

## Research Article

# A Mobile Application Recommendation Framework by Exploiting Personal Preference with Constraints

**Konglin Zhu, Zexuan Liu, Lin Zhang, and Xinyu Gu**

*School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, China*

Correspondence should be addressed to Konglin Zhu; [klzhu@bupt.edu.cn](mailto:klzhu@bupt.edu.cn)

Received 3 February 2017; Accepted 20 March 2017; Published 19 June 2017

Academic Editor: Hao Wang

Copyright © 2017 Konglin Zhu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Explosive mobile applications (Apps) are proliferating with the popularity of mobile devices (e.g., smartphones, tablets). These Apps are developed to satisfy different function needs of users. Majority of existing App Stores have difficulty in recommending proper Apps for users. Therefore, it is of significance to recommend mobile Apps for users according to personal preference and various constraints of mobile devices (e.g., battery power). In this paper, we propose a mobile App recommendation framework by incorporating different requirements from users. We exploit modern portfolio theory (MPT) to combine the popularity of mobile Apps, personal preference, and mobile device constraints for mobile App recommendation. Based on this framework, we discuss the recommendation approaches by constraints of phone power and limited mobile data plan. Extensive evaluations show that the proposed mobile App recommendation framework can well adapt to power and network data plan constraints. It satisfies the user App preference and mobile device constraints.

## 1. Introduction

The advent of the mobile Internet era brings the popularity of mobile phones and explosive growth of users. Along with the wide spread of the Internet and mobile phones, millions of mobile Apps spring up to cover every aspect of mobile users' activity, such as shopping and food. However, not all mobile Apps are well developed and mobile users may only prefer well functional Apps. The majority of App Stores recommend users Apps by either the amount of downloads, or the comments from other mobile users. On the one hand, it opens a door to those fraud Apps by faking downloads and comments [1]. On the other hand, such popular Apps may not fit the preference of mobile users. It is necessary to recommend to mobile users their preferred Apps.

Furthermore, the capability of current mobile phones is restricted by computation, storage, battery volume, network data plan from network service provider, and so forth which limit the usage of mobile devices and associated Apps. Considering the constraints of mobile devices, it will come up with different recommendation results of mobile Apps. For example, a mobile user will not launch a power consuming App when the power of the mobile device is limited. In such

case, it is better to recommend to the mobile user mobile Apps with less power consumption. Similarly, a mobile user hardly launches a network data consuming App when the network data is limited.

The studies of mobile App recommendation have been investigated widely in recent years. The majority of previous works consider the general popularity of mobile Apps. For instance, Zhu et al. [2] consider the popularity of mobile Apps as well as privacy and security concerns of users. They propose recommending popular Apps with privacy and security constraints. Furthermore, many previous studies also take user preference into consideration. Liu et al. [3] conduct the systematic study on incorporating both interest-functionality interactions and users' privacy preferences to perform personalized App recommendations. Appjoy [4] makes personalized mobile App recommendation by analyzing how users actually use their installed Apps, and then it exploits collaborative filtering algorithm for individual recommendation. Nevertheless, it lacks the study that takes the constraints of mobile devices and network resource into consideration.

In this paper, we recommend mobile Apps by considering the popularity of mobile Apps, personal preferences of mobile

users, and mobile device constraints. We first carry out a methodology to formulate the popularity of mobile Apps by considering mobile App usage time duration, frequency, the amount of usage, and the number of users. Then we formulate user's personal preference by Bayesian probability model. Accordingly, we propose a mobile App recommendation framework inheriting from modern portfolio theory (MPT) [5] by considering the popularity of mobile Apps, personal preference of mobile users, and mobile device constraints. Two use cases by taking battery residual power of mobile devices and limited network data plan into consideration are illustrated to show the adaption of the proposed mobile App recommendation framework. Extensive evaluation results show that the proposed framework is effective for mobile App recommendation with different constraints.

The contributions of the paper are summarized as follows:

- (i) We adopt a methodology to formulate the popularity of mobile Apps and the personal preference of mobile users.
- (ii) We carry out a framework to recommend Apps based on MPT by taking the popularity of mobile Apps, the user preferences, and the mobile device constraints into consideration.
- (iii) We explore the recommendation framework by two use cases and evaluate the App recommendation method on our collected data. It shows that the method can reduce traffic and power consumption of mobile phones and meanwhile fit for the personal preference.

The rest of the paper is structured as follows: Section 2 introduces the related work. Section 3 carries out the mobile App recommendation framework with constraints. Section 4 presents two cases that recommend mobile Apps with power and network data constraints. Finally, the conclusion is drawn in Section 5.

## 2. Related Work

In this section, we review the state-of-the-art mobile App recommendation approaches. Most of the App recommendation methods depend on the popularity or personal preference. For instance, Zhu et al. [2] propose two approaches for mining common context-aware preferences based on two different assumptions, namely, context independent and context dependent assumptions, which can fit into different application scenarios. Pu et al. [6] examine how researchers have evaluated design methods that augment recommender system ability to help users find the information or product that they prefer. Zheng et al. [7] present an approach known as user-centered collaborative location and activity filtering, to pull data from users and apply collaborative filtering to find like-minded users and like-patterned activities at different locations. There are also several recommendation systems that take other factors into consideration. For example, Zhu et al. [8] combine both the user preference to mobile Apps and the privacy risk of Apps and propose a mobile App usage recommendation method named AppURank to recommend

the secure Apps for users. Ge et al. [9] provide two cost-aware latent factor models to recommend travel packages by considering both the travel cost and the tourists interests and design a cPMF model that models the tourists cost with a 2-dimensional vector and a GcPMF model where the Gaussian prior is used to express the uncertainty of the travel cost. Liu et al. [10] propose a methodology to learn privacy profiles for permission settings and leverage these profiles in a personalized privacy assistant that actively supports users in configuring their permission settings. Shani et al. [11] think about the long-term effects of each recommendation and the expected value and describe a new model for recommender systems based on Markov decision processes.

