

DYNAMIC LOCATION AREA PLANNING IN CELLULAR NETWORK USING FREQUENT PATTERN MINING

SAMIRAN SAIKIA, SALEENA B, PRAKASH B

School of Computing Science and Engineering, VIT Chennai Campus, Chennai, Tamil Nadu, India. Email: asha.s@vit.ac.in

Received: 19 January 2017, Revised and Accepted: 20 February 2017

ABSTRACT

Frequent pattern (FP) mining algorithms as the name says, mines sets FP from given datasets. These algorithms provide immensely helpful results which have a wide scope of application starting from simple decision problems to complex business intelligence aspects. This paper attempts to apply the same concept of FP mining to solve the location management problem of global system for mobile communication (GSM) networks. In GSM networks, the task of keeping track of a mobile user (MU) and relaying an incoming call is called location management. It basically includes two processes, location update, and paging. Location update deals with managing the current location of the MUs. There are many approaches to do this such as time based, movement based, and distance based. In this paper, the location update procedure relies on the collected data of user movements from one network cell to another cell. These data have a definite pattern, as in daily life a person mostly has a fixed route of traveling, e.g. home to office in the morning and office to home in the evening. During this movement, he crosses a specific set of network cells which remains same throughout the week. Thus, FP mining algorithm can be applied on the user's mobility log and try to find out the most probable location where the MU could be found. Using the results, a dynamic location area for individual user's current location can be created. Thus, minimizing cost related to location update which otherwise involves communication between the mobile handset and the base station, and calculations related to keeping track of the location of MUs.

Keywords: Apriori algorithm, Dynamic location area, Global system for mobile communication, Frequent pattern mining.

© 2017 The Authors. Published by Innovare Academic Sciences Pvt Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>) DOI: <http://dx.doi.org/10.22159/ajpcr.2017.v10s1.19748>

INTRODUCTION

Mobile communication systems have evolved a great deal since its conception. Today, the world is a small place only due to huge advancements in communication and transportation technology. Global system for mobile communication (GSM) systems form an inseparable part of the communication paradigm. With this technology communicating with anyone in any part of the world at our convenience has been possible. The network providers provide the infrastructure through which this is possible. One aspect of GSM systems is to keep track of the mobile phone users to provide the necessary services. Here, the concept called location management comes into play. Location management in cellular networks deals with how to keep track of an active mobile station within the cellular network. There are ample numbers of location management schemes that have been proposed and implemented such as simulated annealing [5], ant colony optimization [6], user based personalization schemes [8], Cartesian coordinate system based [9]. One of the most feasible location management schemes is the location area creation method.

In this scheme, the whole service area is first divided into smaller service areas called location area. Moreover, each location area is further divided into several contiguous subdivisions called cells. Depending on the creation of location areas, it is divided into static and dynamic methods. The static methods do not provide a robust architecture thus dynamic methods had to be adopted. Two important operations in location area scheme are performed to keep track of the mobile user (MU). These are (a) Location update and (b) paging. Location update is an operation performed by the mobile station (cell phone) when it moves from one location area to another. And paging is performed by the cellular network when a call is inbound to a mobile station; the network pages all the possible cells to find in which cell the mobile station may be present. There is a delicate balance between the amount of location updates and paging operation. Considering two extremities which are in one scenario, if the mobile station performs location update for every cell to cell movement, then

the communication traffic will be more and the cell phone power consumption will also be more. Another scenario is where location update is not performed at all and when a call is inbound; the network has to page each and every cell in search of the mobile station thus the load on the network will be huge. Thus, a middle point has to be found, a correct combination of the amount of paging and location updates that need to be done should be considered. The total location management cost depends on these two factors: Location update and paging cost. Minimizing any one will minimize total location management cost.

The paper is organized in the following manner. In section II, description of frequent pattern (FP) mining algorithms namely Apriori algorithm and FP-growth is given and how it is feasible for application in location area creation. In section III, the implementation procedure is addressed. In section IV, the results are analyzed and interpretations are provided finally the paper is concluded in section V along with a note on future enhancements.

