International Conference on Cyber Behavior,
June 18-20, 2014, Taipei, Taiwan

MINING SEQUENTIAL APPLICATION USAGE PATTERNS WITH TIME CONSTRAINT OF SMARTPHONE USERS

Woan-Rou Tseng,

Kuo-Wei Hsu,
National Chengchi University, Taiwan
hsu@cs.nccu.edu.tw

ABSTRACT

Smartphones are mobile phones with extended functionality and have brought changes to our daily lives. The main feature of smartphones is that the developers can create applications that users can use to personalize their smartphones. As both the number of smartphone users and the number of available smartphone applications increase, it is practically important to study how users use the applications on their smartphones. Thus, we apply a sequential pattern mining technique to a set of smartphone logs collected from several users over several months. This paper presents a data preprocessing process that transforms the logs to sessions. It also presents an algorithm designed to mine sequential patterns with constraint on the maximum time interval between applications used in a session. It further presents interesting patterns generated by using different time intervals. This paper contributes to behavior analysis of smartphone users by providing an approach used to discover patterns from smartphone logs and by providing a discussion about the discovered patterns. What presented in this paper could be beneficial to the developers of smartphone applications.

Keywords: Sequential Pattern Mining, Smartphone Application, Smartphone Usage

1. INTRODUCTION

We have seen rapid development of smartphone technology during the past few years. Also we have seen that many kinds of applications, or apps, which can be installed by users and run on smartphones, appear on the market. A smartphone is no longer just a mobile device for voice communication but it could also be, a digital camera, a digital camcorder, a media player, a gaming device, an Internet surfing tool, and a note taking tool. As the number of smartphone apps is increasing, the research attention is certainly increasing and so are the research issues. The major research issue addressed in this paper is behavior analysis of smartphone users, that is, analysis of how users use apps on their smartphones.

There are studies relevant to behavior analysis of smartphone users. However, most of those studies are based on questionnaire surveys or experiments conducted in controlled environments, and thus the results obtained from those studies may not be able to reflect the real behaviors of smartphone users. The questionnaires may possibly be designed under some assumptions that are too strong to be valid in real situations, and the data used in the controlled experiments may possibly be collected only for a short period of time.

The classic association rule mining algorithm cannot be directly applied to the smartphone logs, since the algorithm is designed to mine items relatively frequently appearing together in a transaction but intrinsically there are no transactions defined in the smartphone logs. We take advantage of the concept of transaction, and we introduce the concept of *usage session*, or simply session, which is a time interval in which a user uses apps continuously on his or her smartphone. If we directly apply the classic association rule mining algorithm to sessions, we will discover rules indicating what apps would be frequently used together in a session. However, such rules are not very interesting to us, because we would like to know the order of apps used in a session. Neither can the traditional sequential pattern mining algorithm be directly applied to the smartphone logs, since the algorithm is designed to mine items appearing in sequence among most transactions. If we directly apply the traditional sequential pattern mining algorithm to sessions, which are treated as transactions, we will discover patterns indicating that some app use in a session would be more likely followed by another app use in the successive session. Such patterns present the order of app uses between sessions rather than in a session. Thus, such patterns would not be very meaningful.

In this paper, we present our study in which we use a sequential pattern mining technique to analyze a set of Android-based smartphone logs recording activities of several users over a long period of time. The goal of our study is to discover the users' smartphone app usage patterns that could help improve the design of smartphone apps. The results of our study contribute to a better understanding of the real behaviors of smartphone users.

The rest of the paper is organized as follows: Section 2 gives the background of the work presented in this paper. Section 3 describes how we apply a sequential pattern mining technique to a real data set, reports the experimental results, and discusses patterns observed from the results. Section 4 concludes this paper and indicates the future work.

