Knowledge-Based Systems 258 (2022) 110035

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

DouFu: A Double Fusion Joint Learning Method for Driving Trajectory Representation

Han Wang, Zhou Huang *, Xiao Zhou, Ganmin Yin, Yi Bao, Yi Zhang

Institute of Remote Sensing and Geographical Information Systems, Peking University, Beijing, China
Beijing Key Lab of Spatial Information Integration & Its Applications, Peking University, Beijing, China

ARTICLE INFO

Article history:
Received 11 November 2021
Received in revised form 13 October 2022
Accepted 13 October 2022
Available online 20 October 2022

Keywords:
Trajectory mining
Representation learning
Spatio-temporal analysis
Multimodal fusion
Attention mechanism

ABSTRACT

Driving trajectory representation learning is of great significance for various location-based services such as driving pattern mining and route recommendation. However, previous representation generation approaches rarely address three challenges: (1) how to represent the intricate semantic intentions of mobility inexpensively, (2) complex and weak spatial–temporal dependencies due to the sparsity and heterogeneity of the trajectory data, and (3) route selection preferences and their correlation to driving behaviour. In this study, we propose a novel multimodal fusion model, DouFu, for trajectory representation joint learning, which applies a multimodal learning and attention fusion module to capture the internal characteristics of trajectories. We first design *movement*, *route*, and *global* features generated from the trajectory data and urban functional zones, and then analyse them with an with the attention encoder or fully connected network. The attention fusion module incorporates *route* features with *movement* features to create more effective spatial–temporal embedding. Combined with the global semantic feature, DouFu produced a comprehensive embedding for each trajectory. We evaluated the representations generated by our method and other baseline models on the classification and clustering tasks. The empirical results show that DouFu outperforms other models in most learning algorithms, such as the linear regression and the support vector machines, by more than 10%.

1. Introduction

Enormous amounts of driving data are generated and captured in the form of trajectories as GPS devices become more widely available and numerous location-based applications are developed. Trajectory is traditionally described as a sequence of spatially located points with time stamps [1,2]. As a temporal record of interactions between users and the spatial environment, driving trajectories are capable of demonstrating users' behavioural characteristics and travel intention, which can be exploited further in various applications such as user portrait analysis [3,4], next location recommendation [5–8], and human activity classification [9]. Patterns mined from trajectories can also offer instructions and advice for transportation system optimisation and city planning. The representation of a trajectory inside machines is a critical issue before data mining applications are completed. In addition to spatial and temporal movements, the trajectory is semantically rich in route selection, which can help identify the purpose of drivers. Despite its significance, few current representation models have attempted to incorporate semantic information.

Recently, the learning representation of trajectories in a fixed-length latent space has been studied intensively. Trajectory data of various lengths cannot share the same fixed-length representation space, making it challenging to manage and evaluate the characteristics and relevant relationships. However, in the learned feature space, each trajectory is converted into a fixed length vector called an embedding, which represents this trajectory. An effective representation learning model can maintain the relative relationships between trajectories and map them into a distance between embeddings in the feature space [10]. Deep learning methods have been effectively utilised in many trajectory data mining tasks, including representation learning [11–13]. A variety of downstream tasks can be accomplished using the trajectory embeddings produced by the learning models. Embeddings make it easy to measure similarities between trajectories [14,15]. In the latent feature space, the Euclidean distance between embeddings may reveal the relationship between them, which is helpful for trajectory-group partitioning and clustering. Meanwhile, trajectory analysis can demonstrate users' characteristics and preferences about travelling, which will benefit for location-based recommendation system [3].

* Corresponding author at: Institute of Remote Sensing and Geographical Information Systems, Peking University, Beijing, China.
E-mail address: huangzhou@pku.edu.cn (Z. Huang).

https://doi.org/10.1016/j.knosys.2022.110035
0950-7051/

However, current models mainly concentrate on the spatial and temporal characteristics of trajectories [16] and do not consider rich semantic information. The trajectory should accurately represent the user's purpose to travel as a spatial–temporal interaction sequence for drivers who intend to use the service at the destination along a certain route [17]. In addition to the departure time and destination location, the functions provided by the destination can also help identify user needs. Route selection is also important in the travel process. Different drivers can choose different routes. Some drivers may prefer a faster route, while others may prefer a shorter route. Only a few of them prefer like to take the fastest route [18]. The selection of a route demonstrates the characteristics of users. However, it is rarely mentioned in existing research. Furthermore, several data inputs may not be perfectly independent of each other, especially those which occur in the same spatial–temporal environment. Therefore, it is necessary to investigate the associations between various modalities in the learning model. A correlation model can exploit different attributes to calibrate with each other to achieve for better performance.

Thus, it is possible to analyse the semantic information of trajectories from such perspectives. We can convert a trajectory from these perspectives into different independent inputs and apply methods to extract their features. These modality features can coordinate with each other to improve the performance of downstream tasks. In this study, we aimed to make full use of the multimodal features of a trajectory to produce a better representation.

To address these issues, we propose a novel representation learning model, **DouFu**, for driving trajectories inspired by multimodal fusion and attention mechanisms. The spatial–temporal and semantic information of the trajectory can be merged in the process of multimodal learning to acquire an effective computational representation for driving in a latent feature space, thereby providing a effective foundation for downstream data mining and pattern analysis applications. The main contributions of this study are as follows:

- Considering the spatio-temporal and semantic information in the trajectory data, we decompose the sequence data into several modals, such as movement, route selection, and global semantic characteristics, and process them to obtain independent embeddings, which are fed into our model.
- We present a representation learning model for trajectory embeddings based on a multimodal fusion and attention mechanism that combines the features from several modalities to produce an adequate representation of a trajectory for downstream tasks.
- We conducted extensive experiments, including supervised classification tasks and unsupervised cluster tasks, on real-world trajectory data to evaluate the model. These results justify the ability of our model in representation generation.