FP MINING

FP mining is data mining techniques that are applied on various types of datasets so that various types of patterns or trends, associations etc. can be discovered in the dataset. Two trending pattern mining algorithms are the Apriori algorithm and the FP-growth algorithm. Apriori algorithm was introduced by Agarwal and Srikant in 1994 and FP growth by Han, Per and Yin in 2000.

Apriori algorithm

Apriori algorithm follows a property that any subset of a given FP must also be frequent, and for any in FP, its superset should not be generated or tested. It employs a level-wise search, starting with frequent 1-itemset, at each level k it generates $k+1$ candidates from an FP of length k and prunes (removes) the infrequent items at each iteration. The process continues till no more frequent itemsets can be found.

Apriori algorithm:

Input: Number of transactions (t), i.e., dataset, minimum support (minsup). Suppose $L_1, L_2, L_3, \dots, L_k$ is the itemsets where L_1 is the 1-itemset, L_2 is the 2-itemset and L_k is the k-itemset. D is the dataset. C_k is the K candidate itemset [1,2].

1. First of all find the frequent of 1-itemset, i.e., L_1 from the dataset D
2. For ($k=2; L_k-1 \neq \phi; k++$)
3. {
4. $C_k = \text{Apriori_gen}(L_{k-1}, \text{minsup});$
5. For each transaction $t \in D$
6. {
7. $C_t = \text{subset}(C_k, t); // \text{candidates contained in } t$
8. For each candidate $c \in C_t$ do
9. $c.\text{count}++;$
10. }
11. $L_k = \{c \in C_k | c.\text{count} \geq \text{minsup}\};$
12. }

FP-growth

The disadvantage of Apriori algorithm is that it has to generate candidate sets at each iteration. For large datasets, this candidate generation reaches huge numbers. Moreover, another drawback is the requirement of multiple scans of the database. It needs $n+1$ scans where n is the length of the longest pattern.

To overcome these drawbacks Han, Per and Yin in 2000 proposed another algorithm the FP-growth algorithm. This algorithm generates FP without candidate generation. This algorithm is highly condensed and it avoids a large number of database scans. The logic is to develop an FP-tree and from it extract all the FP. The steps are given in short [3]:

1. Scan database to find frequent 1-itemset
2. Order frequent items in frequency descending order
3. Scan database a second time to construct the FP-tree.

A path in an FP-tree represents a frequent itemset and the nodes are ordered according to the decreasing frequency of the items. FP-growth is faster than Apriori as candidate generation is not carried out. It uses a compact data structure and it has no repeated dataset scan. And counting is the elementary operation rather than pattern matching [4]. This paper deals with implementation of the Apriori algorithm to construct dynamic location area. Use of FP-growth is a future aspect of the project.

Application of FP mining to create dynamic location area

Application of these algorithms to create dynamic location areas in GSM network is feasible due to the following characteristics. A person in his daily life travels from one place to another following a definite pattern. The person follows this trend almost each day of the week say from his home to his place of work he has one or more than one but small and countable number of paths. While traveling, he passes through a specific number of network cells following a pattern. This pattern can be seen more than once in a week. Thus, if FP mining algorithms are applied on the mobility data of a mobile, the pattern of his movements and his correct location can be predicted without having to perform the location update operation. Although location update needs to be performed at first to build up the mobility log of the user. After a definite amount of data has been collected, finding the user location can be relied upon the mined data. A copy of the mined data can be saved on the mobile station, and the mobile station need not perform location update as long as it follows the mined patterns, but if it deviates, it can perform a location update and notify the network about the location of the mobile station.

IMPLEMENTATION PROCEDURE

Master database

This database belongs to the GSM infrastructure. This database keeps logs of user mobility, i.e., it holds the cell ids of all the network cells

that the user has visited. Other than these data, it also holds other information related to the mobile station which is irrelevant to this system. Thus, only the mobility log is extracted.

