www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 4 Issue 5 May 2015, Page No. 12000-12004

Hidden Markov Model With Biclustering Cache Replacement Policy For Location Based Services In MANET

Ms. Bhakti D. Shelar [1], ***Mr. D. K. Chitre*** [2]

[1] Student, M.E. (Computer), Terna Engineering College,
Nerul , Navi Mumbai.
shelar.bhakti@gmail.com

[2]Associate Professor, Terna Engineering College, Nerul, Navi Mumbai.
dkchitre@rediffmail.com

Abstract: ***System performance of mobile client is very important in mobile environment. Frequently accessed data items are cached to improve performance. Cache replacement technique is used to find appropriate data items for eviction from cache. Selection of suitable cache replacement strategy is very important as cache size is limited. Available policies do not take into account the movement patterns of the client. Here, we propose a new cache replacement policy for location dependent data in mobile environment. We use hidden markov model as a location prediction tool and then huge data is clustered as per location and type of data. This makes the policy adaptive to client's movement pattern unlike earlier policies that consider the directional / non-directional data distance only. Simulation results show that the proposed policy significantly improves the system performance in comparison to previous schemes in terms of cache hit ratio.***

Keywords: Hidden Markov Model, Cache replacement Policy, Mobile Adhoc network, temporal sequence, Bi clustering, Location Prediction.

1. Introduction

Recent advances in wireless technology have ushered the new paradigm of mobile computing. With the advent of new mobile infrastructures providing higher bandwidth and constant connection to the network from virtually everywhere and advances in the global positioning technologies, a new class of services referred to as Location Based Services(LBS) has evolved. Mobile-location dependent based services (LBS) have become increasingly popular in recent years. However, data caching strategies for LBSs have thus far received little attention.

1.1 Location Based Services(LBS)

Location-based services (LBS) are a group of computer program-level services that use location related data to estimate the appropriate specialties. It is an informative system and has a number of uses in social life today as an entertainment service. LBS is accessible with mobile devices through the mobile adhoc network and which uses information on the geographical position of the

mobile device. Smart phones are capable to sustain such services, hence LBS Apps can be installed on smart phones. Mobile users can access the data while roaming around.

LBS are used in a variety of areas like object search, personal interests, entertainment and health. As shown in the figure 1, LBS services are provided by the service provider which requires geopositioning. They are able to work with wireless communications. As it is installed on the mobile devices, it supports access of required data from any location. LBS include services can recognize a location of a person, such as finding the nearest service for e.g. Restaurant., or ATM or nearest Railway Station. LBS include parcel tracking and Vehicle Tracking services. LBS can include mobile commerce. when taking the form of coupons or advertising directed at customers based on their current location. They include personalized weather services and even location-based games. They are an example of telecommunication convergence.

LBS is the ability to open and close specific data objects based on the use of location and/or time as (controls and triggers) or as part of complex cryptographic key or hashing systems and the data they provide access to. Location based services today are a part of everything from control systems to smart weapons. They are actively used trillions of times a day and may be one of the most heavily used application-layer decision framework in computing today.

Figure: 1 Location Based Services Concept Diagram

1.2 Cache Replacement in Mobile nodes

Location Based Services are becoming popular day by day. Advances in wireless technology have ushered . But, there are some challenges in providing LBS like limited client power, limited bandwidth, intermittent connectivity, etc. Caching the required data is a solution to meet these challenges. Caching of frequently accessed data item is an effective technique to improve data access cost.LBS provide users with the ability to access impossible to hold all accessed data items in the cache. Thus, there is a need of efficient cache replacement strategy to find out suitable subset of data items. Good cache performance heavily depends on the selection of data items for eviction from the cache. With the help of our portable device we can access frequency, update rate, size of objects etc. There are various categories of cache replacement techniques such as temporal based, frequency based, recency based, current location based, future location based, distance based etc. In temporal based policies, we consider time when data in cache is accessed. Very old data is considered not to be used again, so it may be eliminated.

Frequency based policies consider how many times data is requested from the user. They replace data which is very rarely requested. Recency based policies keep the recent data in the cache and delete the data which is requested before longer period. Current location based policies find the current location of moving client. Then data which is related with this location is maintained in the cache. Some policies predict the possible future location of client, where client is likely to visit and then replace the data unrelated with this location. Some policies consider the distance of data in cache and the location. The data items which are away from the user's location is replaced.

