	messages pics photos make cute friends photo many look apps	music people songs download watch only listen way shows video	sleep helpful helps track every sounds how night light useful
ratio	18%	13%	9%	11%	9%

Table 4.6. List of words for positive feedback I

words	play addicting games playing iphone funny addictive make think should	weather keep recipes track useful list card simple convenient everything	messages accurate need used apps info useful see find there	shopping deals track way money find makes store keep how	bible helpful thank word study words day learn helps daily
ratio	12%	7%	7%	8%	6%

Table 4.7. List of words for positive feedback II

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this thesis, we focus on a measurement study of user feedback in mobile app stores. Specifically, we analyzed tens of millions of user ratings and comments in the Apple app store and Amazon app store. We conducted review spam detection by outlier analysis and root cause analysis for user feedback by machine learning.

In Chapter 2, we first describe the user feedback data obtained from the Apple App Store. Then we identified some patterns from this data by analyzing from 3 different aspects: user, application, and developer. These patterns would help us to discover the trend of user feedback for mobile app store market.

In Chapter 3, we focus on the security issue of the mobile app rating system. Since positive ratings and comments would most probably lead to more downloads which would mean more profit by the app developer, some app developers tend to hire some collusion groups to promote their apps. Then we tried to find some potential candidates for these promoters via detecting the outliers in the rating system. More specifically, we focused on users who give large numbers of reviews. Since ordinary user do not devote too much time to writing many comments, this type of user would be suspected to be a promoter.

In Chapter 4, we explore root causes of comments by topic analysis. We intend to use machine learning techniques to discover why people like or dislike the app. We collected the user comments of mobile apps, then applied the machine learning tool, Lattent Dirichlet Allocation (LDA) model to analyze these user reviews.

More specifically, we uncover the topics which are the root causes for the user's main concerns toward certain mobile apps. Also, we tried to find out the topics distribution for mobile apps, and determine the most critical issues the user praises or criticizes for the app. Further, we show which aspect the users focus on when judging a mobile application.

5.2 Future Work

In Chapter 3, we propose an algorithm to detect the potential promoters through user outlier analysis, but more work can be done from the following two perspectives:

- In Chapter 3, we have mentioned the ground truth problem, which is hard for every online rating system research. How to effectively validate the suspect list? If there is some collaboration with some mobile app store, such as Apple iTunes or Google Play, we could have access to more information to identify the real promoters.
- In Chapter 3, we focused on user outlier analysis. There is another type of outlier, comment outlier. Intuitively, the 5-star comment with high helpfulness rate would lead to more downloads and thus it is also the target of promoters. By analyzing the Amazon dataset, we find that there are some comment outliers for certain unpopular apps. For instance, it is quite suspicious that one comment for an unpopular app has very high helpfulness agree number. It does not imply that the user who makes the comment is suspicious, instead the ones who vote for this one should be concerned. However up to now, we could not conduct related analysis since we lack information about the users who cast their votes for comments.

Bibliography

[1] Felt, A. P., E. Ha, S. Egelman, A. Haney, E. Chin, and D. Wagner (2012) "Android permissions: User attention, comprehension, and behavior," in *Proceedings of the Eighth Symposium on Usable Privacy and Security*, ACM, p. 3.

[2] Xie, Z. and S. Zhu (2014) "GroupTie: toward hidden collusion group discovery in app stores," in *Proceedings of the 2014 ACM conference on Security and privacy in wireless & mobile networks*, ACM, pp. 153–164.

[3] Frank, M., B. Dong, A. P. Felt, and D. Song (2012) "Mining Permission Request Patterns from Android and Facebook Applications." in *ICDM*, pp. 870–875.

[4] Blei, D. M., A. Y. Ng, and M. I. Jordan (2003) "Latent dirichlet allocation," *the Journal of machine Learning research*, **3**, pp. 993–1022.