Data preprocessing

The mobility log will have data in a format which is not directly feedable to the Apriori algorithm, so the data is converted into a readable format by the Apriori algorithm. This contains converting the values to binary format and structuring them to syntax as shown in Fig. 2.

Data mining servers

Fig. 2 shows mobility log of one user; there will be thousands of users each having mobility log of their own thus to process them efficiently server systems are needed to apply the FP mining algorithm on the data. Finally, the FP indicating users mostly visited cell ids is obtained. That is stored in another database which when queried will give the probable location of the mobile station.

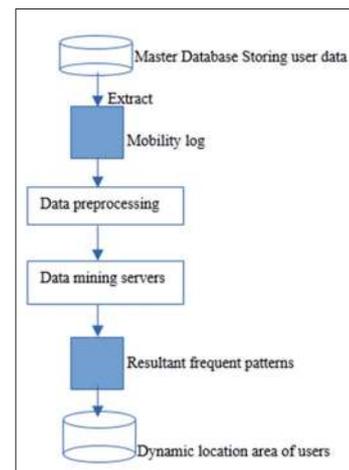


Fig. 1: Architecture of the system

```

relation mobile_user_7

@attribute cell-14 {0,1}
@attribute cell-25 {0,1}
@attribute cell-26 {0,1}
@attribute cell-34 {0,1}
@attribute cell-51 {0,1}
@attribute cell-33 {0,1}
@attribute cell-11 {0,1}
@attribute cell-1 {0,1}
@attribute cell-28 {0,1}
@attribute cell-24 {0,1}

@data
1,1,1,1,1,1,1,1,0,0
1,1,1,1,0,1,1,1,0,0
1,1,1,0,1,1,1,1,0,0
1,1,0,0,1,1,1,1,1,1
1,1,1,1,0,0,1,1,0,0
1,1,1,0,0,0,0,0,0,0
1,1,1,1,0,1,1,1,0,0
1,1,1,1,1,1,1,1,0,0
1,1,1,1,1,1,1,1,0,0
1,1,0,1,1,0,1,1,0,0
  
```

Fig. 2: Preprocessed data of mobile user 7

Procedure

Individual user’s mobility information is recorded and the cell ids of all the visited network cells are stored. This information is shown in Table 1.

Now for each user, the data for each day are represented in Table 2 here, a value 1 means that the MU visited that cell and a value 0 means the user did not visit that cell on that day. Table 2 shows the data for MU 7.

Table 2 is given as input to the Apriori algorithm and result is obtained as shown in Fig. 5. The Apriori algorithm can be applied with different minimum support to create different dynamic location areas. Considering minimum support as 100%, those cells which the MU visit every day is obtained. Using minimum support as 70-80% a good estimate of the cells the user visits mostly could be found. Thus, those groups of cells can be first ones where the network tries to find the MU in case there is an incoming call. Again if the user is not found in those cells, the cells which are obtained using minimum support as 60%, 40%, 20%, etc., could be used. This will act as adaptive search.

Fig. 4 shows the different parameters that have been set to get the FP. Here, the parameter “UpperboundMinSupport” is set to 0.7 so that calculation starts from 70% minimum support. “OutputItemsets” is set to true will give the frequent item sets.

In Fig. 5, the final output for frequent item sets is shown as:

Large itemsets L(6):

cell-14=1 cell-25=1 cell-26=1 cell-11=1 cell-1=1 cell-28=0.7

cell-14=1 cell-25=1 cell-34=1 cell-11=1 cell-1=1 cell-28=0.7

This shows that for 70% of time the user was present in cells 14, 25, 34, 11 and 1. Thus, if the MU needs to be located, the network can start searching in these cells. This set of cell ids represents the dynamically created location area. In this way, the process is carried out for each user and a dynamic location area is created.