2. BACKGROUND

If we consider rapid growth of the smartphone industry, we can see an urgent need for more investigations and studies on smartphones. Among the research topics of the related studies, behavior analysis of smartphone users is an important topic. The goal of the topic is to help researchers have a better understanding of what apps users install and use on their smartphones and how they use these apps. In the paper [1], Chittaranjan, Blom, and Gatica-Perez studied the discovery of personality traits of smartphone users by applying data mining and models to GPS data, call logs, and Bluetooth data. The usage data of smartphones can provide valuable information. In the paper [7], Xie et al. studied the detection of devices infected by malwares by analyzing process state transitions and user operational patterns. In the paper [4], Burguera, Zurutuza, and Nadjm-Tehrani studied the identification of apps containing malware by using a crowdsourcing system to collect the traces of behaviors of applications. In the paper [3], Verkasalo studied a framework for mobile audience measurements that can directly collect data from mobile devices rather than indirectly collect data from user surveys and that can be used to help perform statistical analysis on the collected data. In the paper [10], Xu et al. studied when, where, and how smartphone apps were used by the users in the U.S. by using anonymized network measurements. In the paper [2], Falaki et al. studied the relationship between intentional user behaviors and the network traffic and the power consumption by analyzing the time spent by each user on each of popular apps during every hour of the day. In the paper [5], Kang, Seo, and Hong studied the relationship between usage patterns and battery consumption of smartphones by analyzing the usage data of smartphones collected from some users over a two month period and show that all users have their own usage pattern. The above studies show that the results obtained from analyzing app usage patterns could be beneficial to the network and power management of smartphones.

Given a transaction database in which each transaction contains items purchased by a customer, the association rule mining problem, which was first introduced by Agrawal in the early 1990's [13], concerns how we can efficiently mine or discover rules that indicate sets of items frequently purchased together in a transaction. A set of items is called itemset, and if there are *k* items in a set then we call it a *k*-itemset. In the problem, *support* is the percentage of transactions in the database that contain an itemset of interest. A higher *support* value means a higher percentage of the transactions in which customers purchase an itemset of interest. In the paper [11], Agrawal and Srikant proposed the classic association rule mining algorithm, the Apriori algorithm, to generate association rules from large transaction databases. There are two phases in the Apriori algorithm. First, it generates candidate itemsets by calculating the *support* values of the itemsets. The itemsets whose *support* values are larger than a threshold value, *minimum support*, are called frequent itemsets, and only frequent itemsets will be added as part of candidate itemsets. Second, the Apriori algorithm generates the candidate *k*-itemsets from the frequent *(k-1)*-itemsets by using the property stating that every *(k-1)*-itemsets of the frequent *k*-itemsets must be frequent. The algorithm is efficient in its way to generate candidate itemsets and in its way to remove infrequent itemsets. Association rule mining does not take in account the order of items purchased in a transaction. The rule A → B indicates that if we find item A in a transaction then we would probably find item B in the same transaction. However, we have no idea what item, A or B, would be more likely purchased first. In our study, we are interested in answering the question like "If a user uses a smartphone app, then what would probably be the app that he or she would use next?" Thus, we cannot directly apply the classic association rule mining algorithm to behavior analysis of smartphone users in our study.

Items purchased by a customer are in a transaction, and transactions made by a customer are in a sequence. Given a database of sequences, the sequential pattern mining problem, which was first introduced by Agrawal and Srikant in the mid 1990's [12], concerns how to efficiently generate patterns that indicate the sequences of items that were purchased by a relative large number of customers. For example, a sequential pattern indicates that if customers purchase an item A in their current transactions then, relatively speaking, they would probably purchase an item B in their next transactions. One of the applications of sequential pattern mining is to discover the users' navigation patterns from Web log data. In the paper [14], Iváncsy, and Vajk studied the application of three mining methods to mining the usage data of Web. Several algorithms were proposed to address the sequential pattern mining problem. Mabroukeh and Ezeife reviewed some of these algorithms in the paper [8]. Unlike many other studies on these algorithms whose focus is on improving performance in execution time and memory consumption, the focus of our study is on mining the interesting usage patterns. We design and implement a data preprocessing process to prepare data for the mining task, and further we extend a well-known sequential pattern mining algorithm by taking into account constraint on time intervals between app uses. PrefixSpan, which was proposed by Pei et al. in the early 2000's [6], is a sequential pattern mining algorithm using the pattern growth approach. PrefixSpan divides every sequence into prefix and suffix, and it projects database by evaluating the frequent prefixes. This process runs recursively, and the pattern becomes one item longer after each run. The use of prefix and suffix makes the generation of candidate sequences unnecessary, and the projected databases could become smaller as this process proceeds.

3. THE MINING METHOD

3.1 Data Preprocessing

The set of smartphone logs used in our study is collected in the same way used by Chen et al. in the paper [9]. The logs contain the daily app uses made by 25 users on 26 devices (smartphones) from September 2010 to March 2011. Every log entry includes information about who uses what app, where and when.