The rest of the paper is organized as follows. Section 2 briefly describes survey of the different cache replacement policies. Section 3 details the Hidden Markov Model Prediction, Section 4 explains Bi clustering methodology. Section 5. and section 6 deal with Experimental scenario and Result Analysis receptively. Section 7 concludes the paper.

2. Literature Survey

In 2012, Wesley Mathew, proposed[2] prediction of future location of moving client. For this he used Hidden Markov Model which analyses the sequences of location histories of user. But this method only predicts location based on historical analysis.

In 2002, B. Zheng, proposed[6] cache replacement strategy which considers access probabilities of data object. Also it uses valid scope and distance of cached data from user's location. But this strategy doesn't think about size of the data and doesn't give priority to data from nearest location.

In 2009, S. Dar proposed [8]Manhattan Distance based cache replacement policy which makes cache replacement decisions based on the basis of client's current location and location of each cached data object. But this policy ignores temporal access locality and moving direction of client.

In 2008, Q. Ren proposed [5]Furthest Away Replacement policy which considers direction in which user is moving. So data objects that are from out-direction are evicted from cache and in-direction data are maintained in the cache.

In 2009, E. O'Neil [4]proposed Least Recently used cache replacement policy which considers access time of cached data object. The data which is no queried from long period, is evicted from the cache. Recent queried or accessed data is maintained in the cache. But this policy ignores location of the user.

In 2004, K. Lai, [7] proposed Mobility Aware Replacement Scheme(MARS), which comprised of temporal score, spatial score and cost of retrieving an object. This policy ignores the impact of client's anticipated location while making cache replacement.

3. Hidden Markov Model Prediction

Hidden Markov Models (HMMs) are a well-known approach for the analysis of sequential data, in which the sequences are[1] assumed to be generated by a Markov process with un-observed (i.e., hidden) states. Thus, by using HMMs for modeling location sequences, the states governing the moving agent's decisions are not directly visible, but the visited locations, dependent on the state, are visible. Each state has a probability distribution over the possible locations to be visited, in our case a categorical distribution over the triangles representing each possible area in the Earth's surface. Each state has also a probability distribution over the possible transitions to the state that is going to govern the decision about the next place to be visited in the sequence.

The diagram in Figure 2 shows the general architecture of an instantiated HMM. Each shape in the diagram represents a random variable that can adopt any of a number of values. The random variable x(t) is the hidden state at time t (in the model from the above diagram, x(t) €{x1, x2, x3}). The random variable y(t) is the location visited at time t (with y(t) €{y1, y2, y3, y4}). The arrows in the diagram denote conditional dependencies.

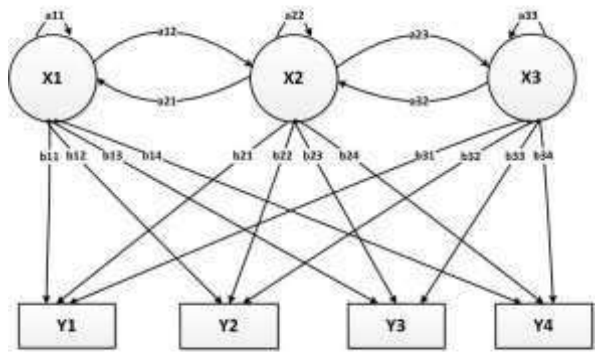


Figure: 2. Example Hidden Markov Model.

From the diagram, it is clear that the conditional probability distribution of the hidden variable x(t) at time t, given the values of the hidden variable x at all times,[1] depends only on the value of the hidden variable x(t1), and thus the values at time t2 and before have no influence. This is called the Markov property. Similarly, the value of the observed location y(t) only depends on the value of the hidden variable x(t), at time t.

Several inference problems can be addressed on top of instantiated HMMs, one of them being the computation of the probability for a given observation sequence (i.e., a sequence of visits to locations). The task is to compute, given the parameters of the instantiated model, the probability of a particular output sequence being observed. This requires computing a summation over all possible state sequences.
The probability of observing a particular sequence in the form Y =< y(1), y(2),…., y(L) >, of length L, is given by:

$$P(Y) = \sum P(Y|X)\, P(X)$$

In the formula, the sum runs over all possible hidden-node sequences X =< x(1), x(2),…, x(L) >. Applying the principle of dynamic programming, this problem can be handled efficiently, using a procedure known as the forward algorithm, which we outline further ahead. As for HMM parameter learning, the task is to find, given an output sequence X or a set of such sequences, the best set of state transition and output probabilities, i.e., the values for $a_{i,j}$ and $b_i(k)$ from Figure 3. The task is usually to derive the maximum likelihood estimate of the parameters of the HMM, given the set of output sequences. No tractable algorithm is known for solving this problem exactly, but a lo-cal maximum likelihood can be derived efficiently using the Baum-Welch algorithm, which is an example of a forward-backward algorithm and a special case of the well-known expectation-maximization (EM) algorithm.