Table 1: Visited cell ids of 10 MUs over a period of 10 days

MU number	Frequently visited cell ids
1	1, 25, 35, 26, 24, 23
2	12, 15, 39, 48, 57, 16, 42, 14, 15
3	3, 8, 9, 5, 6, 4, 15
4	55, 45, 32, 33, 43, 42, 30, 40, 54
5	9, 45, 42, 43, 57, 48, 8, 46
6	1, 9, 8, 16, 7, 10, 6, 14
7	14, 25, 26, 34, 51, 33, 11, 1, 28, 24
8	3, 6, 10, 12, 8, 5, 1
9	1, 5, 6, 8
10	11, 14, 16, 15, 12

MU: Mobile user

Table 2: Per day mobility log of MU 7

Day	Cell 14	Cell 25	Cell 26	Cell 34	Cell 51	Cell 33	Cell 11	Cell 1	Cell 28	Cell 24
1	1	1	1	1	1	1	1	1	0	0
2	1	1	1	1	0	1	1	1	0	0
3	1	1	1	0	1	1	1	1	0	0
4	1	1	0	0	1	1	1	1	1	1
5	1	1	1	1	0	0	1	1	0	0
6	1	1	1	0	0	0	0	0	0	0
7	1	1	1	1	0	1	1	1	0	0
8	1	1	1	1	1	1	1	1	0	0
9	1	1	1	1	1	1	1	1	0	0
10	1	1	0	1	1	0	1	1	0	0

MU: Mobile user

RESULTS

The Apriori algorithm is executed various times on the data with different parameters. The final result of the algorithm is divided into 20%, 60%, and 100% minimum support and shown in Table 3. The values under 100% minimum support are the cell ids where the MU’s visits every day. And values under 60% indicate the cell ids where the MU visits at least 6 days out of 10 days. These cell ids under 60% can be used to locate the mobile in case of an incoming call to that MU. If he is not found in these cells, then the network can search for the user in the cell ids mentioned under the 20% column.

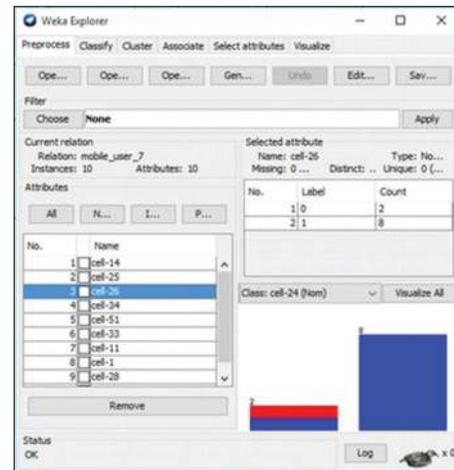


Fig. 3: Importing mobile user 7 data set into Weka

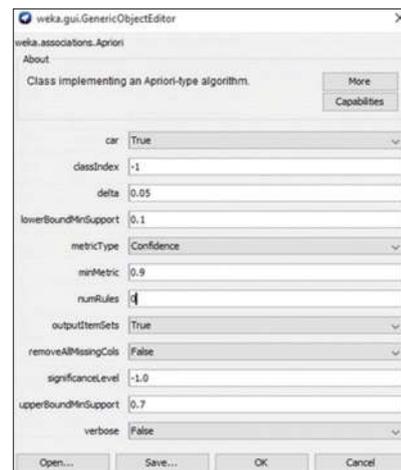
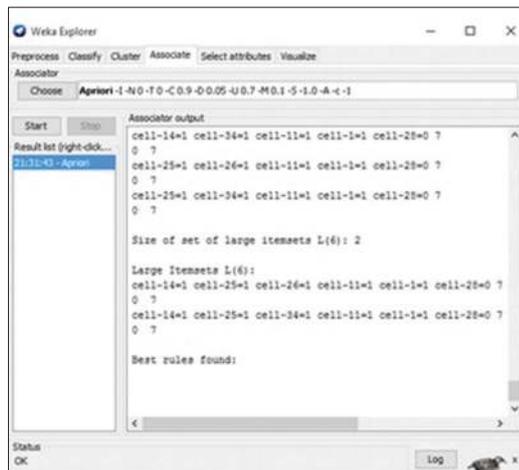