We propose mobile App recommendation framework different from the state of the art by incorporating the popularity of mobile Apps, user personalized preference, and the mobile device constraints. The framework exploits MPT to address mobile App recommendation issue by considering popularity of mobile Apps and user preference as investment and taking mobile device constraints as the risk of investment. The recommended mobile Apps are ranked by the output of MPT.

## 3. Mobile App Recommendation Framework

In this section, we introduce the data set for analysis and mobile App recommendation framework based on popularity of mobile Apps, personal preference, and mobile device constraints.

*3.1. Data Set.* In this work, we use the data set which is introduced in our previous work [8] for analysis. In order to obtain such data, we designed a mobile App named AntTest (<http://www.wandoujia.com/apps/edu.bupt.anttest>) on Android platform. AntTest was developed to measure network speed of mobile device. For such purpose, the data such as App usage pattern of each user was recorded every several seconds, which can be exactly applied for this study as well. We put AntTest in the App store to provide a way for users to download and use. After more than 500 days of data collection, the data set contains over 13,000 mobile users with mobile usage patterns, involving 16,878 mobile Apps and 13.8 TB network traffic. The data set records every user's mobile phone usage activity in every several seconds. It covers over 100 million of records. In each record, the equipment ID is recorded by IMEI, and the mobile App is identified by the name and package of the App. Besides, the timestamp and data traffic (both download speed and upload speed) are also logged in the record. We anonymize users in data set to conduct the experiment for the privacy concerns. The basic characteristics of the data set are shown in Table 1.

*3.2. The Popularity of Mobile Apps.* The popularity of mobile Apps is usually measured by the number of downloads, the active users, or the frequency of usage. These different metrics indeed reflect different usage features of mobile Apps but cannot comprehensively represent the popularity of mobile Apps. Furthermore, such a single metric for measuring the popularity of mobile Apps can be easily confused by ranking

TABLE I: Characteristics of the data set.

Characteristics	Value
Users	13,969
Duration	545 days
Network types	2G/3G/4G WiFi
Applications	16,878
Total traffic	13.8 TB
Number of records	Over 103 million

fraud [1]. Therefore, we consider formulating the popularity of mobile Apps from multiple perspectives, including the number of active users, frequency of usage, usage time, and the amount of usage. Such incorporations can not only avoid the ranking fraud, but also tolerate with the quick change of user preferences with mobile Apps. For example, the game “flappy bird” suddenly popped and then fade out of sight in a short epoch. Simply measuring the popularity of the App by one metric cannot reflect its popularity in general, so we discuss four different metrics that affect the popularity of mobile Apps.

*The Number of Active Users.* One of the most common methods to evaluate the mobile Apps’ popularity is to compare their active users. We measure the number of active users in an epoch  $T$ . For a specific mobile App  $a_i$ , we count the number of users in  $T$ , denoted by  $n_i$ . The popularity ranking by the number of active users is simply by ordering  $n_i$  in a decreasing order. We use  $R_1 = \{a_1, a_2, \dots, a_m\}$  to represent the ranking, where  $a_i$  suggests a rank  $i$ th mobile App with respect to the number of active users.

*The Frequency of Usage.* The frequency of mobile App usage is considered as another factor to measure the popularity of mobile Apps. We measure the number of days in which the mobile App is launched in epoch  $T$  with a total number of  $t_n$  days. Given a mobile App  $a_i$ ,  $t_i$  represents the number of days in which  $a_i$  is launched. Then the frequency of  $a_i$  usage is formulated by  $f_i = t_i/t_n$ . The ranking of Apps is by decreasing order of the frequency usage. The ranking in terms of the frequency of usage is represented by  $R_2 = \{a_1, a_2, \dots, a_m\}$ , where  $a_i$  suggests a rank  $i$ th mobile App.

*The Usage Time.* The popularity of mobile Apps can also be evaluated by the usage time. The longer usage time suggests a more popular mobile App. We measure the usage time by the sum of time intervals in which an App is launched in epoch  $T$ . We define a time interval as the subtraction of the end time of the App usage and the start time of the App usage. Given a specific mobile App  $a_i$ , we use  $I_{ik} = e_{ik} - s_{ik}$  to denote the  $k$ th interval in which  $a_i$  is launched, where  $e_{ik}$  is the end time of the  $k$ th usage of App  $a_i$ , and  $s_{ik}$  is the start time of the  $k$ th usage by user  $i$ . The usage time of  $a_i$  is thus denoted by  $u_i = \sum_k I_{ik}$ . The ranking of Apps is by decreasing order of the usage time. We use  $R_3 = \{a_1, a_2, \dots, a_m\}$  to represent the ranking in terms of the usage time, where  $a_i$  suggests a rank  $i$ th mobile App.

