MyRoute: A Graph-Dependency Based Model for Real-Time Route Prediction

Hanane Amirat[1,2], Nasreddine Lagraa[2], Philippe Fournier-Viger[3], and Youcef Ouinten[2]
[1]Département informatique, Université Kasdi Merbah, Ouargla, Algeria
[2]Laboratoire d'Informatique et de Mathématiques, Université Amar Telidji, Laghouat, Algeria
[3] Harbin Institute of Technology Shenzhen Graduate School, Shenzhen 518055, People's Republic of China
Email: amirat.hanane@univ-ouargla.dz; {n.lagraa, ouinteny}@mail.lagh-univ.dz; philippe.fournier-viger@umoncton.ca

Abstract**—**Mobility prediction is an important problem having numerous applications in mobile computing and pervasive systems. However, many mobility prediction approaches are not noise tolerant, do not consider collective and individual behavior for making predictions, and provide a low accuracy. This paper addresses these issues by proposing a novel dependency-graph based predictor for real-time route prediction, named *MyRoute*. The proposed approach represents routes as a graph, which is then used to accurately match road network architecture with real-world vehicle movements. Unlike many prediction models, the designed model is noise tolerant, and can thus provide high accuracy even with data that contains noise and inaccuracies such as GPS mobility data. To cope with noise found in trajectory data, a *lookahead* window is used to build the prediction graph. Besides, the proposed approach integrates two mechanisms to consider both the collective and individual mobility behaviors of drivers. Experiments on real and synthetic datasets have shown that the performance of the designed model is excellent when compared to two state-of-the-art models.

Index Terms**—**Real-time, route prediction, dependency graph, mobility graph, noise tolerance.

I. Introduction

Predicting movements of moving objects is a task of great importance, having numerous applications in several domains. Obtaining information about a vehicle's locations is key for location based routing of high-speed vehicular ad-hoc networks [1]. Movement prediction is also essential to improve the quality of ITS (Intelligent Transportation Systems), as it can support congestion and trip duration prediction. In other words, if a system can predict future movements of vehicles, it can also be used to predict traffic jams and other traffic hazards [2]. Besides, another important application of mobility prediction is the optimization of hybrid vehicle fuel consumption. For example, Nissan researchers have shown that knowing a vehicle's route in advance can allow reducing fuel consumption by up to 7.8% [3]. Lastly, another important application of movement prediction is supporting Location-Based Services (LBS) such as delivering targeted advertisements to customers who are likely driving toward an area of interest [4].

While various types of movements can be predicted, this paper focuses on route prediction, which consists of predicting the route of a vehicle in terms of the next road segments that it will visit. Route prediction schemes are based on the assumption that routes followed by driver(s) show spatial and temporal regularities that can be used to predict a driver's next location. Two types of mobility behaviors are considered in this paper. The first type, called global (collective) behavior, is based on the assumption that persons may follow similar trajectories and share common mobility patterns when traveling to the same locations. Another example of collective movements is groups of vehicles that follow each other for a long period of time (i.e. platoons) on the highway. The second type of behavior is personal (individual) behavior, which is based on the assumption that a person may exhibit regularities when travelling to the same location. For instance, a person may always follow the same routes to go from home to work.

In recent years, route prediction has attracted the attention of many researchers especially in the networking community. Most route prediction approaches consist of applying machine learning techniques such as neural networks [2], [5] or statistical models such as Markov models [6]–[8] and probabilistic trees [9]. Although these models were shown to perform well, an important drawback is their noise sensitivity toward the mobility patterns that they can learn. The smallest deviation in trajectory data affects the prediction result, and thus the prediction accuracy. This problem becomes worse for noisy mobility datasets such as GPS trajectory data which are prone to many disturbances and inaccuracies. Additionally, some prior proposals were designed without regarding to any constraints on computational and memory limitations of mobile devices which make the real implementation and deployment of these solutions impossible as on-line applications. Besides, many proposals have addressed the prediction problem considering only one type of human mobility behavior and there were only few attempts to study the impact of both individual and collective behaviors on personalized and global mobility prediction.

In this paper, we tackle the above challenges by proposing a real-time graph-based approach for route prediction called *MyRoute*, which adopts the Dependency

Manuscript received August 20, 2017; revised December 19, 2017.
doi:10.12720/jcm.12.12.668-676

Graph (DG) predictor, previously used for web prefetching [10]. Graph-based approaches have been shown to be very efficient in a wide range of domains such as computer science [11], artificial intelligence [12], and communication [13]. Central to our approach is the idea that graphs can perfectly represent the structure of road networks, and therefore accurately model vehicle movements. Moreover, this work is also based on the idea that vehicle mobility is order dependent as a vehicle passes through a sequence of route segments to reach some location of interest, by following a specific order and traversing road segments in specific directions.

