Multimodal Traffic Travel Time Prediction

Shizhen Fan
College of computer science and technology
Qingdao University
Qingdao, China
shizhenfan@126.com

Jianbo Li*
College of computer science and technology
Qingdao University
Qingdao, China
lijianbo@qdu.edu.cn

Zhiqiang Lv
College of computer science and technology
Qingdao University
Qingdao, China
lvzq7614@163.com

Aite Zhao
College of computer science and technology
Qingdao University
Qingdao, China
zhaoaite@qdu.edu.cn

***Abstract*—With the continuous growth of urban population, it is urgent for people to accurately plan the travel time. Therefore, travel time prediction of urban areas has become a key research direction in the field of smart cities. At present, several studies on travel time prediction are only conducted on a single mode, where the prediction process only treats a certain vehicle as an isolated traffic state on the route. However, the factors affecting traffic are extremely complex, thus making it very difficult to produce a comprehensive forecast. Based on this situation, the mixed existing model and mutual influence of multiple modes of transportation in the city are fully considered, and a multimodal deep learning model namely MC-GRU (Multimodal Convoluted Gated Recurrent Unit Network) is proposed. At the same time, to solve the problem of some objective factors, such as departure time and travel distance, we propose an attribute module to deal with these implicit factors. In addition, to explore the interaction between different modes of vehicles, a feature fusion module for obtaining the interaction effect between different modes of vehicles is proposed. Finally, we use GRU to learn the long-term dependence. MC-GRU can realize the accurate prediction of travel time in multimodal traffic state, as well as implement travel time prediction for three types of travel modes. The experimental results show that MC-GRU achieves higher prediction accuracy on a challenging real world dataset as compared with MAE, MAPE and RMSE.**

Keywords—multimodal, attribute module, feature fusion module, GRU

I. INTRODUCTION

Traffic prediction is one of the important research directions of intelligent city. Its main research value is to guide urban construction management and help people make travel plans. The traffic flow forecasting often plays a vital role in guiding urban construction. However, in guiding people's travel planning, the most concerned is the prediction of travel time. A more accurate time prediction model needs to consider a variety of factors [1], including not only external factors, spatio-temporal characteristics and other factors, but also the interaction between vehicles.

Because the travel time has obvious periodicity [2], for example, under normal circumstances, the road conditions of the city during working days are morning and evening peaks, and in other time it is relatively flat. Sometimes we can get the travel time between the two places with the help of self-experience. However, there are two problems difficult for us to solve. First, it is often difficult for us to consider the influence of hidden factors, which will make it difficult for us to estimate travel time under certain conditions, there are several hidden factors including the travel mode, the departure time, the travel distance, the weather and the social emergencies, etc. In fact, the most critical and basic of these factors are the travel mode, the departure time, and the travel distance [3][4][5]. These points determine the lower limit of travel time. Second, it is difficult for us to estimate the traffic conditions in unfamiliar areas. In order to solve the mentioned problems, researchers use machine learning methods such as Support Vector Machines (SVM) [6], K-means clustering [7] and Decision Tree to divide travel attributes to improve the prediction accuracy. The calculation time of this method is usually fast, but it also has the problem of insufficient calculation accuracy. Because these machine learning methods only operate on some objective attributes similar to classification, it is difficult to solve the potential impact factors encountered in the journey, and can not capture the spatial and temporal characteristics, let alone find out the complex nonlinear relationship.

Deep learning has proved to be an effective method to solve complex nonlinear relationships, which is attributed to the good performance of convolution and pool operations in the extraction of spatio-temporal features. The key of time prediction using deep learning is to extract spatiotemporal features from historical vehicle trajectories. On the one hand, any historical trajectory of a vehicle that includes GPS points can well reflect the driving range of the vehicle. For example, buses usually run between the commercial and residential areas of the city; taxis drive more frequently in the commercial areas, transportation hubs and urban expressways; bicycles and electric vehicles often travel around residential areas; and pedestrian paths are often short. On the other hand, the temporal and spatial changes of a traffic trajectory will reflect the characteristics of the traffic condition to a certain extent. For example, the GPS variation characteristics and time variation characteristics of the first half track have important reference value for the prediction of the second half trajectory, which provides a basis for predicting travel time based on historical trajectories. Time prediction based on historical trajectory is further divided into two methods. One is segmented processing and then comprehensively calculates the total time [8] [9]. The advantage of segmented processing is that the travel time prediction for

* Corresponding Author

2021 International Joint Conference on Neural Networks (IJCNN) | 978-1-6654-3900-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/IJCNN52387.2021.9533356

978-0-7381-3366-9/21/$31.00 ©2021 IEEE

sub-segments will be accurate, but the relationship between total time and sub-time is difficult to express, segmentation errors continue to accumulate. Another processing method is to process each trajectory as a whole [10], so that the correlation between each trajectory point will be obtained; but in the processing of long trajectories, the correlation between the front and back parts of the trajectory will be gradually weaken [3]. At present, there are also some studies combining traditional machine learning methods with deep learning methods [11]; for example, capturing urban traffic topological spatial features from a complex network perspective, and then designing deep learning models to obtain temporal features, so that the spatial and temporal features are fused for the accurate time prediction.

The above methods only consider the impact of temporal and spatial features on travel time under a single traffic mode, which extract a large number of features affecting travel time, and then utilize deep neural networks to learn experience for time prediction. However, we have found that with the continuous improvement of urban roads construction, people's travel modes and transportation modes [1] are constantly diversified, having more temporal and spatial features. The characteristics of different modes of transportation also have a certain temporal and spatial correlation. In fact, there is no single mode of transportation in a city. Different modes of transportation usually share urban roads.