A session is a sequence of apps used by a user on a smartphone in a time interval. However, we cannot identify a session of a user directly from the logs in the raw format. A session means a series of app uses, but they are not recorded as a group in the logs. We need to do data preprocessing, which includes cleaning and transformation. First, we remove incomplete log entries. Second, we transform the logs to sessions by using the following definition: A session *s* contains all the log entries corresponding to the same user-machine pair, and the difference in time between two continuous log entries in *s* does not exceed 10 minutes. According to the definition, a session can be longer than 10 minutes, but no two app uses in a session can be far away from each other for more than 10 minutes. We use 10 minutes as the time gap between sessions by referring to the paper [9].

Initially, there are 262,858 log entries. After the data preprocessing process is done, there are 25,880 sessions. There are 1,132 packages or apps in these sessions. Many apps were installed and used by the users, but these apps were not used equally. The users used some apps much more often than others. The support of an app is proportional to the frequency that the app use is recorded in the database. We set the minimum support to 0.5% and we use it to filter out apps that are used less frequently. As a result, there are 33 apps whose support values are larger than 0.5%. Among these 33 apps, which are used more frequently than others, three are launchers. In our study and in this paper, we consider these apps but exclude launchers because the main task of a launcher is to launch other apps and a launcher is used when a user presses the Home button. Thus, it is not surprising to find that launchers are frequent apps. We are more interested in finding patterns that do not contain launchers but other 30 frequent apps.

3.2 Sequential Pattern Mining with Time Constraint

Association rule mining is to find co-occurrence relationship of items in a transaction but not to find the order of apps appearing in a transaction. Sequential pattern mining is to find the order of apps appearing from a transaction to the next one. However, what we want to find is the order of apps appearing in a session. Thus, we need a method that can find patterns in each of which items appear in a certain order and are in the same session. We develop the method by referring to the implementation of PrefixSpan provided by SPMF (Sequential Pattern Mining Framework) [15].

We not only record the order of app uses but also store the timestamps of app uses in the corresponding session entry. If the time difference between two consecutive log entries that belong to the same user-machine pair is in 10 minutes, the app uses corresponding to these two log entries are considered more relevant and are combined into the same session. If two consecutive log entries are away from each other for more than 10 minutes, the app uses corresponding to them are considered less relevant and are divided into different sessions. We assume that the relationship between two app uses in the same session is weak if the corresponding timestamps are away from each other for more than a pre-specified threshold or time constraint.

If two consecutive log entries that belong to the same session are corresponding to two uses of the same app, we do not count the second log entry but record the timestamp of the first log entry in the session entry. This could mean that the user keeps using different functions of the same app in the session.

Unlike the traditional sequential pattern mining algorithm that groups transactions by users and aims at finding inter-transaction relationships, our method does not group sessions and aims at finding intra-session relationships. If a user uses Facebook in the morning and Camera in the afternoon, we usually do not think that these two app uses belong to the same session. These two uses of these two apps may be relevant, but we think that it makes more sense to treat them as two app uses in two sessions.

PrefixSpan is based on the technique of pattern growth, one of major techniques used in sequential pattern mining. The key idea of the technique is to avoid the candidate generation step and to focus on searching a smaller portion of the given database. We extend PrefixSpan due to the following reasons: It is designed for dense databases, and our database is composed of the most frequently used apps and hence is dense. It recursively projects the databases of the frequent prefixes that are generated based on the suffixes, and it adds items to a pattern one at a time and hence allows us to check extra types of constraint when growing patterns. In SPMF, the implementation of PrefixSpan adopts pseudo projection and hence it is efficient. Pseudo projection represents a suffix by a pointer-offset pair in order to avoid constructing a physical projection database. If a projected database can be entirely loaded into main memory, the cost of projection can be greatly reduced.

Some log entries may be more relevant to each other in a session, and this implies that there is a stronger relationship between these app uses. We extend PrefixSpan such that it considers constraint on the maximum time interval between applications used in a session. We have the following assumption: Two app uses can be considered closely relevant to each other if the difference in the timestamps of their corresponding log entries is smaller than a pre-specified threshold or time constraint, 30 seconds for example. When projecting databases, our extension of PrefixSpan only projects data corresponding to the situation in which the difference between the timestamp of an app use and the timestamp of its previous app use is smaller than the pre-specified threshold or time constraint, 30 seconds for example. If an app use does not satisfy pass the threshold, data corresponding to all app uses following that app should not be projected because they are all considered irrelevant to the previous app use.

