

PAPER • OPEN ACCESS

Trajectory Prediction Based on Machine Learning

To cite this article: Lumin Su and Li Li 2020 *IOP Conf. Ser.: Mater. Sci. Eng.* **790** 012032

View the [article online](#) for updates and enhancements.

You may also like

- [Improved predictive performance of cyanobacterial blooms using a hybrid statistical and deep-learning method](#)
Hu Li, Chengxin Qin, Weiqi He et al.
- [Ionospheric vertical total electron content prediction model in low-latitude regions based on long short-term memory neural network](#)
Tong-Bao Zhang, , Hui-Jian Liang et al.
- [Comparison of different predictive models and their effectiveness in sunspot number prediction](#)
Sayed S R Moustafa and Sara S Khodairy



PRIME
PACIFIC RIM MEETING
ON ELECTROCHEMICAL
AND SOLID STATE SCIENCE

HONOLULU, HI
Oct 6–11, 2024

Abstract submission deadline:
April 12, 2024

Learn more and submit!



Joint Meeting of

The Electrochemical Society
•
The Electrochemical Society of Japan
•
Korea Electrochemical Society

Trajectory Prediction Based on Machine Learning

Lumin Su¹ and Li Li^{2,*}

^{1,2}Institute of Sensing Technology and Business, BUPT (Wuxi), China

^{1,2}State Key Laboratory of Networking and Switching Technology, Beijing Laboratory of Advanced Information Networks, Beijing University of Posts and Telecommunications, Beijing, China

*Corresponding author

Abstract. With the rapid development of the Internet and positioning technology, trajectory prediction has become a popular research direction because of its wide range of application scenarios. However, the trajectory points are represented only by a series of incomprehensible numerical labels in current studies, and the extraction of the stay points are also lack of time consistency. On the other hand, conventional trajectory prediction models do not make full use of the trajectory context information, as well as only predict the position of the next stay points, which limits the prediction accuracy. To fulfill the above problems, the trajectory prediction method based on machine learning is studied. In this paper, a spatial information extraction method based on multistage cluster is proposed, as well as the LSTM model and the bidirectional LSTM model in the deep learning are used to predict the next position. Experimental results show that the novel method proposed can better represent the semantic information of trajectory and improve the prediction performance greatly.

1. Introduction

In recent years, mobile Internet technology has developed rapidly. Intelligent mobile devices have gradually become an indispensable part of our life, generating massive trajectory data on different platforms. As an important aspect of trajectory research, trajectory prediction could be applied to a wide range of location-based application scenarios, such as crowd congestion early warning, network resource allocation and mobility management, conducive to provide users better services.

Trajectory prediction is to predict the next position based on the historical trajectory, which could be mainly divided into two parts, the historical trajectory information processing and the trajectory prediction model construction.

The trajectory information processing mainly deals with the spatial information. At present, the research of spatial information extraction is mostly related to clustering algorithms, in which DBSCAN is the main density clustering method^{[1][2]}. Yu Zheng et al. proposed a method of TBHG, which is to detect some stay points where a user has stayed in a certain distance threshold over a time period and using density-based clustering algorithm to hierarchically cluster this dataset into some geospatial regions^[3]. This method is simple and effective, but it is not suitable for sparse trajectory points, and it involves distance threshold and time threshold, which easily affects the accuracy of the extraction of dwell points.

The model construction of trajectory prediction mainly uses the historical trajectory data of the user to explore the movement pattern and calculate the probability of the next position. There are several main types of trajectory prediction methods. One is based on association rule^[4]. The second method is to use the Markov model^[5]. Another trajectory prediction method is based on neural networks^[6].



This paper studies the trajectory prediction based on GPS data. The main attributes include: Firstly, a novel trajectory information processing method is proposed, in which the trajectory data is transformed into more understandable semantic information through trajectory preprocessing, spatial information extraction, as well as time and direction information addition. Secondly, the feature vectors of the processed trajectory information is extracted and a prediction model is built based on the Long Short Term Memory Networks (LSTM)^[7] model and the bidirectional LSTM (Bi-LSTM)^[8] model in deep learning, in which the stay area points and the move points are included, so as to make full use of the context information and improve the prediction accuracy. Finally, the simulation experiments are carried out using the GPS data set in real life, and the performances of the system with different parameters are analyzed. Experiment results show that the proposed methods in this paper can effectively predict the next location of the trajectory with high performance.