To forecast the next location, *MyRoute* utilizes the current location of a vehicle and its historical data, if available. The proposed approach first builds a prediction graph, where nodes (vertices) are road segments and arcs (directed edges) are used to represent the traversal order of road segments by vehicles. Then, the prediction graph is used to predict the next route of a driver by attempting to match its current trajectory with graph paths. However, graph matching is difficult due to the presence of noise in data. To address this issue, the proposed approach creates graph edges not only between consecutive road segments (nodes) in a path but also with the following road segments appearing within a user-defined *lookahead* window. This method allows ignoring noise found in the data to increase prediction accuracy. To provide high prediction accuracy, the proposed model also offers great flexibility in terms of considering the road segments in trajectories when matching a current trajectory of a vehicle with its previous trajectories unlike some prior models which match GPS trajectories. Moreover, to model both global and personal mobility behaviors, two prediction mechanisms are proposed called the *GMG* (Global Mobility Graph) and *PMG* (Personal Mobility Graph), respectively. Note that although this paper focuses on vehicle route prediction, the proposed approach could be used for location prediction of other types of moving objects.

II. Related Work

Due to the numerous applications of route prediction, many researchers have studied this problem in recent years. These studies can be described in terms of various factors such as the types of predictions that are performed (route and/or destination predictions), the techniques used for carrying out predictions, the prediction range (short or long term), etc. Generally, the techniques proposed in previous works can be classified as one of the three following categories:

A. Statistical Models

Most statistical models rely on the Markov assumption that previous trajectories of a person must be considered to perform route prediction. For personal route prediction, Wang *et al.* [14] have used a first order Markov model called PPM (Prediction by Partial Matching). That approach builds a probability transition matrix containing the probabilities of moving from each road segment to the others. To deal with the growing amount of personal mobility data and provide real-time route predictions, data reduction algorithms have been applied on probability matrices. First order Markov model has been also proposed by Krumm [8] for short-term upcoming road prediction. Chen *et al*. [15] have proposed three Markov models for global (GMM), personal (PMM) and regional (RMM) mobility movement predictions. GMM and PMM are used to model, respectively, collective and individual mobility behaviors of moving objects. RMM considers geographic similarities between trajectories by clustering similar trajectories of a person into clusters and then train a Markov model for each resulting cluster. Petróczi et al. [16] have proposed three prediction models: (1) a statistical model based on frequent itemset mining, (2) an n-order Markov model where n<4, and (3) a Pattern Matching Model based on n-order Markov with a flexible number of items to consider. Xue *et al*. [9] have applied a combination of a Variable-order Markov Model (VMM) and Probabilistic Suffix Trees (PSTs). That approach utilizes multiple VMMs with different traffic conditions for daytime driving to mine mobility patterns from real GPS taxi traces. In an earlier work, Simmons *et al*. [17] have proposed to use an HMM (Hidden Markov Model) to simultaneously predict a driver's intended route and destination. Simmons *et al*. have then improved their initial model to consider temporal factors for prediction. To cope with uncertainty in GPS data, HMM has been also applied in [4]. To provide more accurate location prediction, Markov models are often combined with other techniques such as semantic inference [6], [7] for personalized route prediction with discovering the type of visited locations.

B. Data Mining

Data mining techniques have been broadly applied to perform various types of predictions using techniques such as neural networks and sequential pattern mining. Two main neural network architectures have been employed: feed-forward [2] and recurrent bidirectional neural networks [5]. De Brébisson *et al*. [5] have developed a framework to predict the destination of a taxi based on its starting location and associated meta-information such as departure time, driver identifier and client information. A recurrent bidirectional neural network was applied to encode each taxi's path with its relevant metadata. The mean shift clustering technique was used to obtain clusters of destinations representing the training trajectories. To extract route patterns from historical movement data, Ye *et al*. [18] proposed a mining algorithm called CRPM (Continuous Route Pattern Mining) based on the well-known *PrefixSpan* sequential pattern mining algorithm [19]. The proposed approach performs predictions using a pattern tree built from extracted mobility patterns. Chen *et al*. [20] have attempted to predict simultaneously the intended destination and future route of a person. Using real GPS data, the authors have proposed to cluster important places that a person may depart from or go to using the FBM (Forward–Backward Matching) clustering algorithm. Movement patterns are then extracted from abstracted trajectories using an extension of the CRPM algorithm. Merah *et al*. [21] have presented several communication schemes that could be used to collect historical vehicular paths. Movement patterns are then extracted as the most frequent traveled paths and used afterward to generate movement rules that could be used to forecast vehicle future routes.

C. *Trip Matching*

Trip matching consists of finding the most similar trip for a user by comparing current driver trajectory with his/her previous trips. To estimate measure similarity between pairs of trips, Hausdorff metric is used in [22], [23]. Helmholz *et al*. [23] have also extended their model so it considers temporal homogeneity of driving habits.

III. Background

A. *Problem Definition*

Definition 1 (Road segment). A road segment is an abstraction of a vehicle or a driver location. A road segment r_i is a directed edge between two junctions [24]. For example, Fig. 1 depicts road segments r_x and r_y connecting two junctions.

Definition 2 (mobility or movement sequence). A mobility sequence $ms=<r_1,r_2,...,r_n>$ is a sequence of road segments traversed by a vehicle during a trip. For instance, Table I depicts four sample mobility sequences $MS:\{ms_1,ms_2,ms_3,ms_4\}$ performed by four vehicles $V:\{V_1,V_2,V_3,V_4\}$, representing their paths. For instance, the vehicle V_2 has a mobility sequence ms_2 which indicates that V_2 has traversed the road segment r_1 followed successively by r_4 and r_2.

