20th International Conference on Information Fusion
Xi'an, China - July 10-13, 2017

A Hybrid Model Towards Moving Route Prediction Under Data Sparsity

Liang Wang*, Mei Wang*, Tao Ku†, Yong Cheng*‡, Xinying Guo*
*Xi'an University of Science and Technology, Xi'an, China 710054
†Shenyang Institute of Automation (SIA), Chinese Academy of Sciences, Shenyang, China 110016
‡Xi'an University of Technology, Xi'an, China 710048

***Abstract*—Moving route prediction offers important benefits for many emerging location-aware applications such as target advertising and urban traffic management. A common approach to route prediction is to match similar trace recordings from a larger volume of historical trajectories, and return the targeted recorded path as desired answer. However, due to privacy concerns, incentive mechanism and other reasons, especially in small business environment, a limited dataset with sparse trajectories is only available. Actually, the existing sparse dataset cannot cover sufficient query routes, and then the match-based approach may return no results at all. Moreover, the existing sparse dataset may fail many trajectory mining approaches that work well on general environment. In this paper, we investigate moving route prediction from sparse trajectory dataset, and propose a novel hybrid model, namely HMRP, to address the above problem. To avoid sparse distribution over spatial semantic layer, a road network map reconstruction methods are proposed to accommodate the sparse trajectories in semantic transformation. And then, by training historical trajectories, the implicit mobility patterns and Markov transition model are constructed to support route prediction. When a query trajectory arrives, towards its derived potential destination, our proposed HMRP model integrates pattern matching strategy and Markov probability distribution to predict its future route gradually in a complementary way. Experiments on real-life taxicab GPS recorded dataset demonstrate that HMRP method can improve movement prediction precision significantly, comparing with the baseline prediction algorithms. And the response time for each query trajectory is acceptable for most application cases.**

I. Introduction

With the increasing popularity of handheld devices (e.g., GSP, smart phone, PDA etc.), trajectory information of mobile objects can be easily collected to support location-based services. Under many circumstances, relevant applications require predicting future moving routes of users, such as target advertising, location-based services and intelligent transportation, etc.. For example, advertising messages could be pushed to potential consumers who will pass a certain business zone. Besides, moving route prediction can improve the performance of car-sharing system by accurately estimating the demand-supply relationships [1].

By analyzing underlying characteristics of moving behavior from large volume of trajectory dataset, it is possible to predict a potential moving route for an ongoing query trajectory [2], [3], [4], [5]. Specifically, if this given query trajectory can be matched with extracted patterns or recorded trajectories through similarity comparison, the targeted historical path will be returned as the predicted moving route for this query trajectory. However, this common approach may fail to return an answer when it encounters sparse dataset. In sparse dataset, the existing recorded trajectories are far from enough to cover all possible query moving traces. Consequently, query trajectories may not be matched to existing trajectories, and their future moving routes are also unable to be predicted.

In practice, sparse trajectory dataset is inevitable due to privacy concerns, budget constraints, incentive mechanism and many other reasons, especially in small business environment. At present, even the largest available real-life trajectory dataset can not cover all the possible moving routes in a larger metropolitan city [6]. Actually, for moving route prediction problem, sparse trajectory dataset poses a realistic challenge that has not yet been tackled before. In this paper, towards the above-mentioned problem, we propose a novel hybrid model, namely HMRP(**H**ybrid **M**oving **R**oute **P**rediction), to predict moving route using sparse trajectory dataset. Firstly, in order to avoid sparse distribution over spatial semantic layer, we adopt a multi-granularity space division strategy to support moving destination prediction, and devise a road network reconstruction method to achieve map-matching from raw trajectories to semantic sequences. Afterwards, based on transformed semantic sequences, the implicit mobility knowledge, including moving frequent patterns and transition probability model (i.e., Markov), can be trained from learning existing trajectory history. Finally, towards a destination derived by Bayesian inference, the given partial query trajectory grows gradually based on a hybrid route prediction model. Specifically, in the process of prediction, pattern-matching strategy predict near future moving route based on matching moving patterns discovered in advance; and when encountering no-pattern matching, Markov model can be used alternatively to predict next moving location in one step based on transition probability. By this complementary way, the problem of moving route prediction using sparse dataset can be alleviated significantly.

In summary, we make the following contributions in this article:

- Towards the moving route prediction problem under data sparsity, we propose a novel hybrid model to address this problem. By integrating pattern matching approach and Markov probability model, the proposed model can dynamically derive potential future route in a complementary way. By this way, the route prediction accuracy can

978-0-9964-5270-0©2017 ISIF

be improved by long discovered moving patterns; while the prediction robustness can be guaranteed by Markov transition probability distribution.

- To avoid the deterioration of sparse dataset over large road network, we propose a method to simplify and restructure road network by detecting road intersections based on historical trajectory dataset. By leveraging the detected road intersection nodes, we devise intersection-centric route matching and representation strategy to adapt to the problem studied in this paper.
- We conduct extensive experiments by using a sparse GPS dataset collected from a sample of taxicabs in Shenzhen city, China to evaluate the prediction accuracy and runtime efficiency of our proposed HMRP approach. Compared with baseline algorithms, the results demonstrate that our proposed approach can consistently achieve better prediction accuracy of moving routes. Moreover, the hybrid prediction model is also efficient.