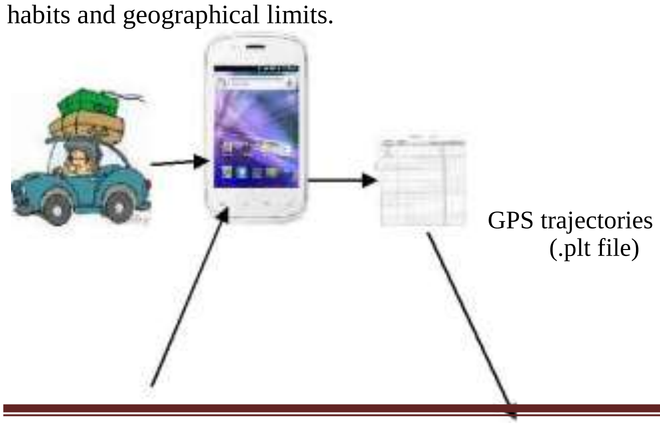
4. Biclustering Methodology

GPS data is used in location-based services (*LBSs*) to provide location recognitions. We observe that most existing LBSs provide location recommendations by clustering the User-Location matrix. User-Location matrix created based on GPS data is huge. First problem is, the number of similar locations that need to be considered in computing the recommendations can be numerous. As a result, the identification of truly relevant locations from numerous candidates is challenging. Second, the clustering process on large matrix is time consuming. Thus, when new GPS data arrives, complete re-clustering of the whole matrix is infeasible. To tackle these two problems, we propose the Bi Clustering. By considering activities (i.e., temporal preferences) and different user classes (i.e., Pattern users, Normal Users, Travelers) in the recommendation process, CLR is capable of generating more precise and refined recommendations to the users compared to the existing methods. One of the major functions of a location-based service (LBS) is to recommend interesting locations to the users. Most existing LBSs recommend all points of interest (POIs) that are near the users. These POIs can be numerous. Thus, it is difficult for a user to pick the most relevant suggestions from the large number of recommendations. In order to provide more accurate location suggestions according to the users' interests, many existing LBSs provide location recommendations by first clustering the User-Location matrix, which represents the locations visited by each user, and then making location recommendations based on the user and location clusters. We get information that

(1) locations exhibit spatial properties as they contain geographical information.

(2) activities exhibit temporal-spatial properties as they represent temporal sequences of visited locations.

(3) users exhibit long-term spatial properties as different users tend to visit different locations according to their long term habits and geographical limits.



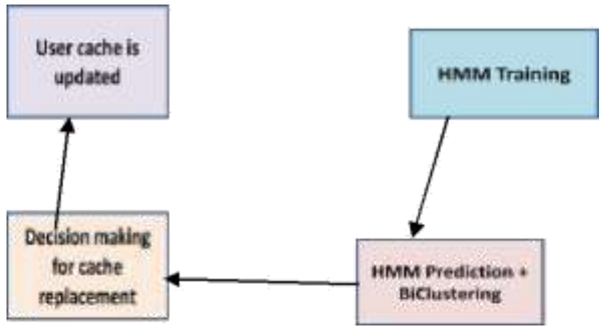


Figure:3. System Architecture

As shown in the figure 3, moving client fires the query from its smart phone LBS application. Client's location is trapped in the .plt file of GPS trajectory. Then HMM training takes place by using available output sequences of locations. HMM predicts the probable future location and then Biclustering is applied on the data of predicted location. Data items are selected for the eviction from cache. The client is provided with the updated cache.