*The Amount of Usage.* Finally, we also consider the amount of usage as one factor to affect the popularity of mobile Apps. The higher amount of usage suggests more popularity of the App. We count the number of launches in time epoch  $T$  for the measurement of the amount of usage. Given a mobile App  $a_i$ , we use  $c_i$  to denote the amount of usage of  $a_i$  in  $T$ . We use  $R_4 = \{a_1, a_2, \dots, a_m\}$  as the App ranking with respect to the amount of usage, where  $a_i$  suggests a rank  $i$ th mobile App.

Incorporating the above-mentioned four metrics, we formulate the popularity of mobile App by the average of the weighted rankings with respect to four metrics. It is denoted by

$$R(a_i) = \frac{1}{4} \sum_{k=1}^4 \omega_k * R_k, \quad (1)$$

where  $\omega_k$  is the weight for  $k$ th metric. After obtaining  $R(a_i)$  for all Apps, the popularity ranking is the increasing order of  $R(a_i)$ .

*3.3. Personal Preference.* Personal preference reflects the preference of different individuals to mobile Apps. As the sparsity of data, it always happens that many users do not have sufficient data trajectory to mine his or her App preference. Firstly, each system has many new users, and there is no data about these new users. Secondly, if a user goes to a new environment, their old data would be disabled. We investigate personal preference based on the distribution of common preference, as presented in [2]. Specifically, we investigate the common preference of all users and then represent each user’s preference by a distribution of common preference. If the common preferences are presented by  $\mathbf{z}$ , the conditional probability that a user  $u_m$  prefers the category  $\mathbf{a}$  of Apps given the set of all Apps  $\mathbf{A}$  can be represented as

$$P(\mathbf{a} | \mathbf{A}, u_m) = \frac{P(\mathbf{a}, \mathbf{A} | u_m) * P(u_m)}{P(\mathbf{A}, u_m)}. \quad (2)$$

For given Apps  $\mathbf{A}$  and user  $u_m$ ,  $P(u_m)$  and  $P(\mathbf{A}, u_m)$  are constant. Therefore, we have

$$\begin{aligned} P(\mathbf{a} | \mathbf{A}, u_m) &\sim P(\mathbf{a}, \mathbf{A} | u_m) \sim \sum_{\mathbf{z}} P(\mathbf{a}, \mathbf{A}, \mathbf{z} | u_m) \\ &\sim \sum_{\mathbf{z}} P(\mathbf{a}, \mathbf{A} | \mathbf{z}) P(\mathbf{z} | u_m), \end{aligned} \quad (3)$$

where the preference of user  $u_m$  to the category of Apps  $\mathbf{a}$  is determined by the common preferences of users (i.e.,  $P(\mathbf{a}, \mathbf{A} | \mathbf{z})$ ) and also his or her personal preference conveyed by a distribution of common preferences (i.e.,  $P(\mathbf{z} | u_m)$ ). The calculation of common preferences of other users and the distribution of common preferences of user  $u_m$  can be presented by the normalized number of Apps launched by user  $u_m$  and other users.

*3.4. MPT for App Recommendation.* Modern portfolio theory is a mathematical framework for assembling a portfolio of assets so that the expected return is maximized for a given level of risk. For instance, stock holders expect the maximum

of benefits and minimum of risks from the invested stocks. In the context of mobile App recommendation, we exploit the popularity of mobile Apps and personal preference of mobile users as the investment. The mobile device constraints such as residual power and network data plan are considered as the risk of the investment. For instance, a mobile device with low residual power has more risk to be terminated if it launches a mobile App with higher power consumption. Similarly, a mobile device with a low network data plan has more risk to be out of network if it launches a mobile App with higher network data consumption. For a mobile user, we consider the composition of  $n$  Apps as an App portfolio  $\Gamma$ , which consists of  $n$  Apps and the weight  $w_i$  for each App, is denoted as

$$\Gamma = \{(a_i, w_i)\}, \quad 0 \leq w_i \leq 1, \quad \sum_i w_i = 1. \quad (4)$$

The weight  $w_i$  is considered to be the degree that the recommendation system hopes users to give to the corresponding App  $a_i$ . The App recommendation ranking is established by learning the right weight to maximize the expected return of App investment portfolio (i.e., minimize the risk at the same time). Suppose the asset on App  $a_i$  is a random variable, denoted as  $r_i$ , and the user's investment for a portfolio is denoted as  $R$ ; then

$$R = \sum_i w_i r_i. \quad (5)$$

In the context of mobile App recommendation, the asset  $r_i$  is composed of the popularity of mobile Apps, user personalized preference, and mobile device constraints for mobile App  $a_i$ . The portfolio return is the weighted combination of the assets' returns, and the portfolio volatility is a function of the correlations of the assets for all pairs  $i$  and  $j$ . Therefore, we obtain the sum of variance as

$$\sigma^2 = \sum_i \sum_j w_i w_j \text{cov}_{ij}, \quad (6)$$

where  $\sigma$  is the standard deviation and  $\text{cov}_{ij}$  is the covariance between the assets of  $r_i$  and  $r_j$ . The portfolio with uncorrelated investments has lower volatility and thus is more likely to be recommended. The target of the MPT for mobile App recommendation is to look for a portfolio of mobile Apps satisfying the condition that no other portfolio of mobile Apps exists with a higher expected return but with the same standard deviation of return. Then the selection of the portfolio is to find the "efficient frontier" with minimum risk which is defined as

$$\min \mathbf{w}^T \Sigma \mathbf{w}, \quad (7)$$

where  $\mathbf{w}$  is a vector of portfolio weights,  $\Sigma$  is the covariance matrix for the returns on the assets in the portfolio of mobile Apps. The optimization solution is known as Markowitz's objective and has been solved in [12].