The remainder of this article is structured as follows. Sect.2 presents related work on moving route prediction. The preliminaries of this paper, is described in Sect.3. In Sect.4, we restructure urban road network from trajectory history to avoid more sparse distribution over road network. The hybrid moving route prediction model is proposed in Sect.5. Experimental results are reported and discussed in Sect.6. Finally, Sect.7 concludes this paper.

II. Related Work

From a technological perspective, the most popular schemes to moving route prediction are similarity measurement-based strategy [2], [7] and pattern mining-based strategy [8], [9], [10], [11], [12]. The former approach calculates similarity between a current query trajectory and history trajectories. According to the obtained similarity measurements, it returns a recorded similar moving path as its future route. While the latter one executes frequent pattern mining on existing recorded trajectories to find moving patterns and association rules with a given confidence. For a query partial trajectory, it retrieves pattern repository to find matched moving patterns. And one of the matched popular patterns would be chosen as the future moving path. However, all of the above mentioned techniques suffer form sparse dataset problem. As the available historical trajectory dataset could only cover a portion of moving routes for various original-destination pairs. These two common approaches may return no prediction results due to insufficient information. Thus, the approaches could not be applied into sparse dataset problem we studied in this work. In addition, a few research studies leverage Markov model to extract mobility transition probability and construct prediction model [13], [14], [15]. The research [13] employs a K-bounded VMM to determine the transition probability by knowing at most K previous states. But how to choose an appropriate order of a higher-order Markov model remains an open question. And it is generally harder to find longer sequences in trajectory history to train a higher-order model. Most importantly, according to the existing works [14], [15], even a second-order Markov model require a large volume of training data. Hence, it is impossible to achieve a robust high-order Markov model by using sparse and limited training dataset.

In addition to moving route prediction, a most relevant research field is destination prediction. Recently, some research effort has been made on the moving destination prediction [6], [16], [17], [18], [19], [20], [21]. For example, Alvarez-Garcia et al. extract clustered destination locations form historical journey data, and a Hidden Markov Model is employed to predict the possible destination point [16]. Kostov et al. propose a moving destination prediction method incorporating external information, such as day and time [17]. Marmasse, N. et al. predict a personalized destination by employing a probability model learned from historical moving information [18]. Tanaka et al. propose a car navigation system to predict the destination information on the basis of vehicle driving trajectory and additional contexts information [19]. Xue et al. study the problem of data sparsity in destination prediction, and propose a novel decomposition-synthesis mechanism to enlarge the space of matched trajectories [6], [20], [21].

III. Preliminaries

In this section, we give essential preliminaries, including problem definition and destination prediction strategy.

A. Problem Definition

A complete moving trip which contains n GPS samplings is often represented in the form of a sequence of time-order points, such as $Tr = \{(x_1, y_1, t_1), (x_2, y_2, t_2), ..., (x_n, y_n, t_n)\}$, where (x_i, y_i), $1 \leq i \leq n$, denotes a geospatial coordinate point, and t_i is the corresponding timestamp. In the scenario of moving route prediction, online query trajectory is ongoing and uncompleted. Formally, the online partial query trajectory Tr_q can be represented as $Tr_q = \{(x_1, y_1, t_1), (x_2, y_2, t_2), ..., (x_m, y_m, t_m)\}$. Compared with the above complete trip Tr, the length of query trajectory (i.e., m) is less than Tr's, that is $m < n$. For a query trajectory, in which the origin location and partial moving route is given, the task of route prediction is to predict its unknown trajectory based on the given information.

Definition 3.1 (*Trip Complete Percentage*) Trip complete percentage is defined as the ratio between the length of query trajectory and the length of its actual and complete trajectory. For the above-mentioned query trajectory Tr_q, if we assume that its real and complete trace is Tr=$\{(x_i, y_i, t_i), 1 \leq i \leq n\}$, the trip complete percentage of Tr_q can be calculated as: m/n. It is noticeable that the greater the trip complete percentage, more information can be known.

Definition 3.2 (*Moving Pattern*) A moving pattern is a sequential pattern of spatial elements (e.g., spatial regions or road nodes and so on) with a support which is no less than a predefined threshold value.

Next, we formally define the moving route prediction problem as follows:

Moving Route Prediction: Given a historical trajectory dataset TD, for an ongoing partial query trajectory Tr_q=$\{(x_i, y_i, t_i), 1 \leq i \leq m\}$, our task is to derive a most likely moving path as this query trajectory's future route.

B. Destination Prediction by Bayesian Inference

In practice, the determination of moving route is strongly related with origin and destination locations. For a given query partial trajectory, as the origin location is known, its moving destination will play an essential role in the choice of moving route. In other words, when a potential moving destination can be determined, the search space of its possible routes will be shrunk dramatically. Moreover, in order to terminate the process of moving route prediction, it is necessary to obtain a possible destination in advance. Based on this discussion, moving destination of query trajectory should be firstly predicted.

Currently, the most popular approach to moving destination prediction is to utilize a set of discrete regions to represent the covered space, and perform probability calculation to derive a potential destination based on trajectory history [6], [18], [20], [22]. In this work, we also follow the paradigm. In this paper, we employ *k-d* tree-based division method to partition covered space into multi-granularity regions. The *k-d* tree-based division method partitions the covered space by using either vertical lines or horizontal lines of equal number of location points. By repeating the partition process recursively for vertical and horizontal dimensions, a set of divided regions can be achieved, that is $RE = \{Re_i, 1 \leq i \leq n\}$, where Re_i is a divided region.