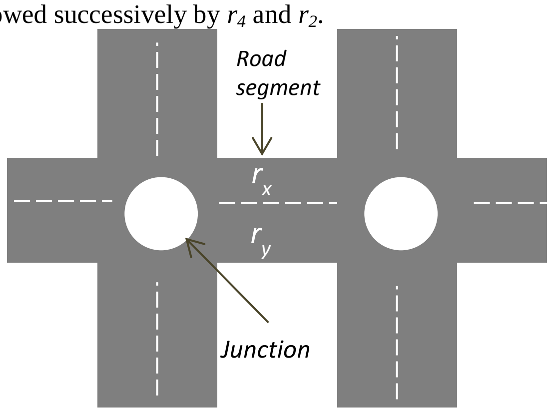


Fig. 1. Illustration of a road segment.

TABLE I. A Sample of Mobility Sequences.

Mobility sequence ID	*Vehicle ID*	*Mobility sequence*
ms_1	V_1	$ms_1= <r_0, r_1, r_2>$
ms_2	V_2	$ms_2=<r_1, r_4, r_2>$
ms_3	V_3	$ms_3=<r_0, r_1, r_3, r_4>$
ms_4	V_4	$ms_4=<r_3, r_4, r_2, r_5>$

Given a mobility sequence $ms=<r_1,r_2,…,r_n>$ of a vehicle V_i, the problem of route prediction consists of predicting the next road segment V_i that will be visited by vehicle V_i.

B. *Mobility Graph Model*

To address the problem of route prediction, this paper proposes a mobility graph model inspired by the dependency graph (DG) predictor. DG is a graph based sequence prediction model initially proposed for web-prefetching [10], which represents order dependencies as edges between graph nodes. Formally, a mobility graph *MG* is a pair of sets (*R*, *A*) where *R* is a set of road segments (nodes) and $A \in R \times R$ is a set of directional arcs (edges) representing movements on road network. An arc $a(r_x\ r_y) \in A$ connects two road segments r_x and r_y in *MG*. The road from which the movement *a* starts is called the source of *a*, and is denoted as $Source(a)=r_x$, whereas the other node r_y is called the destination of *a*, and is denoted as $Dest(a)= r_y$. An arc $a(r_x\ r_y)$ is created in *MG* if and only if r_y appears within *w* movements after r_x in a mobility sequence, where *w* is a user-defined parameter called the *lookahead* window size. Moreover, a weight value ($w(a)$) is associated to each arc *a* in *MG*, which indicates the number of times that *Dest(a)* was traversed by drivers after *Source(a)*.

In the context of mobility prediction, a *MG* allows representing order dependencies among traversed roads in mobility sequences by drivers where the source of a given arc $a_i \in A$ must appears before its destination. The mobility prediction graph is built by gradually inserting mobility sequences in the graph, where each mobility sequence is represented as a path comprising the set of roads that it contains. In the case where sequences share common roads, the weights of the shared arcs are incremented rather than creating new arcs. Consequently, significant space reduction can be achieved using the *MG* representation. For instance, Fig. 2 depicts the mobility graph constructed using the mobility sequences of Table I with a *lookahead* window defined by $w=2$.

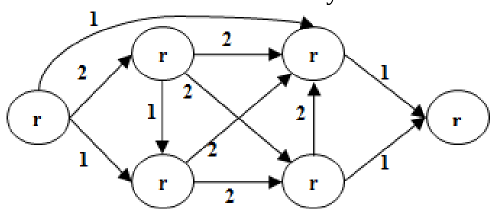


Fig. 2. A mobility graph.

IV. My Route Model

Having introduced the mobility graph representation, this section first illustrates the architecture of the proposed prediction approach, and then described the prediction models that regard human mobility behaviors.

A. *System Architecture*

The *MyRoute* framework is designed to perform real-time route prediction by adopting a client-server architecture where the client is the vehicle whereas the server is an authority infrastructure such as a RSU (roadside unit). *MyRoute* comprises three main modules as depicted in Fig. 3.

1. *Data preparation.* The first module periodically collects driver location data (GPS records) and sends it to a server site. During collection, location data is split into trips by defining stay points. A stay point is a geographic area expressed as a set of consecutive GPS records where the distance from the first and last GPS records exceeds a distance threshold D_{thre} and the driver spent more time than a threshold T_{thr}. The resulting trips are then converted into mobility sequences by map-matching GPS trajectories using a cloud map-matching based API [25].

2. *Graph construction.* After obtaining mobility sequences, the server incrementally updates its mobility graph. The mobility graph is initially built and then extended by inserting new road segments as graph nodes, whereas vehicle movements between pair of road segment within the *lookahead* window size are represented by arcs. The weight of each new arc is set to 1. In the case where a road segment appears on newly collected mobility sequences, the weight of the corresponding arc is incremented accordingly. Note that, in some situations where movement sequences are highly heterogonous and do not overlap (have no common road segments), the mobility graph is a disconnected graph. In this paper, we assume that the mobility graph is connected, comprises a least two nodes, and has no isolated nodes. These assumptions hold in real-life for representing the mobility of vehicles in active urban areas.

