IOSR Journal of Engineering
Apr. 2012, Vol. 2(4) pp: 787-793

Secured Transaction with Efficient Mining and Prediction in Mobile Commerce

G. Umamageshwari
Dept. of Information and Communication Engineering, Sri Venkateswara College of Engineering Chennai, India

Asst. Prof. P. Rajalakshmi
Dept. of Information and Communication Engineering, Sri Venkateswara College of Engineering Chennai, India

***Abstract*— In this paper, we explore a new data mining capability and security for a mobile commerce environment. M-commerce is a major application domain for mobile devices, enabling users to perform commercial transactions wherever they go. However, these applications require a high level of security. To achieve effective mobile commerce security and ultimately consumer trust the security mechanisms will constitute as a security risk. So, we identify the special characteristics of M-commerce and reflect on some important security issues using Random Number Generator (RNG) with efficient mining and prediction techniques. A novel framework, called Mobile Commerce Explorer (MCE), for mining and prediction of mobile users' movements and purchase transactions under the context of mobile commerce. The MCE framework consists of three major components like Similarity measuring model for measuring the similarities among stores and items. Frequent-Pattern Growth method avoids repeated database scanning of Apriori algorithm and Mining Frequent Patterns for efficient discovery of mobile users and Behavior Predictor (BP) for prediction of possible mobile user behaviors. A randomized algorithm uses a random number at least once during the computation to make a decision during transaction by providing token numbers only to the particular mobile user after purchasing each and every item.**

***Keywords* - Data mining, Mobile commerce, security.**

I. INTRODUCTION

Security transaction over the web browser, when a customer is using mobile transaction through a web browser the customer is protected by inactivity lock out, these technology loges out the user. Mobile commerce information security and privacy issues are a very important fact, which needed to be considered by mobile, m-commerce and other electronic commerce developers Consumers are very aware of those issues; therefore it will directly have an influence the services provided by business such as money transactions and other services. There are about three billion subscribers using mobile phones worldwide. On the other hand, there are only one billion users to the Internet, this enormous wide spread of mobile phones technologies and the significant number of mobile devices which is increasing rapidly, will provide more opportunities for mobile commerce. Mobile commerce was first found in 1997 in Finland, it was enhanced in a vending machine to serve Coca-Cola by using SMS. M-commerce is a new area arising from the marriage of electronic commerce with emerging mobile and pervasive computing technology. The newness of this area and the rapidness with which it is emerging makes it difficult to analyze the technological problems that m-commerce introduces and, in particular, the security and privacy issues. This situation is not good, since history has shown that security is very difficult to retro-fit into deployed technology, and pervasive m-commerce promises (threatens) to permeate and transform even more aspects of life than e-commerce and the Internet has.

In this paper, the entire concepts move on through mobile commerce fashion over which any of the user try to login with the help of a mobile and also lot more misbehaves opportunities also there in the network. So, proper administration control will be there to maintain a recognized users records and its personal information to keep is as privacy one using various techniques for mining, predication and authentication for the mobile user.

Similarity measuring model: To compute the store and item similarities automatically from the mobile transaction database, this captures mobile users' moving and transactional behavior. From the database we have the following information available: 1) for a given store 2) for a given item. The information can help us to infer which stores or items are similar. A parameter-less data mining model, named SIM to tackle this task of computing store and item similarities. Before computing, we derive two databases, namely, Store-Item Database (SID) and Item-Store Database (ISD), from the mobile transaction database**.**

Mining Frequent Patterns:
Compress a large database into a compact, Frequent-Pattern tree (FP-tree) structure highly condensed, but complete for frequent pattern mining avoid costly database scans and Develop an efficient, FP-tree-based frequent pattern mining method are A divide-and-conquer methodology: decompose mining tasks into smaller ones and also Avoid candidate generation: sub-database test only. Then two way of prediction might be happened in Behavior Prediction concept. If a user already reaches a

minimal support mean no problem to predict user behavior it's fully based on same user previous behavior happenings and very easy to recommend stores and items for users. But, if its fresh user means based on whole user information we predict recommendations for a particular user and prediction of mobile users' commerce behaviors in order to recommend stores and items. To provide a high-precision mobile commerce behavior predictor, we focus on personal mobile pattern mining. Besides, to overcome the predictions failure problem, we incorporate the similarities of stores and items into the mobile commerce behavior prediction.