To tackle the above challenges, it is valuable to study the time correlation between different modes of transportation. This paper we propose a novel deep learning model: Multimodal Convoluted Gated Recurrent Unit Network (MC-GRU). This paper mainly consists of five parts: introduction, related work, methodology, experiment and conclusion. In the third chapter of the methodology part, MC-GRU model is introduced in detail, including multimodal convolution module, feature fusion module, implicit feature processing module and other key points; in the fourth chapter of the experiment part, some advanced machine learning and deep learning model comparison experiments are listed. The main contributions of this paper are summarized as follows:

- A novel convolution method is proposed, named Multimodal Convolution Module, which convolves the historical trajectories of different vehicles respectively for spatio-temporal feature extraction of different vehicles.
- We propose Feature Fusion Mechanism fused these multimodal features to obtain the influence factors between different traffic modes.
- We design an Implicit Feature Processing Module to deal with several significant implicit factors. Due to the various ways of expressing these factors, the attribute processing module represents different factors numerically and standardizes them. Then obtain a unified trajectory feature.

[1] People's travel mode refers to different travel options, including walking, cycling, riding, etc., while the transportation means that people travel by physical means such as bus, taxi and subway.

II. REALTED WORK

In recent years, multimodal feature learning has made great progress in various fields, especially in audio-visual integration, emotional analysis, etc. In this section, we mainly discuss the deep learning methods of traffic prediction, and then further discuss the current multimodal deep learning methods in the field of transportation.

Several studies propose to use LSTM to extract the temporal and spatial correlation of vehicles to predict the historical vehicle trajectory in combination with weather, date, and driver number [12]. Chen et al. proposed a prediction algorithm that can denoise and handle long-term dependence of the historical vehicle trajectory; simultaneously, the prediction effect of the hybrid model and the prediction accuracy were significantly enhanced [13]. Yu et al. proposed a short-term traffic flow prediction model based on GRU-RNN using historical traffic flow data.[14]. Jin et al. constructed a two parameters convolution theoretical model based on density partition using taxi passenger and trajectory data and vehicle speed [15]. In a fixed time period, the trajectory data is deeply analyzed by considering whether the taxi is carrying passengers or not, combined with the vehicle speed construct a two-parameter convolution theoretical model based on density partition, and use the obtained final velocity value to calculate the transit time. However, most of the above methods only analyze one vehicle or a single mode. Considering the complexity of the traffic scene, multimodal learning will be more conducive to traffic prediction.

In the field of multimodal learning of traffic prediction, Dong et al. divided the traffic state into six levels (categories) by road service level, and learned the corresponding relationship between different modes and traffic flow, forecasted the traffic flow [16]. Wang proposed a method employing machine vision and multimodal feature fusion, which focused on the driver fatigue detection in actual driving environment [17]. Li et al. used the multimodal trajectory data of users in Shenzhen, including mobile phone trajectory data, traffic card trajectory data, taxi trajectory data and bus trajectory data, to establish the connection between blocks for physical space perception [18]. However, the interaction between different modes of traffic prediction has not been considered. Zhang designed a travel information maximization generation countermeasure network by modeling the joint distribution of travel time of two continuous links for travel time estimation [19]. Adetiloye T. et al. proposed a big data fusion framework based on homogenous and heterogeneous data for traffic congestion prediction. [20]. Ma proposed a new road segment-based bus travel time prediction method combining two different influencing factors of bus and taxi to establish traffic impact models, which enhance the prediction ability of bus travel time [21].

III. METHODOLOGY

In this section, we describe our multimodal deep learning model MC-GRU in detail. The overall structure is shown in Fig. 1.

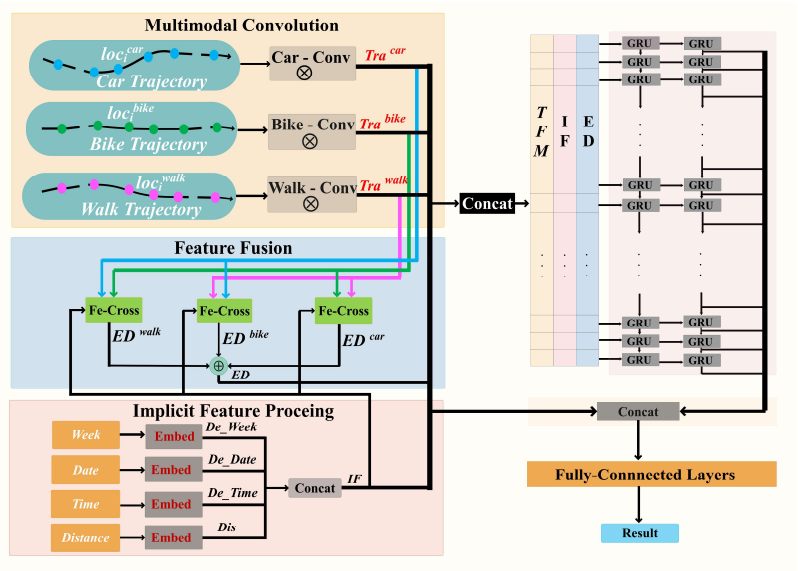


Fig.1. The model structure of MC-GRU

$$mean^{type} = \frac{\sum_{i=1}^{T-1} dist\ (loc_i,\ loc_{i+1})}{T-1} \quad (1)$$

MC-GRU mainly includes three parts: the multimodal convolution module, the implicit feature processing module and the feature fusion module. The multimodal convolution module is designed to extract spatio-temporal features based on the historical trajectories of different vehicles. The implicit feature processing module processes several key implicit factors and basic information of the itinerary, such as travel time, travel distance, and travel date. The feature fusion module merges the spatio-temporal features obtained by the multimodal convolution module, computing the effect degree of other traffic modes. Then these features are fed into the GRU recurrent neural network to learn temporal dependence. Finally output the prediction result by the fully connected layers.