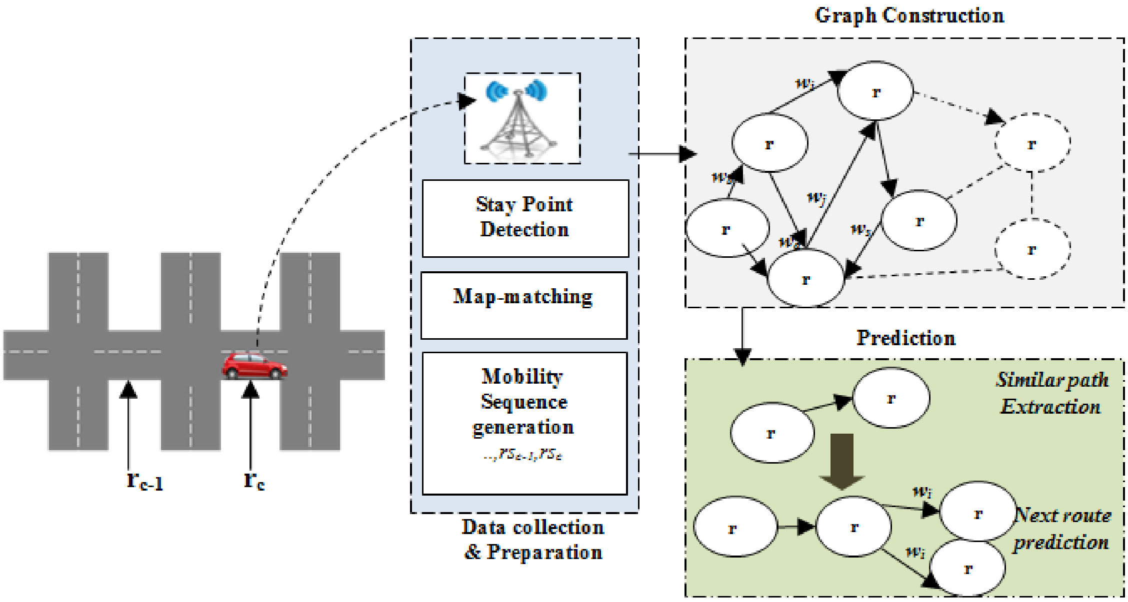


Fig. 3. The overall architecture of *MyRoute*.

3. *Prediction.* Once a mobility graph has been built, predictions can be performed using it. To predict the next route segment that will be visited by a driver *D*, its current trajectory, denoted as *CT* (Current Trajectory), is required. It contains the current road segment where *D* is located in addition to its previous locations for the same trip, if available. Formally, let $RS=\{r_1,r_2,\ldots,r_n\}$ be the set of all road segments in a road network *RN*. $CT=\{p_j,p_{j+1},..,p_c\}$ is a sequence of road segments traversed by *D* where *pc* is the current road segment of *D*. Having a trajectory *CT*, the prediction of the next route segment is performed in two steps.

3.1. *Graph matching.* The first step consists of finding a path $SP=\{s_i,s_{i+1},\ldots,s_m\}$ in the mobility graph that matches with *CT* where $s_i \in$ RS. We say that *SP* matches *CT* if and only if each road segment in SP appears in the same order in *CT*, that is $\forall i, s_i = p_i$ and m =c. Note that graph matching is noise sensitive. Finding the path that exactly matches a trajectory *CT* can be challenging since erroneous positions may appear in location data. Using *MG*, built according to *lookahead* window, more flexibility to handle noisy data could be obtained. The *MG* allows creating arcs not only between consecutive road segments (which may be noise) but also to the following road segments within the *lookahead* window. In other words, if the first road segment *Ne* that comes after a given node N_x in a mobility sequence *s* is considered as noise, another arc will be created that skips *Ne* and go directly to next road segment, given that $w \geq 2$. Unlike other Markov-based predictors such as PPM, the noise tolerance strategy achieved by using *MG* permits *MyRoute* to forecast the future route of *CT* that have not been previously seen in mobility sequences.

3.2. *Next road extraction.* The second step is to find the next road segment that a driver will visit following *SP*, denoted as *Nr*. This road segment is predicted as the destination of the arc having the highest weight emanating from the last road segment in *SP*. More formally, let $E=\{a_1,a_2,\ldots,a_n\}$ be the set of outgoing arcs for the last node of $SP(s_m)$. Then, *Nr* is defined as: $Nr=Dest(a_k)$ such that $W(a_k) \geq W(a)$ for $a \in E$ and $\forall a_k \in E$ and $source(a_k)=s_m$. For example, consider that the current trajectory of a driver is $CT=\{r_0,r_3,r_4\}$. Based on the mobility graph of Fig. 2, two candidate arcs are considered, which are $a_1(r_4\ r_2)$ and $a_2(r_4\ r_5)$. These arcs have weights of 2 and 1, respectively. Therefore, the next route segment is predicted to be $Nr=r_2$ (the destination of a_1). In case where candidates segments have equal weights, many selection criterions could be employed such as retaining the road with highest frequency in mobility data.

B. Prediction Models

Having presented the proposed prediction scheme for the designed mobility graph structure, this subsection explains how the mobility graph is adapted to consider both global (collective) and personal (individual) movement behaviors of drivers. Two models are proposed:

1. Global MG (GMG). This model is used to represent global mobility behavior of a set of persons. The prediction graph is constructed from mobility data of all drivers. The *GMG* model is employed to perform predictions for a driver when no prior knowledge about his mobility pattern could be found such that drivers newly seen in prediction framework.