Authentication Tokens:

Random Number Generator (RNG): This is a computational or physical device designed to generate a sequence of numbers or symbols that lack any pattern, i.e. appear random is used to improve security during transaction of money from the bank by providing token numbers only to the authenticated user after purchasing each and every items. Several computational methods for random number generation exist, but often fall short of the goal of true randomness — though they may meet, with varying success, some of the statistical tests for randomness intended to measure how unpredictable their results are (that is, to what degree their patterns are discernible)

II. RELATED WORK

Authentication requires that the claimant shall prove through a secure authentication protocol that he or she controls the token.[13] Long-term shared authentication secrets, if used, shall never be revealed to any party except the claimant and CSP, however session (temporary) shared secrets may be provided to verifiers by the CSP. Each of the three token types has somewhat different utility and security properties. Soft token solutions are easily realized in "thin clients" with TLS and client certificates. Moreover this solution allows not only initial authentication of claimants, but also allows the entire session, or as much of it as is security critical, to be cryptographically authenticated by a key created during the authentication process.

Hard token solutions provide the additional assurance of a physical token, and users should know if their token has been stolen. Like soft tokens, hard tokens allow not only initial authentication of claimants, but also allows the entire session, or as much of it as is security critical, to be cryptographically authenticated by a key created during the authentication process. One-time password device token systems are commercially available, portable and work easily with any browser client. Like hard tokens, one-time password device tokens have the security advantage that the token is a tangible, physical object. Subscribers should know if their token is stolen, and the key is not vulnerable to network, shoulder-surfing or keyboard sniffer attacks. Unlike soft tokens or hard tokens, a session key is not created from the authentication process to authenticate subsequent data transfers. All three token types present the eavesdroppers with similar strong cryptographic protection. Each has its advantages and disadvantages against various types of attacks. All three offer considerably greater strength than Level 2 solutions. Application implementers with specific Level 3 authentication requirements, who need to select a particular technology should chose the one that best suits the functional needs and risks of their application. In the near future, it is expected that tens of millions of users will carry mobile phones or portable devices that use wireless connection to access a worldwide information network for business or personal use from anywhere at any time, making the mobile commerce (MC) a reality [3], [1], [2]. For example, e Network Web Express [5] enables mobile users to use commercial Web applications over wide-area wireless networks (WANs). Bluetooth technology [7] allows terminals and cash registers to talk directly to each other for the purpose of mobile commerce. The Wireless Access Protocol (WAP) [6] brings the MC environment a world-wide standard for providing Internet communications to digital mobile phones. In an MC environment, customers can make any transaction from anywhere at any time with the payment mechanism provided by banks or credit card companies [2].

In addition, some kinds of Nokia mobile phones provide the wallet application that enables customers to get easy access to mobile services and to make convenient online mobile transactions [4].In the wallet, customers can store sensitive personal information, such as payment and loyalty card details, delivery addresses, and notes, as well as service profiles. In addition, with the wallet application, the Nokia mobile phones have the capability of storing the transactions with moving patterns and purchasing patterns of customers.[9] Mark N. Gasson, the authors discuss how to open the possibilities to profile people based on the places they visit, people they associate with, or other aspects of their complex routines determined through persistent tracking. It is possible that services offering customized information based on the results of such behavioral profiling could become common place. Here, in detail on a short case study tracking four people, in three European member states, persistently for six weeks using mobile handsets. The GPS locations of these people have been mined to reveal places of interest and to create simple profiles. The information drawn from the profiling activity ranges from intuitive through special cases to insightful. [11] A new data mining capability for a mobile commerce environment, he studied the problem to better reflect the customer usage patterns in the mobile commerce environment, we propose an innovative mining model, called mining mobile sequential patterns, which takes both the moving patterns and purchase patterns of customers into consideration. How to strike a compromise among the use of various knowledge to solve the mining on mobile sequential patterns is a challenging issue. Mobile E-Commerce provides location-based services to mobile users in web environments. One of the best ways to personalize mobile services is based on location. The authors propose a new algorithm called the Distributed Pattern Miner (DPM), for mining location-aware service request patterns from distributed databases on a Data Grid. The location and service request patterns represent frequently requested services and the corresponding location of mobile users in mobile web environments [12].