A. Multimodal Convolution Module

In this section, we divide travel modes into three types: car, bicycle and walking. These three travel modes are also the most common travel modes and have the most predictive value. Although most cities consider subways as the efficient means of public transportation, they have a relatively fixed operating time and generally running on time and run below the road or have separate bridge tracks, therefore, the ground traffic is less affected by the subway. In addition, buses are also a common way for traveling, the data on public transportation is generally sparse, and it is difficult for us to obtain relevant data on urban buses. Therefore, in this article we only consider three transportation modes: car, bike, and walking.

After analyzing the trajectories of the three vehicles, under the same sampling interval, the trajectory sparsity is quite different as shown in Fig. 2. We use (1) to calculate the average distance between two points at the same time interval. In the (1), loc represents the i-th sampling point; T represents the length of the trajectory. The result shows that $mean^{car} : mean^{bike} : mean^{walk} \approx 40:8:2 = 20:4:1$. This provides a basis for us to extract spatial features by convolution of different patterns.

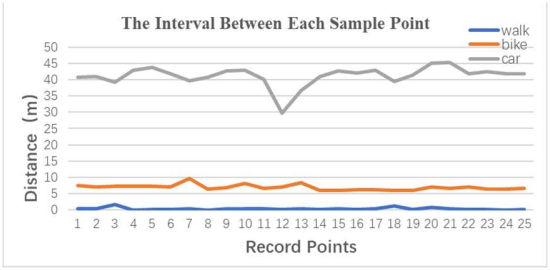


Fig.2. Sparse difference of track points

At the same time, we depict different trajectories in the same picture with diverse colors, which show that there are more or less intersections between various trajectories in Fig.3. This crisscross feature provides a basis for us to consider the mutual influence factors between different trajectories.

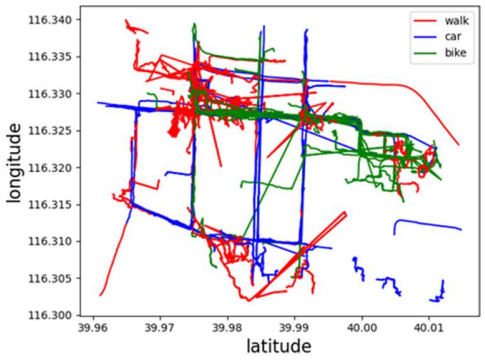


Fig.3. Intersection of different vehicles

For each kind of trajectory, in order to fully obtain its spatial features, we first merge longitude and latitude with (2), and then map the trajectory to a larger width through a nonlinear

transformation [3]. The structure of multimodal convolution as shown in the Fig.4.

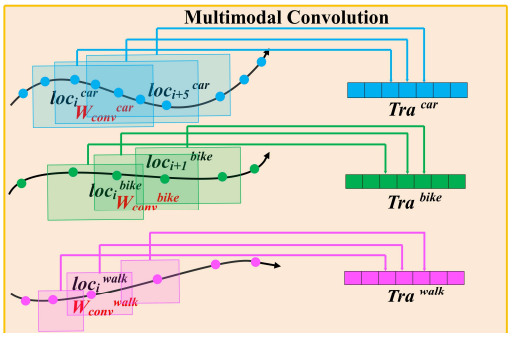


Fig.4. The structure of Multimodal Convolution

$$\begin{cases} loc_i^{car} = f(W_{loc} \cdot [p_i^{car}.lat \circ p_i^{car}.lng]) \\ loc_i^{bike} = f(W_{loc} \cdot [p_i^{bike}.lat \circ p_i^{bike}.lng]) \\ loc_i^{walk} = f(W_{loc} \cdot [p_i^{walk}.lat \circ p_i^{walk}.lng]) \end{cases} \quad (2)$$

In these Equation, ○ represents a merge operation, f is a nonlinear mapping function, W_{loc} stands for learning weight matrix, p_i represents the track point of the trajectory; *lat* and *lng* represent latitude and longitude respectively. Here, we map the width to 16 dimensions, Therefore, our trajectory width has changed from $R^{2\times|T|}$ to $R^{16\times|T|}$, where $|T|$ is the length of the track (i.e. the number of points contained in each track). After expanding the dimensions, the spatial features are extracted by multimodal convolution which is shown in (3):

$$\begin{cases} Tra^{car} = \sigma_{cnn}\ (W_{conv}^{car} * loc_{i:\,i+k-1}^{car} + b^{car}\ ) \\ Tra^{bike} = \sigma_{cnn}\ (W_{conv}^{bike} * loc_{i:\,i+l-1}^{bike} + b^{bike}\ ) \\ Tra^{walk} = \sigma_{cnn}\ (W_{conv}^{walk} * loc_{i:\,i+m-1}^{walk} + b^{walk}\ ) \end{cases} \quad (3)$$

In these Equations, Tra^{car} represents the convolution results of car track, or called Trajectory Feature Map (*TFM*); σ_{cnn} means the activation function (we share one activation function in three modes), W_{conv} is the weight matrix, we call it convolution kernel here, b is the offset value, and $loc_{i:\,i+k-1}$ is a segment of the trajectory, where k, l and m is the length of the convolution kernel. After the convolution operation, the characteristic size of the trajectory is changed to $R^{16\times(|T|-k+1)}$.

B. Implicit Feature Processing Module

In this paper, we consider several key travel information and important attribute features, which will greatly affect the accuracy of travel time prediction. For example, the time taken for the same journey will be greatly different from the departure time 10 am to 6 pm, and the travel distance further affects the travel time based on the departure time. Moreover, several implicit attributes of travel will also have an important impact on travel time. For instance, the date of the trip and whether the trip is on a working day play a significant role in travel time prediction.