[5] Blei, D. M. and J. D. Lafferty (2006) "Dynamic topic models," in *Proceedings of the 23rd international conference on Machine learning*, ACM, pp. 113–120.

[6] Lin, C., Y. He, C. Pedrinaci, and J. Domingue (2012) "Feature lda: a supervised topic model for automatic detection of web api documentations from the web," in *The Semantic Web–ISWC 2012*, Springer, pp. 328–343.

[7] Hong, L. and B. D. Davison (2010) "Empirical study of topic modeling in twitter," in *Proceedings of the First Workshop on Social Media Analytics*, ACM, pp. 80–88.

[8] Fu, B., J. Lin, L. Li, C. Faloutsos, J. Hong, and N. Sadeh (2013) "Why people hate your app: Making sense of user feedback in a mobile app store," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 1276–1284.

[9] Joshi, M., D. Das, K. Gimpel, and N. A. Smith (2010) "Movie reviews and revenues: An experiment in text regression," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Association for Computational Linguistics, pp. 293–296.

[10] Chahuneau, V., K. Gimpel, B. R. Routledge, L. Scherlis, and N. A. Smith (2012) "Word salad: Relating food prices and descriptions," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Association for Computational Linguistics, pp. 1357–1367.

[11] Mukherjee, A. and B. Liu (2012) "Modeling review comments," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, Association for Computational Linguistics, pp. 320–329.

[12] Ding, X., B. Liu, and P. S. Yu (2008) "A holistic lexicon-based approach to opinion mining," in *Proceedings of the 2008 International Conference on Web Search and Data Mining*, ACM, pp. 231–240.

[13] Ghose, A. and P. G. Ipeirotis (2011) "Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics," *Knowledge and Data Engineering, IEEE Transactions on*, **23**(10), pp. 1498–1512.

[14] Hu, M. and B. Liu (2004) "Mining and summarizing customer reviews," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 168–177.

[15] Archak, N., A. Ghose, and P. G. Ipeirotis (2011) "Deriving the pricing power of product features by mining consumer reviews," *Management Science*, **57**(8), pp. 1485–1509.

[16] Hu, N., J. Zhang, and P. A. Pavlou (2009) "Overcoming the J-shaped distribution of product reviews," *Communications of the ACM*, **52**(10), pp. 144–147.

[17] Archak, N., A. Ghose, and P. G. Ipeirotis (2007) "Show me the money!: deriving the pricing power of product features by mining consumer reviews," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 56–65.

[18] Chevalier, J. A. and D. Mayzlin (2006) "The effect of word of mouth on sales: Online book reviews," *Journal of marketing research*, **43**(3), pp. 345–354.

[19] Felt, A. P., K. Greenwood, and D. Wagner (2011) "The effectiveness of application permissions," in *Proceedings of the 2nd USENIX conference on Web application development*, USENIX Association, pp. 7–7.

[20] Jindal, N. and B. Liu (2008) "Opinion spam and analysis," in *Proceedings of the 2008 International Conference on Web Search and Data Mining*, ACM, pp. 219–230.

[21] Li, F., M. Huang, Y. Yang, and X. Zhu (2011) "Learning to identify review spam," in *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Three*, AAAI Press, pp. 2488–2493.

[22] Mukherjee, A., B. Liu, and N. Glance (2012) "Spotting fake reviewer groups in consumer reviews," in *Proceedings of the 21st international conference on World Wide Web*, ACM, pp. 191–200.

[23] Wang, G., S. Xie, B. Liu, and P. S. Yu (2011) "Review graph based online store review spammer detection," in *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, IEEE, pp. 1242–1247.

[24] Xie, S., G. Wang, S. Lin, and P. S. Yu (2012) "Review spam detection via temporal pattern discovery," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 823–831.

[25] Levenshtein, V. I. (1966) "Binary codes capable of correcting deletions, insertions and reversals," in *Soviet physics doklady*, vol. 10, p. 707.