The framework can be implemented on either App Store servers or personal devices. If the framework is implemented on App Store servers, it collects the usage pattern and

constraints of mobile devices from users in an anonymized manner to preserve user privacy and then calculate the portfolio weights for each anonymized user. Then mobile Apps are ranked by the weights of mobile Apps for individuals to build the recommendation list. In the case in which mobile users do not want to reveal any privacy information to App Store, the framework can be implemented on the personal mobile device. The App Store servers only need to send the popularity of mobile Apps and the common preferences of mobile users to the mobile device, and the device computes the portfolio weights according to the popularity of mobile Apps and the common preferences acquired from App Store servers. Then mobile Apps are ranked locally based on the weights of mobile Apps. Indeed, the recommendation can not only be used to recommend new Apps to be downloaded, but also be used to recommend Apps that have been installed on the mobile device for launching.

#### 4. Use Cases and Performance Evaluation

In this section, we introduce two device constraints (e.g., residual battery power and network data plan) as use cases to show the incorporation of App recommendation framework with constraints. Accordingly, we also conduct the performance evaluation to show the effectiveness of the proposed recommendation framework.

*4.1. App Recommendation with Power Constraints.* Nowadays mobile devices are power consuming while the battery technique cannot fully satisfy the power requirement of mobile devices, and it becomes one severer constraint for the usage of mobile devices. In this subsection, we illustrate the way to combine the framework with the constraint of phone power.

In spite of the fact that the data set provides sufficient data for the formulation of popularity of mobile Apps and personal preference of mobile users, it lacks data for power consumption of different mobile Apps and effective method to quantify the power consumption of different mobile Apps.

For mobile devices, the power consumption varies in different statuses [13]. For instance, the power consumption for phone calls differs from the power consumption for WiFi connection. As we mainly focus on the power consumption on mobile Apps, we measure it with the minimum differences of other settings on the mobile device. For instance, we keep the network connection, display, voice, and so forth in the same configuration when we measure the power consumption of different mobile Apps.

The power measurement for different mobile Apps is difficult for the turbulence of power consumption as well as the large number of measurements. We recruit 2 students and use 3 months to measure the power consumption of over 700 mobile Apps. To guarantee the accuracy of the measurement, each App is measured 5 times and every time with 5 minutes. The distribution of the power consumption for all Apps is illustrated in Figure 1. In practice, the power consumption measurement can be conducted by the App provider or App Stores when the mobile App is ready for the public. It is once and for all work and will not bring too much burden to either App providers or App Stores.

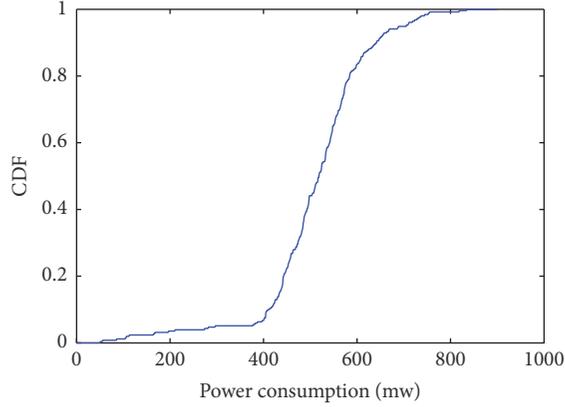


FIGURE 1: The cumulative distribution function (CDF) of power consumption.

After obtaining the power consumption of mobile Apps, we exploit the MPT framework for mobile App recommendation with power consumption constraints. Assume the power consumption of each time unit for App  $a_i$  is  $p_i$ . The mobile device has  $\mathcal{P}$  amount of residual battery power. The investment of each mobile App  $r_i$  is composed of three parts including the popularity of mobile Apps, the personal preference of mobile users, and the power consumption of Apps. Therefore, the investment of each mobile App is formulated as

$$r_i = \Delta_i^{-1} + \Theta_i^{-1} - q * \nabla_i^{-1}, \quad (8)$$

where  $\Delta_i$  is the popularity ranking of  $a_i$ ,  $\Theta_i$  is user's personal preference to  $a_i$ , and  $\nabla_i$  is the power consumption ranking of  $a_i$ . We use  $q \in (0, \infty)$  to denote the weight of power consumption for mobile recommendation. The higher value suggests power consumption takes a more important role for the mobile recommendation. Then the objective of MPT is to find the "efficient frontier" by

$$\begin{aligned} \min \quad & \mathbf{w}^T \Sigma \mathbf{w} \\ \text{s.t.} \quad & \sum_i p_i * t_i \leq \mathcal{P}, \end{aligned} \quad (9)$$

where  $t_i$  is the planned launch time of  $a_i$ . The objective is constrained by the power consumption in this case.

**4.2. App Recommendation with Network Data Constraints.** In spite of the fact that the development of mobile networks is dramatically improving the network supply for the usage, mobile users still have a higher demand on network traffic owing to the large volume network consumption from mobile Apps. We discuss the mobile App recommendation restricted by the network data plan.