*2. Personal model (PMG).*This model consists of creating a mobility graph for each driver comprising his previous trajectories. Since each *PMG* only considers a single driver, the prediction graph is only trained with his mobility sequences rather than the data of all drivers. A PMG is a sub-graph of the *GMG*. Therefore, a *PMG* can be considerably smaller than a *GMG*.

By default, to forecast the next location of a given driver, the *GMG* model is used unless matching personal data is found in the *PMG* of the driver. In such situation, the driver's *PMG* is used for prediction.

V. Experimental Evaluation

To evaluate the proposed *MyRoute* framework, an extensive experimental evaluation was carried out. This section first describes the datasets and experimental settings, followed by the evaluation metrics to measure *MyRoute*'s performance. Finally, this section presents the conducted experiments and the corresponding results.

A. Datasets

Two vehicular mobility datasets have been used: *SanFransisco* cabs [26], and *Lust* [27]. *SanFransisco* cabs contains GPS mobility traces of 500 taxi cabs collected during one month in *SanFransisco*, USA whereas *Lust* is a dataset generated using *SUMO* [24] to simulate driving based on traffic data of Luxembourg. Both datasets are prepared and then divided into training and testing sets. Experiments were carried using a 10-fold cross-validation.

B. Experimental Setting

Experiments were performed on a computer equipped with a dual Core Intel CPU, 3GB of RAM and 250GB of Hard Disk. The proposed approach was implemented in Java by using the DG implementation available in the SPMF open-source data mining library [28].

C. Parameter Settings

To generate trips from GPS taxi trace, both T_{thre} and D_{thre} were set to 20 which indicates that a stay point is detected if and only if a vehicle remains stationary (or exhibits a slow mobility) within a geographic area of 20 meters for 20 minutes. Among the generated trips, only those containing at least two road segments were considered. For mobility graph construction, a *lookahead* window set to 2 was used.

D. Evaluation Metrics

To measure the performance of *MyRoute*, the two following metrics were computed.

Overall Accuracy. It is defined as the number of successfully predicted routes, divided by the total number of test mobility sequences.

$$Overall\ Accuracy = \frac{number\ of\ successful\ predictions}{number\ of\ \text{test}\ mobility\ sequences}$$

Coverage. It is the number of test mobility sequences where a matching path was found for each current trajectory of a driver divided by the total number of test sequences.

$$\text{Coverage} = \left(\frac{number\ of\ matching\ paths}{number\ of\ test\ mobility\ sequences}\right)$$

E. Experimental Evaluation

Two experiments have been carried to assess the performance of *MyRoute* and to compare its performance with the state-of-the-art PPM Markov model widely used in the literature for this type of prediction problem [8], [12], [13] and LZ prediction model used for the task of location prediction in [29]. LZ is similar to k-order Markov predictor except that the *k* is a parameter that can grow to infinity.

Experiment 1: Global and personal prediction

The goal of the first experiment is to compare the performance of *MyRoute* with PPM and LZ. Besides, this experiment also considered individual and collective mobility behaviors of persons using the *PMG* and *GMG* prediction models on the *SanFransisco* dataset. For personalized prediction, *PMG* was tested with trajectories of two taxi drivers denoted as *Driver1* and *Driver2*. For *GMG*, the mobility graph was constructed using the mobility sequences of 10 taxi cabs.

Results

For *PMG*, obtained results are shown in Fig. 4 and Fig. 5. They indicate that the overall accuracy generally increases as the number of mobility sequences is increased. This could be ascribed to the fact that as more trips are considered the more individual driver's trips are repeated and therefore intra-trips similarity is increased.

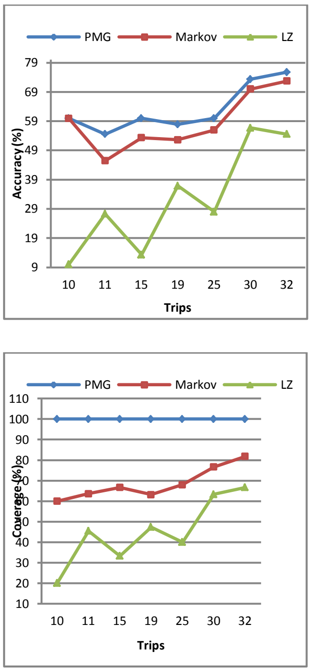


Fig. 4. Performance evaluation of *Driver* 1.

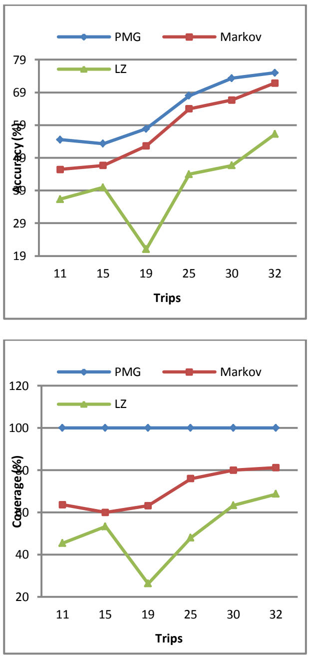
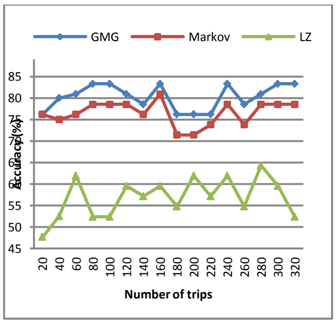


Fig. 5. Performance evaluation of *Driver2*.