III. PROPOSED ARCHITECTURE DESIGN

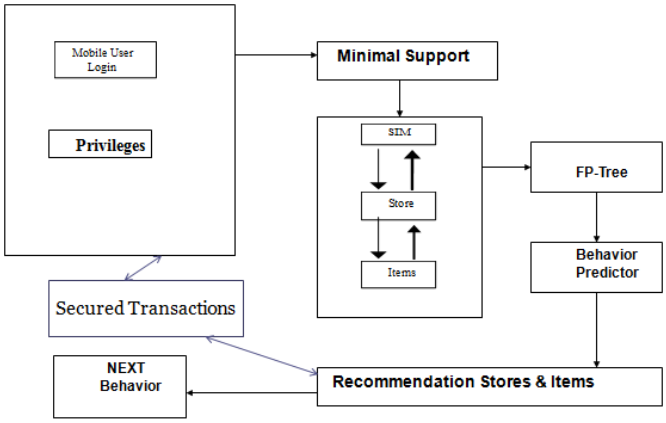
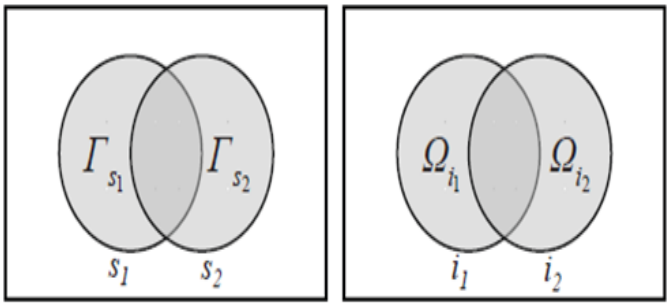


Figure 1: Mobile Commerce Expolrer Architecture

In this section, we describe our design of a personal mobile commerce mining and prediction framework, called Mobile Commerce Explorer, which incorporates three innovative techniques, including 1) Similarity Measuring model for measuring the similarities among stores and items, which are two basic mobile commerce entities considered in this paper; 2) FP-tree for efficient discovery of mobile users' Personal Mobile Commerce Patterns; and 3) BP for prediction of possible mobile user behaviours. Along with these techniques authentication tokens are used for secured transactions.

A. Similarity Measuring model



(a)Store Similarity Inference (b) Item Similarity Inference

Figure 2: Basic concept of Similarity Measuring model

As shown in Figure 2(a), s1 and s2 are two stores and Γ_{s1} and Γ_{s2} are two item sets which are sold in stores s1 and s2, respectively. In Figure 2(b), i1 and i2 are two items and Ω_{i1} and Ω_{i2} are two store sets where users have purchased i1 and i2, respectively. Based on our observations, we identify two basic heuristics to serve as the basis of our Similarity Measuring model: 1) s1 and s2 are more similar, if Γ_{s1} and Γ_{s2} are more similar.

For example, McDonalds and Burger King are similar since their provided items are similar, e.g., hamburgers, French fries, and Cokes. 2) i1 and i2 are more dissimilar, if Ω_{i1} and Ω_{i2} are more dissimilar. For example, the stores {McDonalds, Burger King, and KFC} and {Hang Ten, Giordano, and G2000} are dissimilar since the former is food-related stores and the latter is clothes-related stores. The items sold between these two kinds of stores are usually different, e.g., hamburger and shirt. Although SimRank which is similar to SIM has been proposed in [8], it is not applicable to the store similarity inference.