We first measure how much traffic different mobile Apps need based on the data set, which provides the network download and upload speed. It allows calculating the average network traffic consumption of each mobile App in a time unit. For mobile App  $a_i$  and its  $k$ th launch period  $t_{ik} = e_{ik} - s_{ik}$  with  $s$  records in the data set, where  $e_{ik}$  is the end time of

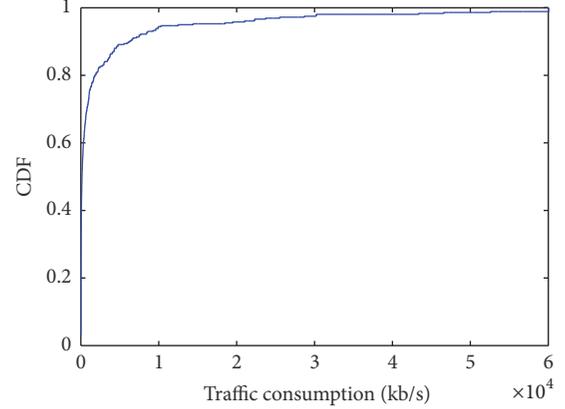


FIGURE 2: The cumulative distribution function (CDF) of network data consumption.

the  $k$ th launch and  $s_{ik}$  is the start time of the  $k$ th launch. The average flow rate of  $a_i$  for the  $k$ th launch is calculated by

$$m_{ik} = \frac{\sum_{j=1}^s (m_{uj} + m_{dj}) * t_{ik}/s}{t_{ik}}, \quad (10)$$

where  $m_{uj}$  and  $m_{dj}$  are the upload and download speed in the record. The average flow rate of  $a_i$  then takes the average of all launches and is denoted as  $m_i$ . The distribution of network data consumption by mobile Apps is shown in Figure 2. In practice, similar to the power consumption measurement, the traffic consumption of different mobile Apps can be measured by App providers or App Stores before the App is issued to the public.

For network constraint with residual data plan  $\mathcal{M}$ , we exploit MPT for mobile App recommendation. Then the investment of each mobile App  $r_i$  is composed of the popularity of mobile Apps, the personal preference of mobile users, and the flow consumption of Apps. Therefore, the investment of each mobile App is formulated as

$$r_i = \Delta_i^{-1} + \Theta_i^{-1} - x * \Upsilon_i^{-1}, \quad (11)$$

where  $\Upsilon_i$  is the flow consumption ranking of  $a_i$ . Similar to power consumption, we use  $x \in (0, \infty)$  to denote the weight of flow consumption for mobile recommendation. The higher value of  $x$  indicates flow consumption takes a more important role in the mobile recommendation. Then the objective of MPT is to find the "efficient frontier" that is constrained by the flow consumption.

**4.3. Performance Evaluation.** In this section, we show the effectiveness of the mobile App recommendation framework by incorporating popularity of mobile Apps, personal preference of mobile users, and the power and network constraints of mobile devices.

**4.3.1. The Popularity of Mobile Apps.** We measure the ranking of mobile Apps by different metrics, to show the differences of popularity ranking in terms of the number of active users, the frequency of usage, the usage time, and the amount of usage.

TABLE 2: The popularity ranking by different metrics.

Ranking	Active users	Frequency	Time	Usage	Comprehensive
1	FMI Zhangyue ireader	FMI Content scripts	Content scripts	FMI	FMI
2	—	—	FMI	Content scripts	Content scripts
3	Content scripts	QQ reading	Easou novel	Easou novel	Baisi Budejie
4	Baisi Budejie Easou novel LRTS QQ reading Shuqi novel Zhihu	Reading bus	Baisi Budejie	Baisi Budejie	Easou novel
5	—	Douban reading	Reading bus	Reading bus	QQ reading
6	—	Shuqi novel	Douban reading	Douban reading	Reading bus
7	—	Baisi Budejie	QQ reading	Baozou cartoon	Zhangyue ireader
8	—	Zhihu Zhangyue ireader	Cosme	Secret	Douban reading
9	—	—	Baidu reading	QQ reading	Zhihu
10	Baozou cartoon He reading Reading bus Tyread	Screte	Zhihu	Bilibili	He reading
11	—	Easou novel	He reading	Baidu reading	LRTS
12	—	LRTS He reading Cosme NetDragon	JDNDSE	He reading	Shuqi novel
13	—	—	Zhangyue ireader	Cosme	Baozou cartoon
14	Bilibili Duokan reading Secret Taobao reading	—	Tyread	Zhihu	Cosme
15	—	—	LRTS	LRTS	Secret
16	—	Tyread	Baozou cartoon	Zhangyue ireader	Tyread
17	—	Baozou cartoon Idduu	Shuqi novel	Tyread	Baidu reading
18	Baidu reading Cosme Douban reading Idduu JDNDSE NetDragon	—	Bilibili	Shuqi novel	Bilibili
19	—	Secret	Idduu	JDNDSE	JDNDSE
20	—	Bilibili Taobao reading	Duokan reading	Duokan reading Taobao reading	NetDragon
21	—	—	NetDragon	—	Taobao reading
22	—	Duokan reading JDNDSE	Taobao reading	NetDragon	Duokan reading
23	—	—	Secret	Idduu	Idduu