From the results in Fig. 6, it can be observed that *GMG* achieves an overall accuracy of about 83.5%, even outperforming *PMG*.



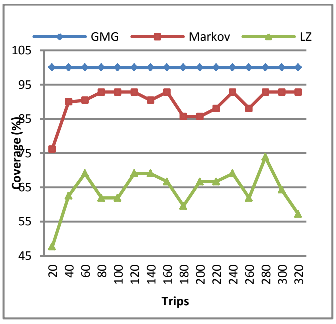


Fig. 6. Performance evaluation of *GMG*.

It is important to mention that taxi destinations and trajectories are client dependent. Thus, the collective mobility behavior of drivers is confirmed. The spatial regularity captured is ascribed to the fact that multiple cabs may share common driving paths covering specific geographic areas and traveling to them to wait for new clients such as airports, hotels, touristic locations, and therefore a high degree of similarity could be found in their movement exhibiting predictable mobility patterns. Results have also shown that *MyRoute* outperforms the PPM Markov model and LZ for both *PMG* and *GMG*. This may be ascribed to the fact that the Markov model requires a full match of a driver trajectory with the prediction model. However, real-life trajectory data such as GPS trajectories often contain location errors or miss data points, which may result in performing incorrect predictions. In contrast, the proposed mobility graph structure, used in *MyRoute* is more resistant to noise. Incomplete or noisy trajectories are handled by creating transitions to several upcoming road segments, using a *looakhead* window, which allow skipping erroneous or missing data.

How noise is handled in *MyRoute* also explain the high coverage obtained by *MyRoute*, compared to other models. The reason for the high coverage of the proposed approach is that *lookahead* transitions allow creating additional links between road segments, which often permits performing predictions by skipping segments when otherwise no prediction could be done. This, considerably increase the coverage. It is also observed, in this experiment, that the number of trips is relatively small. This is because taxi driving is characterized by active mobility with few stationary states that exceed the stay point detection thresholds. Note that prediction times are not reported in this experiment as they were negligible for all models.

Experiment 2: Model size and prediction time

The second experiment consisted of studying and comparing the scalability of *MyRoute* against the PPM Markov model in terms of model size and prediction time by increasing the number of trips. This experiment is important since *MyRoute* is designed to be deployed in an online manner using mobile devices with low processing and storage capacities. Therefore, measuring the prediction time and the occupied storage space was done.

To study scalability, the *Lust* dataset, characterized by its large number of mobility sequences was employed.

Results: Concerning model size, results shown in Fig. 7 indicate that *MyRoute* requires more storage space than the PPM Markov model. This is understandable since the number of nodes in both models is the same. However, more space is required in *MyRoute* due to the additional lookahead-transitions created and stored with their associated weights in mobility graph. Even though *MyRoute* is the largest model, the mobility graph size remains small and suitable for real-life use as many mobile devices are currently equipped with several gigabytes of memory. Moreover, results also demonstrate that the number of trips does not have a great impact on model size. This is explained by the fact that as more trips are added to a mobility graph, many of them overlap and thus no additional edges need to be added. In other words, novel arcs are added to a mobility graph only if they represent movements between road segments that were not previously seen. Otherwise, only weights are updated, thus preserving the size of the mobility graph.

Concerning the processing time, Fig. 8 shows that *MyRoute* requires less time to perform a prediction compared to PPM, and that prediction time for both models takes only a few milliseconds. This experimental finding confirms the feasibility of deploying *MyRoute* for real-time prediction. These results can be explained by the fast incremental graph construction process that permits reusing the same graph components and only inserting novel connections and updating weights for movements in the *lookahead* window. This accelerates and facilitates the matching process

Discussion. Based on the experiments, it can be concluded that the proposed *MyRoute* framework has good performance and scalability comparing to PPM and LZ predictors. In particular, it outperforms PPM Markov model in terms of accuracy, coverage, prediction time, and scalability, but not in terms of model size. This is however understandable as PPM is not noise tolerant, whereas *MyRoute* stores additional arcs based on its *lookahead* window to provide noise tolerance. Note that several experiments have been carried to evaluate the impact of the *lookahead* on performance. It was found that using a *lookahead* window of two segments provided the best results. Detailed results of those experiments are not presented in this paper due to space limitation.

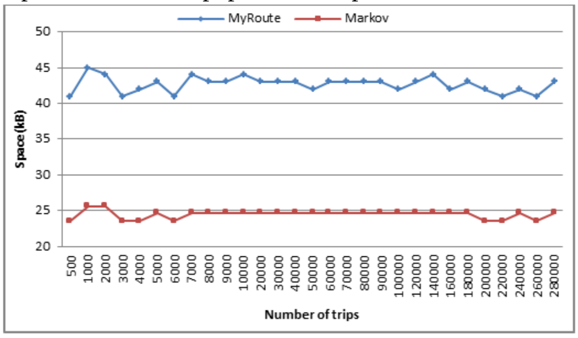


Fig. 7. Model storage space.

