Prediction System for Traffic Congestion using GPS Data on Hadoop Cloud Storage

Hnin Thant Lwin, Thinn Thu Naing
University of Computer Studies, Yangon, Myanmar
hninthantlwin@gmail.com,thinnthu@gmail.com

Abstract

The high values of vehicles, the inadequate infrastructure cause traffic congestion. Congested roads can be avoided by determining the travel-time for a particular road ahead of time. Traffic prediction and travel time estimation has traditionally relied on expensive measuring methods such as loop detectors, vehicle identification devices. In this paper, we use mobile GPS equipments on vehicles to gather data for cheaper and real time travel-time estimation. We use this data to develop the prediction system for traffic congestion in order to improve the quality and safety of vehicle movement and for minimization the time and costs when vehicles are moved at the specified routes. We collect the GPS data and classify them with K-Means algorithm. Moreover, framework based on Markov model is used to predict traffic and Hadoop is used as cloud storage and platform, to accelerate the processing computing speed and allow handling of large-scale data.

Key Words : Traffic Prediction, GPS, Markov, Hadoop, MapReduce, K-Means.

1. Introduction

Millions of people waste time waiting in car queues to get to or from work, resulting in money loss as work time is wasted in everyday. As traffic volumes increase, the needs for precise travel-times estimates grow. Traditionally, estimating travel times has relied on slow and costly methods such as loop detectors, observations vehicles or automatic vehicle identification or floating car observers. The increasing popularity of mobile phones embedded with positioning functionality such as GPS is allowing users to easily acquire their own locations and collect their own trajectories, which can be used for various purposes such as location-based service applications. Therefore, new possibilities have opened for cheaper travel-time prediction [1,2].

Although dedicated moving observer or floating car vehicle-based methods can provide precise estimations, they require that an instructed driver collected the data needed. This is both time consuming and costly as the driver must be paid. This method also provides less data as a relatively small number of vehicles are usually used. Since road networks are ever changing and traffic volumes fluctuate, travel-time estimates must be recalculated occasionally or continually using current data to reflect these changes. For this reason, we use GPS data from mobile for the faster and cheaper methods.

Previous approaches to travel-time estimation include algorithms based solely on more or less educated guesses calculated from the permitted speed on a particular road segment, on finding weighted average given single observations, on data collected using expensive moving observer methods, or on the experience of traffic experts [2,1].

In this paper, we collect data from mobile GPS on the vehicles. We then cluster these GPS data using K-Means clustering algorithm. We then use Markov model to predict the traffic on a particular road ahead of time. We use Hadoop map reduce, a cloud computing platform, to accelerate the processing speed which can be used with real-world datasets rather than sampling data.

2. Related Work

Several papers are concerned with prediction travel- time estimation and actual path finding. Kanolus et al. [14] propose a method for finding the fastest path through a road network given the constraints of a time interval at either the start of or destination of the trip. Ku et al. [16] propose an adaptive nearest-neighbour query based on travel time instead of Euclidian or network distance.

In [12], Nielsen presents methods for using data recorded by GPS devices mounted in cars to analyze congestion. It is argued that using GPS data provides more knowledge than traditional methods as routes can be inferred from the stream of GPS observations. Hansen presents methods for analyzing congestion continually over extended periods of time, using GPS data.

Advances in GPS and tracking technology have motivated large efforts in classifying trajectories. Rajput *et al.* [13], proposed a basic framework by integrating the hypothesis of rough set theory (reduct) and k-means algorithm for efficient clustering of high dimensional data.

In [2, 12, 15] GPS equipped vehicles are used to collect samples at regular intervals, which are then used for estimating travel times. Many different sampling intervals are used 30 seconds in [10] and one second in [9]. Different systems will provide data recorded with different sampling rates, a solution independent of sampling rates has not been proposed to our knowledge. In [11] Quiroga et al. study the impact of changing sampling rate and road segment length using GPS.