TABLE I: SID AND ISD

Store	Items
A	i_1, i_3
B	i_1, i_5
C	i_3, i_5
D	i_2, i_4, i_6, i_7
E	i_1, i_3
F	i_3, i_4
I	i_2, i_5, i_6, i_8
K	i_2, i_5

Item	Stores
i_1	A, B, E
i_2	D, I, K
i_3	A, C, E, F
i_4	D, F
i_5	B, C, I, K
i_6	D, I
i_7	D
i_8	I

In SimRank, the similarity between two given objects is measured based on the average similarities between other objects linked with the given two objects. As a result, two supermarkets selling a number of different items may be considered as dissimilar in SimRank. If we apply the same similarity inference heuristics to both of stores and items, various types of items may be seen as similar since different supermarkets are seen as similar. Based on our heuristics, if two stores provide many similar items, they are likely to be similar; if two items are sold by many dissimilar stores, they are unlikely to be similar. Since the store similarity and item similarity are inter-dependent, we compute them iteratively. In the following, we discuss the computational model. For the store similarity, we consider that two stores are more similar if their provided items are more similar. Given two stores sp and sq, we compute their similarity sim (sp, sq) by calculating the average similarity of item sets provided by sp and sq. For every item sold in sp (and respectively sq), we first find the most similar item sold in sq (and respectively sp). Then, the store similarity can be obtained by averaging all similar item pairs.

$$sim(s_p, s_q) = \frac{\sum_{\varphi \in \Gamma_{s_p}} MaxSim(\varphi, \Gamma_{s_q}) + \sum_{\gamma \in \Gamma_{s_q}} MaxSim(\gamma, \Gamma_{s_p})}{|\Gamma_{s_p}| + |\Gamma_{s_q}|} \quad (1)$$

Where Maxsim (e, E) =Max e' ϵ^E sim(e, e') represents the maximal similarity between E and the element in E. Γ_{sp} and Γ_{sq} are the sets of items sold in sp and sq, respectively. On the other hand, for the item similarity, we consider that two items are less similar if they are sold by many dissimilar stores. Given two items i_x and i_y, we compute their similarity sim(i_x, i_y) by calculating the average dissimilarity of store sets that provide i_x and i_y. For every store providing i_x (and respectively i_y), we first find the most dissimilar store that provides i_y (and respectively i_x) to obtain the item similarity by averaging all dissimilar store pairs. Therefore, sim(i_x, i_y) is defined as (2).

$$sim(i_x, i_y) = 1 - \frac{\sum_{\omega \in \Omega_{i_x}} MinSim(\omega, \Omega_{i_y}) + \sum_{\psi \in \Omega_{i_y}} MinSim(\psi, \Omega_{i_x})}{|\Omega_{i_x}| + |\Omega_{i_y}|} \quad (2)$$

Where Minsim (e, E) =Min e' ϵ^E (1-sim (e, e')) represents the maximal similarity between E and the element in E. Ωi_x and Ωi_y are the sets of stores sell i_x and i_y, respectively.

B. FP-Growth Method: Construction of FP-Tree

First, create the root of the tree, labeled with "null".
Scan the database D a second time. (First time we scanned it to create 1-itemset and then L).

The items in each transaction are processed in L order (i.e. sorted order).

A branch is created for each transaction with items having their support count separated by colon.

Whenever the same node is encountered in another transaction, we just increment the support count of the common node or Prefix.

To facilitate tree traversal, an item header table is built so that each item points to its occurrences in the tree via a chain of node-links.

Now, the problem of mining frequent patterns in database is transformed to that of mining the FP-Tree.