We are given a database storing sessions and each session contains an ordered list of app uses. The workflow of the mining method used in our study is described below:

1. Calculate the support value for each app and find apps that are frequently used by finding apps whose support values are larger than the pre-specified minimum support.
2. For the growing pattern, which initially contains no app, do the following:
3. For each frequent app, if the time difference between the app and the last app appearing in the growing pattern is smaller than the pre-specified or time constraint, do the following:
4. Append the frequent app to the growing pattern, and calculate the support value of the new pattern.
5. If the support value is larger than the pre-specified minimum support, do the following:
6. Add the new pattern generated in the last step to the result set, which is a set of sequential patterns.
7. Use the new pattern as the prefix, and do prefix-based projection to have a database in which each session contains only the postfix.
8. Use the new database and the new pattern as the input, and recursively call the function.

In the next section, we report as well as discuss results that we obtain when setting that minimum support to 0.5%.

3.3 Experimental Results

The typical scenario in which people use smartphones is the scenario in which people make phone calls. We report the support values for the sequential patterns regarding the use of the app Phone in Table 1, in which we report support values generated by using different values of the pre-specified threshold or time constraint. The actions such as dialing number as well as opening contacts are highly relevant to making phone calls. These actions can be done quickly. According to the first pattern in Table 1, users use Htcdialer and then Phone in about 6.4% of sessions in five seconds. According to the second pattern in Table 1, users use Htccontacts and then Phone in about 0.7% of sessions in five seconds. It is surprising to find that, when some users want to make a phone call, they prefer dialing the number directly rather than using the contacts list. In about 9.6% of sessions, users use Phone after using Htcdialer within three minutes, and in about 2.0% of sessions, users use Phone after using Htccontacts within three minutes. This may suggest the developers to study in depth the users' satisfaction with the user interface of the contact list.

Table 1. Support values for sequential patterns regarding Phone

Sequential Pattern \ Time constraint in seconds	5	30	60	180
Htcdialer → Phone	6.4%	9.2%	9.4%	9.6%
Htccontacts → Phone	0.7%	1.5%	1.7%	2.0%

In Table 2, we report the support values for the sequential patterns regarding the use of the app GMail. Email is also an important function of smartphones. Surprisingly, email apps such as GMail are not used for a long period of time. After about one minute, users begin to use another app. This may imply that users use email apps on smartphones for quickly checking emails and making short replies. In 2.3% of sessions, users use Facebook within three minutes after they use Gmail.

Table 2. Support values for sequential patterns regarding GMail

Time constraint in seconds / Sequential Pattern	5	30	60	180
GMail → Browser	-	0.7%	1.0%	1.5%
GMail → Facebook	-	1.2%	1.7%	2.3%
GMail → GTalk	-	-	0.6%	0.8%
GMail → WhatsApp	-	0.6%	0.9%	1.3%

Since almost all smartphones are equipped with camera and many photography apps are available, more people use smartphones as point-and-shoot digital cameras. We report the support values for the sequential patterns regarding the use of the app Camera in Table 3. Taking photo can be done quickly. Using Camera and then Album can be done in about 0.6% of all sessions in five seconds. However, the support for the pattern indicating that users use Camera and then Facebook is not larger than the minimum support, 0.5%, when the time constraint is set to five seconds. This may be because Camera provides a direct link to Album but there was no such a direct link when the logs were collected. Using Camera and then Album can be done in about 1.9% of all sessions in three minutes, and using Camera and then Facebook can be done in about 1.2% of all sessions in three minutes. If we consider all the possible combinations of uses of apps, we would not think such support values low. Table 3 shows how uses love to use their smartphones to take photos and share photos through Facebook, which is probably the most popular social network service.

