EFFICIENT MINING BEHAVIOR PATTERNS OF USER MOVEMENT IN A MOBILE SERVICE ENVIRONMENT

C. Nivedhitha*

S. Muthu Kumar*

Abstract

Mobile service systems offer useful information to users through mobile devices. Mobile service systems have the capability of effectively mining a special request from abundant data based on user movement behavior patterns. UMBPs which includes U(mobile user), L(location), T(timestamp) and S(service request). Effective mining can be performed by introducing standard graph-matching algorithms along with the primitives of a database management system, which comprises grouping, sorting, and joining operations. Moreover, by mining the associated structure via maximum weight bipartite graph matching, a prediction mechanism, based on the model of UMBPs, is used to find strong relationships among U, L, T and S. This paper aimed at mining mobile User Movement Behavior Patterns such that movement locations associated with suitable services could be predicted and recommended for users.

Keywords----Data mining, mobile services, mobility prediction, mobile access patterns.

* PSN College of Engineering and Technology, Tirunelveli

I. INTRODUCTION

The advances in wireless communications and mobile device technologies not only accentuate various wireless communications applications but also enable the provision of plentiful kinds of mobile services for users[3],[6]. In mobile service environments, mobile users may request various kinds of services and applications through mobile devices.

To achieve quick response from the system, the four characteristics are used in mobile services environment. This is called as User Movement Behavior Patterns (UMBP). The UMBP characteristics are mobile Users (U), movement Locations (L), Timestamps (T), service requests(S).Here data mining is to be used to an efficient and excellent performance result in mobile service responses. The project contains both spatial and temporal mining, also data mining approach is proposed for efficiently discovering UMBPs in a mobile service environment. Traditional mobile service systems are inadequate in handling complex UMBPs without taking *U* (mobile users), *L* (movement locations), *T* (dwell time in timestamps), and *S* (service requests) into serious consideration.

Over the past few years, some studies have employed data mining techniques to discover interesting patterns from Web logs or large databases [1]. Although some recent studies have made progress on data mining in mobile service systems, they were mostly focused on issues such as moving path mining or service request log mining. Mainly faced the problems of location tracking and resource allocation. Moreover, by modeling UMBPs via maximum weight bipartite graph matching [9], this prediction mechanism provides appropriate recommendations for users in terms of the spatial movement locations of each mobile user with requested services in different time intervals. Nevertheless, for acquiring an appropriate recommendation and a precise prediction, the four relational patterns of U, L, T, and S must be considered simultaneously.

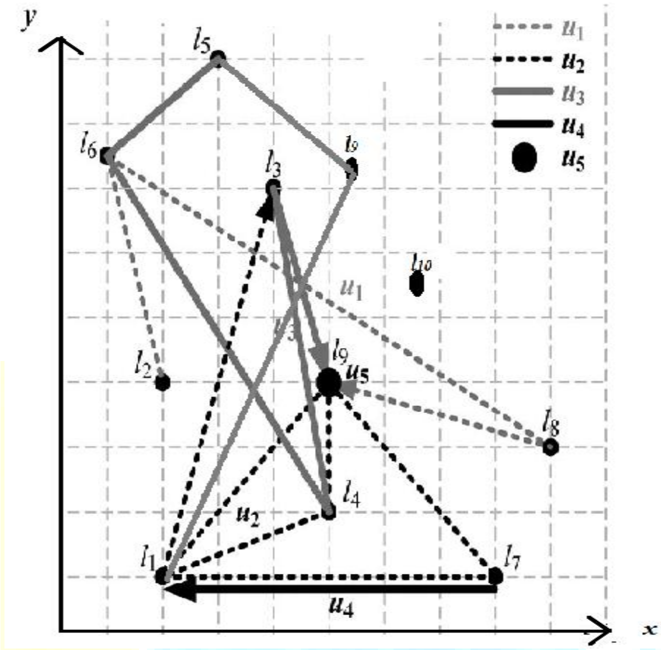


Fig: Example of the deployed locations and user movements

Fig. shows that there are ten different locations are scattered randomly in a spatial area. In addition, five users are denoted with different trajectories and with different plotted line styles.

A data mining approach is proposed for efficiently discovering user movement behavior patterns in a mobile service environment, namely matching mobile access pattern, which consists of user movement along with requested services in different time intervals. The mobile access patterns consists of three databases, User Movement database (UMD), Location Movement Database (LMD), Service Request Database (SRD). The UMD, which is a 2-D database, is considered the original raw database. The LMD is obtained after validation and conversion mapping from the UMD. The SRD is used to store the service request sequences of mobile users into a structure that is suitable for data mining. A family of match join weights yields a maximum match by MJSC. In addition, this associated tree-based technique is used to determine the relationships among data in the structure. An associated tree-based pattern, extracted from candidate cardinality matching tuples, is obtained based on the given minimum support value.