Fig. 4: Parameters while running the Apriori algorithm

Table 3: Mobile unit's visited cells based on minimum support

MU	Min support 20%	60%	100%
1	1, 25, 35, 26, 24, 23	1, 25, 35, 26, 24	1, 25, 26, 24
2	12, 15, 39, 48, 57, 16, 42	12, 15, 39, 48, 16	12, 15, 39
3	3, 8, 9, 5, 6	3, 8, 9, 5, 6	3, 8, 9, 6
4	55, 45, 32	55, 45, 32	55, 45, 32
5	9, 45, 42, 43, 57, 48, 8	9, 45, 42, 43, 48, 8	9, 45, 43
6	1, 9, 8, 16, 7, 10	1, 9, 8, 16, 7	1, 9, 8
7	14, 25, 26, 34, 51, 33, 11, 1	14, 25, 26, 34, 33, 11, 1	14, 25
8	3, 6, 10, 12, 8, 5, 1	3, 6, 12, 8	3, 6, 8
9	1, 5, 6, 8	1, 5, 6, 8	1, 5, 6, 8
10	11, 14, 16, 15	11, 14, 16	11, 14, 16

MU: Mobile user

**Fig. 5: Result after Apriori algorithm is applied**

CONCLUSION AND FUTURE WORK

Apriori algorithm has been used to create dynamic location area for individual MUs. By changing the minimum support, different location area sets with different probability of availability of the MUs could be

found. Each user has his own location area profile based on which the network can route an incoming call. 100% minimum support gives the cell ids in which the user visits every day. If he is not available in those cell ids, then the cell ids which are obtained using the minimum support as 60% or 70% could be used as alternate search areas.

As FP-growth algorithm is better than Apriori, as a future work instead of Apriori, FP-growth can be implemented. Another aspect is to consider the most frequented cell ids irrespective of the MU and form clusters to indicate in which cell ids users are more frequent and in which they are not.

REFERENCES

1. Agrawal R, Imielinski T, Swami AN. Mining association rules between sets of items in large databases. Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data. Washington, DC, May; 1993. p. 207.
2. Aggrawal R, Srikant R. Fast algorithms for mining association rules in large databases. Santiago, Chile: Proceedings of the 20th International Conference on Very Large Data Bases (VLDB), September; 1994.
3. Wu B, Zhang D, Lan Q, Zheng J. An efficient frequent patterns mining algorithm based on apriori algorithm and the FP-tree structure. Third International Conference on Convergence and Hybrid Information Technology; 2008.
4. Borgelt C. An Implementation of the FP-growth Algorithm. Germany: Department of Knowledge Processing and language Engineering; 2005.
5. Prajapati NB, Agravat RR, Hasan MI. Simulated annealing for location area planning in cellular network. Int J Appl Graph Theory Wirel Ad Hoc Netw Sensor Netw Graph Hoc. 2010;2(1):???
6. Prajapati NB, Agravat RR, Hasan MI. Comparative study of various cooling schedules for location area planning in cellular networks using simulated annealing. Le Royal Meridian, Chennai, India, 27-29 December; 2009.
7. Mudaliar K, Swadas P, Prajapati N. Location management for cellular network using ant colony optimization. International Conference on Recent Trends on Computer Technology in Academia, 21-23 April; 2012.
8. Mudaliar K, Prajapati N. User based personalization scheme for dynamic location management. International Conference "ICNICT 11" at Karpagam University, Coimbatore, 15-16, December; 2011.
9. Markande SD, Bodhe SK. Cartesian coordinate system based dynamic location management scheme. Int J Electron Eng Res 2009;2:63-9.
10. Rong C, Senmiao Y. Distributed and Dynamic Location Area for PCS. IEEE, 1-4244-0463-0/06.