Table 2 shows the popularity ranking by different metrics for a group of reading Apps. The ranking differs from one metric to another. For instance, the App “QQ reading” ranks 4th by the number of active users; it becomes 8th ranking when considering the frequency of usage. In the perspective

of the usage time, “QQ reading” ranks 7th, but by considering the amount of usage, it ranks 9th. Such different rankings by different types of metrics also appear in other mobile Apps. The popularity ranking is hard to determine by a single metric. By our combined comprehensive metric, “QQ reading”

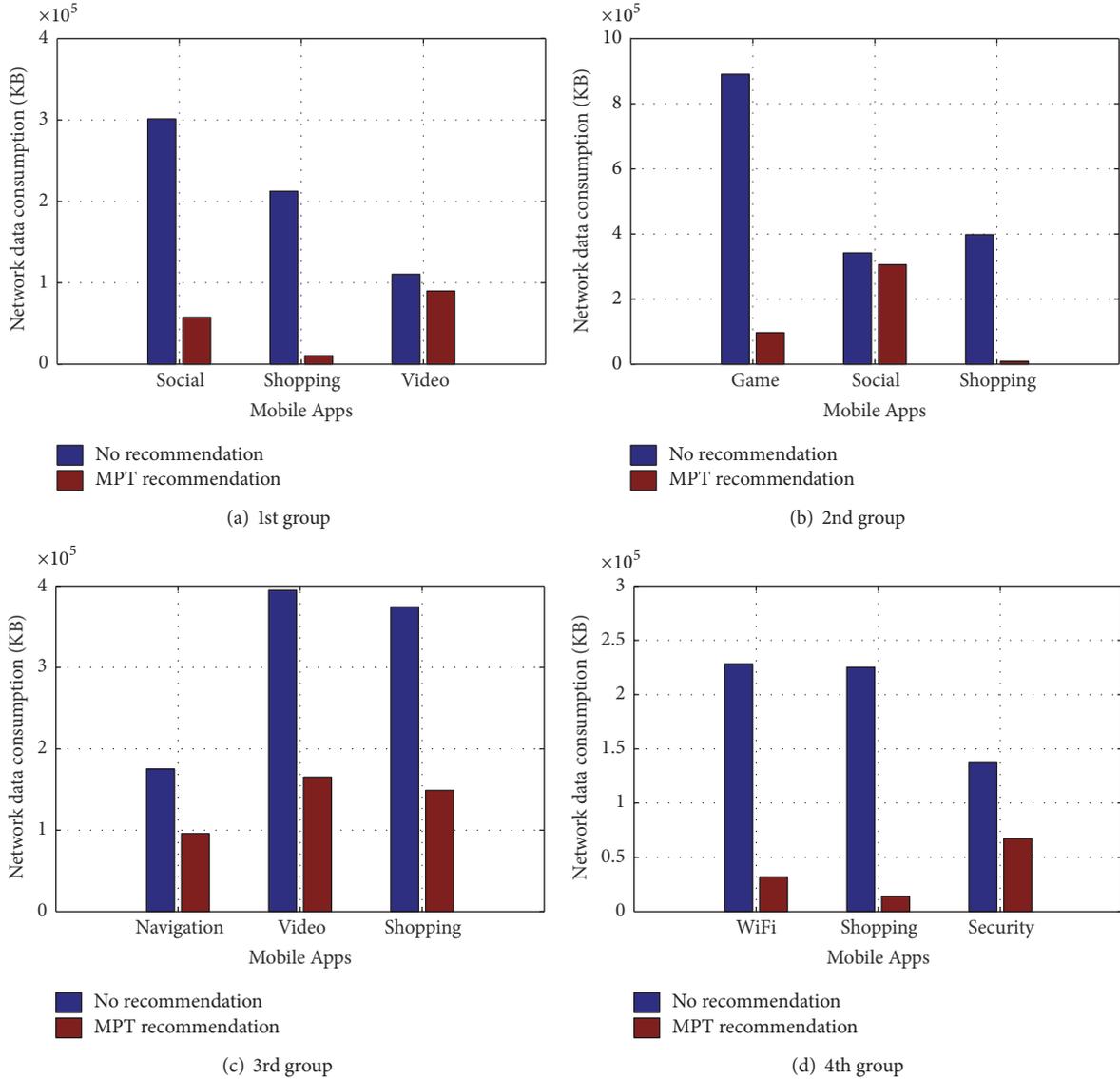


FIGURE 3: The network data consumption comparison before and after recommendation.

shows a rank of 5. By observation of Table 2, it shows that the comprehensive metric not only combines different features of metrics, but also provides a distinguishable ranking list.

**4.3.2. Recommendation Results.** We show the mobile App recommendation framework with power and network data constraints saving battery power and network data dramatically.

For the experiments, we put the users into four groups by their mobile App usage patterns and then measure the power and network data consumption before and after recommendation. Figure 3 shows the network data consumption of four groups of users before and after recommendation. Due to the limitation of space, we only show the first 3 types of recommended Apps. First of all, different groups have varied preferences resulting in the fact that their recommended Apps vary from each other. With respect to the network data consumption, the recommended mobile Apps help to

reduce the network data consumption. In particular, the first group of users are given recommendation of mobile Apps for social, shopping, and video, as shown in Figure 3(a). The recommendation reduces more than 70% for social Apps and more than 90% for shopping Apps, while it only reduces 10% of network data for videos. Similarly, the recommendation reduces the network data for the other groups of users as well. It shows the effectiveness to overcome the mobile device constraints.