Whether it is the travel information or the attribute features, their original data representation is very different. The original values of these implicit features are classified values, which makes them unable to be trained directly in neural networks. Firstly, the embedded method is used to map the travel information and hidden features to the numerical vector. In this paper, we use a nonlinear mapping method to map each data into a vector consistent with the length of the trajectory. We transform the initial classification value from $v \in R^d$ to $v \in R^{D\times1}$. Here we calculate the mapping matrix $W \in R$. d represents the dimension of the original classification value data, and D represents the dimension of the mapping space. *Time* is the time label of the first point; $Distance = \sum_{i=1}^{T-1} |p_i, p_{i+1}|$; *Week* and *Date* are recorded as the value 1-7 and 1-30, respectively. The mapping method can not only convert the classification value into a deeply learned numerical value, but also can be embedded in the similar position [22].

$$\begin{cases} De\_Time = Time * W^{time} \\ Dis = Distance * W^{distance} \\ De\_Week = Week * W^{week} \\ De\_Date = Date * W^{date} \end{cases} \quad (4)$$

$$IF = De\_Time \circ Dis \circ De\_Week \circ De\_Date \quad (5)$$

In the (4), * represents matrix multiplication, ○ in the (5) denotes merge operation. *Time* means the departure time of the trajectory, which can be represented by $p_1.time$, *De_time* represents the data after being embedded, *Dis*, *De_Week* and *De_Date* are processed as the same as *Time*. In (5), we combine the above elements to get implicit features (*IF*).

After the implicit feature module, the implicit feature flow into different directions. On the one hand, the feature information and spatial information obtained by multimodal convolution module are combined into recurrent neural network to learn and extract temporal features. On the other hand, these implicit features input Feature Fusion Module, we use the implicit features of the current traffic mode to extract the trajectory feature map of other modes, and then obtain the influence degree.

C. Feature Fusion Module

By the Multimodal Convolution Module, we obtain the spatial information of different vehicles. Further, as shown in Fig. 3, we can see the complex intersection between different traffic trajectories. In other words, there are some spatio-temporal correlations between different vehicles.

In this paper, a Feature Fusion Module is proposed to capture the relationship between trajectory space features, as shown in the Fig.6. The travel time of the current mode will be affected by other means of transportation. We define the influencing factors of other modes as the current mode uses own implicit feature, such as time and date, to obtain the characteristics of other transportation modes. Therefore, the impact of various modes of transport is defined as Equation group (6),which named Feature Cross (Fe-Cross):

$$\begin{cases} ED = IF \otimes Tra \oplus IF \otimes Tra \\ ED^{bike} = IF^{bike} \otimes Tra^{walk} \oplus IF^{bike} \otimes Tra^{car} \\ ED^{car} = IF^{car} \otimes Tra^{bike} \oplus IF^{car} \otimes Tra^{walk} \end{cases} \quad (6)$$

$$\otimes \qquad \otimes \qquad \otimes \quad (7$$

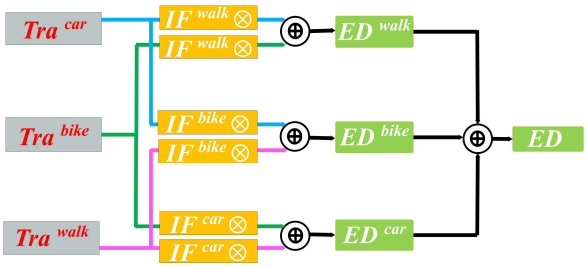


Fig.5. The structure of Feature Fusion

Among Equation group (6), $ED^{walk}, ED^{bike}, ED^{car}$ are three effect degree, which indicates the impact of other vehicles on the current trip; *IF* is the implicit feature corresponding to the current travel mode, which we call the influence factor here; Tra^{bike} represents the bicycle track feature map extracted by multimodal convolution; Tra^{walk} represents the pedestrian trajectory feature map extracted by multimodal convolution; and Tra^{car} represents the vehicle feature map extracted by multimodal convolution. ⊕ means XOR operation; means convolution. We use the current mode's implicit feature as the kernel, to convolute the feature map calculated by multimodal convolution module. In this way, we can see the influence of other trajectories on the current trajectory. Finally, in the (7), we merge the different *ED* of other modes, to get the effecting degree to the current mode; *IF* represents the implicit feature, which must be one of the IF^{walk}, IF^{bike} and IF^{car}.

After the feature fusion mechanism, we combine the effect degree, implicit features with the historical trajectory feature to form the input value of recurrent neural network.

Next, we have all the basis to obtain the spatial dependence. The recurrent neural network can effectively obtain the time dependence in the sequence and update the current state information based on the new input information and the previously obtained hidden state.

The recurrent neural network GRU has the advantage of obtaining short-term and long-term memory features efficiently with shorter training time. Using GRU to deal with the spatio-temporal characteristics has good prediction results [23] [24]. Hence GRU is used to process the spatio-temporal feature output by feature fusion mechanism to achieve the purpose of prediction. The renewal formula of the recurrent neural network can be expressed as follows:

$$h = \sigma\left(W \cdot loc + W_h \cdot h + W \cdot attr + W \cdot Ef\right) \quad (8)$$

After extracting the temporal features from the recurrent neural network, we obtain the complete spatio-temporal features. Finally, we design a fully connection layer and integrate the travel information features to get the prediction of the travel time.

IV. EXPERIMENT

In this section, we test our model on a real data set and evaluate the accuracy of the indicators, including MAE, MAPE and RMSE. The following experiments include the prediction of MC-GRU and some comparative experiments.