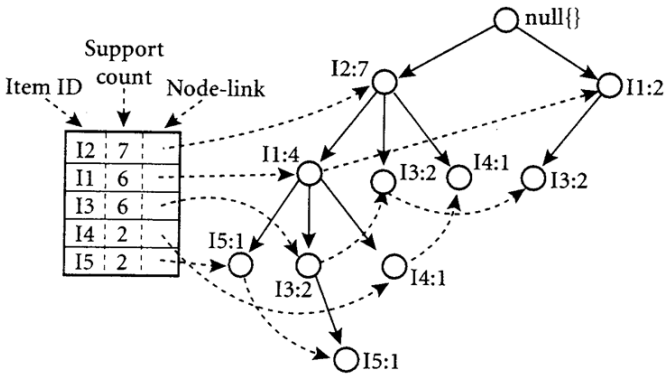


Figure 3: An FP-tree that registers compressed, frequent pattern information

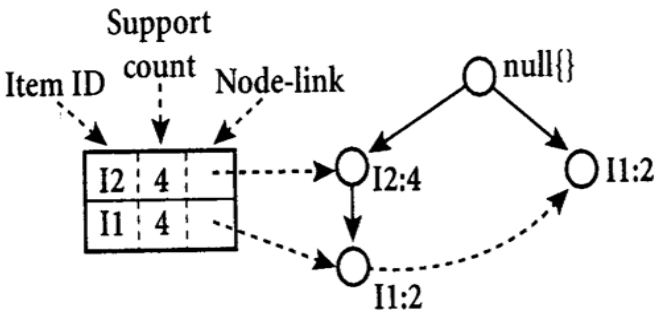


Figure 4: The conditional FP-Tree associated with the conditional I3

Mining the FP-Tree by Creating Conditional (sub) pattern bases

Steps:

1. Start from each frequent length-1 pattern (as an initial suffix pattern).
2. Construct its conditional pattern base which consists of the set of prefix paths in the FP-Tree co-occurring with suffix pattern.
3. Then, Construct its conditional FP-Tree & perform mining on such a tree.
4. The pattern growth is achieved by concatenation of the suffix pattern with the frequent patterns generated from a conditional FP-Tree.
5. The union of all frequent patterns (generated by step 4) gives the required frequent item set.

TABLE 3: Mining the FP-Tree by creating conditional (sub) pattern bases

Item	Conditional pattern base	Conditional FP-Tree	Frequent pattern generated
I5	{(I2 I1: 1),(I2 I1 I3: 1)}	<I2:2 , I1:2>	I2 I5:2, I1 I5:2, I2 I1 I5: 2
I4	{(I2 I1: 1),(I2: 1)}	<I2: 2>	I2 I4: 2
I3	{(I2 I1: 1),(I2: 2), (I1: 2)}	<I2: 4, I1: 2>,<I1:2>	I2 I3:4, I1, I3: 2 , I2 I1 I3: 2
I2	{(I2: 4)}	<I2: 4>	I2 I1: 4

Now, Following the above mentioned steps:

1. Lets start from I5. The I5 is involved in 2 branches namely {I2 I1 I5: 1} and {I2 I1 I3 I5: 1}.
2. Therefore considering I5 as suffix, its 2 corresponding prefix paths would be {I2 I1: 1} and {I2 I1 I3: 1}, which forms its conditional pattern base.
3. Out of these, Only I1 & I2 is selected in the conditional FP-Tree because I3 is not satisfying the minimum support count.
4. For I1 , support count in conditional pattern base = 1 + 1 = 2
5. For I2 , support count in conditional pattern base = 1 + 1 = 2
6. For I3, support count in conditional pattern base = 1
7. Thus support count for I3 is less than required min_sup which is 2 here.
8. Now , We have conditional FP-Tree with us.
9. All frequent pattern corresponding to suffix I5 are generated by considering all possible combinations of I5 and conditional FP-Tree.
10. The same procedure is applied to suffixes I4, I3 and I1.

Note: I2 is not taken into consideration for suffix because it doesn't have any prefix at all.