5. Experimental Evaluation

The training of the HMM models took between 21 and 318 minutes on a standard laptop PC, with results varying ac-cording to the number of states. In the proposed method, the previously collected location histories are first clustered according to their characteristics (i.e., according to the temporal period in which they occurred). Afterwards, the clusters are used to train different Hidden Markov Models (HMMs) corresponding to the different types of location histories (i.e., one HMM for each cluster). Given a new sequence of visits, from which we want to discover the particular location that is more likely to be visited next, we start by finding the cluster that is more likely to be associated to the particular sequence of visits being considered in the prediction task, and then we use inference over the corresponding HMM in order to discover the most probable following location. Each of the places in a given sequence of visits is associated to a timestamp and to the corresponding HMM in order to discover the most probable following location. Each of the places in a given sequence of visits is associated to a timestamp and to the corresponding geospatial coordinates of latitude and longitude. The initial clustering of sequences is based on the temporal period associated to each sequence, and we use the timestamp associated to the last place visited in the sequence, in order to group sequences ac-cording to three clusters.

We have compared Prediction methodology of HMM with temporal based cache replacement policy, Least Recently Used (LRU) and distance based cache replacement policy, Furthest Away Replacement(FAR).

5.1 Performance Metrics

Primary performance metric is cache hit ratio. Cache hit ratio is defined as ratio of number of queries answered by client's cache to the total number of queries generated by client. Higher the cache hit ratio, higher is the local data availability. The cache size is given in terms of bytes of data objects.

5.2 Geolife Database

We implemented the proposed approach for pedestrian movement prediction through an existing implementation of the algorithms associated with Hidden Markov Models (i.e., the Baum-Welch and the forward-backward procedures), latter

Validating proposed ideas through experiments with the Geolife dataset, i.e. a GPS trajectory dataset collected in the context of the Geolife project from Microsoft Research Asia, by 178 users in a period of over three years from April 2007 to Oct. 2011. In the Geolife Dataset , a GPS trajectory is represented by a sequence of time-stamped points, each of which containing latitude and longitude coordinates. The full dataset contains 17,621 trajectories with a total distance of about 1.2 million kilometers and a total duration of 48000+ hours.[15][16][17]

6. Result Analysis

In the experiments, we considered the cache size in terms of bytes and then vary the cache size 5,10,15 and 20bytes. Cache hit ratio is calculated for every size of cache. Cache hit ratio for LRU,FAR,HMM and HMM+Biclustering are given in tabular form.

Table 1 Comparison table of various Cache Replacement Policies

CACHE SIZE(in terms of no. of entries)	CACHE HIT RATIO			
	LRU	FAR	HMM	HMM+Biclustering
5	0.1	0.12	0.14	0.17
10	0.13	0.14	0.16	0.24
15	0.16	0.15	0.23	0.25
20	0.2	0.22	0.24	0.29

Figure 4 indicates the performance of the four cache replacement policies. Blue line indicates cache hit ratio of HMM + Biclustering, green line indicates cache hit ratio of HMM, red line indicates cache hit ratio of FAR and light blue line indicates cache hit ratio of LRU.

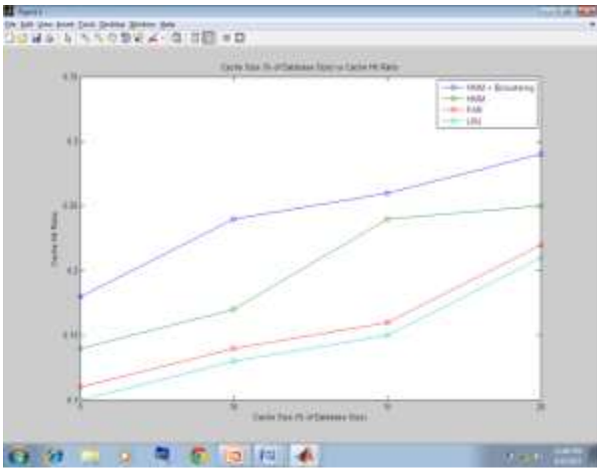


Figure:4. Cache hit ratio obtained by LRU, FAR, HMM and HMM with Biclustering policies.

7. Conclusion and Future Scope

In proposed cache replacement policy we emphasized on predicting region according to client's current position apart from considering only user's direction or distance. HMM with Biclustering approach gives the best results in terms of improved cache hit ratio. The cache hit ratio obtained by proposed approach is near 0.35. This cache hit ratio is stable though we vary size of cache. For future work, we would also like to experiment the Mixed Markov Model and extension of Conditional Random Fields(CRFs) which can be used in association with Bi Clustering. This is kept for future. Also HMM can also be corporated with Conditional Random Fields(CRFs). This model deals with pattern recognition of the frequently used data. This is a state space model which attempts to capture variations in spatial sequences.