The remainder of the study is organised as follows. Section 2 reviews the related work. After introducing the preliminaries in Section 3, we propose the DouFu framework in Section 4. Section 5 presents experimental evaluations of our model, and Section 6 provides the conclusion.

2. Related work

We briefly introduce three major parts of the related work on trajectory representation learning, multimodal learning, and the attention mechanism.

2.1. Representation learning in trajectory

Representation learning is the basis of intelligent computational models. A good representation can effectively reveal the internal relationships and differences between original factors. Recently, research on representation learning has extended to trajectory sequence data. Various models have been developed to capture internal representations. Deep learning models such as recurrent neural networks (RNN) and one-dimensional convolutional neural networks have been applied to time series data created by sliding windows [11,12]. Kieu et al. [13] considered trajectory data as an image that was fed to a convolutional neural network to learn embeddings to represent trajectories. Ren et al. [19] extracted manually defined patterns from data and employed Siamese networks, which are capable of training a metric to measure the similarity between historical trajectories. This line of research is limited to the physical variables. None of these studies have recognised the importance of functional semantic information in trajectories, which may contribute significantly to the performance of embedding generation. In fact, it is more important to learn patterns, such as the longest stay point and daily commuting schedule. Gao et al. [20] developed a sequence split model to group GPS location points into subsequences. Adapted from the word2vec method, this model creates check-in embeddings for each subsequence in order to learn a representation. In addition to spatial patterns from location information, extensive researches has focused on utilising semantic data to produce understandable representations. Ying et al. [21,22] presented semantic trajectory mining methods for location predictions. Most current models prefer to incorporate spatial–temporal records with POI to extract more explicit information [4,9]. However, more data sources can provide more detailed information. In addition to POI, other mobility contexts can be applied to improve the quality of the embeddings. Fu et al. [23] proposed a learning model, Trembr, based on road segment embeddings obtained from geometric map matching. Trembr uses the prediction of road segments co-occurrence in the same trajectory to learn its representation. Subsequently, only the road segments sequence is fed to an encoder–decoder model. Zhou et al. [24] designed a contrastive mobility learning model with data augmentation for self-supervised spatial–temporal context learning. Siami et al. [25] applied self-organised mapping and a deep autoencoder to analyse the driving style patterns of various trajectories. VAMBC [26] detects the stay points and extracts the corresponding geographical contexts from the trajectories. After being processed by the long short term memory module, VAMBC leverages a variational autoencoder to model the individualised latent embedding. Tabatabaie et al. [27] leveraged inverse reinforced learning techniques to recover the reward and policy from trajectories by modelling drivers as agents. Furthermore, the trajectories were resampled to mobility tensors with different resolutions. Combined with the reward, POI, and weather features, final embeddings are generated from multi-scale convolution network modules.

However, these methods rarely mention the route selection semantics. More features, including geometries and semantics, can be exploited in an inexpensive manner than in the previous models. In terms of mobility, time series information matters in semantic analysis, which can reveal the drivers' personalised route choice paradigm. In addition to statistical spatial features, dynamic temporal data provides more flexible details and a new perspective for inspecting driving behaviour and mobility pattern.

2.2. Multimodal learning

Building models using multimodal data is a significant topic that has been extensively studied in recent years [28]. Generally, a modality refers to a way to capture information about an object independently. More practically, modalities refer to different information sources from the same event, which can help us to know more about it. Different specific methods are applied to the data captured from different modalities, which are be mapped to similar feature spaces. Furthermore, it is convenient to use convolutional neural networks to process spatial visual information, such as videos, while using RNNs to process sequential sound information, such as audio. Multimodal methods have a variety of applications, especially in multimedia areas such as human emotion classification [29,30] and video-subtitle alignment [31,32], because of massive multi-source data.

Recently, a considerable amount of research has been conducted which combines multimodal data to solve the trajectory representation learning problem. GPS, cameras, motion sensors, and many other devices, such as visual and auditory data sources in multimedia, collect mobile information about trajectories. It is possible to use multimodal learning methods in trajectory analysis. Visual images and driving operation recorders can also contribute to modality coordination and fusion [33]. Cui et al. [34] applied convolutional neural networks to the a sky-view scene with driving states to make an online prediction of the next trajectory location. Chen et al. [35] employed surveillance camera data to track a vehicle and recover a sparse driving trajectory. Nevertheless, the majority of the present methods involve the use of more sensor devices, such as video cameras, which is not very convenient and inexpensive. It is easier to use existing spatial records, such as mapsm instead of sensors as a modality. DiversityGAN [36] attempted to integrate trajectory features with a map embedding vector generated from nearby lane coefficients. Feng et al. [37] proposed a model called DeepMove that employs a jointly multimodal embedding module with manually designed spatiotemporal and personal features to learn a dense representation. Inspired by DeepMove, VANext [38] uses the variational attention method instead of the traditional attention mechanism to learn patterns in the historical trajectory data. Essentially, features from different modalities are projected into similar feature spaces and then coordinated into the same one.