C. Behavior Prediction

The proposed MCBP which measures the similarity score of every FP-tree with a user's recent mobile commerce behavior by taking store and item similarities into account.

In MCBP, three ideas are considered:

The premises of FP-tree with high similarity to the user's recent mobile commerce behavior are considered as prediction knowledge. More recent mobile commerce behaviors potentially have a greater effect on next mobile commerce behavior predictions.

FP-tree method with higher support provides greater confidence for predicting users' next mobile commerce behavior. Based on the above ideas, we propose a weighted scoring function to evaluate the scores of FP-tree method.

$$w_k = \frac{k}{\sum_{i=1}^{max(m,n)} i} \tag{3}$$

Equation (3) gives more weight to recent movement. We can calculate their pattern score by the weighted scoring function. The consequence of FP-tree with the highest score is used to predict the next mobile commerce behavior.

D. Authentication Tokens

Authentication is the process of determining if a user or identity is who they claim to be. Authentication is accomplished using something the user knows (e.g. password), something the user has (e.g. security token) or something of the user (e.g. biometric). The authentication process is based on a measure of risk. High risk systems, applications and information require different forms of authentication that more accurately confirm the user's digital identity as being who they claim to be than would a low risk application, where the confirmation of the digital identity is not as important from a risk perspective. This is commonly referred to as "stronger authentication". Security includes authenticating business transactors, controlling access to resources such as Web pages for registered or selected users, encrypting communications, and, in general, ensuring the privacy and effectiveness of transactions. The many applications of randomness have led to the development of several different methods for generating random data. Many of these have existed since ancient times, including dice, coin flipping, the shuffling of playing cards, the use of yarrow stalks (by divination) in the I Ching, and many other techniques. Because of the mechanical nature of these techniques, generating large amounts of sufficiently random numbers (important in statistics) required a lot of work and/or time. Thus, results would sometimes be collected and distributed as random number tables. Nowadays, after the advent of computational random number generators, a growing number of government-run lotteries, and lottery games, are using RNGs instead of more traditional drawing methods. RNGs are also used today to determine the odds of modern slot machines. Pseudo-random number generators (PRNGs) are algorithms that can automatically create long runs of numbers with good random properties but eventually the sequence repeats (or the memory usage grows without bound). The string of values generated by such algorithms is generally determined by a fixed number called a seed. One of the most common PRNG is the linear congruential generator, which uses the recurrence to generate numbers.

$$X_{n+1} = (aX_n + b) \bmod m \tag{4}$$

The maximum number of numbers the formula (4) can produce is the modulus, m. To avoid certain non-random properties of a single linear congruential generator, several such random number generators with slightly different values of the multiplier coefficient a can be used in parallel, with a "master" random number generator that selects from among the several different generators. We use the term RNG in this document to mean a cryptographically-secure PRNG, deterministic RNG or deterministic RBG (DRBG). All these terms mean the same thing for our purposes.

RNG ≡ RBG &equiv DRBG &equiv PRNG

PRNGs work by keeping an internal state. Typically this is a seed and a key, which are kept secret. When a consumer requests random data, a cryptographic algorithm operates on the seed and the key to produce pseudo-random output. The internal state is then updated so that the next request does not produce the same data. Ferguson and Schneier describe a simple generator using AES-256 and a 128-bit counter. NIST specifies a whole smorgasbord of generators using message digest hashes, HMACS, block ciphers and even elliptic curves. The idea is that designers can use whichever cryptographic function is already available to them. Some typical pseudo-code for a PRNG generator might be:

INPUT: (Key, Seed)

OUTPUT: random_data, (Key', Seed')

random_data = F (Key, Seed)

Key' = F (Key, Seed+1)

Seed' = F (Key', Seed)

return random_data

Where F is a cryptographic function. All the generators are essentially some variant of this. Continuous RNG Test to the consumer requests a set of random data, we generate an extra 64-bit value which is not output but is stored for comparison purposes. If this matches the first 64 bits of the next about-to-be-output data, then we throw a catastrophic error. If the length of the requested random data is less than 64 bits, then we pad the about-to-be-output data to 64 bits with zeroes before comparison. On start-up, we generate a 64-bit block that is not used for output but is saved for comparison with the next request.