Based on the divided regions, we can build corresponding mobility association relationship from learning historical trajectory dataset, including origin-destination association and mobility state transition relationship. Following the approaches prorposed by Xue et al in [6], [20], a possible destination region of given query trajectory can be derived by leveraging Bayesian inference, The deduction is formalized as follows:

$$P(Re_{des}|Tr_q) \propto \frac{p_{cur \to des}}{p_{org \to des}} * P(Re_{des}|Re_{org}). \qquad (1)$$

In above equation, for query trajectory Tr_q, symbol $p_{cur \to des}$ denotes transition probability from current location to predicted destination region, and $p_{org \to des}$ is the transition probability from original location to predicted destination region. The value of $P(Re_{des}|Re_{org})$ can be calculated by the established origin-destination association. And the probabilities from current and original location to the predicted destination can be derived by mobility state transition relationship. Based on the above-mentioned Bayesian inference representation, we can derive the probability of possible regions being destination for a given query trajectory Tr_q, and the corresponding predicted destination can be achieved.

IV. Road Network Restructure from Moving Trajectory

On the condition of sparse dataset, as the scale of road network elements is usually too larger, especially in larger metropolitan city, the transformed trajectory-sequences will be distributed over a more sparse space. For instance, the number of road segments and nodes in Shenzhen city in which our training dataset is collected are about 26,928 and 24,324 respectively. In that case, the sparse dataset problem would deteriorate even further if we use the complete road elements to represent original trajectories. In order to adapt to the sparse dataset, we consider to adopt some small-scale but representative road network elements to translate raw trajectories, and represent moving pathes accordingly. Based on these representative elements, a simplified road network can be reconstructed to approximate the real road network map. By this way, the problem of sparse trajectories can be alleviated remarkably.

At road intersections, there are some obvious characteristics which can be reflected by recorded moving trajectories. In reality, vehicles often keep their moving direction unchanged until meeting intersections on road network. That is to say, vehicle turnings can be observed near road intersections. Accordingly, this event can also be reflected in the moving historical trajectories. On the basis of this common sense, it is feasible to identify road intersections by detecting direction changes of moving trajectories.

First of all, the Douglas-Peucker algorithm is hired to execute line simplification on raw trajectories to filter fluctuant points in original trajectories. And then, for each complete moving trip Tr, we link two consecutive locations p_i and p_{i+1}, where $p_i = (x_i, y_i)$ and $i = 1, 2, ..., n$, successively to yield a set of moving segments $L = \{l_i = \overrightarrow{p_i p_{i+1}}\}$, $i = 1, 2, ..., n-1$.

Definition 4.1 (*Moving Segment Direction*) Moving direction of a trajectory segment l_i is defined as an angle ϕ_i between l_i and a horizontal line.

Definition 4.2 (*Moving Direction Change*) Moving direction change between two consecutive moving segments is defined formally as $\Delta\phi = \phi_{i+1} - \phi_i$. In other words, the difference between a former segment l_i's moving direction ϕ_i and its subsequent segment l_{i+1}'s direction ϕ_{i+1} is defined as $\Delta\phi$.

As defined above, the symbol ϕ_i is used to depict the moving direction for two consecutive trajectory locations. In fact, on the one hand, moving direction change $\Delta\phi$ indicates the corresponding change of moving route, and on the other hand, it reflects the underlying structure of road network by which objects are restricted in the moving process.

By investigating $\Delta\phi$ for m consecutive trajectory segments, an aggregate value $\rho = \sum_{i=1}^{m} \Delta\phi_i$ can be obtained. Considering the sampling interval, moving speed and practical traffic facilities, we select $m = 3$ as a default value to detect intersections in urban road network. Then, we decide candidate road intersections by comparing the aggregate value ρ with an empirical value γ. In this article, the empirical value γ is defined as two interval values $[80°, 100°]$ and $[260°, 280°]$.

The role of the empirical value γ is to restrict ρ approximate to a right-angle 90° or 270° with a margin of 20°. That is to say, if the aggregate value ρ falls into interval value $[80, 100]$ or $[260, 280]$, a potential road intersection may exist in this zone which is covered by the relevant $m + 1$ trajectory locations. Among these $m + 1$ location points, the first and last one are used to generate a candidate road intersection. Specifically, the horizontal and vertical coordinate values of the candidate road intersection can be determined as the horizontal value of the first location and vertical value of the last location point, respectively. This relationship is represented as $p_1 \bigotimes p_{i+m}$ in this work.

Let's use a toy example to demonstrate the process of road intersections detection. A partial moving trajectory which comprises four successive sampling locations p_1, p_2, p_3 and p_4 are shown in Figure 1. By linking trajectory location p_i with its subsequent location point p_{i+1}, we can get three consecutive moving segments l_1, l_2 and l_3, where $l_1 = \overrightarrow{p_1p_2}$, $l_2 = \overrightarrow{p_2p_3}$ and $l_3 = \overrightarrow{p_3p_4}$, in which the symbol of arrow represents the moving orientation. The moving direction of these three moving segments are defined as ϕ_1, ϕ_2 and ϕ_3, respectively. And then, these three segments' moving direction changes $\Delta\phi_1$, $\Delta\phi_2$ and $\Delta\phi_3$ can be computed separately. As the aggregate value $\rho = \sum_{k=1}^{3} \Delta\phi_k$ falls into interval value $[80, 100]$, a candidate road intersection, which is marked in blue in Fig. 1, can be produced by $p_1 \bigotimes p_4$.