A. Experimental Setup

1) Data Set

The experimental dataset 1 is the real traffic data of Beijing. The original data set contains three types: travel mode, GPS location sequence and time series. Among them, GPS sequence represents the travel path. In order to obtain the travel information and hidden features mentioned above, the original data set is preprocessed. We calculate the day of the week by the date of the original data, and use the distance between two adjacent points of GPS to get the travel distance, and the departure time is directly expressed by the first record of the track.

In addition, there are a few tracks with large span and a few tracks with very short span in the original data set. We sampled these tracks and viewed them in the actual map, and found that there were two destinations in these tracks loc_0^{type} and loc (*type* represents one of the three types: walk, bike or car), which were not residential or commercial areas that might exist in the real world, but two meaningless locations. Therefore, short trajectories were considered to be caused by incomplete sampling of data sets , and we exclude these trajectories. For the track with a large span, part of the track was beyond the scope of the city, because the proposed model was used to predict the travel time of the city, so this part of track segmentation was only retained in the city. Then the effective trajectories can be expressed as follows:

$$Tra^{type} = \left\{loc_0^{type} \sim loc_k^{type} \middle| 4 < k < 125\right\} \quad (9)$$

k is the serial number of GPS points.

Because it is difficult to find a suitable multi-modal dataset, according to the characteristics of dataset 1, experimental dataset 2 is an improved data set based on the actual data set of Chengdu taxi. The specific measures are as follows: we divide the data set into three subsets; regard subset 1 as the walking data set; regard subset 2 as the single vehicle data set; and regard subset 3 as the vehicle data set without processing. According to Fig.2, we randomly shorten the distance interval of the first track of subset by 20 times, and divide the original length by the coefficient 20 α (α is a random number, $0 < \alpha < 1$); we randomly shorten the distance interval of the second track of subset by 4 times, and divide the original length by the coefficient 4 β (β is a random number, $0 < \beta < 1$).

We use 90% of the data set as the training set and the remaining 10% as the test set.

2) Parameter Settings:

In the light of the relationship of the average distance between the two adjacent points of the modal trajectory, we test our model under different values of *k*, setting the convolution kernel size *k* as $k = 5, k = 2, k = 1$ respectively. After repeated experiments, the above settings can achieve the best prediction. We set up 32 convolution kernels respectively. Considering the *TFM* will be used to calculate the effect degree, we use ELU as activation functions [25], which can better maintain robustness. The ELU functions are defined as follows:

$$ELU(x) = \begin{cases} e^x - 1, & x \le 0 \\ x, & x > 0 \end{cases} \quad (10)$$

In the attribute module, we map the departure time to R^{16} and the day of the week in which the trip is located to R^3

In the recurrent neural network, we set the hidden layer neurons as 128, the hidden layer activation function as tanh x, and the tanh x is defined as:

$$\tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (11)$$

According to many different experiments, we set the learning rate to 0.001; batch size is set to 64 and epoch is set to 400.

We train the model with Python 3.7, a popular Python library. Our code is deployed on the CPU with NVIDIA GeForce RTX 2080Ti, 8-core i7.

3) Evaluation Index:

We used MAE, MAPE and RMSE to evaluate the prediction effect of the model:

$$MAE = \sum_{i=1}^{n} \left| \frac{(\hat{T}_i - T_i)}{n} \right| \quad (12)$$

$$MAPE = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{(\hat{T}_i - T_i)}{T_i} \right| \quad (13)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\hat{T}_i - T_i)^2} \quad (14)$$

In the (12) to (14), n means the number of the trajectory, $\hat{T}_i$ means the prediction, T_i means true value.

B. Performance Comparison

In order to evaluate the accuracy of MC-GRU for predictio n, we conducted comparative experiments on the proposed model. In this paper, we compare our model with other state-of-the-art algorithms, as well as compare the performance of the split components: multimodal convolution module, implicit feature processing module and feature fusion mechanism in our model.

a) Baseline Comparison: There are six models compared in time series and spatial feature extraction in traffic prediction.

SVM

SVM has strong generalization ability and can guarantee the global minimum of given training data [26]. We use the nonlinear SVM algorithm and radial basis function (RBF) as the kernel function for prediction.

DECISION TREE

Gradient Boosting Decision Tree (GBDT) is an iterative decision tree algorithm and an algorithm with strong generalization ability. In this paper, we have established the GBDT prediction model of different modes to carry out classification prediction as a comparative experiment, in which the depth of tree is set to 6.

LSTM

As a classical model for time series processing, LSTM has strong temporal modeling ability [12]. In this paper, the discriminative mode is selected for convolution operation, so that directly feed the feature map obtained by multimodal convolution into LSTM for predicting the travel time.

CONV-NET

According to the time prediction model proposed in the paper [27], the model without recurrent neural network has the advantage of fast convergence. Thus, we do not use any recurrent neural network unit in CONV-Net. In this way, the extracted feature information will obtain spatial features and incomplete time series features for travel time prediction.

ENCODER-DECODER

It is an end-to-end learning algorithm. This model can deal with the variable length sequence efficiently, and has excellent performance in dimensionality reduction of high-dimensional data [28]. In this paper, we also compare this baseline based on LSTM. First, the time sequence feature map is encoded and decoded by LSTM. The final output of decoding is to obtain the spatio-temporal features, and then the travel time prediction results are output through the full connection layer.

DEEPTTE

DeepTTE [3] is an end-to-end deep learning travel time estimation framework, which can directly estimate the travel time of the whole path. In this model, several components are designed to extract spatio-temporal features, and estimate interval and global time, etc. At the same time, this model also takes many implicit factors into account, which is an effective method for travel time prediction.