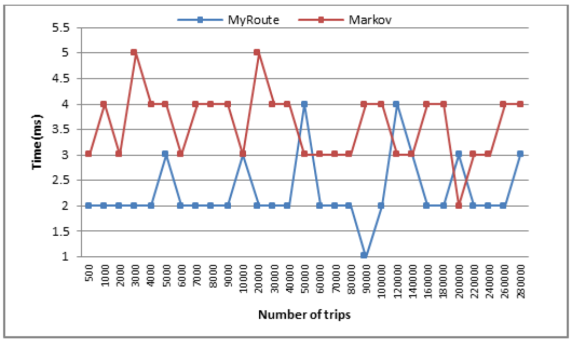


Fig. 8. Prediction time estimation.

VI. Conclusion

In this paper, we have proposed *MyRoute*, a novel framework for real-time route prediction. The central idea behind *MyRoute* is the assumption that human mobility exhibits regularity and periodicity. The framework utilizes a graph representation of mobility sequences where nodes are road segments and each arc represents an ordering of two road segments. A distinctive characteristic of the proposed mobility graph structure is the creation of additional links to upcoming road segments using a *lookahead* parameter. This provides noise tolerance for route prediction. Moreover, *MyRoute* implements two prediction schemes called *GMG* and *PMG* to model both collective and individual human mobility behaviors, respectively. An extensive experimental evaluation has demonstrated that *MyRoute* outperforms the state-of-the-art PPM Markov model and LZ, and achieves an accuracy of 76% for *PMG* and 83.5 for *GMG* with 100% coverage. For future work, several improvements will be considered such as extending the proposed mobility graph to regard temporal regularities of human movements and road conditions by considering temporal factors such as time-of-day and day-of-week and other contextual factors (i.e. weather, traffic congestion level, etc). Therefore, it will become possible to distinguish between trips relying not only on spatial data but also time and other contextual data. Finally, we intend to predict not only the next route (short term prediction) but also other upcoming routes (long-term prediction) in addition to the person's final destination.

References

[1] G. Xue, Y. Luo, J. Yu, and M. Li, “A novel vehicular location prediction based on mobility patterns for routing in urban VANET,” *EURASIP J. Wirel. Commun. Netw.*, vol. 2012, p. 222, 2012.

[2] T. Miklušćák, M. Gregor, and A. Janota, “Using neural networks for route and destination prediction in intelligent transport systems,” *Commun. Comput. Inf. Sci.*, vol. 329, pp. 380–387, 2012.

[3] Y. Deguchi, K. Kuroda, M. Shouji, and T. Kawabe, “HEV charge / discharge control system based on navigation information,” in *Proc. Convergence International Congress & Exposition on Transportation Electronics*, 2004, vol. 1.