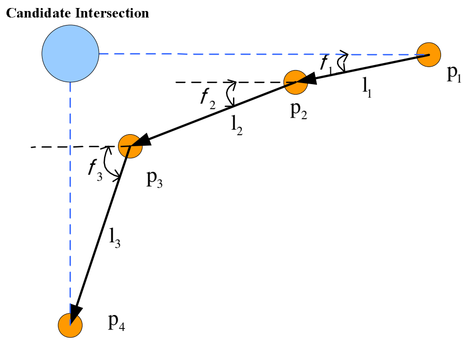

Fig. 1. Detecting the road intersections from GPS data

Finally, we refine the set of candidate intersections to improve the performance of final results. Concretely, we leverage density-based spatial clustering DBSCAN algorithm to group the candidate intersection points. By tuning parameters $MinPts$ and ε in DBSCAN algorithm, a group of clustered candidate nodes can be obtained. In this way, the hot road intersections which occurs frequently in historical trajectories can be identified successfully. At last, the center of each cluster is picked out as the final desired road intersection Ri.

V. Moving Route Hybrid Prediction Model

As explained above, pattern matching-based prediction approach can exploit all the information in query trajectory Tr_q, and improve the prediction accuracy by matching long discovered patterns with non-fixed steps. However, this approach may fail to return results due to no matching, especially for sparse trajectory dataset studied in our work. A potential reason is that the discovered frequent patterns only concerns moving behaviors whose frequency is not less than a predefined support threshold, rather than all the moving behaviors in existing recorded trajectories. Compared with the frequent patterns, Markov model can characterize the complete moving behavior represented by probability distribution in trajectory data. Nevertheless, the moving transition step (i.e., the order of Markov model) is usually fixed and limited. So to speak, frequent patterns explore the global moving characteristics; while Markov model depicts local characteristics with short and fixed steps in moving traces.

In this section, we propose a hybrid route prediction model by integrating pattern matching-based approach and a first-order Markov probability model. Based on the hybrid prediction model, the query partial trajectory can grow gradually towards its predicted destination. In the process of route prediction, pattern matching-based approach is used to predict future route based on movement patterns discovered in advance. When encountering no-pattern matching, the Markov model can be employed to predict the next moving location in one step. The role of pattern-matching approach is to take advantage of given partial trajectory information and predict route with more than one steps. While the Markov model is to overcome no-pattern matching problem in prediction stage. Via this complementary mechanism, the data sparsity problem in route prediction can be solved in the dynamic process. In order to improve the prediction performance, we modify the pattern matching strategy and Markov probability model to make these approaches more adapted to the problem we studied.

A. Movement Frequent Pattern Index and Pattern Matching Degree

First of all, using the reconstructed road network, original moving trajectories are transformed into semantic sequences represented by detected road intersections. The transformation process is based on distance measurement, and a distance threshold is selected as 200 m based on the average distance among detected road intersections. The classical sequence pattern mining algorithm PrefixSpan is employed to discover hidden frequent patterns in historical trajectories. In order to index discovered patterns, a *FP-Tree* is built according to these patterns' prefixes[23]. The *FP-Tree* is constructed in the process of mobility pattern discovering with a pattern growth way. Each discovered frequent mobility pattern is inserted by scanning the *FP-Tree* to find that if there is any branch corresponds to the prefix of the pattern. When the frequent pattern mining process finishes, all the discovered patterns will be indexed in the *FP-Tree*.

In pattern-based prediction approach, moving pattern matching procedure is implemented to find the candidate patterns whose prefix can matches the query trajectory sequence. Generally speaking, there are three common matching strategies, such as **complete matching**, **last matching** and **longest last matching** [24]. However, the complete matching strategy often fail to return a matching movement pattern due to its strict constraints, especially in sparse dataset. While the prediction results from the last matching strategy is difficult to achieve satisfactory performance, as it does not consider given trajectory information except current location. The longest last matching can be regarded as a compromise between the two above mentioned strategies. It focuses on the relative coverage of the query trajectory with respect to the recorded patterns to be matched. In addition to the relative matching coverage, we argue that the distance between current location in query trajectory also play a very important role in prediction performance. Thus, we propose a novel matching strategy which integrates matching coverage, matching distance and pattern support value to evaluate matching degree of a discovered pattern. According to matching degree of candidate patterns, an appropriate movement pattern could be picked out as predicted route for a query trajectory.

For a given query trajectory Tr_q, matching degree of a candidate matched pattern $CmPatt_i$ can be defined as follow:

$$degree_i = \frac{cov_i}{dis_i} \times sup_i, \tag{2}$$

where cov_i denotes the matching coverage which is calculated by the length of matched part in $CmPatt_i$, dis_i is aggregated distance between query trajectory Tr_q's current location and candidate pattern $CmPatt_i$, sup_i is support value of $CmPatt_i$. The symbol dis_i is formalized as follow:

$$dis_i = \sum_{k=1}^{cov_i} \left| e_k - e^{'}_{end} \right|, \tag{3}$$

where e_k is an matched item in $CmPatt_i$ with regard to query trajectory Tr_q, and $e^{'}_{end}$ is current location item in Tr_q. As shown in equation 2, the larger the matching coverage cov_i, the higher the matching degree of candidate pattern $CmPatt_i$. While the distance dis_i is in inverse proportion to candidate pattern's matching degree, as if a matched pattern $CmPatt_i$ is far away from current location in Tr_q, the probability that Tr_q follow $CmPatt_i$ will decay accordingly. Note that, as the discovered frequent patterns are usually short-length, only a chosen pattern cannot lead Tr_q reaching its predicted destination. So the prediction process will repeated in many steps until meeting its predicted destination.