TABLE I. PERFORMANCE COMPARISON ON DATASET 1

Method	Walk			Bike			Car		
	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE
SVM	12.88	9.29	12.52	13.91	10.88	8.57	8.13	13.42	10.33
GDBT	13.44	8.26	9.29	19.98	10.43	9.32	9.65	17.47	23.73
LSTM	5.17	4.70	3.81	8.83	11.52	9.51	4.28	13.84	7.85
CONV-Net	15.94	5.13	11.61	17.41	7.92	13.62	12.85	16.58	20.08
Encoder-Decoder	5.13	4.39	3.97	7.03	7.91	5.59	4.59	12.86	8.40
DeepTTE	5.24	6.37	4.76	10.00	9.68	8.54	3.42	11.10	6.40
MC-GRU	5.10	5.13	3.73	5.80	7.92	4.54	4.01	10.58	6.11

As demonstrated in table I, we can see that our proposed model MC-GRU has better prediction effect than other models. Under the three indicators, the prediction results of each model have been improved. The three indicators in the walking data set reached 5.10%, 5.13 and 3.73 respectively; in the biking data set reached 5.80%, 7.92 and 4.54 respectively; in the busing data set reached 4.01%, 10.58 and 6.11 respectively, which are much better than baseline methods. Because we process the data sets of different modes separately, and in the model, the parameters of different modes are initialized for training respectively, so the results are stable results based on probability and statistics.

By contrast, machine learning method does not perform well in multimodal prediction, especially in car data set. This shows that SVM and other machine learning methods are difficult to establish a stable nonlinear relationship between different modes. LSTM model also has good effect in predicting walking travel time, whose error rates are only 5.17%, 4.7 and 3.81. But LSTM has some shortcomings in the prediction of bike and car time. The RMSE is too large and the degree of dispersion is high,

which is caused by insufficient extraction of spatial features. Without the extraction of spatial features by convolution, it is difficult to accurately predict the travel time which is obviously affected by spatial factors .

From the prediction results of CONV-Net, we can see that in the MAPE index, prediction results of the three modes are quite different from MC-GRU; in fact, convolution can obtain the spatial information of a single track, and hard to obtain the temporal information of trajectories and the interaction between different traffic modes. Therefore, the three indicators in the car data set reached 12.85%, 16.58 and 20.08 respectively.

As for the prediction results of the Encoder-Decoder model, just like the LSTM model, the prediction result of the walking time is better than bicycle and car; because the encoder-decoder model can't effectively learn the spatial features, it can only use the limited spatio-temporal features extracted by the cyclic neural network in the encoding process for prediction, and the bicycle and car are affected by the spatial characteristics more obviously. The DeepTTE model is proposed for the prediction of vehicle travel time, and it also shows excellent performance; however, the performance of travel time prediction under multimodal conditions is insufficient. In the walking and biking time prediction, the error rate of DeepTTE reaches 5.24%, 6.37, 4.76 and, 10.00%, 9.68, 8.54 respectively. It can not flexibly cope with the spatial characteristics of different travel modes, nor can it take full advantage of the interaction between different vehicles. But it achieves high prediction rate in the car set, the indicators almost equal to MC-GRU.

TABLE II. PERFORMANCE COMPARISON ON DATASET 2

Method	Walk			Bike			Car		
	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE
SVM	11.42	8.66	10.70	14.01	11.17	8.56	8.29	13.19	10.54
GDBT	9.14	7.45	12.37	18.82	10.62	18.89	8.63	16.75	18.30
LSTM	5.86	4.69	3.84	8.34	11.06	9.44	4.22	12.93	6.76
CONV-Net	13.25	15.93	11.61	13.13	14.88	13.54	11.05	14.82	16.17
Encoder-Decoder	5.71	4.68	3.64	6.84	6.97	5.70	4.27	10.78	6.09
DeepTTE	5.75	5.99	4.43	5.65	9.82	7.26	3.21	9.38	5.41
MC-GRU	5.03	5.26	3.32	5.39	7.84	4.13	3.91	7.05	5.32

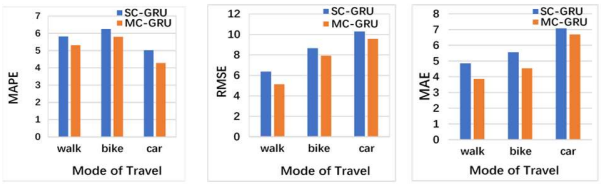
According to the experimental results in Table II, the performance of MC-GRU in the simulation data set is generally better than that in the real data set 1. Because in dataset 2, we compress the track distance of subset 1 and subset 2, which is only the simulation of spatial characteristics, and lack of targeted imitation for the implicit factors mentioned in our experiment and the potential characteristics of different traffic modes. Therefore, the multimodal prediction is relatively single, and the prediction accuracy is high. But in the same case, MC-GRU also shows better performance than other comparative experiments.

b) Module Comparison: In order to evaluate the prediction contribution of each part of MC-GRU model on the data set, we conducted experiments on three modules respectively to get the performance differences between them

EFFECT OF MULTIMODAL CONVOLUTION MODULE

In the first experiment, we transform the multimodal convolution module into the traditional convolution (single mode convolution, SC), that is, all tracks are processed by the same convolution process in the three modes. The predicted results are demonstrated in the Fig.6.

Compared with the MC-GRU, we found that the prediction effect of convolution processing using a single mode is much worse. Our multimodal convolution module is proposed to solve this contradiction, through the comparison of graphs the multimodal convolution module can effectively improve the prediction effect.