In this paper, the rest of the structure is organized as follows. The proposed trajectory information processing method and the prediction model based on LSTM model and the Bi-LSTM model is presented in Section2. Section3 shows the parameter settings and the experimental results of simulation. Section4 summarizes the whole paper and gives the research direction in the future.

2. Trajectory Prediction Method Based on Spatio-temporal Coherency Extended

The process of the trajectory prediction method proposed in this paper are shown in Figure 1, includes the following steps: trajectory spatial information extraction, trajectory time and direction information addition, feature vector extraction, and trajectory prediction. By the means of spatial information extraction, the sliding window and the region coherency expansion algorithm, the stay points with relative temporal and spatial coherency is extracted and the trajectory data is converted into semantic information, which makes the trajectory easier to understand. After that, the feature vectors of the processed trajectory information are extracted and the prediction model based on the LSTM algorithm and the Bi-LSTM algorithm in deep learning are set up, in which the stay area points and the move points are considered, making full use of the context information and improving the prediction accuracy.

2.1. Trajectory Spatial Information Extraction Method Based on Multistage Clustering

The trajectory data preprocessed is still very large. It is difficult to find the movement rule implied by the trajectory since the granularity is too small. Therefore, trajectory information extraction is needed. The process of it include the following steps. First, the trajectory points satisfying the conditions are extracted at a certain time interval or distance interval. Second, the stay points are extracted based on the spatio-temporal consistency extension. Then, the stay area points are extracted based on the heuristic incremental clustering method. Finally they are merged with the move points and converted to the name of the place in life.

2.1.1. Stay point extraction. The regional consistency extended stay point extraction method base on DBSCAN is looked as the baseline in this paper. The core definitions in it include^[9]:

Definition 1 Area Coherency Weight: Used to determine the regional correlation of two trajectory points. The left half of the formula uses distance to calculate coherency, and the right half uses speed to calculate coherency.

$$coh(i,j) = exp \left(- \left(\frac{distance(p,q)}{\delta} \right) - \left(\frac{distance(p,q)}{duration(p,q)} \right) \right) \quad (1)$$

Definition 2 Directly Coherency-correlative: If the coherency weights $coh(p,q)$ between the trajectory points p and q are greater than a given weight threshold ρ , p points and q points are directly coherency-correlative.

Definition 3 Coherency-correlative: If there is a trajectory sequence $Traj = \{p_1, p_2, \dots, p_n\}$, from point p_i to p_{i+1} is directly coherency-correlative, then points p_i and p_n are coherency-correlative.

Definition 4 Cluster: A stay point cluster Sp is a non-empty subset in the trajectory $Traj$. All the trajectory points in the Sp are coherency-correlative, and the number of trajectory points included is greater than the set threshold $MinNum$.

In the regional consistency extended stay point extraction method base on DBSCAN, clusters points that are close and slower, both of the speed and density are taken into account. It can be known from the cluster definition that any two points in the cluster S_p are coherency-correlative, and then can be extended from one point to another according to the directly coherency correlation, so that the extracted trajectory points have spatial coherency, belong to the same stay area. But there are two shortcomings of this method:

- i) The trajectory points is a sequence with spatio-temporal information. It is necessary to consider not only spatial coherency but also time coherency. The same place represents different meanings at different times, but the stay point extraction method based on the regional coherency extension still aggregates them together.
- ii) The amount of calculation is large. Each time the coherency weight between the current points and other coherency points is obtained, it needs to be read to the end of the trajectory. In the actual situation, when the user leaves the stay area, the calculation of the stay point value should be ended.

Therefore, an improved method is proposed in this paper, which is based on spatiotemporal coherency extension. The sliding window is combined with the region coherency expansion algorithm to make the extracted stay points have spatio-temporal coherency and therefore reduce the overall calculation.