The rest of the paper deals with the following sections: Section II discuss about a review of moving objects in the database. In section IV, both match join algorithms MJMF and MJSC are described. Modules described in Section V. Finally, conclusions are presented in Section VII.

II. MOBILE ACCESS PATTERNS

The various terminologies used are :

Time vertical—the total duration the data have been captured.

Movement---the mobile user moves from one base station to another base station.

Interesting location---the mobile user stays at a location longer than the predefined maximum duration.

Generic location--- the mobile user stays at a location shorter than the predefined maximum duration.

T_m is the ID number of the stamp, U_i is the ID number of the user, L_j is the ID number of the location.T_s is the duration that mobile users dwell at location U_i. $S_n = \{s_1, s_2, s_3,\ldots,s_n\}$ is the set of services requested by mobile users. Each element in S_n is given a designated optimum time of using this service. If a mobile user uses the service for a longer duration than the optimum time, then the service is regarded as an interesting service. Each element in S_n is the ID number of the service; T_r is denoted as the duration of using a service.

Table: Example of raw data for MJSC

U_i	L_j	LC_n	T_m	T_s	S_n	T_r
1	1	11	23	24	1	3
1	4	12	34	6	2	5
1	5	1	54	4	3	23
1	7	5	12	13	4	13
2	8	6	54	27	5	17
2	6	12	23	19	1	28
3	4	14	27	4	1	2
3	7	2	14	8	1	5
4	7	7	15	17	4	9
5	9	7	18	13	2	16
4	9	9	29	12	3	27

III. BASIC DEFINITIONS

The match joins of matching mobile access patterns from the three databases, *UMD*, *LMD*, and *SRD*. The basic definitions are given as follows:

Definition 1—[Mobile User]: U = {u1, u2, u3, . . . , ui} is the set of mobile users. Each mobile user represents a person who carries a mobile device that has the capability of receiving services from the mobile environment, can be identified and tracked.

Definition 2—[Location]: Two special locations used to identify the regularities of the visiting locations are the generic location and the interesting location. The generic location is a collective term for one or more interesting locations, and the interesting location is a subset of the generic location. The generic location can be defined as L = {l1, l2, l3, . . . , lj}, where each element *lj* represents a generic location. The interesting location can be defined as the user *ui* staying at a location *li* longer than the maximum duration.

Definition 3—[Timestamp and Maximum Duration]: The timestamp *Tm* have an equal period and a uniform unit. The maximum duration is considered to be 30 min, in general.

Definition 4—[Service]: S = {s1, s2, s3, . . . , sn} is the set of services requested by mobile users. Each element represents an individual service ID. An optimum time is set for each service requested.

Definition 5—[Weight Value w-Join]: The weight value *w* is used to identify the importance of an extracted pattern. The w is dependent on three factors, the weight values of match joins $U \bowtie L$, $L \bowtie T$ and $T \bowtie S$.

weight value of $U \bowtie L = [LC_1/Maxf, LC_2/Maxf, \ldots, LC_n/Maxf]$

weight value of $L \bowtie T = [Ts_1/d_{Max}, Ts_2/d_{Max}, \ldots, Ts_m/d_{Max}]$

weight value of $T \bowtie S = [Tr_1/Ot_1, Tr_2/Ot_2, \ldots, Tr_l/Ot_l]$

LCi is the count of arriving at location *li* for each user.

Maxf is the maximum *LCi* for each user.

Tsi is the duration a mobile user dwells at location *li*.

dMax is the maximum *Tsi*.

Tri is the duration of requested service at each location *li*.

Oti is the given optimum time of using the *i*th service.

IV. MATCHING MOBILE ACCESS PATTERNS

Mining matching *UMBPs* can be done by operating on relational flows to generate the max flow of match joins by means of *MJMF*. The transformation from a matching problem to the max flow problem can be divided into three phases. They are grouping nodes, building the reduced graph, and exercising the max flow algorithm. The first step in the grouping node is to find tuples whose relations have different weight values on the join columns. Second, the weight values of match joins $U \bowtie L, L \bowtie T$ and $T \bowtie S$ are to be calculated. With the weight values and the designated min_sup value, the relative intensity(RI) can be carried out to construct the reduced max flow graph.