Figure 4 shows the power consumption of four groups of users before and after recommendation. Similar to network data constraint, different groups have varied preferences resulting in the fact that their recommended Apps vary from each other. With respect to power consumption, the recommended mobile Apps help to reduce the power consumption compared with mobile Apps before recommendation. In particular, Figure 4(a) shows the power consumption comparison before and after recommendation. It presents the

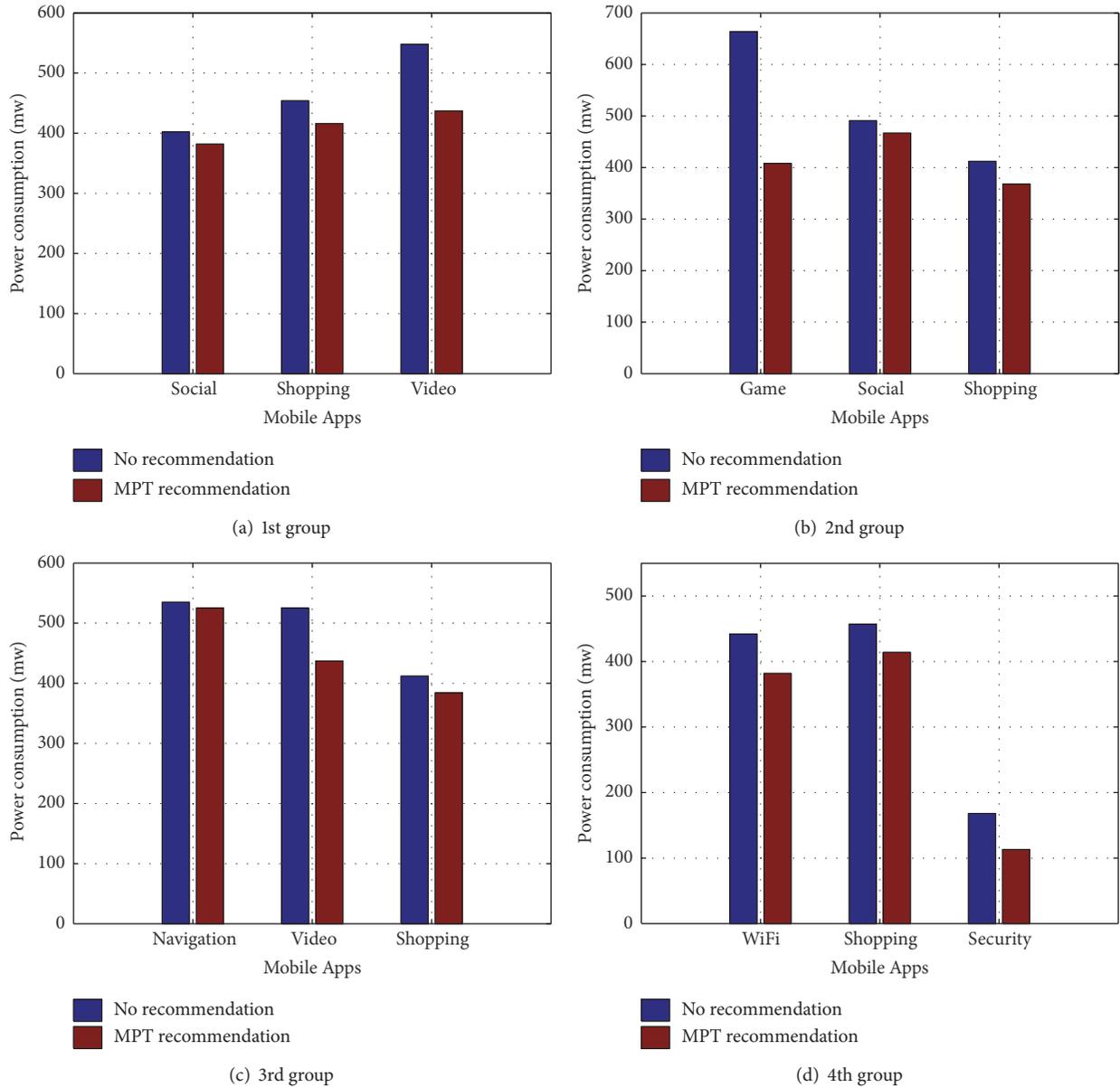


FIGURE 4: The power consumption comparison before and after recommendation.

fact that the mobile App recommendation framework with power constraint reduces about 10% of power consumption for social and shopping Apps. It reduces about 20% of power consumption for video Apps. The similar phenomenon can also be found in the other groups of users where the mobile App recommendation approach reduces the power consumption of mobile devices. Therefore, the mobile App recommendation framework with power constraint is effective for mobile App recommendation with power constraints.

Comparing Figures 3 and 4, although the preferences for the same group of people are the same, different recommendation results subject to power and network constraints result in the consumption of network data and power quite differently. The observation shows that the gap of power consumption between recommended Apps and Apps before

recommendation is smaller than that of network data consumption between recommendation Apps and Apps before recommendation. We believe this is mainly attributed to the fact that the basic power consumption (e.g., screen) takes the dominant role in the whole power reduction.