The trajectory is a set of points arranged in time order. When the algorithm traverses each trajectory and finds the trajectory points which is directly coherency-correlative to the trajectory point p , a sliding window of a specified length is added, which could keep the time between the trajectory points relatively coherency. The core steps of the algorithm are shown in Figure 2. p_1, p_2, \dots, p_n compose a trajectory. Take point p_1 as an example. First, add a sliding window. Then look for the trajectory points in the window that are directly coherency-correlative to point p_1 , and mark them as seed points, indicated by the symbol \blacktriangle . Then, when the algorithm traverses the trajectory points in the window, if there is a seed point in the window, the window slides forward one space in turn to compare the point p_1 with the next trajectory point. Finally, when there is no seed point in the sliding window, the calculation stops. The same method is used for the extended seed points to extract the stay points.

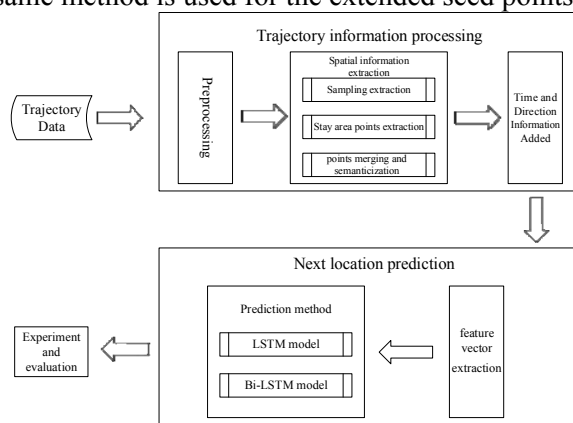


Figure 1. Overall framework.

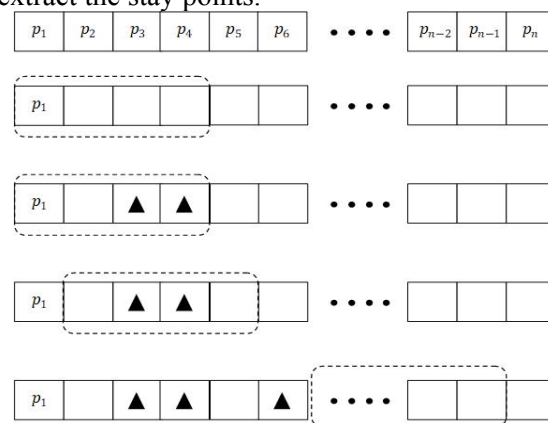


Figure 2. Example of the core steps of stay point extraction.

2.1.2. Stay area points extraction. After the stop point extraction process, a sequence of trajectories with a certain meaning is obtained, but it does not represent the actual scene. Due to the different scope of each activity area, multiple stops will occur. In real life, all the stay area points on the same trajectory are time-disjoint, that is, the time ranges of the two stay area points are disjoint. Therefore, an incremental cluster method is proposed in this paper, which uses the distance threshold and time relationship to extract the stay area points. Compare the time relationship between two stay points. If the relationship is included, merge them together. If not included, the distance between the two points is calculated, and if the distance threshold is satisfied, combine them together. The method is depicted in Algorithm 1.

Algorithm 1: Stay area points extraction based on incremental cluster**Input:** Stay points sequence SP; Distance threshold value D_{max} **Output:** Stay area points sequence SRP

```

1:  $i = 1$  ,  $SP\_num = |SP|$ 
2:  $SrpCluster.add(p_i)$ 
3: for  $i = 1$  to  $SP\_num$  do
4:    $include = TimeInclude(p_i, p_{i+1})$ 
5:    $dist = distance(p_i, p_{i+1})$ 
6:   if  $include == \text{ture}$  or  $dist \geq D_{max}$  then
7:      $SrpCluster.add(p_{i+1})$ 
8:   else  $SrpCluster.coord = CoumputMeanCoord()$ 
9:      $SrpCluster.arrT = MinTime()$ 
10:     $SrpCluster.leaT = MaxTime()$ 
11:     $SRP.add(SrpCluster)$ 
12:     $SrpCluster.clean()$ 
13:     $SrpCluster.add(p_i)$ 
14:   end if
15: end for
16: return SRP