B. Markov Transition Probability Model

To avoid the problem of returning no result, we consider using Markov model as a alternate option when encountering no matching. As explained above, a first-order Markov model is used to model moving trajectories by associating a state to each detected road intersection. And the transition probability is represented as moving from one intersection to another, which is denoted by $p_{i,j}$. The transition probability $p_{i,j}$ can be calculated as the number of trajectories which contain moving sequence $Ri_i \rightarrow Ri_j$ divided by the number of trajectories which include the element Ri_i. The $p_{i,j}$ can be formalized as follow:

$$p_{i,j} = \frac{\left| T_{Ri_i \rightarrow Ri_j} \right|}{\left| T_{Ri_i} \right|}, \tag{4}$$

where T_{Ri_i} denotes the total number of transformed trajectory sequences which include Ri_i, and $T_{Ri_i \rightarrow Ri_j}$ is the number of sequences containing subsegment $Ri_i \rightarrow Ri_j$. The transition probability $p_{i,j}$ for each intersection pair Ri_i and Ri_j can be derived by traversing history trajectory dataset. The value of $p_{i,j}$ is recorded as an entry in a two dimensional $n \times n$ matrix M_{Trj}, where n is the number of extracted intersection nodes. During transition probability calculation, each complete trip sequence is decomposed into a set of subsequences of length 1. For example, a trip sequence Tr which is represented as $\langle Ri_3, Ri_{15}, Ri_9, Ri_{32}, Ri_{46} \rangle$ can be decomposed into four subsequences, such as $Ri_3 \rightarrow Ri_{15}$, $Ri_{15} \rightarrow Ri_9$, $Ri_9 \rightarrow Ri_{32}$ and $Ri_{32} \rightarrow Ri_{46}$. These subsequences can contribute to transition probabilities $p_{3,15}$, $p_{15,9}$, $p_{9,32}$ and $p_{32,46}$, respectively. This process is effectively decomposing each complete trip sequence in dataset. Finally, the Markov transition matrix M_{Trj} can be constructed.

Based on transition matrix M_{Trj}, moving route prediction can be conducted in one step. For a given query trajectory Tr_q, assuming its current location is mapped to intersection Ri_i, the entries in $i{-}th$ row of matrix M_{Trj} are the probability distribution of next location. From the probability distribution, a candidate intersection node can be picked out as the query trajectory's next location. To enhance the prediction accuracy, we devise a prediction strategy which considers transition probability, the distances from current location to original and predicted destination location. For the current location in Tr_q, we firstly choose *top-k* possible intersection nodes according to probability distribution of $i - th$ row in M_{Trj}. For each selected candidate intersection, we calculate the distances to Tr_q's original location and predicted destination. Assuming the distance between candidate intersection Ri_i and Tr_q's original location is d^i_{orig}, distance between Ri_i and the predicted destination is d^i_{dest}, we define Ri_i's score is listed as follows:

$$score(Ri_i) = \frac{d^i_{orig}}{d^i_{dest}} \times pro(Ri_i), \tag{5}$$

where $pro(Ri_i)$ is probability of Ri_i in Markov model matrix M_{Trj}. The role of equation (5) is to lead the predicted route getting close to the predicted destination and keeping away from its original location. Then, the selection of final intersection node can be represented formally as follow:

$$argMax : Score(Ri_i), i = 1, 2, ..., k. \tag{6}$$

C. Hybrid Moving Route Prediction Approach

Based on the pattern matching approach in Sec. 5.1 and Markov probability model explained in Sec. 5.2, we propose our hybrid moving route prediction method as follow. When query trajectory Tr_q arrives, the transformation process is implemented to converted it into sequences represented by the discrete divided regions. And then based on the Bayesian destination inference paradigm explained above, a possible destination Re_q of this query trajectory Tr_q is obtained. When the destination location Re_q can be determined, the hybrid prediction model will lead the partial trajectory growing towards Re_q to generate its future moving route in road-intersection representation. More specifically, we firstly retrieve ***FP-Tree*** to find candidate patterns which can be matched with the partial trajectory sequence. By calculating matching degrees of candidate patterns from equation 2, a final appropriate pattern will be selected as the predicted near future route. If no-pattern matching happened, the established Markov model could be used to predict its next moving location. Among *top-k* candidate intersections, the intersection having maximal score is selected as the predicted next location. It should be noted that, after attaching a predicted next location, query trajectory Tr_q may be able to match with stored frequent patterns as it extends with a one length item. And then, the pattern-matching based approach would be leveraged again to predict its future route. By this way, we can gradually predict future moving route for a given trajectory, until reaching its predicted destination or a predefined traveling distance. The procedure of route prediction is presented in Algorithm 1 as follow. It is note that the symbol $CmPatt_{max}$ is candidate pattern having maximum matching degree.