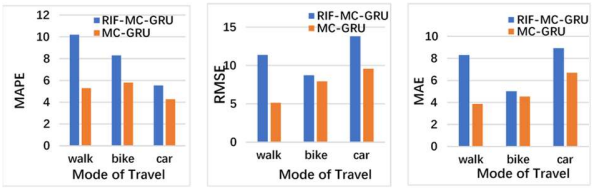
(a) MAPE Compare (b) RMSE Compare (c) MAE Compare

Fig.6. Comparison between single convolution and multimodal convolution

EFFECT OF IMPLICIT FEATURE PROCESSING MODULE

Whether the implicit feature processing module model can accurately predict travel time is a core issue in this topic. In this comparative experiment, we temporarily remove the implicit feature processing module, only feeding the extracted spatial features from multimodal convolution into the feature fusion module, which is called the Remove Implicit Features GRU(RIF-GRU).

As illustrated in Fig.7, it can be seen that removing the implicit feature processing module has a great impact on the accuracy of the prediction results. This shows that the time point and date of departure have important reference significance for predicting travel time, which is consistent with the periodicity of traffic state.

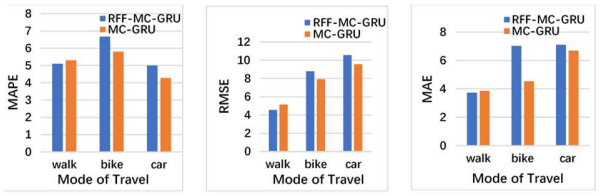


(a) MAPE Compare (b) RMSE Compare (c) MAE Compare

Fig.7. Effect of implicit feature processing module

EFFECT OF FEATURE FUSION MECHANISM

In order to verify the function of feature fusion module in MC-GRU model, we remove the feature fusion module in this comparative experiment. The spatial features extracted by multimodal convolution module and the features obtained by implicit feature processing module are combined together, which is directly sent to the cyclic neural network for temporal feature processing, called the Remove Feature Fusion MC-GRU (RFF-MC-GRU). The prediction results are compared with those of the MC-GRU model. The results are shown in Fig. 8.



(a) MAPE Compare (b) RMSE Compare (c) MAE Compare

Fig.8. Influence of feature fusion module

Compared with the MC-GRU, the prediction results of the RFF-MC-GRU are worse than the original experimental results in bike and car travel mode, but the indicators show little difference in walking mode. The results show that the pedestrian travel time is less affected by other traffic modes.

In order to verify the relationship between multimode convolution module and feature fusion module, we compare the SC-GRU and RFF-MC-GRU results, as shown in Fig.9. Comparing the three modes, we can see that the SC-GRU model and RFF-MC-GRU model have similar effect on the prediction of vehicle travel time and bike travel time, which is particularly significant in the prediction of vehicle travel time. Combined the previous comparative experiments, we can draw a conclusion that the multimodal convolution module and the feature fusion module should be utilized together to show their respective advantages. In the prediction of walking time, the prediction effect of RFF-MC-GRU outperforms SC-GRU, which shows that spatial feature extraction and implicit factors are indispensable in walking prediction.

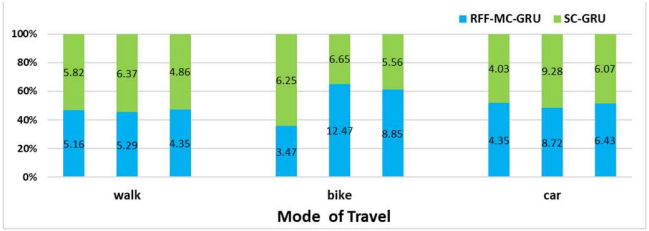


Fig.9. The influence relationship between SC-GRU and RFF-MC-GRU

V. CONCLUSION

In this paper, we propose a MC-GRU deep learning framework to predict the travel time under multimodal conditions. The experimental results show that the travel time prediction based on the influencing factors of different traffic modes is very meaningful. Compared with the existing deep learning traffic prediction model, Multimodal Convolution Module has achieved better results in different traffic modal feature extraction. Furthermore, the Feature Fusion Module of MC-GRU fuses the features and calculates the Influence Degree between different traffic modes, which plays an important role in the final output of the model. However, the MC-GRU model still has great room for improvement in terms of prediction index, and the relationship between vehicles requires further analysis and research. In the future, we will focus more on the feature fusion principle of MC-GRU and external influencing factors to improve pression the travel time prediction.

ACKNOWLEDGMENT

This research was supported in part by National Key Research and Development Plan Key Special Projects under Grant No. 2018YFB2100303, Shandong Province colleges and universities youth innovation technology plan innovation team project under Grant No. 2020KJN011, Shandong Provincial Natural Science Foundation under Grant No. ZR2020MF060, Program for Innovative Postdoctoral Talents in Shandong Province under Grant No. 40618030001, National Natural Science Foundation of China under Grant No. 61802216, and Postdoctoral Science Foundation of China under Grant No.2018M642613.

REFERENCES

[1] N. Liu, S. Zhao and N. He, "Study about Factors Influencing Travel Time Based on Factor Analysis," *In Proceedings of the 2011 International Conference on Information, Services and Management Engineering* (ISME 2011) (Volume 3), 462-466.

[2] J. Tang, F. Liu, Y. Zou, W. Zhang and Y. Wang, "An Improved Fuzzy Neural Network for Traffic Speed Prediction Considering Periodic Characteristic," *In IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 9, pp. 2340-2350, doi: 10.1109/TITS.2016.2643005, Sept. 2017.

[3] D. Wang, J. Zhang, W. Cao, J. Li and Y. Zheng, "When Will You Arrive? Estimating Travel Time Based on Deep Neural Networks," *In AAAI*, 2018.

[4] D. Ettema, M. Friman, L. E. Olsson, T. Gärling, "Season and weather effects on travel-related mood and travel satisfaction," *Frontiers in Psychology*, 8. doi:10.3389/fpsyg.2017.00140, 2017.