## 5. Conclusion

We propose a mobile App recommendation framework by incorporating the popularity of mobile Apps and the user preference of mobile users as well as the mobile device capacity constraints. We formulate the popularity of mobile Apps by a combination of four different metrics to comprehensively present those different features of the metrics. Then we formulate the personal preference to show the personalized

characteristics of mobile users. To illustrate the effectiveness of the proposed mobile App recommendation framework, we use two use cases by considering power consumption and network data consumption as the constraints of mobile devices. Extensive evaluation results show that the proposed mobile App recommendation framework can reduce traffic data up to 90% and reduce over 30% of power consumption.

## Conflicts of Interest

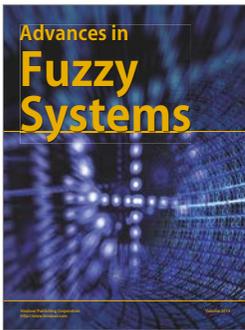
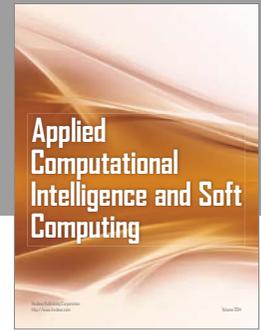
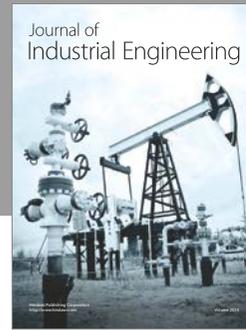
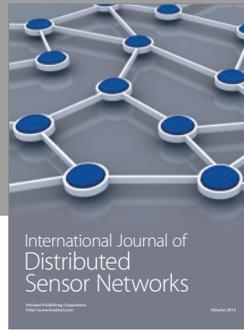
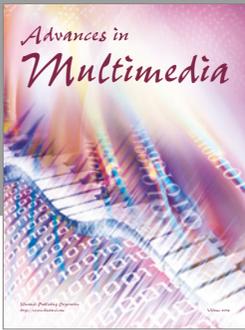
The authors declare that they have no conflicts of interest.

## Acknowledgments

This work has been partially sponsored by National Natural Science Foundation of China (no. 61502045), the National Key Research and Development Program of China (2016YFB0100902), and the III Project (no. B08004).

## References

- [1] H. Zhu, H. Xiong, Y. Ge, and E. Chen, "Discovery of ranking fraud for mobile apps," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 1, pp. 74–87, 2015.
- [2] H. Zhu, E. Chen, H. Xiong, K. Yu, H. Cao, and J. Tian, "Mining mobile user preferences for personalized context-aware recommendation," *ACM Transactions on Intelligent Systems and Technology*, vol. 5, no. 4, pp. 1–27, 2014.
- [3] B. Liu, D. Kong, L. Cen, N. Z. Gong, H. Jin, and H. Xiong, "Personalized mobile app recommendation: reconciling app functionality and user privacy preference," in *Proceedings of the 8th ACM International Conference on Web Search and Data Mining (WSDM '15)*, pp. 315–324, ACM, Shanghai, China, February 2015.
- [4] B. Yan and G. Chen, "AppJoy: personalized mobile application discovery," in *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services (MobiSys '11)*, pp. 113–126, ACM, New York, NY, USA, July 2011.
- [5] J. Wang and J. Zhu, "Portfolio theory of information retrieval," in *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '09)*, pp. 115–122, ACM, Boston, Mass, USA, July 2009.
- [6] P. Pu, L. Chen, and R. Hu, "Evaluating recommender systems from the user's perspective: survey of the state of the art," *User Modeling and User-Adapted Interaction*, vol. 22, no. 4-5, pp. 317–355, 2012.
- [7] V. W. Zheng, B. Cao, Y. Zheng, X. Xie, and Q. Yang, "Collaborative filtering meets mobile recommendation: a user-centered approach," in *Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI '10)*, Atlanta, Ga, USA, July 2010.
- [8] K. Zhu, X. He, B. Xiang, L. Zhang, and A. Pattavina, "How dangerous are your Smartphones? App usage recommendation with privacy preserving," *Mobile Information Systems*, vol. 2016, Article ID 6804379, 10 pages, 2016.
- [9] Y. Ge, Q. Liu, H. Xiong, A. Tuzhilin, and J. Chen, "Cost-aware travel tour recommendation," in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '11)*, pp. 983–991, San Diego, Calif, USA, August 2011.
- [10] B. Liu, M. S. Andersen, F. Schaub et al., "Follow my recommendations: a personalized privacy assistant for mobile app permissions," in *Proceedings of the Symposium on Usable Privacy and Security*, 2016.
- [11] G. Shani, D. Heckerman, and R. I. Brafman, "An MDP-based recommender system," *Journal of Machine Learning Research*, vol. 6, no. 1, pp. 1265–1295, 2005.
- [12] W. Zhang, J. Wang, B. Chen, and X. Zhao, "To personalize or not: a risk management perspective," in *Proceedings of the 7th ACM Conference on Recommender Systems (RecSys '13)*, pp. 229–236, ACM, Hong Kong, October 2013.
- [13] A. Carroll and G. Heiser, "An analysis of power consumption in a smartphone," in *Proceedings of the USENIX Annual Technical Conference*, vol. 14, p. 21, Boston, Mass, USA, June 2010.



# Hindawi

Submit your manuscripts at  
<https://www.hindawi.com>