ALGORITHM 1: Hybrid Moving Route Prediction

```
Input: A query trajectory $Tr_q$, top-k parameter,
discovered pattern base $PattBase$, Markov transition
model $M_{Trj}$ and User-specified predicted distance $DP$.
Output: A predicted moving route $PreRoute$.
Predicted Destination $Re_q$ = PredictDes($Tr_q$);
$PreRoute$ = $Tr_q$;
Repeat
    $CmPatt$ = PattSearch($PreRoute$,$PattBase$);
    If $CmPatt \neq$ then
        For each candidate pattern in $CmPatt$ do
            $CmPatt_{max}$ = MatchDegree($CmPatt$);
        End
        $PreRoute$ = $PreRoute \cup CmPatt_{max}$;
    else
        Calculate probability distribution of $M_{Trj}$;
        Compute $Score$ for next possible $Ri$;
        $PreRoute$ = $PreRoute \cup Ri_{max}$;
    End
Until $Dis(PreRoute) > DP$ or $PreRoute$ = $Re_q$;
```

VI. Performance Evaluation and Discussion

A. Experiment Set-up

We use a real-life GPS dataset which consists of more than 30,000 practical trips recorded over a period of ten day from a sampled number of taxicabs (i.e., 200) in Shenzhen city, China. It contains a total of 0.172 million kilometers of distance traveled, and 0.90 million GPS sampling points. Among the raw recorded trajectories, 5,000 complete trajectories are selected to be the tested trajectories, while the remaining trajectories are used as the training dataset. Our experiments and latency observations are conducted on a standard server (Windows), with Intel Core i3-3110M CPU, 2.40 GHz and 4 GB main memory.

In our experiments, we use the following two means of measurement to evaluate the performance of our proposed route prediction approach, such as Prediction Route Deviation (**PRD**) and Response Time for Each query trajectory (**RTE**). The PRD index is to calculate the average deviation between predicted moving route and the real moving trajectory; while the RTE is to measure the efficient of online prediction.

To the knowledge of our best, no existing work can be directly applied to the problem studied in this work. As described above, the common approach, pattern matching-based method, will fail to return suggestions when encountering no-pattern matching. For fair comparison, this approach is not hired as baseline prediction algorithm. Nevertheless, its performance will be analyzed in the subsequent experiments. In our experiments, we adapt the idea of Markov model, and generate two baseline prediction algorithms: first-order Markov-based and second-order Markov-based method, respectively.

B. Experiment Results and Analysis

The performance of our proposed route prediction algorithm HMRP is evaluated by aforementioned two measurements against varying three parameters one at a time. First of all, we vary the trip completed percentage (10-50% with 10% increment). The second parameter is *top-k* candidate next moving intersection node in Markov model. When we use Markov model to predict the next moving route, at least *k* candidate locations are suggested by the model. In experiments, the parameter *k* is set from 1 to 3 to evaluate its impact on prediction accuracy and efficiency. The last parameter is predicted distance parameter DP which is used to restrict the predicted distance of future moving route. DP is varied from 3 km to 10 km in our experiments. Many other parameters are listed as follows: the average partitioned area is $9.9645km^2$, $MinPts = 20$, $\varepsilon = 0.16km$, and the final number of obtained road intersections is 806.

1) Varying the percentage of trip completed: Fig. 2 shows the performance of moving route prediction versus the trip completed percentages for 1000 query trajectories, the *top-k* parameter is set 3 and the predicted distance DP equals $5km$. It is observed that our proposed HMRP approach markedly outperforms the baseline Markov-based prediction algorithms. Specifically, the prediction average deviation PRD of HMRP algorithm outperforms the first-order Markov algorithm by 37.49%, the second-order Markov algorithm by 44.50%. Remarkably, the prediction performance of high-order Markov approach is less than the first-order Markov approach. The reason is that, in order to train a robust high-order Markov model, much more training trajectories are required as high-order Markov model incorporating more previous state information. While in the condition of sparse dataset, this is

TABLE I
PREDICTION RESULTS OF VARYING TRIP COMPLETED PERCENTAGE

Trip Completed Per.	10%	20%	30%	40%	50%
RTE of 1st Markov	0.086s	0.073s	0.067s	0.054s	0.048s
RTE of 2st Markov	0.091s	0.085s	0.079s	0.067s	0.060s
RTE of HMRP	0.097s	0.123s	0.147s	0.175s	0.206s
Ave. Steps of HMRP	17.0	16.3	16.0	15.6	14.5

impossible, and thus much more zero entries are presented in high-order Markov matrix. Too sparse Markov model will result in performance degradation in prediction accuracy. This result is also consistent with the explanation above mentioned.

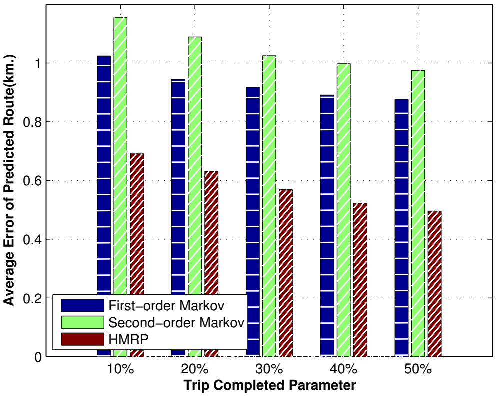

Fig. 2. Route prediction precision versus trip completed percentage

The following Table 1 shows that the average predicted step in HMRP drops noticeably from 17.0 to 14.5 as the trip completed percentage varies from 10% to 50%. For the efficiency of these three algorithms, the RTE of HMRP algorithm is longer than baseline algorithms, as additional pattern retrieval and matching operation is implemented in HMRP approach. Furthermore, the RTE of baseline algorithms reduce as the trip completed percentage increases, on the contrary, RTE of HMRP algorithm increases in this process. The reason is that as the trip completed percentage increases, the baseline algorithms will predict less steps for each query trajectory. So the RTE becomes short accordingly. While longer query trajectory (higher trip completed percentage) may match more discovered movement patterns recorded in repository, the time expenditure on pattern search and matching will increase accordingly. However, the maximum response time in HMRP is 0.206 second for each query trajectory, that is sufficient for most application cases.