[5] G. Gao, Z. Wang, X. Liu, Q. Li, W. Wang and J. Zhang, "Travel behavior analysis using 2016 Qingdao's household traffic surveys and Baidu electric map API data," *J. Adv. Transp*, vol. 2019, Mar. 2019.

[6] X. Chen, H. Gong and J. Wang, "BRT Vehicle Travel Time Prediction Based on SVM and Kalman Filter," *Journal of Transportation Systems Engineering and Information Technology*, vol. 12, pp. 29 – 34, 2012.

[7] R. P. D. Nath, H.-J. Lee, N. K. Chowdhury and J.-W. Chang, "Modified k-means clustering for travel time prediction based on historical traffic data," *Proc. Int. Conf. Knowl.-Based Intell. Inf. Eng. Syst.*, pp. 511-521, 2010.

[8] B. Yang, C. Guo, and C. S. Jensen, "Travel cost inference from sparse, spatio temporally correlated time series using markov models," *Proceedings of the VLDB Endowment* 6(9):769–780, 2013.

[9] Y. Wang, Y. Zheng, and Y. Xue, "Travel time estimation of a path using sparse trajectories," *In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 25–34. ACM, 2014.

[10] E. Jenelius, H. N. Koutsopoulos, "Travel time estimation for urban road networks using low frequency probe vehicle data," *Transportation Research Part B: Methodological* 53:64–81, 2013.

[11] L. Kang, G. Hu, H. Huang, W. Lu, and L. Liu, "Urban Traffic Travel Time Short-Term Prediction Model Based on Spatio-Temporal Feature Extraction," *Journal of Advanced Transportation* Vol.2019. doi: 10.1155/2020/3247847.

[12] Y. Duan, Y. L.V. and F. Wang, "Travel time prediction with LSTM neural network," *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Rio de Janeiro, pp. 1053-1058, doi: 10.1109/ITSC.2016.7795686, 2016.

[13] Q. Chen, D. Wen, X. Li, D. Chen, H. Lv and J. Zhang, "Empirical mode decomposition based long short-term memory neural network forecasting model for the short-term metro passenger flow," *PLoS ONE 14(9): e0222365*. https://doi.org/10.1371/journal.pone.0222365 PMID: 31509599, 2019.

[14] D. Yu, S. Qiu, H. Zhou and Z. Wang, "Research on short-term traffic flow prediction at intersections based on GRU-RNN model," *Highway Engineering*, 45(04): 109-114, 2020.

[15] N. Jin, M. Liu, X. Gu and Y. Han, "Two parameter convolution model for traffic time prediction." *Computer engineering and application*,56 (20): 258-263, 2020.

[16] H. Dong, X. Sun, L. Jia and Y. Qin, “Multimodal traffic flow prediction model,” *Journal of Jilin University (Engineering Edition)*,41 (03): 645-649, 2011.

[17] G. Wang, “Research on driver fatigue detection method based on multimodal feature fusion,” *Shandong University*, 2020.

[18] C. Li, “Urban spatial perception based on multimodal trajectory data,” *Shenzhen University*, 2019.

[19] K. Zhang, N. Jia, L. Zheng, and Z. Liu, “A Novel Generative Adversarial Network for Estimation of Trip Travel Time Distribution with Trajectory Data,” *Transportation Research* Part C: Emerging Technologies 108: 223–244. doi: 10.1016/j.trc.2019.09.019, 2019c.

[20] T. Adetiloye and A. Awasthi, “Multimodal Big Data Fusion for Traffic Congestion Prediction,” *In: Seng K., Ang L., Liew AC., Gao J. (eds) Multimodal Analytics for Next-Generation Big Data Technologies and Applications. Springer, Cham*,https://doi.org/10.1007/978-3-319-97598-6_13, 2019.

[21] J. Ma, J. Chan, G. Ristanoski, S. Rajasegarar and C. Leckie, “Bus travel time prediction with real-time traffic information,” *Transp. Res. C Emerg. Technol.*, vol. 105, pp. 536-549, Aug. 2019.

[22] Y. Gal, “A theoretically grounded application of dropout in recurrent neural networks,” arXiv:1512.05287, 2015.

[23] G. Dai, C. Ma and X. Xu, “Short-Term Traffic Flow Prediction Method for Urban Road Sections Based on Space–Time Analysis and GRU,” *in IEEE Access*, vol. 7, pp. 143025-143035, 2019, doi: 10.1109/ACCESS.2019.2941280.

[24] D. Zhang and M. R. Kabuka, “Combining weather condition data to predict traffic flow: A GRU-based deep learning approach”, *IET Intell. Transp. Syst.*, vol. 12, no. 7, pp. 578-585, Mar. 2018.

[25] D.-A. Clevert, T. Unterthiner and S. Hochreiter, “Fast and accurate deep network learning by exponential linear units (elus).” *arXiv preprint* arXiv:1511.07289, 2015.

[26] C. Wu, J. Ho and D. Lee, “Travel-time prediction with support vector regression,” in IEEE Transactions on Intelligent Transportation Systems, vol. 5, no. 4, pp. 276-281, Dec. 2004, doi: 10.1109/TITS.2004.837813.

[27] X. Ran, Z. Shan, Y. Fang and C. Lin, “A Convolution Component-Based Method with Attention Mechanism for Travel-Time Prediction.” *Sensors 2019*, 19, 2063.

[28] S. H. Park, B. Kim, C. M. Kang, C. C. Chung and J. W. Choi, “Sequence-to-Sequence Prediction of Vehicle Trajectory via LSTM Encoder-Decoder Architecture,” *2018 IEEE Intelligent Vehicles Symposium (IV)*, *Changshu*, pp. 1672-1678, doi: 10.1109/IVS.2018.8500658, 2018