Jyoti R. Patole use K-Means algorithm to cluster the wireless sensor network with Map Reduce algorithm [2]. We not only use the K-Means algorithm to cluster GPS data from mobile phone on vehicles with MapReduce algorithm but also predict using Markov chain to predict traffic from these clustering results.

3. Overall Architecture for Traffic Prediction

The general architecture for traffic prediction is described in Figure1. Android Device on vehicles is used for capturing GPS location data and transmitting the data to data center. GPS data provide with longitude, latitude, a timestamp, direction, speed and a unique identification for the recording GPS device. The data transmitted by android devices deployed in vehicles is collected and organized in a computing cloud. These data are clustered using K-Mean algorithm based on timestamp, direction, longitude and latitude. Traffic conditions are predicted using Markov model using these clustered data. Then, we display the predicted traffic congestion to the user's Mobile. We use hadoop for computing massive data analysis.

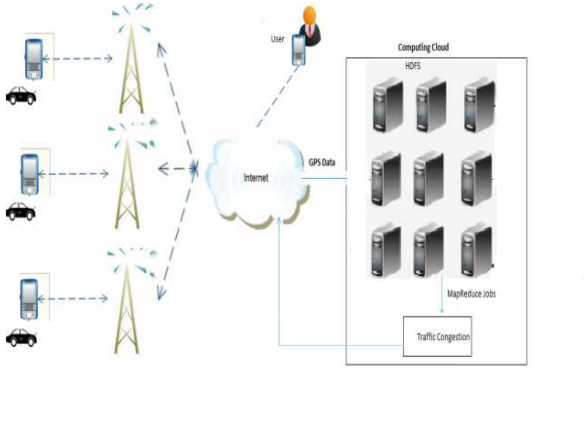

Figure 1: General architecture for Traffic Prediction.

3.1. Hadoop

Hadoop [7] is a java open source implementation of Map Reduce sponsored by Yahoo! The Hadoop project is a collection of various subprojects for reliable, scalable distributed computing. The Hadoop Distributed File System (HDFS) allows storing large files as multiple blocks which are replicated on multiple nodes to provide reliability. The scale of GPS data is so large that is not possible to fit the data on a single machine's disk. GPS data comes from large number of vehicles in the form of data streams are preprocessed and store it in HDFS.

Hadoop allow parallel processing of data and data clustering algorithms can be implements as MapReduce jobs. MapReduce has two phases: Map and Reduce. In the Map phase, GPS data stored in HDFS is read, partitioned them among a set of computing nodes in the cluster, and sent to the nodes as a set of key-value pairs. The Map tasks process the input records independent of each other and produce intermediate results as key-value pairs. The intermediate results are stored on the local disk of the node running the Map task. When all the Map tasks are completed, the Reduce phase begins in which the intermediate data with the same key is aggregated. The process of data input and output is as follows [5]:

Map (key_1, $value_1$) → List(key_2, $value_2$)
Reduce (key_2, $value_2$) → List($value_3$)

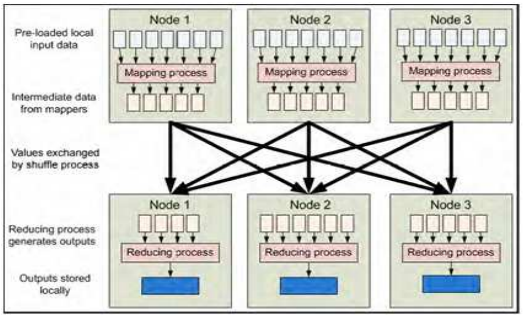

Figure: MapReduce illustration

3.2. K-Means Algorithm with Hadoop MapReduce

Clustering techniques are general tools for data analysis that are also valuable for visualizing trajectory data by building summaries that reduce overplotting. In this system, we group GPS data based on the weekdays and weekends, directions and time periods in every 15 minutes. Then, each group is clustered using K-Means algorithm with mapreduce.