Nevertheless, these studies have ignored the relationship between modalities in feature engineering. The models learn the weights to map modality features into several similar feature spaces and then coordinate them to build a unified feature space for downstream usage. However, it is difficult to extract knowledge from spatial–temporal modals independently because of sparsity and long-term dependencies. Potentially, the relationships between modalities help correct biased perceptions of objects while learning the mapping weights, especially for sparse spatial–temporal data.

2.3. Attention mechanism

Recently, the self-attention mechanism has attracted much attention [39] and performs well in many learning tasks, especially in the natural language processing methods, such BERT [40] and XLNet [41]. The attention module aims to simply compute the correlation between the query and value to produce weight, and then apply attention weights to origin data in order to emphasise more important regions in information flow, which can improve the model performance. Compared to trivial RNNs, attention computation can resolve long-term dependencies on time series data. In the field of spatial–temporal data mining area, RNNs may fail to process the risk of gradient vanishing caused by objects such as long trajectories due to the data sparsity. Therefore, numerous emerging methods with the attention mechanism have been proposed. Gao et al. [38] proposed a generative model to extract the attention of POIs from historical trajectories in order to learn representations. DeepMove [37] designed a historical attention module to select the most similar historical trajectory to match the current trajectory. The STDN [42] adopt a shifted attention mechanism to obtain periodic temporal information in trajectory mining.

Several studies have been conducted using different attention information from various independent modalities. ST-LBAGAN [43] uses a bidirectional attention method to learn a feature map for missing traffic data imputation. Yu et al. [44] designed a modular co-attention network which utilises the attention from video input to decode the answer from the attention of given questions to complete the video question answering task. In addition, LXMERT [45] employs a cross-modality encoder to extract mutual cross-attention by encoding. It is possible to analyse the multimodal trajectory data in a similar manner because of the extensive spatio-temporal and semantic dependencies

3. Preliminaries

In this section, we first define several essential concepts and then formally formulate the trajectory representation problem.

Definition 1 (*Driving Trajectory*)**.** The GPS device on the vehicle can record the driver's location during the travel at a specific sampling rate. Usually, a GPS point p consists of a location point with longitude lat and longitude lng and the current time stamp t, i.e.$p = \{lat, lng, t\}$. A driving trajectory τ is a sequence composed of GPS points from such a GPS recorder, i.e. $\tau = \{p_1, p_2, \ldots, p_n\}$.

Definition 2 (*Trajectory Set*)**.** Given a user u and a set of historical trajectories $\mathcal{T} = \{\tau_1, \tau_2, \ldots, \tau_n\}$, the trajectory set $\mathcal{S} = \{u, \mathcal{T}\}$. A user-trajectory classification task aims to identify the true user of a new trajectory with the historical trajectory set as training data.

Definition 3 (*Road Segment*)**.** In an urban traffic environment, all roads are connected to one another to construct a directed network $G = \{V, E\}$. A road segment is an edge with several attributes from the edge set E, which can be denoted as $r = \{l_{in}, l_{out}, a\}$ where l_{in} is the in-edge set and l_{out} means the out-edge set. In addition, road interaction i is a vertex from the vertex set V.

Definition 4 (*Route*)**.** Driving from the origin to the destination is actually along the road segments in the network while recording the location using GPS. Hence, a route R can be defined as a sequence of road segments, i.e. $R = \{r_1, r_2, \ldots, r_n\}$.

Definition 5 (*Functional Zone*)**.** Travellers may want to obtain services at the destination. The proportion of function types in the origin/destination neighbourhood can represent the type of service at the current location. A functional zone vector $z_o = \{f_1, f_2, \ldots, f_n\}$ where f_i is the ith function, and $\sum_i^n f_i = 1$.

Problem (*Driving Trajectory Representation Learning*)**.** Given trajectory sets $\{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_n\}$ from user group $\mathcal{U} = \{u_1, u_2, \ldots, u_n\}$, train a model M which can transform trajectories into vectors in a specific feature space $\mathcal{H}$. For each trajectory τ_u belonging to user u, we can output an embedding $\epsilon_\tau = M(\tau_u) \in \mathcal{H}$ where $\mathrm{Min}_{i \in \mathcal{U}} \sum_{\tau \in \mathcal{S}_u} Dist(\epsilon_\tau, c_i) = u$, where $Dist(\epsilon_\tau, c_i)$ indicates the distance between embedding ϵ_τ and the trajectory clustering centre c_i for driver i.

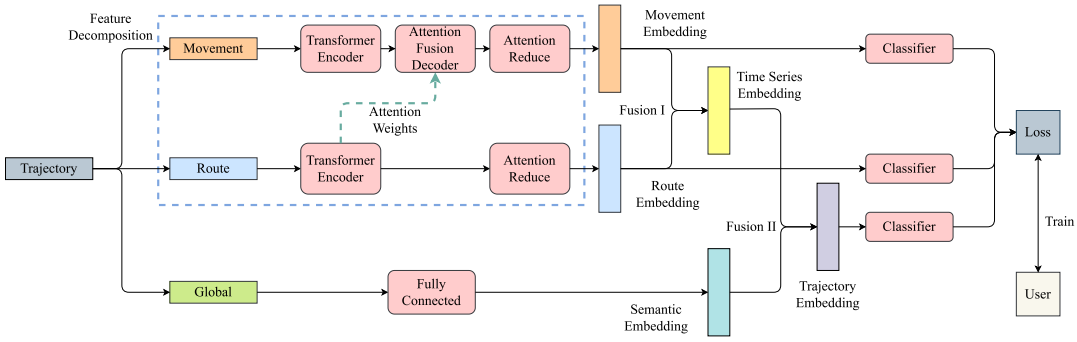


Fig. 1. Overview of the DouFu framework. The *movement* and *route* features are fed into the transformer encoder to output time series embeddings with an attention fusion module. The Global feature is fed into the fully connected network to produce a semantic embedding. All embeddings are incorporated to generate the final representation of the trajectory.