Table 3. Support values for sequential patterns regarding Camera

Time constraint in seconds / Sequential Pattern	5	30	60	180
Camera → Album	0.6%	1.2%	1.6%	1.9%
Camera → Facebook	-	0.6%	0.9%	1.2%

We report the supports for the patterns relevant to Facebook in Table 4. From the column for the time constraint of five seconds, we can see that the supports for the patterns shown in Table 4 are not larger than the minimum support. From the column for the time constraint of 30 seconds and the column for the time constraint of 60 seconds, we can see that we can see that the supports for most of the patterns are not larger than the minimum support. This implies that users usually use Facebook for one or two minutes before they switch to another app. There are two patterns indicating that users use Facebook in a repeated way (rather than a continuous way). In about 2.6% of all sessions, after using Facebook, users would use it again within three minutes. In about 0.9%, after using Facebook, users would use Facebook for the second time within three minutes, and then users would use Facebook for the third time within three minutes. Since we consider only 30 frequent apps, there may be the uses of other apps between these uses of Facebook. Nevertheless, these results just show how much users are addicted to Facebook. The support values shown in Table 4 seem low, but it they may not be so if we consider all the possible combinations of uses of apps. We can see that the use of Facebook appears in many sessions if we add the support values up together. Facebook serves or is used as the center of social connections, since users use Facebook and then GTalk, Plurk, and WhatsApp, in about 1.1%, 0.7%, and 0.8% of all sessions in three minutes, respectively. The use of Facebook would trigger the use of Browser, Camera, and Phone within three minutes in about 1.8%, 0.8%, and 1.5% of all sessions. The results suggest that Facebook has the potential to be the center of the use of smartphone. Remember that the time frame for when the logs were collected was from September 2010 to March 2011. In early 2013, Facebook Inc. released Facebook Home, a product specific to Android-based smartphones[1]. The main feature of the product is to integrate Facebook with Home Screen, and an important feature of the product is to help users launch apps. The results that we find from our study make the release of Facebook Home not surprising.

Table 4. Support values for sequential patterns regarding Facebook

Sequential Pattern \ Time constraint in seconds	5	30	60	180
Facebook → Browser	-	0.7%	1.2%	1.8%
Facebook → Browser → Facebook	-	-	-	0.8%
Facebook → Camera	-	-	-	0.8%
Facebook → Facebook	-	0.8%	1.5%	2.6%
Facebook → Facebook → Browser	-	-	-	0.6%
Facebook → Facebook → Facebook	-	-	-	0.9%
Facebook → GTalk	-	-	0.7%	1.1%
Facebook → Phone	-	-	0.7%	1.5%
Facebook → Plurk	-	-	-	0.7%
Facebook → WhatsApp	-	-	-	0.8%

In Table 5, we report the support values for the sequential patterns regarding the use of the app WhatsApp, one of the popular instant messaging apps. From Table 5, we can see that users use WhatsApp and then Facebook in about 1.0% of sessions in three minutes. When using communication apps such as WhatsApp, users usually spend 30

[1] http://en.wikipedia.org/wiki/Facebook_Home

to 60 seconds before switching to another app. We find patterns indicating two, three, and even four repeated uses of WhatsApp. This may reflect users' high retention on WhatsApp. In 2.7% of sessions, users use WhatsApp for a conversation and then use it for another conversation within three minutes. Users spend more time on actions when using apps for social network services such as Facebook. Each of users' uses of such apps can last for several minutes before users use another app. Most of the support values of patterns shown in Table 5 are not large enough when the pre-specified threshold or time constraint is not larger than 1 minute.

Table 5. Support values for sequential patterns regarding WhatsApp

Time constraint in seconds / Sequential Pattern	5	30	60	180
WhatsApp → Facebook	-	-	0.6%	1.0%
WhatsApp → WhatsApp	-	0.8%	1.5%	2.7%
WhatsApp → WhatsApp → WhatsApp	-	-	-	1.3%
WhatsApp → WhatsApp → WhatsApp → WhatsApp	-	-	-	0.7%

Most smartphones are equipped with GPS (Global Positioning System) capability, and some people use smartphones as digital maps or even navigation devices. In Table 6, we report the support values for the sequential patterns regarding the use of the app GoogleMaps. When users use GoogleMaps, they tend to keep using it for several minutes. This may be explained by the situation in which users read a map on screen for directions while walking and finding their ways or points of interests. Such a situation shows a relatively longer period of time for the use of GoogleMaps. It is understandable that users may switch between browser or communication apps for address information or directions while using GoogleMaps, but it is not very common according to the results obtained from our study.