To cluster GPS data from mobile on the vehicles using K-Means, we define a timestamped point to be a pair (t, p) consisting of a time t > 0 and a point p Є R^d. A trajectory is a sequence T of n timestamped points $\{(t_1, p_1), (t_2, p_2),\ldots,(t_n, p_n)\}$ ordered so $t_i < t_j \forall i < j$. A subtrajectory of T is a subsequence S = iT j ⊆ T containing elements $\{(t_i, p_i), (t_{i+1}, p_{i+1}),\ldots,(t_j, p_j)\}$; 1 ≤i ≤j ≤n. We attempt to separate the trajectories into a small number of clusters according to time, directions and longitude, latitude so that each of the resulting subtrajetories $S_1= 1T_{C1}$, $S_2= {}_{C1}T_{C2,}$ ….,$S_{nc}= {}_{Cnc}T_{n.}$ We then use the K-Means algorithm to cluster each subtrajectory.

In traditional K-Means algorithm, it randomly selects k of the objects first. Each of which initially represents a cluster mean or center. For each of remaining objects, an object is assigned to the cluster to which it is the most similar, based on the distance between the object and the cluster mean. It then computes the new mean for each cluster. This process iterates until the criterion function converges.

The distance between the data points is calculated using Euclidean distance as follows. The Euclidean distance between two points or tuples, X1 = (x_{11}, x_{12} ,….., x_{1n}) X2 = (x_{21}, x_{22} ,….., x_{2n}) .

$$Dist(X_1, X_2) = \sqrt{\sum_{i=1}^{n}(x_{1i} , x_{2i})^2}$$

In this system, we apply the goodness of K-Means along with MapReduce. The operation of GPS data starts with the cluster set up phase, in which clusters of the GPS data are formed, followed by the data transmission phase, in which cluster nodes will transmit the collected data to cluster head. Each cluster head aggregates the data received from cluster nodes and relays to the base station. The cluster set up phase is divided into two sub clustering phases.

In first sub clustering phase, the base station has to cluster the GPS data and assign the proper roles to them. This operation is referred as MAP protocol. In second sub clustering phase, if the energy of the cluster node is getting down it will try to second out the better cluster head. This phase is referred as REDUCE protocol.

Cluster Setup Phase

As we are using MAP-REDUCE [4, 5] algorithm, the types of key and value for proposed method are as follows.

key_1 ⟶List of initial set of selected k centroids.
$value_1$ ⟶ List of all other nodes along with their location information of GPS data.
key_2 ⟶ List of new set of k centroids.
$value_2$ ⟶ List of all other nodes with their cluster heads.
key_3 ⟶ List of new k ′ ≤ k centroids:
$value_3$ ⟶ List of all other nodes with new cluster heads.

Map / First-clustering Phase

Table 3.1 shows the process of a Mapper. Input to the mapper is list of initial set of randomly selected k centroids as key_1 and list of all other nodes along with GPS data as $value_1$. By using mapper (key_1, $value_1$) protocol, the Map phase would produce list of new set of k centroids as key_2 and list of all other nodes with their cluster heads known to them as $value_2$.

Table 3.1: MAP Protocol

1. BS → N nodes : Requesting number of nodes
2. BS: KMEANS(key_1, $value_1$)
3. BS assigns role (Cluster Head /member node) to each GPS data
4. Each cluster sends one hop communication about the cluster to member nodes
5 Generate output key_2, $value_2$

Reduce / second-clustering Phase

Table 3.2 shows the process of a Reducer. The intermediate results produced by Map protocol are given as input to the reducer i.e. list of new set of k centroids as key_2 and list of all other nodes with their cluster heads known to them as $value_2$. By using reducer (key_2, $value_2$) protocol, the Reduce phase would produce final clusters with their cluster heads and other nodes in that cluster as $value_3$.The term reduce is used in Reduce phase, which is meant for

optimizing the output and not for reducing the size of the output.

Original Map is parallel in nature. We use a centralized MAP algorithm at BS but REDUCE is parallelized to optimize the final clusters. This parallization reduces the time of clustering the GPS data.