MJSC computes the match join of the four input relations by first dividing the relations into groups of candidate matching tuples of *U*, *L*, *T*, and *S*, and, second, by operating the match join within each group. The steps of the algorithm are as follows:

Step 1: Perform an external sort of four input relations on all attributes involved with *w*.

Step 2: Iterate through the relations and generate the next group *G* of the tuples of *U*, *L*, *T*, and *S*.

Step 3: Within *G*, combine the four subsets of *U*, *L*, *T*, and *S* tuples. Compared to the operation of combine-join, the iterators operating in the table can be advanced as soon as matches are found.

To determine the relative intensity the related parameters of three sorts, M(ui, lj), M(lj, tm), and M(tm, sn), which correspond to $\alpha 1$, $\alpha 2$, $\alpha 3$ values have to be given in advance. The three parameters follow the relation of $\alpha 1 + \alpha 2 + \alpha 3 = 1$. Second, the weights of the join *U LTS* are normalized to identify a useful relation, which corresponds to the weight value. If the weight value is greater than or equal to the min_sup value, then the relation is denoted as useful. Then the *RI* is calculated with related parameters $\alpha 1$, $\alpha 2$, and $\alpha 3$.Finally, according to different users, grouped these tuples (ui, lj, tm, sn) into *i*-groups. The operation through the table within a group can then retrieve the maximum match. As a result, each group has its maximum match based on the RI. Based on the RI of match join weights *w*, the maximum flow can be obtained by *MJSC*.

The *RI* can be determined as follows:

$RI = E1(ui, lj) \times \alpha 1 + E2(lj, tm) \times \alpha 2 + E3(tm, sn) \times \alpha 3.$

The min_sup value, which has been set to 0.2 in advance. Meanwhile, $\alpha 1$, $\alpha 2$, and $\alpha 3$ are the related parameters of edges (ui, lj), (lj, tm), and (tm, sn), whose values are 0.2, 0.4, and 0.4, respectively. E1(ui, lj), E2(lj, tm), and E3(tm, sn) are the corresponding normalized weight values of (ui, lj), (lj, tm), and (tm, sn), respectively.

V. MODULES DESCRIPTION

A. Mobile Network Model

Mobile Networks refer to a family of routing protocols developed to route traffic through mobile wireless networks. These networks place special requirements on routing protocols due the unpredictable nature of the radio links and changing network topology due to node mobility. As the mobile user movement locations are sampled over a long-term historical data, the movements of the mobile user are tracked as an n-length sequence S of spatial locations of the form {(l_1,t_1) , (l_2,t_2) ,...,(l_j,t_j) }, where l_j is the j -location and t_j is the time the mobile user arrives at the j -location.

B. Mobile Transaction Sequence Dataset

In a mobile transaction database, users in the different user groups may have different mobile transaction behaviors. The user give the information i.e. location, time, type of service, user Id to the mobile service station for getting the requested service. We assume that the user movement database (UMD) is the raw database for mobile access patterns mining. It consists of a 2-dimensional database in which each column represents the individual mobile user and each row represents individual time, and the duration between each time unit represents an equal duration, whether it is one minute, or one hour.

C. Maximum Join Sort Combination(MJSC)

Match joins using sort combination (MJSC) that computes the match join of the four relations by first dividing up the relations into groups of candidate matching tuple of U, L, T, and S and then computing the match join within each group.

The main steps of the algorithm are as follows:

1. Perform an external sort of four input relations on all attributes involved in w.

2. Iterate through the relations and generate the next group G of U, L, T, and S tuple.

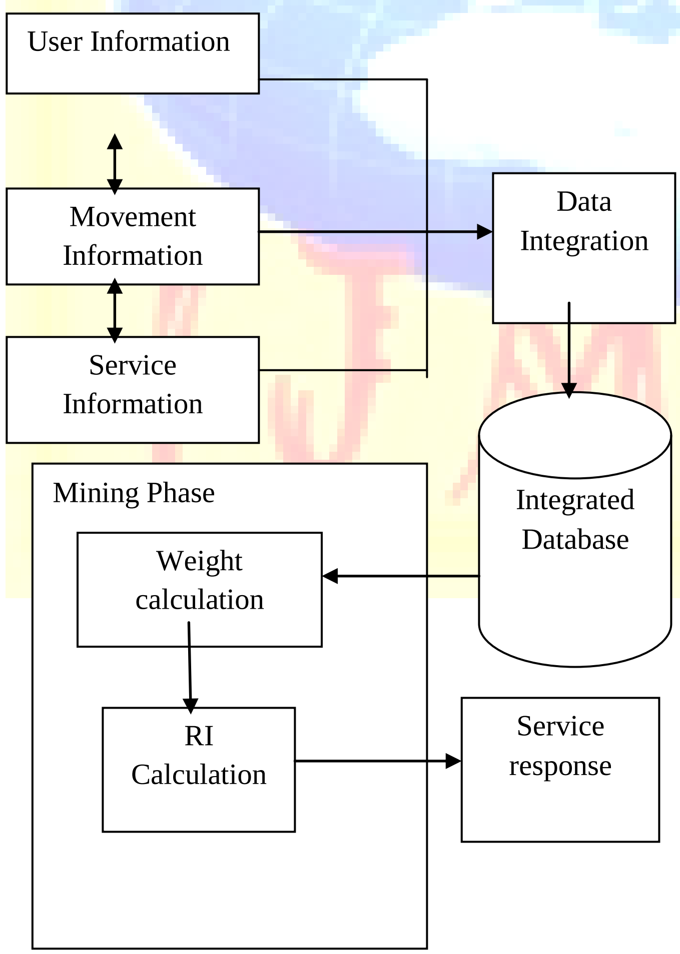
3. Within G, combine the four subsets of U, L, T, and S tuple, just as in combine join, except that iterators on the table can be advanced as soon as matches are found. The related intensity of a conventional sort combined join is proportional to the product of its input relations. The weightage values of the dataset is to be calculated such as w1,w2,w3. Then calculate the relative intensity.

D. Service Response and Performance Analysis

According to the weight values w1.w2,w3 and its relative intensity value the service is to be provided. The services are provided to the particular user based on high relative intensity values.

When compared to the existing system, the proposed system has less execution time with more candidate data set. When the min_ sup value of the generated candidate data sets increased from 0.2% to 1%, the size of the candidate data sets decreased from 50 to 10.

E. Block Diagram



VI. EXPERIMENTAL RESULTS

Simulations were conducted as follows. The objective of this simulation was to compare the performance of the proposed algorithms with the Apriori-like algorithm and evaluate their sensitivity by varying the parameters of various data characteristics. The change in the strong relationships among these nodes (ui, lj, tm, sn) was examined. Then, the support values of match joins were compared with different candidates, including the min_sup value and the number of nodes obtained with the length of *k*-tree candidates. Finally, the performance of the three synthetic databases with parameters in varying degrees was tested.

In the simulation, generated 5-10 mobile users and each mobile user had a distinct movement path. To simulate the movement patterns of the mobile users, the probability model was utilized to model user movement behaviors. Each mobile user had a unique movement behavior, it means that the user would frequently move in certain locations and request certain services. In given locations where a mobile user usually stayed and moved among the locations, movement paths, denoted as the shortest paths, for the user were generated. To generate the movement patterns of the mobile users, the approximate locations for the maximal frequent patterns needed to be determined . As the number of users and locations were inputted, random maximal patterns were selected, and several random trajectories were generated in space. Thus, the maximal frequent patterns were obtained according to the preset parameter $Maxf_l$ to determine the maximal frequency of each user arriving at locations each day. To generate the dwell time in timestamps *T*, the parameter *Tn* was used to determine the number of timestamp patterns. Dwell time patterns were generated randomly; each dwell time pattern was attached to timestamps *T*. Each mobile user had a location sequence *S*, which was a randomly consecutive location. The dwell time pattern was a subsequence of *S*. To generate service request patterns, each mobile user requested services in different locations, and the time of each service request pattern was generated randomly. Each service request in a location contributed to its dwell time of using each service pattern in timestamps *T*. Given the number of services, the parameter *Maxfs* was used to determine the maximal frequency of each user requesting services each day, and the parameter *Oti* was used to determine the optimum time of using each service.

It was observed that the proposed approach obtained more candidate patterns than the Apriori-like algorithm. The proposed algorithms is much better than the Apriori-like algorithm. That is, as more candidates were generated for selection, the less the execution time of the proposed methods.