In pseudo-code:

```
INPUT:  nbits, previous_block
OUTPUT: random_data, previous_block'
if (previous_block!= Null)
{
  random_data = GenerateRandomData (nbits)
  if (nbits < 64)
    tocompare = random_data || 0^ (64 - nbits)
  else
    tocompare = LeftMostBits (random_data, 64)
  if (tocompare == previous_block)
    return "catastrophic error"
}
previous_block' = GenerateRandomData (64)
return random_data
```

A strict reading of FIPS 140-2 would seem to require a check of every successive 64-bit block generated by the DRBG_ Generate function where the requested number of bits is greater than 64. We assert that this pointless unless the HMAC-SHA-1 function is corrupted. Furthermore, and far more serious, storing every generated block to compare with the next would expose a huge security hole. Using RNG, authentication tokens are provided to the user for transactions by avoiding the intruders. Transaction verification is the internet-based security method of verifying that the actual content of a transaction has not been altered by fraudulent techniques. Online fraud caused by identity or content theft becomes one of the major issues in the future development of the e-society. Identy PASS can provide a fully mobile and secured token management system for end-point-authentication and transaction security. The token allows a remote multi-factor authentication of users and a verification of transactions over the internet without the need of any peripheral installation. An optical interface allows a one-way communication from a remote server directly into the token. Encrypted messages are transmitted by a server that generates a flickering code that is read by the optical interface of the token on any screen displays.

IV. PERFORMANCE EVALUATION

The goal of this experiment is to compare SIM and ranking and suggesting popular as the suggesting popular items seems problematic due to potential popularity disorder, why make suggestions in the first place? This is for several reasons; for example, suggestions may help recall what candidate items are. A fix to avoid popularity skew would be to suggest all candidate items and not restrict to a short list of few popular items [11]. This is often impractical for reasons such as limited user interface space, user ability to process smaller sets easier, and the irrelevance of less popular items. So, the number of suggestion items is limited to a small number.The main reason is that the similarity measure among stores and items for Similarity Measuring model is more accurate than ranking and suggesting popular items. The similarity between two stores can be accurately measured by Similarity Measuring model, even if there is no common item sold in these two stores.

Here we comparing Normality Mining with behavior predication method [9] Which can used to predict the users current location and there is no such options to the user to search their desired objects or places via their mobile itself. No persistent server for maintaining the history of each user's information. It is possible that services offering customized information based on the results of such behavioral profiling could become commonplace. However, it may not be immediately apparent to the user that a wealth of information about them, potentially unrelated to the service, can be revealed. Further issues occur if the user agreed, while subscribing to the service, for data to be passed to third parties where it may be used to their detriment. To provide a high-precision mobile commerce behavior predictor, we focus on FP-growth. Besides, to overcome the predictions failure problem, we incorporate the similarities of stores and items into the mobile commerce behavior prediction. So it is that behavior prediction is more efficient than normality mining. Performance study shows FP-growth is an order of magnitude faster than Apriori, and is also faster than tree-projection. Reasoning are No candidate generation, no candidate test, Use compact data structure, Eliminate repeated database scan.

Basic operation is counting and FP-tree building

TABLE 2: Comparison between PRNGs and TRNGs

Characteristic	Pseudo-Random Number Generators	True Random Number Generators
Efficiency	Excellent	Poor
Determinism	Deterministic	Nondeterministic
Periodicity	Periodic	Aperiodic

The characteristics of TRNGs are quite different from PRNGs. First, TRNGs are generally rather inefficient compared to PRNGs, taking considerably longer time to produce numbers. They are also nondeterministic, meaning that a given sequence of numbers cannot be reproduced, although the same sequence may of course occur several times by chance. TRNGs have no period. The table below sums up the characteristics of the two types of random number generators.