2) Varying the top-k parameter in Markov model: We also investigate the effect of the *top-k* parameter on the performance of prediction algorithms by varying the value of k from 1 and 3, respectively. The experiments are conducted on tested trajectories from 1000 to 5000 with 1000 increment, the trip completed percentage and predicted distance parameter are set to 30% and $5km$. The experimental results are shown in Fig. 3. From the results we can see, our proposed HMRP algorithm shows a more accurate prediction performance than baseline algorithms. And the *top-k* parameter can improve the prediction performance noticeably for both HMRP algorithm and baseline algorithms. The reason is that $k = 1$ means that a road intersection with maximum probability value is chosen directly as the next moving location, some potential intersections are excluded in the prediction process. While if the parameter k is set to 3, the first three possible intersections are selected according to their scores computed by equation (5). By this way, the selection space of potential locations (i.e., 3 intersection nodes) can be extended and the prediction precision can be increased.

Moreover, it is obvious that the effect of *top-k* parameter on HMRP algorithm is less than that on the baseline algorithms. The reason may be that not all the prediction procedure is conducted by Markov model in HMRP, pattern matching strategy is also implemented by about 37% in the experiments. That is to say, the common pattern matching-based approach may fail to return a prediction result for about 63% query trajectories in the test data set due to the data sparsity problem in these experiments. This also explain that pattern matching-based prediction approach can not applied to the sparse dataset circumstances.

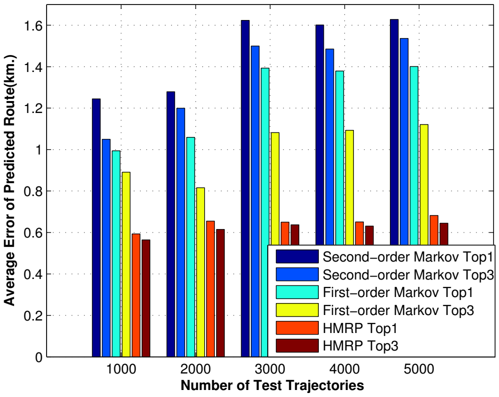

Fig. 3. Route prediction precision versus top-k parameter

3) Varying the predicted distance parameter: We also study the effect of the predefined predicted distance on the final prediction results. As shown in Fig. 4, by varying the predicted distance from $3km$ to $10km$, the performance of the baseline prediction algorithms deteriorate distinctly; while little impact is observed for the proposed HMRP algorithm. Meanwhile, the pattern matching strategy is implemented averagely for about 32.7% for the 1000 testing trajectories in the experiments. It indicate that if the common pattern-based prediction approach is solely used to predict moving route for the testing trajectories, about 67.3% trajectories will encounter no-pattern matching and return no suggestions finally. This proves that our proposed HMRP algorithm can overcome the data sparsity problem while maintaining a stable performance. Furthermore, RTEs and the average predicted steps in all the algorithms are shown in Table 2. As can be seen from the table, the average predicted step increases when the predicted distance parameter

TABLE II
PREDICTION RESULTS OF VARYING PREDICTED DISTANCE PARAMETER

Predicted Distance	3km	5km	7km	9km	10km
RTE of 1st-order Markov	0.059s	0.065s	0.083s	0.096s	0.121s
RTE of 2st-order Markov	0.066s	0.076s	0.091s	0.110s	0.133s
RTE of HMRP	0.105s	0.166s	0.213s	0.298s	0.320s
Ave. Steps of HMRP	9.39	15.64	21.9	28.01	31.06

grow from 3 km to 10 km. The reason is obvious that long distance brings more predicted steps in the process.

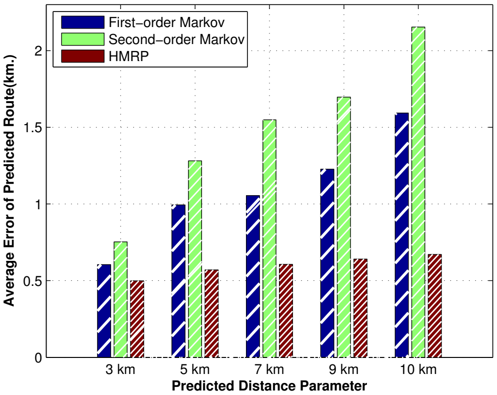

Fig. 4. Route prediction precision versus predicted distance parameter

VII. CONCLUSION

In this paper, we investigate the moving route prediction problem with sparse dataset, and propose a novel approach termed HMRP to overcome this problem. In view of the sparse distribution of existing historical trajectories, we employ a multi-granularity space division and devise road network reconstruction strategy to represent the original GPS data on region and road node level, respectively. As the common prediction approach may fail to return a suggestion result for a query due to no-pattern matching, we alternatively leverage pattern matching method and Markov prediction model, the proposed hybrid prediction frame lead the query trajectory growing gradually towards its inferred destination in a complementary way. Experiments based on real-life dataset show that the performance of HMRP algorithm outperforms the baseline algorithms.