4. The DouFu model

4.1. Framework overview

The RNN is an effective tool for analysing time series data but may be insufficient for he task of driving trajectory representation learning. Several challenges must be addressed. First, the driving trajectory captured from vehicle GPS devices often has a long sequence record with an average sampling rate of approximately 10 s or more per point, with significant noise. Under such circumstances, RNNs are less likely to learn long-term and sparse dependencies. Furthermore, an itinerary can be converted into modal data of various lengths, such as spatially located points and POI check-ins. However, it is difficult for RNNs to obtain correlations between these sequences of data during processing. We propose a novel double fusion model for trajectory representation learning that is inspired by multimodal learning and the attention mechanism.

Before learning about the model, it is fundamental to recognise that a single trajectory can be regarded as encompassing multiple modalities. The raw data of each modality must be processed using detailed feature engineering. Previous studies have tended to focus on how to design computational inputs, usually multi-dimensional matrices and sequences, to the model from massive trajectory data and the complex spatial–temporal context referred to as a feature engineering process. It is necessary to design individual feature inputs for different independent types of data, including spatial–temporal trajectories, land-use types, and road segment information. An effective feature engineering method can improve the performance of data-driven models. In our model, a trajectory can be decomposed into three independent modalities: *movement*, *route* and *global* features (See Fig. 1), the details of which are explained in Section 4.2.

Similar to other multimodal models, we initially handle these modalities using different methods. The *movement* and *route* features, as sequential inputs, contain rich time series information about the driving characteristics and route selection preferences. Although most existing studies note that the geographical contexts of trajectories contain rich semantic information, they simply concatenate features as a unified vector, and rarely consider the correlations between different input features [36,37]. However, there is a high correlation between them because driving behaviour is limited to the current road segment. The traffic conditions, lane number, and geometry of the road can affect the driving process. Thus, the road network structure constrains the driving behaviour which can be identified from the *movement* features. Additionally, inspired by Xu et al. [31], who used deep learning techniques to align video and subtitles, we leverage the attention mechanism to compute the correlation degree for each unit of two sequences, the driving characteristics and the road segments, and coordinate them. We use a transformer encoder to extract the attention weights from them and then design an attention fusion decoder to combine these two attention weights to create a mixed time series embedding for the trajectory. With the *global* feature, which is semantically rich, a fully connected neural network was used to generate a semantic embedding to explore the mobility pattern 245 of users. In this section, we use two information fusion methods: attention fusion and embedding fusion. Then, the time series and semantic feature are merged to produce a final comprehensive embedding for the trajectory, which explores the driving behaviour, route selection preferences and mobility patterns. Next, we train three independent linear classifiers that identify the driver to which trajectory belongs to for the *movement* feature, *route* feature, and final feature to learn a better representation in a supervised learning manner.

4.2. Feature modality decomposition

As mentioned in Section 4.1, the trajectory can be viewed independently as a spatial movement sequence, road segment sequence, and the global semantic information. We introduce feature engineering in detail as follows.

Movement: Similar to the method proposed in [11], it is appropriate to apply a sliding window to the raw GPS data points to generate a physical movement feature sequence. We then calculated the statistical variables of the location points in the window as a unit, including the mean, minimum, maximum, standard deviation, and quantile (25%, 50%, and 75%) values of the speed norm, acceleration norm, speed difference norm, acceleration difference norm, and angle speed norm to construct the *movement* vector. The *movement* feature stores detailed information on the driving characteristics.

Route: In general, the trajectory can be converted into a sequence of road segments by extracting each road segment that passes through one by one. As Table 1 shows, for each road segment, a fixed length feature vector is composed of road attributes which include road start/end position, road length/direction, bounding box length/area, start/end point intersection number, road level and lane number. Using the sequence of road segments as road features, we were able to analyse the route selection preferences of drivers.

Global: Trajectories reveal the intentions of travellers who may need the services of the destination [46]. The description of the trajectory, regardless of spatial and temporal information, can also represent travel patterns. In Table 2, the fixed length vector, as a global feature, includes departure time, O/D locations and some statistical results of the *movement* and *route* features. Most importantly, the proportion of land use types [47–49] in the

Table 1
Variables of road segment features. The designed variables include geometric, geographic, and semantic attributes of a road segment. Categorical variables like road class are encoded in the form of one-hot code.

Feature	Variable
Bounding box	Location Edge length Area Direction
Location	Origin Destination
Intersection Number	In Out
Other attribute	Function zone of buffer Road segment length/width Point Number Lane Number One hot code of road class

Table 2
Variables of global features. The variables include statistical data of the movement feature and the route feature. Categorical variables like departure time are encoded in the form of one-hot code.

Feature	Variable
Statistic	Speed Difference of speed Acceleration Difference of acceleration Angle Speed
Location	Origin Destination
Bounding box	Edge length Area
Other attribute	Direction Length One hot code of departure time Duration Function zone of buffer
Road segment (Mean)	Lane number Road length/width Point number In/Out intersection number

O/D functional zone which represents the functional difference between origin and destination. This demonstrates the mobility intention of the driver in a less expensive way than demographic data or other sensor data.