```

2.1.3. Trajectory points merging and semanticization. Many research methods directly take the extracted stay area points as input to build a trajectory prediction model. In real life, people will go to different places through different roads. The movement process often has a certain impact on the trajectory prediction and should not be abandoned. On the other hand, the distance between the stay area points is large, so that the uncertainty is big, which affects the performance of the trajectory prediction model. Therefore, in this paper, the stay area points are compared with the remaining points of the extracted trajectory to merge the trajectory points. If the remaining trajectory points are within the time zone of stay area points, they would be ignored, otherwise they are inserted into the stay area points sequence in chronological order. After the trajectory points are merged, the trajectory sequence still cannot be directly applied to the trajectory prediction model, because the extracted trajectory points, even if they correspond to the same place, have different latitude and longitude and cannot be directly compared. In this paper, the trajectory points are semantically mapped into natural language, and the context environment is introduced. The trajectory data is no longer a series of numerical labels. It can represent the user's active place and move process, which has practical significance, and also brings trajectory prediction and other work convenience.

2.2. Trajectory Time and Direction Information Added

Through the previous processing, a specific spatial information trajectory is obtained, and the stay area points have arrival time and departure time. For the move points, the original time is set as the arrival time, and the time of the next trajectory point is set as the departure time. Calculate the difference between the arrival time and the departure time of each trajectory point and add it to the trajectory data as the time used. The trajectory of the user is a dynamic change, and the direction indicates the trend of the trajectory. The same historical trajectory, going east or west, lead to the completely opposite next position. Therefore, according to the latitude and longitude calculation of the azimuth angle between the trajectory points, the motion direction information of the trajectory as well as the relationship between the trajectory points can be obtained.

2.3. Feature Vector Extraction

A trajectory is a set of points with continuity, and the front and back positions are related, similar to the context in natural language. This paper adopts the popular Word2Vec coding model in natural language processing^[10], and transforms the semantic trajectory points into word vectors to realize word embedding. The time of day is divided into several time periods according to the given time interval.

The arrival time and the leave time of each trajectory point are mapped into corresponding time segments. At the same time, the used time is normalized and added as a one-dimensional vector to the feature sequence. The direction information is similarly processed. It is divided into 8 directions by the difference of 45 degree. The azimuth is mapped into the corresponding direction to obtain a discrete sequence of directional features.

The trajectory is divided based on the sliding window. Set a fixed length sliding window at the beginning of the trajectory. The window slides forward one position in turn, then set the last trajectory point in the window as the label of the next position, and the remaining trajectory points training features set.

2.4. Trajectory Prediction Method Based on Deep Learning

2.4.1. Trajectory prediction based on LSTM model. Trajectory prediction is mainly based on LSTM model. By constructing a multi-layer network structure, the relationship between the input trajectory training features and the label of the next position is learned, and the future possible arrival position is predicted. The LSTM is used to keep the closer trajectory point information and forget the far trajectory point information through the gating unit and the linear connection. The storage state of the trajectory information is controlled by the input gate, the forget gate and the output gate.

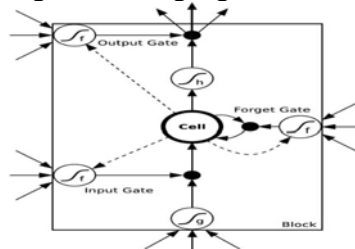


Figure 3. LSTM model.

2.4.2. Trajectory prediction based on bidirectional LSTM model. The current position of the user is not only related to the historical trajectory sequence, but also has a context relationship with the future trajectory sequence. Therefore, the bidirectional LSTM(Bi-LSTM) model is also used for trajectory prediction. The Bi-LSTM trains two instead of one LSTM on the input feature vector sequence. In the forward calculation, the output value of the current hidden layer is related to the previous position. In the reverse calculation, the output value of the current hidden layer is related to the latter position, and the final output depends on the weighted sum of the forward and reverse calculations. Therefore, the Bi-LSTM can fully learn the context information of the trajectory and obtain a better vector sequence, thereby improving the performance of the trajectory prediction.