Table 6. Support values for sequential patterns regarding GoogleMaps

Time constraint in seconds / Sequential Pattern	5	30	60	180
GoogleMaps → Facebook	-	-	-	0.5%
GoogleMaps → GoogleMaps	-	-	-	0.7%

4. CONCLUSIONS AND FUTURE WORK

In this paper, we presented our work on the mining method for the analysis of a set of smartphone logs that are collected from 25 users for about seven months. We discussed why the classic association rule mining algorithm and the traditional sequential pattern mining cannot be directly applied to the data. We presented a data preprocessing process that performs data cleaning and data transformation, and we also presented an algorithm used to mine sequential patterns with constraint on the maximum time interval between app uses in a usage session. One of the contributions of this paper is to discover interesting patterns hidden in the data. The discovered patterns could be beneficial to the designers of smartphone apps. The patterns suggest that the functions of camera apps could be improved for better browsing, viewing, and editing photos as well as for better integrating with social network services such that the users do not need to spend more time than necessary on switching apps. The patterns also suggest that social apps could add more photo-taking and photo-editing functions.

Our future work includes the following: First, we plan to analyze the discovered patterns even more. Second, we plan to investigate other mining methods in order to discover more interesting patterns. Third, we plan to analyze how users' behaviors change over time by designing mining methods to discover changes in patterns that are discovered from data collected from different time frames.

5. REFERENCES

[1] G. Chittaranjan, J. Blom, and D. Gatica-Perez, Mining large-scale smartphone data for personality studies. *IEEE International Symposium on Wearable Computers, San Francisco, CA, USA*, 2011.

[2] H. Falaki, R. Mahajan, S. Kandula, D. Lymberopoulos, R. Govindan, and D. Estrin, Diversity in Smartphone Usage. *International Conference on Mobile Systems, Applications, and Services, San Francisco, CA, USA*, 2010.

[3] H. Verkasalo, Analysis of smartphone user behavior. *International Conference on Mobile Business and Global Mobility Roundtable*, *Athens, Greece*, 2010.

[4] I. Burguera, U. Zurutuza, and S. Nadjm-Tehrani, Crowdroid: behavior-based malware detection system for Android. *ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices, Chicago, IL, USA*, 2011.

[5] J. Kang, S. Seo, and J. Hong, Usage pattern analysis of smartphones. *Asia-Pacific Network Operations and Management Symposium*, *Taipei, Taiwan*, 2011.

[6] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U.Dayal, and M. Hsu, PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth. *International Conference on Data Engineering, Heidelberg, Germany*, 2001.

[7] L. Xie, X. Zhang, J. Seifert, and S. Zhu, pBMDS: A behavior-based malware detection system for cellphone devices. *ACM Conference on Security and Privacy in Wireless and Mobile Networks, Hoboken, NJ, USA*, 2010.

[8] N. Mabroukeh and C. Ezeife, A taxonomy of sequential pattern mining algorithms. *Journal of ACM Computing Surveys*, 43(1), 1-41, 2010.

[9] P. Chen, H. Wu, C. Hsu, W. Liao, and T. Li, Logging and Analyzing Mobile User Behaviors. *International Symposium on Cyber Behavior*, *Taipei, Taiwan*, 2012.

[10] Q. Xu, J. Erman, A. Gerber, Z. Mao, J. Pang, and S. Venkataraman, Identifying diverse usage behaviors of smartphone apps. *ACM SIGCOMM Conference on Internet Measurement Conference, New York, NY, USA*, 2011.

[11] R. Agrawal and R. Srikant, Fast algorithm for mining association rules in large databases. *International Conference on Vary Large Data Bases, Santiago de Chile, Chile*, 1994.

[12] R. Agrawal and R. Srikant, Mining sequential patterns. *International Conference on Data Engineering, Taipei, Taiwan*, 1995.

[13] R. Agrawal, T. Imielinski, and A. Swami, Mining association rules between sets of items in large databases. *ACM SIGMOD Conference, Washington DC, USA*, 1993.

[14] R. Iváncsy, and I. Vajk, Frequent pattern mining in web log data. *Acta Polytechnica Hungarica*, 3(1), 77-90, 2006.
[15] Viger Philippe Fournier. SPMF-A Sequential Pattern Mining Framework, http://www.philippe-fournier-viger.com/spmg/

ACKNOWLEDGEMENTS

The work presented in this paper was supported in part by the National Science Council of Taiwan under Grant Numbers NSC 101-2221-E-004-011 and NSC 102-2221-E-004-013. This paper was also supported in part by the X-Mind Research Group at the National Chengchi University, Taipei, Taiwan, sponsored by “Aim for the Top University Plan” of the university and the Ministry of Education of Taiwan. Their support is gratefully acknowledged.