Generally, the variables in tables are arranged as computable numerical vectors, which are then fed into the model. The road segment features in Table 1 were designed to describe the characteristics of each segment through which the trajectory passes. For example, we developed m independent features as described in Table 1 for a road segment and compacted them into a vector. For a trajectory with n road segments, an $m \times n$ matrix was formulated as the input of the transformer encoder. In addition, a positional encoder module is applied to assign a positional value which represents the order of the road segment in the sequence. For the *global* feature in Table 2, a fully connected layer is applied to produce a latent semantic embedding vector.

4.3. Road segment geometric feature pre-training

From the driver's perspective, the route can be converted into the form of a road segment sequence. However, in an urban transportation environment, all roads connect to build a network. Other road segments not in the driver's route can also affect the current route selection decision. Hence, it is necessary to consider segments other than those in the route when constructing a route model. Inspired by word2vec [50] method, it may be preferable to create a fixed-length pre-trained feature for road segments, such as word vectors. To solve this problem, we apply a variational graph autoencoder (VGAE) [51] to generate a geometric embedding instead of a hand-designed feature as the model input.

First, a road network must be created from segments of historical trajectories. As Fig. 2 shows, there is a trajectory set $T = \{\tau_1, \tau_2, \tau_3, \tau_4\}$ and road segment set $R = \{R_1, R_2, R_3, R_4, R_5, R_6\}$ where $\tau_1 = \{R_1 \rightarrow R_2 \rightarrow R_3\}, \tau_2 = \{R_1 \rightarrow R_2 \rightarrow R_4\}, \tau_3 = \{R_1 \rightarrow R_5 \rightarrow R_2\}, \tau_4 = \{R_2 \rightarrow R_4 \rightarrow R_5\}$. Then, a graph $G = \{V = \{R_1, R_2, R_3, R_4, R_5\}, E, X\}$ that uses road segments as vertex set V and interactions as edge set E with road segments features X is generated.

Second, the embedded road segment is encoded using a variational graph autoencoder that can be trained further during the subsequent modelling process. In our method, a two-layer graph convolution network is adopted to estimate the parameters μ, δ of the conditional Gaussian probability distribution p for each vertex. Subsequently, a latent variable z is created to represent such a vertex by sampling from this distribution. For the edge link from vertex i to vertex j, the dot product probability of z_i, z_j:

$$p\left(E_{ij} = 1 \mid \mathbf{z}_i, \mathbf{z}_j\right) = \sigma\left(\mathbf{z}_i^{\top} \mathbf{z}_j\right) \tag{1}$$

where $\delta(\cdot)$ is the logistic sigmoid function that indicates the link probability value between i and j. Prediction edge set is as follows:

$$p(\mathbf{E} \mid \mathbf{X}) = \prod_{i=1}^{N} \prod_{j=1}^{N} p\left(E_{ij} \mid \mathbf{x}_i, \mathbf{x}_j\right) \tag{2}$$

which can be trained using the ground truth edge adjacency matrix later. After training, all the road segments are converted into fixed-length vectors with geometric information.

In summary, the road segment pre-training method is an additional module independent of the main network architecture that aims to improve the performance of DouFu. In the *route* feature process, it is possible to use the originally designed features as the input of the transformer encoder. However, considering the geometric connection properties of road networks, we leverage a variational graph autoencoder that takes the original feature and road network geometry as input to generate better latent embedding for each road segment. The latent embedding then replaces the original road segment feature, which is then fed to the model.

4.4. Attention fusion & reduction

After being processed in Sections 4.2 and 4.3, the movement and route features are fed into the transformer encoder, respectively, to calculate the movement attention A_m and route attention A_r of dimension d. Clearly, the situation on the road is greatly influenced by the location of the vehicle. All routes are restricted by roads and can only travel along them. Therefore, the movement attention from GPS sampling points, as a continuous signal with intensive noise, could be improved by route attention from a deterministic finite road set. Fig. 3 shows the architecture of the attention fusion decoder (see Fig. 4).

Before the fusion procedure, a consistent layer is applied to map the movement feature into the shape of the route feature. We then apply a fusion transformer decoder to decode movement attention with route attention and learn a fusion feature by adding the original movement vector:

$$\text{FusionAttn} = \text{LN}\left(\text{SM}\left(\frac{\text{C}(A_m) \cdot A_r}{\sqrt{d}}\right) A_m + A_m\right) \tag{3}$$

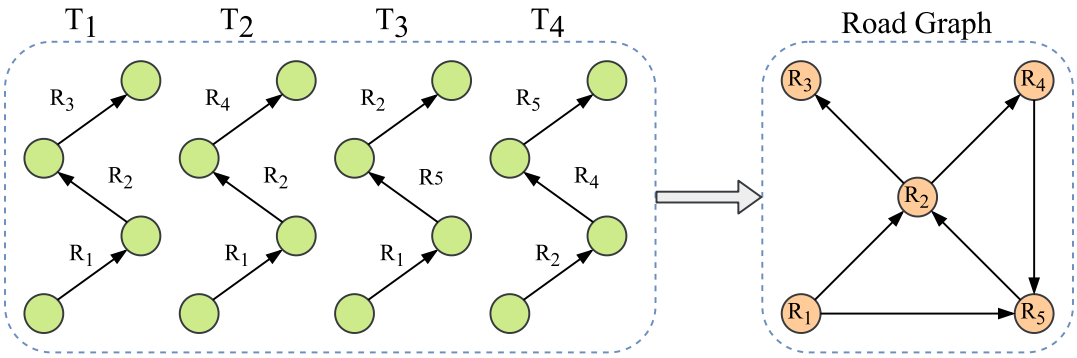


Fig. 2. Generation of road graph. The road graph uses road segments as vertices. By connecting each road segment vertex via historical trajectory data, we build a directed road graph to model the complex geometric relationship.