3. Experiments

3.1. Experimental Settings

In the simulation, a GPS trajectory dataset collected by the Microsoft Asia Research Institute Geolife^[3] is used. The dataset contains historical trajectory information for 182 users from April 2007 to August 2012, with a total of 17,621 trajectories. Each trajectory consists of a series of GPS points, each of which contains information such as latitude, longitude, altitude, date and time of collection. 70% of each user's data is taken as the train set and 30% as the test set, 10% randomly extracted from each training set as the validation set.

3.2. Experimental Results

In this paper, LSTM model and Bi-LSTM model are used for trajectory prediction. The performances of the two models with different parameters are simulated. To verify the validity of the LSTM model and the Bi-LSTM model, the next location of the trajectory predicted by ANN in the same data set was used for performance comparison. ANN is a basic single layer network structure.

3.2.1. Epoch. Epoch is training rounds, which refers to the period in which the model needs to learn the data training. Figure 4 shows the results of the prediction accuracy in different Epoch during the learning of the LSTM model and the Bi-LSTM model. The experiments were conducted on the training set.

As can be seen from Figure 4, the accuracy of the model increases with the increase of the epoch. The LSTM model and the Bi-LSTM model are gradually stable after Epoch is 60 and 100, respectively. To prevent overfitting and experimental comparisons, Epoch was chosen to be 100 for the other experiment.

3.2.2. Batch_Size, num_units. Batch_Size is the gradient update block size, the number of samples used for a single training. Num_units is the number of neurons in the hidden layer and is also the output dimension of the LSTM layer. The experiments were conducted on the test set. Figure 5 and Figure 6 show the results of the prediction accuracy and train time of different models on the test set in the different Batch_Size. Figure 7 show the prediction results for different models with different num_units. It can be observed from these figures that the prediction effect of the Bi-LSTM model is generally better than the LSTM model. For different parameters, as the num_units increases, the models effect is improved. As the Batch_Size increases, the number of samples per training increases. Although the model training time decreases, the model prediction accuracy has dropped slightly. After continuous debugging parameters, the Bi-LSTM model with the Epoch is 100, the Batch_Size is 32, and the num_units is 64 has a relatively high prediction effect, and the accuracy rate reaches 87.43%. The prediction effects of the LSTM model and the Bi-LSTM model are all better than those of the ANN model.

3.2.3. The feature of time and direction. The trajectory prediction method in this paper not only considers the spatial features, but also adds other features such as time and direction. The LSTM model with shorter training time is selected, and the basic parameters of Epoch is 100, Batch_Size is 128, and num_units is 32. The experiments were conducted on the test set and the result is shown in Figure 8. As the feature increases, the prediction accuracy of the model is gradually improved, and the validity of the time and direction features is verified.

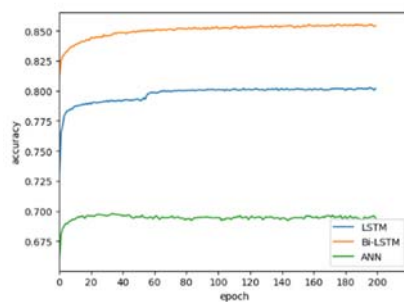


Figure 4. Results of different models in different epoch.

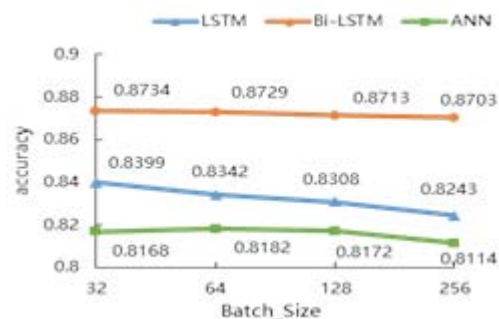


Figure 5. Results of different models in different Batch_Size.