These characteristics make TRNGs suitable for roughly the set of applications that PRNGs are unsuitable for, such as data encryption, games and gambling. Conversely, the poor efficiency and nondeterministic nature of TRNGs make them less suitable for simulation and modeling applications, which often require more data than it's feasible to generate with a TRNG.

V. CONCLUSIONS AND FUTURE WORK

In this paper is to facilitate mining and prediction of mobile users' commerce behaviours based on FP-growth and Behaviour Predictor (BP) by providing secured transaction using authentication tokens which contains products and benefits like Small and compact token (size of a credit card), Strong identity verification of holder with a two- or three factor authentication (token + pin and/or fingerprint),Low-power fingerprint swipe sensor enclosed Protect privacy (biometric storage and match on-device).The experimental results show that our proposed framework and components are highly accurate under various conditions.

For the future work, we plan to explore more efficient mobile commerce pattern mining algorithm, design more efficient security in mobile transactions, similarity inference models, and develop profound prediction strategies to further enhance the MCE framework. In addition, we plan to apply the MCE framework to other applications, such as object tracking Sensor networks and location based services, aiming to achieve high precision in predicting object behaviors.

REFERENCES

[1] U. Varshney, R. J. Vetter, and R. Kalakota, "Mobile commerce: A new frontier," IEEE Comput., vol. 33, no. 10, pp. 32–38, Oct. 2000.

[2] J. Veijalainene, "Transactions in mobile electronic commerce," in Proc.8th Int. Workshop on Foundations of Models and Languages for Data and Objects, Sep. 1999, pp. 203–224.

[3] [Online] Available:http://www.mobilecommerceworld.com

[4] Wallet Application [Online]. Available: http://www.forum.nokia.com.

[5] R. Floyd, B. Housel, and C. Tait, "Mobile web access using e network web express," IEEE Pers. Commun., vol. 5, no. 5, pp. 47–52, Oct. 1998.

[6] WAP Forum Wireless Application Protocol. [Online]. Available: http://www.wapforum. org/

[7] Bluetooth Overview, 1999 [Online].Available: http://www.bluetooth. Com

[8] G. Jeh and J. Widom, "SimRank: A Measure of Structural- Context Similarity," Proc. Int'l. Conf. on Knowledge Discovery and Data Mining, pp. 538-543, Jul. 2002.

[9] Mark N. Gasson, Eleni Kosta, Denis Royer, Martin Meints, and Kevin Warwick (2010) 'Normality Mining: Privacy Implications of Behavioral Profiles Drawn From GPS Enabled Mobile Phones' IEEE transactions on systems, man and cybernetics—part c: applications and reviews 1.

[10] Ching-Huang Yun and Ming-Syan Chen (2007) 'Mining Mobile Sequential Patterns in a Mobile Commerce Environment' IEEE transactions on systems, man and cybernetics—part c: applications and reviews Vol. 37, No. 2, pp. 273–566.

[11] Milan Vojnovic, James Cruise, Dina Gunawardena, and Peter Marbach (2009) 'Ranking and Suggesting Popular Items' IEEE transactions on knowledge and data engg Vol.8 No.8, 15–37.

[12] U. Sakthi Research Scholar, Department of Computer Science, Anna University Chennai, Raghuvel S. Bhuvaneswaran Raghuvel S. Bhuvaneswaran, Ramanujan Computing Centre Anna University Chennai (2011) 'Data Grid Mining of Mobile User Behaviours in Web Environments' European Journal of Scientific Research ISSN 1450-216X Vol.49 No.4 , pp. 555-566S

[13] William E. Burr, Donna F. Dodson, W. Timothy Polk, April 2006 'Electronic Authentication Guideline' Recommendations of the National Institute of Standards and Technology Special Publication 800-63 Version 1.0.2