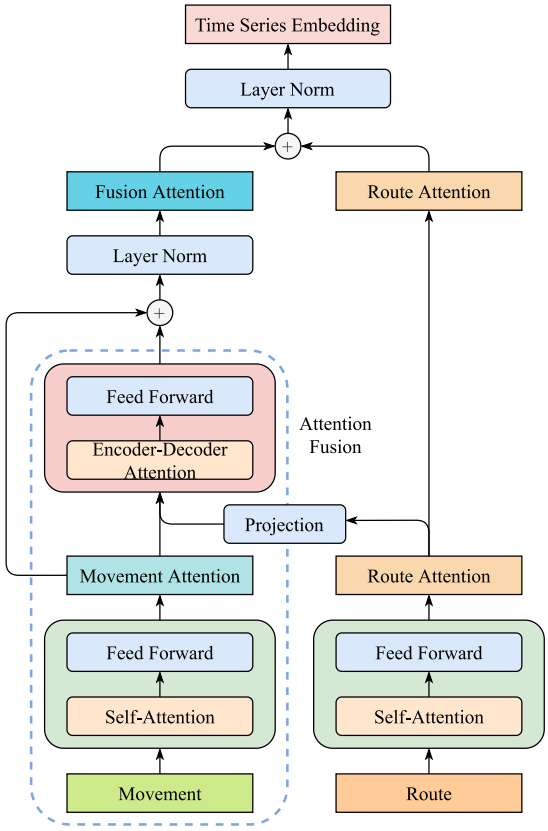


Fig. 3. Overview of the attention fusion decoder. We first leverage the self-attention encoder module to extract *movement* and *route* attention. Then, *route* attention is projected to be consistent with the *movement* attention and fed to the decoder module to generate fusion attention. A time series embedding is produced by adding fusion attention and route attention.

where LN is a layer norm operator, SM is a softmax operator, and C. is a consistent layer. After fusion, the sequence length of the *movement* feature is the same as that of the routes. Inspired by attention reduce [44], multiple linear layers are adopted to learn the weight of each feature to produce a fixed length embedding from the fusion features:

$$w_i = \text{softmax}\left(\text{MLP}_i\left(X\right)\right) \tag{4}$$

$$\tilde{x} = \sum_{i=1}^{n} \text{Layer Norm}\left(\sum_{j=1}^{m} \alpha_{ij} x_j\right) \tag{5}$$

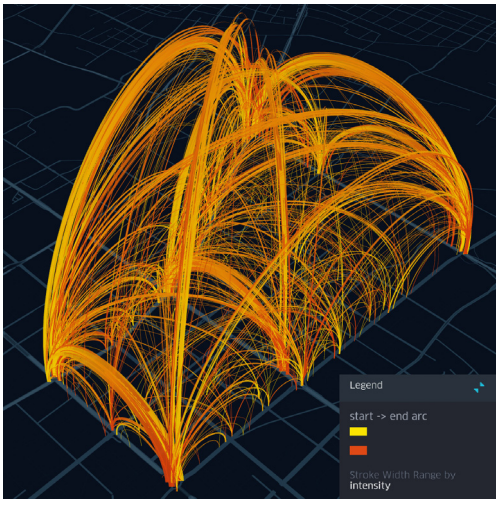


Fig. 4. Evaluation data area which includes 10,575 trajectories from 100 users.

where w_i is the ith weight for feature $X = \{x_1, x_2, \ldots, x_m\}$. The weighted sum of the features can be obtained as a representation result. We used cross entropy as the loss function to train three classifiers to identify the drivers of the current trajectory on top of the comprehensive feature:

$$\mathcal{L} = \mathcal{L}_{\text{fusion}} + \alpha \mathcal{L}_{\text{move}} + \beta \mathcal{L}_{\text{route}} \tag{6}$$

where α and β are loss weight adjustment factors. These classifiers contribute to representation learning in different training processes. In the early process, the *movement* embedding classifier and the route embedding classifier learn the internal characteristics of features faster than fusion embedding, which helps the entire model converge quickly. However, the performance of the model depends only on the result of fusion embedding. Thus, we must guarantee that the backpropagation of the fusion error is dominant in all error terms. The specific setting of the adjustment factors depends on the capacity and quality of the dataset and the experimental results.

5. Experimental evaluation

5.1. Experimental details

In this section, a private dataset is adopted from a navigation service company to train and evaluate our model using several

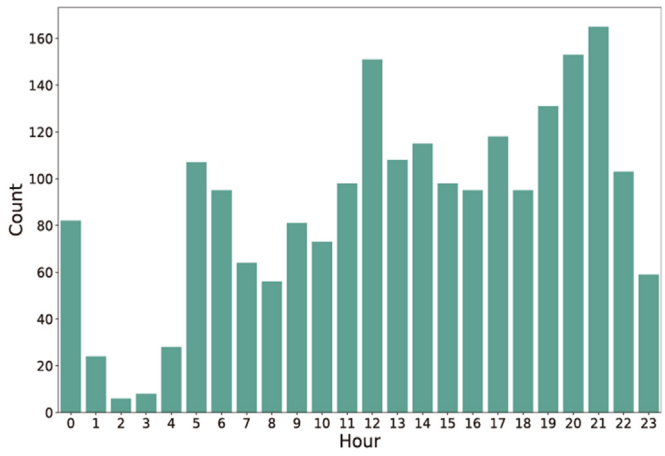


Fig. 5. Distribution of departure time. Most of the records are concentrated in the daytime, with the least from 0–5 o'clock.

baselines. This dataset collected a large number of driving trajectories in a subarea of Beijing in December 2018, as Fig. 3 shows. The departure times of these trajectory data are distributed over a 24-hour period in Fig. 5. The trajectories were firstly divided into training sets and test sets according to the user. We selected 100 drivers with 10,575 trajectories as the training set. No trajectory of users in the training set was in the test set. We then selected 21 drivers with 2113 trajectories as the testing set. The training set was split into 8:2 for training and 315 for validation. From the trajectories data, we created a road segment graph G with 4158 vertices and 33,731 edges for geometric learning.