Table 3.2: REDUCE Protocol

1.Read ($value_2$) /* Build second Clustering *
2. Place k′ (k′ ≤ k) nodes represented as initial cluster heads
3 Repeat a) If the member node is losing the energy below the threshold, it will start searching for better Cluster Head b) Or the cluster head is running out of energy new Cluster Head will be assigned to the node. c) Update Cluster Head i.e. calculate the mean value for each cluster d) Until no change
4 Produce $value_3$

K Means algorithm will be called by Map and Reduce Protocol (See table 3.3).

Table 3.3: K Means Algorithm

1. BS will arbitrarily chooses k nodes as initial cluster heads having closer to the centriod node
2. Repeat
3. (Re)assign each node to the cluster with the nearest Cluster Head.
4. Calculate the mean value of the Cluster.
5. Until no change

The effectiveness of clusters are evaluated based on uniformity of node distribution.

Intra-Cluster Distance

This is the distance between the cluster nodes to its cluster centres to determine whether the clusters are compact [4].

$$\text{intra} = \frac{1}{N}\sum_{i=0}^{K}\sum_{x\in Ci} \| x - Zi \|$$

where N is the number of nodes in the network, K is the number of clusters, and Z_i is the cluster centre of cluster C_i.

Inter-Cluster Distance

This is the distance between clusters [30]. We calculate this as the distance between cluster centres, and take the minimum of this value, defined as

$$\text{inter} = (\| Zi - Zj \|^2)$$

i=1,2 . . .K -1 and j= i+1 . . .K
we take only the minimum of this value.

3.3. Markov Model

We use the Markov model to predict the traffic flow of near term future road by using the resulting clustered GPS data. The Markov Chain model is used to employ current and recent values of the traffic flow from GPS data and describes the future value. This future value is described by transition probability which is approximated by the Gaussian Mixture Model (GMM) whose parameters are acquired by Expectation Maximization (EM) algorithm.

A. Markov Chain model for traffic flow

An N-order Markov Chain is employed to describe a series which has the following attribution: given the current and N-1 preceding states, the future state is independent to the states prior to the given states. If we denote x(t) as the state of the traffic flow at time interval t, then the probability density function of the next time traffic flow$(Y|X) = p(x(t+1)|x(t), x(t-1), \ldots, x(t-N+1))$ describes the transition probability of next time x(t + 1) given the current and N - 1 previous states of traffic flow, where multidimensional random vector X =[x(t), x(t - 1),….., x(t - N + 1)]^T , random variable Y = x(t + 1) and N is the order of Markov Chain Model. If probability density function of the next traffic flow $p(Y|X)$ is known, then the optimal estimation of next traffic flow Y would be given as

$$Y\hat{} = E(Y|X) \qquad (1)$$

under the criterion of Minimum Mean Square Error (M.M.S.E.) [8].

If we obtain joint probability distribution p(Y, X), then probability density of the future time traffic flow p(Y|X) can be easily got by the Bayesian Theorem. That is

$$P(Y|X) = \frac{p(Y,X)}{p(X)} = \frac{p(Y,X)}{\int p(Y,X)dY}$$

In modeling the traffic flow, we use a Markov Chain model. Therefore, we can approximate the joint probability $p(Y,X)$ based on the clustered GPS data by using Gaussian mixture model whose parameters are estimated with CEM algorithm to predict probability density of the traffic flow for next road segment.

B. Gaussian Mixture Model and the parameter estimation

How to deduce p(Y|X) with the given GPS clustering data is a problem. In this paper the joint probability distribution p(Y,X) is approximated through a Gaussian Mixture Model (GMM) which can approximate an arbitrary probability density function with enough accuracy. We prefer the Gaussian Mixture model mainly based on the following three reasons:

1). Many events or phenomena in the natural world *per se* obey Gaussian distributions.

2). The Gaussian function has its convenience in mathematical processing which can be seen below.

3). We can approximate an arbitrary probability distribution with the combination of sufficient Gaussian functions.