ACKNOWLEDGMENT

The research is supported partially by the National Natural Science Foundation of China (Grant No. 61402360, 61401015), the Foundation of Shaanxi Educational Committee(Grant No. 16JK1509).

REFERENCES

[1] Karbassi, A, and M. Barth. Vehicle route prediction and time of arrival estimation techniques for improved transportation system management. Intelligent Vehicles Symposium, 2003. Proceedings. IEEE IEEE Xplore, 2003:511-516.

[2] Froehlich, Jon, and John Krumm. Route prediction from trip observations. No. 2008-01-0201. SAE Technical Paper, 2008..

[3] Chen, Ling, M. Lv, and G. Chen. A system for destination and future route prediction based on trajectory mining. Pervasive and Mobile Computing 6.6(2010):657-676.

[4] Chen, Ling, et al. A personal route prediction system based on trajectory data mining. Information Sciences 181.7(2011):1264-1284.

[5] Ye, Qian, L. Chen, and G. Chen. Predict Personal Continuous Route. International IEEE Conference on Intelligent Transportation Systems IEEE Xplore, 2008:587-592.

[6] Andy Yuan Xue, Jianzhong Qi, Xing Xie, Rui Zhang, Jin Huang and Yuan Li. Solving the data sparsity problem in destination prediction. The VLDB Journal 24.2 (2015): 219-243.

[7] Kim, Sang-Wook, et al. Path prediction of moving objects on road networks through analyzing past trajectories. International Conference on Knowledge-Based and Intelligent Information and Engineering Systems. Springer Berlin Heidelberg, 2007.

[8] Monreale, Anna, et al. Wherenext: a location predictor on trajectory pattern mining. Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2009.

[9] Morzy, M. Prediction of moving object location based on frequent trajectories. International Symposium on Computer and Information Sciences. Springer Berlin Heidelberg, 2006.

[10] Ying, Josh Jia-Ching, Wang-Chien Lee, and Vincent S. Tseng. Mining geographic-temporal-semantic patterns in trajectories for location prediction. ACM Transactions on Intelligent Systems and Technology (TIST) 5.1 (2013): 2.

[11] Győző Gidófalvi and Christian Borgelt and Manohar Kaul. Frequent route based continuous moving object location-and density prediction on road networks. Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. ACM, 2011.

[12] Tseng, Vincent S., and Kawuu W. Lin. Efficient mining and prediction of user behavior patterns in mobile web systems. Information and software technology 48.6 (2006): 357-369.

[13] Guangtao Xue and Zhongwei Li and Hongzi Zhu and Yunhuai Liu. Traffic-known urban vehicular route prediction based on partial mobility patterns. Parallel and Distributed Systems (ICPADS), 2009 15th International Conference on. IEEE, 2009.

[14] Ashbrook, Daniel, and Thad Starner. Using GPS to learn significant locations and predict movement across multiple users. Personal and Ubiquitous computing 7.5 (2003): 275-286.

[15] Amiya Bhattacharya and Sajal K. Das. LeZi-update: An information-theoretic framework for personal mobility tracking in PCS networks. Wireless Networks 8.2/3 (2002): 121-135.

[16] J.A. Alvarez-Garcia, J.A. Ortega, L. Gonzalez-Abril and F. Velasco. Trip destination prediction based on past GPS log using a Hidden Markov Model. Expert Systems with Applications 37.12 (2010): 8166-8171.

[17] Kostov, V. and Ozawa, J. and Yoshioka, M. and Kudoh, T. Travel destination prediction using frequent crossing pattern from driving history. Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE.

[18] Marmasse, Natalia, and Chris Schmandt. A user-centered location model. Personal and ubiquitous computing 6.5-6 (2002): 318-321.

[19] Kohei Tanaka and Yasue Kishino and Tsutomu Terada and Shojiro Nishio. A destination prediction method using driving contexts and trajectory for car navigation systems. Proceedings of the 2009 ACM symposium on Applied Computing. ACM, 2009.

[20] Andy Yuan Xue, Rui Zhang, Yu Zheng, Xing Xie, Jin Huang and Zhenghua Xu. Destination prediction by sub-trajectory synthesis and privacy protection against such prediction. Data Engineering (ICDE), 2013 IEEE 29th International Conference on. IEEE, 2013.

[21] Andy Yuan Xue, Rui Zhang, Yu Zheng, Xing Xie, Jianhui Yu and Yong Tang. Desteller: A system for destination prediction based on trajectories with privacy protection. Proceedings of the VLDB Endowment 6.12 (2013): 1198-1201.

[22] Krumm, John, and Eric Horvitz. Predestination: Inferring destinations from partial trajectories. International Conference on Ubiquitous Computing. Springer Berlin Heidelberg, 2006.

[23] Borgelt, C. (2005, August). An Implementation of the FP-growth Algorithm. In Proceedings of the 1st international workshop on open source data mining: frequent pattern mining implementations. ACM, 2005: 1-5.

[24] Morzy, M. Mining frequent trajectories of moving objects for location prediction. International Workshop on Machine Learning and Data Mining in Pattern Recognition. Springer Berlin Heidelberg, 2007.