To demonstrate the performance of the trained DouFu, we used embeddings of the same size from the test set to conduct the classification and the clustering tasks. In the classification task, some simple machine learning classifiers, such as linear regression and SVM, were adopted to complete the trajectory user classification with supervised training. In the clustering task, unsupervised methods, such as K-means were applied to group all trajectory representations into clusters. We will evaluate both of the experimental results

5.2. Evaluation metrics & baselines

Evaluation Metrics

Classification This task is intended to predict the candidate users for the input trajectory. We use accuracy and macro-F1 to measure the performance, which are common metrics for classification tasks.

- **Accuracy**:

$$ACC = \frac{\#\text{true prediction of trajectories}}{\#\text{trajectories}} \tag{7}$$

where $\#(\cdot)$ indicates the number of elements that meet the condition.

- **Macro-F1**:

$$\text{macro-F1} = \frac{2 \times P \times R}{P + R} \tag{8}$$

where P is average precision and R is recall value.

Clustering The clustering task attempts to divide the trajectories into groups. We apply the Davies–Bouldin index, normalised mutual information score, and adjusted rand score to compare the clustering result with the original trajectory sets $\mathcal{S}$.

- **Davies–Bouldin Index** calculates the ratio of within-cluster distances to between-cluster distances in order to measure the similarity of clusters which means a better clustering result is supposed to have a lower index value:

$$\text{DB} = \frac{1}{n}\sum_{i=1}^{n}\max_{j\neq i}\left(\frac{\sigma_i + \sigma_j}{d\left(c_i, c_j\right)}\right) \tag{9}$$

where c_i is the centre of ith class, σ_i is the mean distance between the sample in the ith class and c_i, and $d\left(c_i, c_j\right)$ is the distance between c_i and c_j.

- **Normalised Mutual Information Score** can estimate the correlation between two cluster results:

$$\text{NMI}\left(\Omega, C\right) = \frac{I\left(\Omega; C\right)}{\left(H\left(\Omega\right) + \frac{H(C)}{2}\right)} \tag{10}$$

where I is the mutual information and H is the entropy. Similar objects Ω and C result in a high NMI value.

- **Adjusted Rand Score** regards cluster as a decision process:

$$\text{ARI} = \frac{\text{RI} - E\left[\text{RI}\right]}{\max\left(\text{RI}\right) - E\left[\text{RI}\right]} \tag{11}$$

where RI is the original Rand Score:

$$RI = \frac{TP + TN}{TP + FP + TN + FN} \tag{12}$$

where TP is the number of true positives, FP is the number of false positives, TN is the number of true negatives, and FN is the number of false negatives. A higher Rand score indicates that the clustering results are consistent with the ground truth.

Baseline Algorithms

It is not appropriate to compare our multimodal model, with other models that use a single input, such as DeepMove [37]. Hence, we design several baseline algorithms to demonstrate the performance of the modules in our method.

- **RNN Move** [11] only applies a traditional RNN to the movement feature to generate an embedding without the input of route and global features.
- **RNN route**, which borrows the idea from RNN Move, feeds only the route feature to the RNN to produce a time series feature for route selection without the application of movement and global features.
- **Global** employs a fully connected layer neural network to capture internal characteristics from the *global* features designed in Section 4.2, without the input of *route* and *global* features.
- **DeepMove** [37] applies the attention mechanism which compares the current trajectories and historical trajectories. The original DeepMove is used to predict the next possible location for a POI sequence. However, we extract the vector result of the output layer as the trajectory embedding instead of feeding it into the softmax layer for the prediction task.
- **VANext** [38] uses the variational attention method to learn patterns from the historical trajectories data. Similar to Deepmove, we use the embedding result as the trajectory representation for the downstream tasks.
- **VAMBC** [26] leverages a variational autoencoder to model individual latent embeddings. In the experiment, we use the comprehensive embedding generated from Gumbel Softmax and the variational autoencoder to the representation of the current trajectory.
- **RNN Fusion** simply feeds the *movement* and *route* sequence features to an independent RNN, respectively, and combines them with a dense linear layer.

Table 3
The evaluation results of the user prediction task. Our proposed method outperforms baseline algorithms.