The GMM [17] representation of joint probability distribution p(Y,X) is as follows:

$$P(Y,X) = \sum_{m=1}^{M} \alpha_m \; G(Z; \mu_m, \Sigma_m) \qquad (2)$$

where Z = [Y X] is multiple random variable. $G(Z; \mu_m, \Sigma_m)$ denotes the m^{th} clustered data component of the GMM, which is a Gaussian density function with mean μ_m and covariance matrix Σ_m. α_m is the mixing coefficient of the m^{th} component. M is the number of clustered data components.

Because of its appealing accuracy and low requirement for the amount of training data, usually we use Maximum Likelihood Estimation (MLE) to carry out parameter estimation of μ_m, Σ_m and α_m with clustered GPS data. Although the Expectation Maximization EM algorithm is an effective method to carry out MLE, it usually converges to local maxima. To find a global maximum is more significant in most cases. The Competitive Expectation Maximization (CEM) algorithm overcomes the drawbacks of basic EM algorithm; besides, it is also capable of automatically choosing the number of mixing components and is insensitive to initial configuration of the number of the mixture components and model parameters [18]. Therefore, we use CEM algorithm to carry out MLE in this paper.

If we rewrite mean of m^{th} clustered data component $\mu_m = [\mu_{my}, \mu_{mx}]^T$ and covariance matrix $\Sigma_m = \begin{bmatrix} \Sigma_{myy} & \Sigma_{myx} \\ \Sigma_{myx} & \Sigma_{mxx} \end{bmatrix}$ of GMMs model, we can approximate joint probability p(Y,X) in the equation 2

$P(Y,X) = \sum_{m=1}^{M} \alpha_m \; G(Z; \mu_m, \Sigma_m)$ as

$$= \sum_{m=1}^{M} \alpha_m \; G(Z; \mu_m, \Sigma_m) G(Y; \mu_{my|x}, \Sigma_{my|x})$$

and probability density function of the future time traffic flow p(Y |X) can also calculate as

$$P(Y|X) = \sum_{m=1}^{M} \beta_m \; G(Y; \mu_{my|x}, \Sigma_{my|x}) \quad (3)$$

Where

$$\beta_m = \frac{\alpha_m \; G(X; \mu_{mx}, \Sigma_{mx})}{\sum_{j=1}^{M} \alpha_j \; G(X; \mu_{jx}, \Sigma_{jx})},$$

$\mu_{my|x} = \mu_{my|x} - \Sigma_{myx} \Sigma_{mxx}^{-1} (\mu_{mx} - X)$ and

$$\Sigma_{my|x} = \Sigma_{myy} - \Sigma_{myx} \Sigma_{mxx}^{-1} \Sigma_{mxy}$$

With this model, we can neatly get the optimal estimation traffic prediction future time of Y given X. The optimal forecasting of future time traffic flow estimation Y^ can be represented as the following form:

$$Y^{\wedge} = E(Y|X)$$

$$= \int Y \, p(Y|X) dY$$

$$= \sum_{m=1}^{M} \beta_m \int Y \; G(Y; \mu_{my|x}, \Sigma_{my|x}) dY$$

$$= \sum_{m=1}^{M} \beta_m \; \mu_{my|x} \qquad (4)$$

The flow chart of in this forecasting traffic flow procedure using clustered GPS data can be described as follows.

1). Approximate the joint probability distribution p(Y, X) by GMM and CEM algorithm using the methods.

2).Deduce probability density function of the next time traffic flow $(Y|X)$ of the clustered GPS data.

3). Carry out the optimal estimation of Y in the form of equation 4.

4. Conclusions

The application of GPS in traffic analysis is proving to be the most effective solution compared to other existing traffic management methods like safety cameras, human inspection, speed governors and tachographs. We use the K-Means algorithm to cluster the GPS data. As MapReduce is the best programming model for large data sets to parallel the task. We tried to use this functionality of MapReduce. K-Means is widely used for clustering in data mining, but it is best suitable for smaller data sets. The Lager data set of mobile GPS data on vehicles becomes the smaller data set of K and for it the K Means works best. And then we use the Markov chain algorithm to predict the traffic flow. Markov model is a simple, effective way to predict near-term, future road traffic flow.