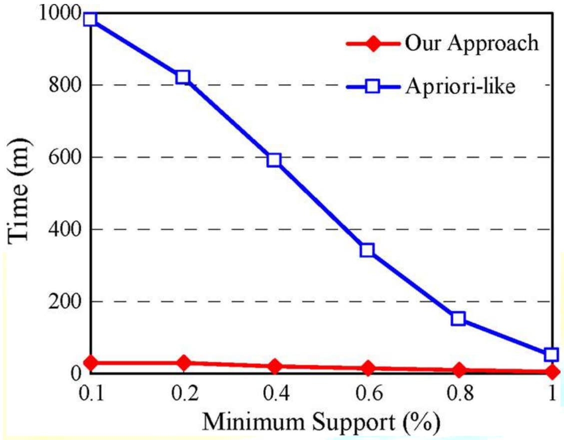


Fig. Execution time in response to changes in different minimum support values.

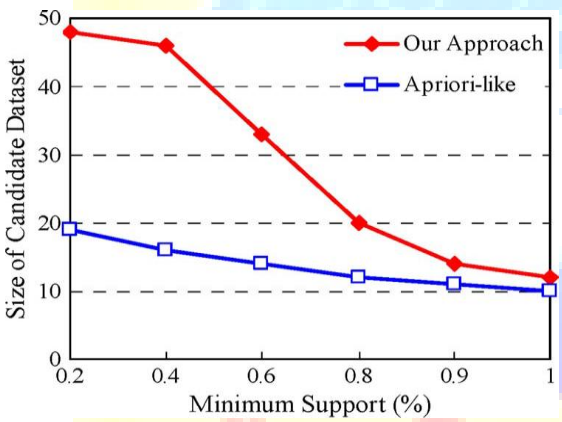


Fig. Size of candidate datasets in response to changes in different minimum support values.

VII. CONCLUSION

The proposed method was based on User Movement Behavior Patterns(UMBPs).The UMBPs can be regarded as the match joins of U, L,T and S based on their weight values. The simple approach of computing the full *w*-joins and then applying standard graph-matching algorithms and the DBMS primitives of grouping, sorting, and joining could be utilized to yield efficient match join operations. Finally, through experimental evaluation under various simulation conditions, and using synthetically generated data, the proposed methods produced excellent performance results in terms of execution efficiency and scalability. This paper aimed at mining mobile UMBPs such that movement locations associated with suitable services could be predicted and recommended for users.

REFERENCES

[1] H. Cheng, X. Yan, and J. Han, “IncSpan: Incremental mining of sequential patterns in a large database,” in Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Seattle, WA, Aug. 2004, pp. 527–532.

[2] J. L. Huang, M. S. Chen, and W. C. Peng, “Exploring group mobility for replica data allocation in a mobile environment,” in Proc. ACM Int. Conf.Inf. Knowl. Manage., New Orleans, LA, Nov. 2003, pp. 161–168.

[3] I. J. Perez, F. J. Cabrerizo, and E. Herrera-Viedma, “A mobile decision support system for dynamic group decision-making problems,” IEEE Trans. Syst., Man, Cybern. A, Syst., Humans, vol. 40, no. 6, pp. 1244–1256, Nov. 2010.

[4] V. Terziyan and O. Vitko, “Bayesian metanetworks for modelling user preferences in mobile environment,” in KI 2003: Advances in Artificial Intelligence. Berlin, Germany: Springer-Verlag, 2003, pp. 370–384.

[5] H.-W. Tsai, C.-P. Chu, and T.-S. Chen, “Mobile object tracking in wireless sensor networks,” Comput. Commun., vol. 30, no. 8, pp. 1811–1825, Jun. 2007.

[6] A. Pashtan, A. Heusser, and P. Scheuermann, “Personal service areas for mobile web applications,” IEEE Internet Comput., vol. 8, no. 6, pp. 34–39, Nov./Dec. 2004.

[7] W. C. Peng and M. S. Chen, “Developing data allocation schemes by incremental mining of user moving patterns in a mobile computing system,”IEEE Trans. Knowl. Data Eng., vol. 15, no. 1, pp. 70–85, Jan./Feb. 2003.

[8] T. Li, W. Peng, C. Perng, S. Ma, and H. Wang, “An integrated data-driven framework for computing system management,” IEEE Trans. Syst., Man,Cybern. A, Syst., Humans, vol. 40, no. 1, pp. 90–99, Jan. 2010.

[9] J. Monnot and S. Toulouse, “The path partition problem and related problems in bipartite graphs,” Oper. Res. Lett., vol. 35, no. 5, pp. 677–684, Sep. 2007.

[10] G. Yavas, D. Katsaros, Ö. Ulusoy, and Y. Manolopoulos, “A data mining approach for location prediction in mobile environments,” Data Knowl.Eng., vol. 54, no. 2, pp. 121–146, Aug. 2005.