[4] D. Qiu, P. Papotti, and L. Blanco, "Future locations prediction with uncertain data," *Lect. Notes Comput. Sci.*, vol. 8188, pp. 417–432, 2013.
[5] A. de Brébisson, É. Simon, A. Auvolat, P. Vincent, and Y. Bengio, "Artificial neural networks applied to taxi destination prediction," *Computer Science*, Jul. 2015.
[6] F. D. N. Neto, C. de S. Baptista, and C. E. C. Campelo, "A user-personalized model for real time destination and route prediction," in *19th International Conference on Intelligent Transportation Systems*, 2016, pp. 401–407.
[7] F. D. N. Neto, C. D. S. Baptista, and C. E. C. Campelo, "Prediction of destinations and routes in urban trips with automated identification of place types and stay points," in *Proc. Brazilian Symposium on Geoinformatics*, pp. 80–91, 2015.
[8] J. Krumm, "A markov model for driver turn prediction," SAE World Congr., 2008.
[9] G. Xue, Z. Li, H. Zhu, and Y. Liu, "Traffic-known urban vehicular route prediction based on partial mobility patterns," in *Proc. Int. Conf. Parallel Distrib. Syst.*, 2009, pp. 369–375.
[10] V. N. Padmanabhan and J. C. Mogul, "Using predictive prefetching to improve world wide web latency," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 26, no. 3, pp. 22–36, 1996.
[11] M. Taheriyan, C. A. Knoblock, P. Szekely, and J. L. Ambite, "A graph-based approach to learn semantic descriptions of data sources," Springer, Berlin, Heidelberg, 2013, pp. 607–623.
[12] M. Perez-Ortiz, P. A. Gutierrez, C. Hervas-Martinez, and X. Yao, "Graph-Based approaches for over-sampling in the context of ordinal regression," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 5, pp. 1233–1245, May 2015.
[13] T. Moscibroda, R. Wattenhofer, and Y. Weber, "Protocol design beyond graph-based models," in *Proc. 5th ACM SIGCOMM Work. Hot Top. Networks*, 2006, pp. 1–6.
[14] X. Wang, Y. Ma, J. Di, Y. L. Murphey, S. Qiu, J. Kristinsson, J. Meyer, F. Tseng, and T. Feldkamp, "Building efficient probability transition matrix using machine learning from big data for personalized route prediction," *Procedia Comput. Sci.*, vol. 53, no. 1, pp. 284–291, 2015.
[15] M. Chen, X. Yu, and Y. Liu, "Mining moving patterns for predicting next location," *Inf. Syst.*, vol. 54, pp. 156–168, 2015.
[16] A. I. Petróczi and C. Gáspár-Papanek, "Route prediction on tracking data to location-based services," *Lect. Notes Comput. Sci.*, vol. 5733, pp. 69–77, 2009.
[17] R. Simmons, B. Browning, Y. Z. Y. Zhang, and V. Sadekar, "Learning to predict driver route and destination Intent," in *Proc. IEEE Intell. Transp. Syst. Conf.*, 2006, pp. 127–132.
[18] Q. Ye, L. Chen, and G. Chen, "Predict personal continuous route," in *Proc. IEEE Conf. Intell. Transp. Syst. Proceedings*, 2008, pp. 587–592.
[19] J. J. Pei, J. J. Han, B. Mortazavi-Asl, *et al*., "Mining sequential patterns by pattern-growth: The PrefixSpan approach," *IEEE Trans. Knowl. Data Eng.*, vol. 16, no. 11, pp. 1424–1440, Nov. 2004.
[20] L. Chen, M. Lv, and G. Chen, "A system for destination and future route prediction based on trajectory mining," *Pervasive Mob. Comput.*, vol. 6, no. 6, pp. 657–676, 2010.
[21] A. F. Merah, S. Samarah, A. Boukerche, and A. Mammeri, "A sequential patterns data mining approach towards vehicular route prediction in VANETs," *Mob. Networks Appl.*, vol. 18, no. 6, pp. 788–802, 2013.
[22] J. Froehlich and J. Krumm, "Route Prediction from Trip Observations," *Proc. SAE World Congr*., vol. 2193, p. 53, 2008.
[23] P. Helmholz, E. Ziesmann, and S. Robra-Bissantz, "Context-awareness in the car: Prediction, evaluation and usage of route trajectories," *Lect. Notes Comput. Sci.*, vol. 7939, pp. 413–419, 2013.
[24] Simulation of Urban MObility download | SourceForge.net. [Online]. Available: https://sourceforge.net/projects/sumo/
[25] F. Marchal, J. Hackney, and K. Axhausen, "Efficient map matching of large global positioning system data sets: Tests on speed-monitoring experiment in zürich," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 1935, pp. 93–100, Jan. 2005.
[26] M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser, "CRAWDAD dataset epfl/mobility (v. 2009-02-24)," CRAWDAD Wirel. Netw. data Arch., Jan. 2009.
[27] L. Codeca, R. Frank, and T. Engel, "Luxembourg SUMO Traffic (LuST) Scenario: 24 hours of mobility for vehicular networking research," in *Proc. IEEE Vehicular Networking Conference*, 2015, pp. 1–8.
[28] P. Fournier-Viger, A. Gomariz, T. Gueniche, A. Soltani, C. W. Wu, and V. S. Tseng, "SPMF: A java open-source pattern mining library," *J. Mach. Learn. Res.*, vol. 15, pp. 3569–3573, 2014.
[29] A. Bhattacharya and S. K. Das, "LeZi-Update: An information-theoretic framework for personal mobility tracking in PCS networks," *Wirel. Networks*, vol. 8, no. 2-3, pp. 121–135, 2002.

Hanane Amirat received her engineer degree from the University of Ouargla, Algeria, in 2010 and the M.S. degree from University of Laghouat (UL), Algeria, in 2013, both in computer science. She is currently pursuing the Ph.D. degree with the Department of computer science, UL. Her research interests include mobility mining, data mining, and intrusion detection in wireless ad-hoc networks.

Nasreddine Lagraa received his PhD in automation from the Ecole Nationale Polytechnique, Algeria (2008). He is currently the director of mathematics and computer science laboratory, University of Laghouat, Algeria. He is teaching various courses on computer architecture, multimedia, mobile networks and network security. His research interests are: Vehicular networks, network security, decentralized control, and fuzzy and artificial neural networks. He is serving in the technical program committees of many international conferences, Wireless and Mobile Computing, Networking and Communications (WiMob), and Innovations in Information Technology (IIT).

Philippe Fournier-Viger received his PhD in computer science from University of Quebec in Montreal (Canada) in 2010. He is currently a full professor in the school of Natural Sciences and Humanities and the Harbin and Institute of Technology in Shenzhen, China. He is Editor-in-Chief of Data Science and Pattern Recognition journal and Program Committee of the International Conference on Genetic and Evolutionary Computing. He is a member of review board of many journals Journal of Digital Information Management, International Journal of Advanced Pervasive and Ubiquitous Computing. He is a member of editorial review board of Interdisciplinary Journal of E-Learning and Learning Objects. He is serving on the technical program committees of many international conferences Pacific-Asia Conference on Knowledge Discovery and Data Mining(PAKDD), ACM SAC , International Conference on Knowledge Discovery and Information Retrieval (KDIR), international Conference on Advanced Data Mining and Applications ADMA ,etc. His research interests are: Data Mining, Big Data, Artificial Intelligence, Pattern Mining, Itemset Mining, Sequential pattern Mining, Sequence Prediction, Intelligent Tutoring Systems, and Applications of Data Mining.

Youcef Ouinten received his M.S degree in 1984, and his PhD in 1988, both in operational research from the Southampton university, United Kingdom. He is currently an assistant professor in the Department of Computer Science at Laghouat University, Algeria. His research interests are: Applied Mathematics, Data Mining, modeling and simulation, data processing and statistical analysis, Information and Social Networks Analysis, Text Mining, decisional information systems.