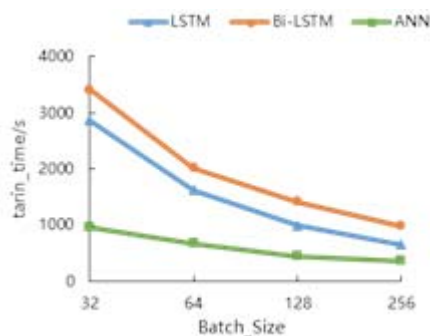


Figure 6. Results of different models in different Batch_Size.

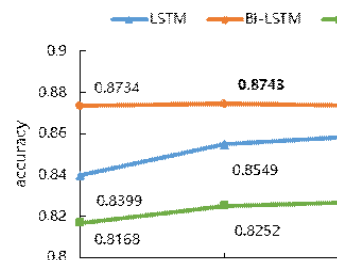


Figure 7. Results of different models in different num_units.

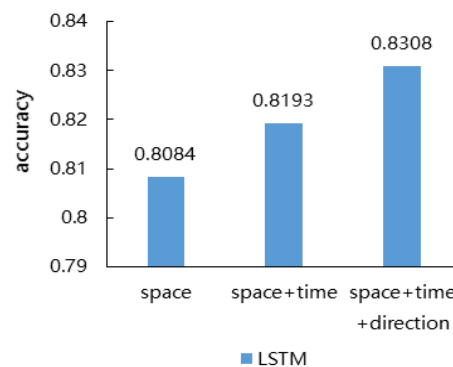


Figure 8. Results of LSTM model in different features.

4. Conclusion and Future Works

Trajectory prediction is an important part of trajectory research and has a wide range of application scenarios. In this paper, the trajectory prediction method is studied, and the GPS trajectory data is used to predict the user's next position. The method of spatial information extraction method based on multistage cluster is proposed, and trajectory prediction is realized by combining LSTM model in deep learning and Bi-LSTM model. The experimental results show that the research results in this paper improve the effect of trajectory prediction. Using the Bi-LSTM model, the prediction accuracy rate is 87.43%. Adding different features helps the model to better understand the trajectory and improve performance.

This paper basically achieves the goal of trajectory prediction, and future research work can be carried out from multiple angles. The current method also requires manual input of threshold parameters, and the method of training without inputting threshold parameters needs further study. By constructing the model to predict the next position of the trajectory, the problem of where the user will go is solved, and further consideration can be given to when the user reaches the position and predict the specific time when the user reaches the next position.

Acknowledgments

The research work is supported by the project of National Science and Technology Major Project of the Ministry of Science and Technology of China (No: 2018ZX03001014-003).

References

- [1] Birant D, Kut A . ST-DBSCAN: An algorithm for clustering spatial-temporal data[J]. Data & Knowledge Engineering, 2007, 60(1):208-221.
- [2] Wen L. Research on the position prediction method of moving object under dynamic relation [D]. China university of mining and technology, 2017.
- [3] Zheng Y, Xie X, Ma W Y. Geolife: A collaborative social networking service among user, location and trajectory[J]. IEEE Data Eng. Bull., 2010, 33(2): 32-39.
- [4] Morzy M. Mining Frequent Trajectories of Moving Objects for Location Prediction[C]// International Conference on Machine Learning and Data Mining in Pattern Recognition. Springer-Verlag, 2007:667-680.
- [5] Qiao S, Shen D, Wang X, et al. A Self-Adaptive Parameter Selection Trajectory Prediction Approach via Hidden Markov Models[J]. IEEE Transactions on Intelligent Transportation Systems, 2015,16(1):284-296.
- [6] Leca C L, Nicolaescu I, Rîncu C I. Significant location detection & prediction in cellular networks using artificial neural networks[J]. Computer Science and Information Technology, 2015, 3(3): 81-89.
- [7] Hochreiter S , Schmidhuber, Jürgen. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8):1735-1780.

- [8] Graves A , Jürgen Schmidhuber. Framewise phoneme classification with bidirectional LSTM and other neural network architectures[J]. Neural Netw, 2005, 18(5):602-61
- [9] Yang J, Xu J, Xu M, et al. Predicting next location using a variable order Markov model[C]// ACM Sigspatial International Workshop on Geostreaming. ACM, 2014:37-42.
- [10] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.