REFERENCES

[1]Ajey Kumar, Manoj Misra, Anil K. Sarje, "A Predicted Region based Cache Replacement Policy for Location Dependent Data in Mobile Environment", I. J. Communications, Network and System Sciences. 2008, 1: 1-103

[2] W. Mathew, Ruben Raposo,"Predicitng Future Locations with Hidden Markov Models", 2012 ACM 978-1-4503-1224-0/12/09 [3]D. Barbara and T. Imielinski, "Sleepers and Workaholics: Caching Strategies in Mobile Environments," In the Proceedings of the ACM SIGMOD Conference on Management of Data, Minneapolis, USA, pp. 1-12, 1994.

[4]E. O'Neil and P. O'Neil, "The LRU-k page replacement algorithm for database disk buffering", In the Proceedings of the ACM SIGMOD, Vol. 22, No. 2, pp. 296-306, 1993.

[5]Q. Ren and M. H. Dhunham, "Using Semantic Caching to Manage Location Dependent Data in Mobile Computing," In the Proceedings of 6th ACM/IEEE Mobile Computing and Networking (MobiCom), Boston, USA, pp. 210-221, 2000.

[6]B. Zheng, J. Xu and D. L. Lee, "Cache Invalidation and Replacement Strategies for Location-Dependent Data in Mobile Environments," IEEE Transactions on computers, Vol. 51, No. 10, pp. 1141-1153, October 2002.

[7]K. Lai, Z. Tari and P. Bertok, "Location–Aware Cache Replacement for Mobile Environments," IEEE Global Telecommunication Conference (GLOBECOM 04), Vol. 6, pp. 3441-3447, 29th November- 3rd December 2004.

[8]S. Dar, M. J. Franklin, B. T. Jonsson, D. Srivastava and M. Tan, "Semantic Data Caching and Replacement," In the Proceedings of the 22nd International Conference on Very Large Databases(VLDB), pp. 330-341,1996.

[9]Balamash and M. Krunz, "An Overview of Web Caching Replacement Algorithms," IEEE Communications Surveys & Tutorials, Vol. 6,

[10]D. L. Lee, W.-C. Lee, J. Xu and B. Zheng, "Data Management in Location-Dependent Information Services," IEEE Pervasive Computing, Vol. 1, No. 3, pp. 65-72, July 2002.

[11]J. Jing, A. Helal and A. Elmagarmid, “Client-Server Computing in Mobile Environments,” In the Proceedings of the ACM Computing Surveys, Vol. 31, No. 2, pp. 117-157, June 1999.
[12]T. Camp, J. Boleng and V. Davies, “A Survey of Mobility Model for Ad Hoc Network Research,” Wireless Communication & Mobile Computing (WCMC): Special Issue on Mobile AdHoc Networking: Research, Trends and Applications, Vol. 2, No. 5, pp. 483-502, 2002.
[13]A. Balamash and M. Krunz, “An Overview of Web Caching Replacement Algorithms,” IEEE Communications Surveys & Tutorials, Vol. 6, No. 2, pp. 44-56, 2004.
[14]I. A. Getting, “The Global Positioning System,” IEEE Spectrum, Vol. 30, No. 12, pp. 36-47, December 1993.
[15] Yu Zheng, Quannan Li, Yukun Chen, Xing Xie, Wei Ying Ma. Understanding Mobility Based on GPS Data. In Proceedings of ACM conference on Ubiquitous Computing (UbiComp 2008), Seoul, Korea. ACM Press: 312-321.
[16] Yu Zheng, Xing Xie, Wei-Ying Ma, GeoLife: A Collaborative Social Networking Service among User, location and trajectory. Invited paper, in IEEE Data Engineering Bulletin. 33, 2, 2010, pp. 32-40.
[17] B. Zheng, J. Xu and D. L. Lee, “Cache Invalidation and Replacement Strategies for Location-Dependent Data in Mobile Environments,” IEEE Transactions on Computers, Vol.51 oct 2002.

Author Profile

Bhakti Shelar received the B.E.(Computer) degree from Terna Engineering College, Nerul , Navi Mumbai. Worked in S. S. Jondhle College of engineering and now working in Dilkap Engineering College, Nerul. Pursuing M.E.(computer) from Mumbai University.