5. References

[1]B.S. Yoo, S.P. Kang, and C.H. Park. Travel time estimation using mobile data. In Proceedings of theEastern Asia Society for Transportation Studies, Vol.5, pages 1533–1547, 2005.

[2]Barbara Frith, David Pearce, and Tom Sutch. The highways agency journey time database. Road Transport Information and Control, pages 98–105, 2004.

[3] Baibo Zhang, Changshui Zhang, Xing Yi, ompetitive EM Algorithm for Finite Mixture Models, *Pattern Recognition*, Volume: 37, Issue: 1, January, 2004, pp. 131-144.

[4]Isra_l Tamim, Asif Khan ,Emdad Ahmed, Muhammad Abdul Awal Multiple Parameter Based Clustering (MPC): Prospective Analysis for Effective Clustering in Wireless Sensor Network (WSN) Using K-Means Algorithm Wireless

[5]Jyoti R. Patole, "Clustering the Wireless Sensor Network using K-Means and MapReduce algorithm", 2012. Sensor Network,4, 18-24,2012.

[6] Lasgouttes J M ,Furtlehner C and Fortelle A D L 2007 A belief-propagation approach to traffic prediction using probe vehicles Proc. 10th IEEE Conf. of Intelligent Transportation Systems pp10221027

[7]Hadoop project, http://lucene.apache.org/hadoop (2011).

[8]Ming-Hui Chen, Qi-Man Shao, Joseph G. Ibrahim, *Monte CarloMethods in Bayesian Computation*, Springer, New York, 2000.

[9] Michael A.P. Taylor, Jeremy E. Woolley, and Rocco Zito. Integration of the global positioning system and geographical information systems for traffic congestion studies. Transportation research. Part C : Emerging technologies, pages 257–285, 2000.

[10] Nectaria Tryfona, Dieter Pfoser, and Agnes Voisard. Dynamic travel time maps - enabling efficient navigation. In SSDBM '06: Proceedings of the 18th International Conference on Scientific and Statistical Database Management (SSDBM'06), pages 369–378, 2006.

[11]N. Quiroga Cesar A and Darcy Bullock. Travel time studies with global positioning and geographic information systems: an integrated methodology. Transportation research. Part C : Emerging technologies, pages 101–127, 1998.

[12] Otto Anker Nielsen. Analyse of traengsel og hastigheder vha. gps-data. Trafikdage, pages 1–21, 2003.

[13] Rajput D., Singh P., and Bhattacharya M., "An Efficient and Generic Hybrid Framework for High Dimensional Data Clustering," *in Proceedings of International Conference on Data Mining and Knowledge Engineering*, *World Academy of Science, Engineering and Technology*, Rome, pp. 174-179, 2010.

[14] Tian Xia, Evangelos Kanoulas, Yang Du, and Donghui Zhang. Finding fastest paths on a road network with speed patterns. In ICDE '06: Proceedings of the 22nd International Conference on Data Engineering (ICDE'06), pages 1–10, 2006.

[15] Wang, Xuan, "Clustering in the Cloud: Clustering Algorithms to Hadoop Map/Reduce Framework" http://ecommons.txstate.edu/cscitrep/19 Paper 19, 2010

[16] Wei-Shinn Ku, Roger Zimmermann, Haojun Wang, and Chi-Ngai Wan. Adaptive nearest neighbor queries in travel time networks. In GIS '05: Proceedings of the 13th annual ACM international workshop on Geographic information systems (GIS'05), pages 210–219, 2005.

[17] X.David Doria," Expectation-Maximization: Application to Gaussian Mixture Model Parameter Estimation", April 23, 2009

[18] Xing Yi, Baibo Zhang, Changshui Zhang, Competitive EM Algorithm for Finite Mixture Models, *Pattern Recognition*, Volume: 37, Issue: 1, January, 2004,