Model	Linear regression		Ridge regression		Naive Bayes		SVM		KNN		MLP		Decision tree	
	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1
RNN Move	28.67	25.93	27.99	22.34	27.31	23.79	20.15	17.73	26.37	24.83	29.35	28.03	22.45	21.96
RNN route	17.78	14.07	17.38	11.83	16.84	9.97	9.46	6.78	13.39	11.79	16.84	14.53	12.64	11.39
Global	12.70	11.84	10.95	6.94	9.73	7.02	7.91	5.71	6.82	5.32	13.92	11.84	7.43	6.65
DeepMove	29.87	26.03	28.01	22.22	27.82	23.98	20.82	17.99	26.68	25.25	30.05	28.72	22.98	22.05
VANext	30.29	26.55	28.33	22.54	28.01	24.77	21.09	18.13	27.92	26.02	32.36	30.35	22.82	21.99
VAMBC	30.87	27.02	28.93	23.02	28.92	25.02	21.23	18.76	28.45	26.88	33.12	32.12	23.18	22.32
RNN fusion	31.68	29.38	28.91	23.57	25.39	22.78	21.19	18.91	28.64	26.61	30.40	28.80	22.35	22.00
Semantic fusion	40.64	39.05	37.73	32.03	33.61	30.51	30.90	31.15	30.56	28.40	35.23	34.43	21.43	21.64
Attention fusion	42.26	41.20	37.19	31.57	37.46	34.25	33.06	32.42	34.01	31.79	37.39	36.40	24.38	24.42
Double fusion	43.81	42.03	41.45	35.33	39.01	35.85	31.10	29.47	34.41	32.34	40.56	36.85	24.47	24.61

- **Semantic Fusion** is a RNN based model which can be considered as a simplified version of the proposed method. Instead of using a transformer to extract the attention weight of features, it applies RNN to process the time series embedding. It is necessary to compare the performances of the RNN and the attention module.
- **Attention Fusion** only applies the attention mechanism and attention reduce module to process time series data and then only uses the fully connected layer to integrate them without the attention fusion module.

The baseline algorithms are simplified variants of the proposed method. The RNN Move, RNN Route and Global baselines only leverage only one of the three designed features without fusion techniques. The RNN Fusion, Semantic Fusion and Attention Fusion combine at least two feature inputs in a multimodal learning manner.

Implementation details

We leveraged Pytorch to implement DouFu and other baseline algorithms. All experiments were conducted on an NVIDIA TITAN GPU with 24 GB memory. In the training process, we set the data batch size to 32 and the number of epochs to 200. The initial learning rate was 0.006. Additionally, we used the warm-up and cosine annealing scheduler to adjust the learning rate. The loss weight adjustment factor α was set to 0.005, while β was set to 0.001, as the section of the sensitivity study shows.

5.3. Result analysis

Classification

We first trained Linear Regression, Ridge Regression, Naive Bayes, SVM, KNN, MLP, and Decision Tree with embeddings from models to be evaluated in a 10-fold cross validation manner to complete the trajectory-user linking task. And then Table 3 evaluates them using ACC and macro-F1.

In all learners, differentiable models, such as MLP and linear regressions work better than non-differentiable ones, such as decision trees with the embeddings. This is because differentiable models are more capable of handling continuous and uniform signal inputs, such as well-trained computational representations, than non-differentiable categorical values. Furthermore, decision trees can detect the importance of input fields but may fail if the fields are equally important. Hence, Table 3 shows that the proposed method learns a more uniform feature space.

The *movement* and *route* features obtain a higher score, while the *global* feature contributes less to representation learning which indicates that a general pattern of trajectories may be insufficient to describe spatial–temporal streaming data. However, this does not mean that the semantic information is good for nothing. Compared with a simple RNN fusion model, the semantic fusion model outperformed considerably. Mobility routines and intentions from functional zone differences are important in trajectory pattern mining. In addition, the attention module works better than RNNs. Among most machine learning approaches, Double Fusion models achieve better performance than others.

Clustering

We utilise embeddings from such models for K-means clustering directly in unsupervised learning. Then, they were evaluated using metrics. Fig. 6 shows the results of the clustering evaluation. A lower Davies–Bouldin index value indicates a better within-class clustering distribution, where all embedding elements of the same class in the feature space tend to be closer to their centre. And DouFu outperforms the other models, which discover more internal associations between trajectories of the same driver than others. A better Normalised mutual information score and adjusted Rand score indicate a better partitioning result for the trajectories of different drivers. The clustering evaluation results show that our method can provide an effective representation which is appropriate for driver similarity and pattern analysis.

The classification and clustering evaluation results demonstrate that the joint learning method with the multiple inputs mentioned above can improve the quality of embeddings. Inputs from different modalities can complete the task. However, they tend to focus only on one aspect of the data from a moving vehicle. Clearly, a comprehensive fusion model can capture more information. In addition, the attention module can compute the correlations between independent modules, which enhances the performance of DouFu.

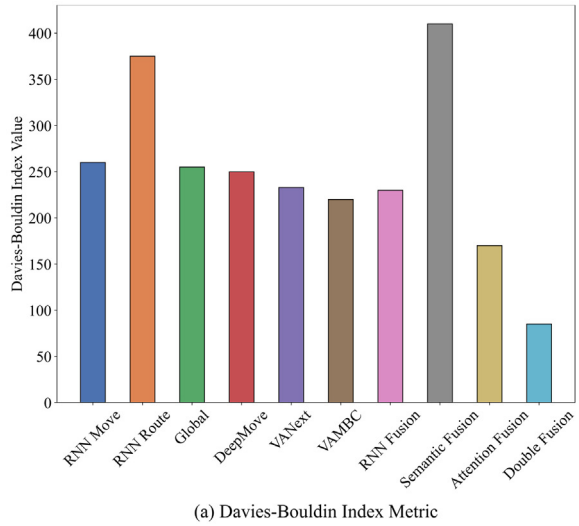
Fig. 7 shows the embedding TSNE results of the models with 5 selected users. The embeddings from the DouFu model provide a better understanding of the features of trajectories than the others.

Ablation analysis

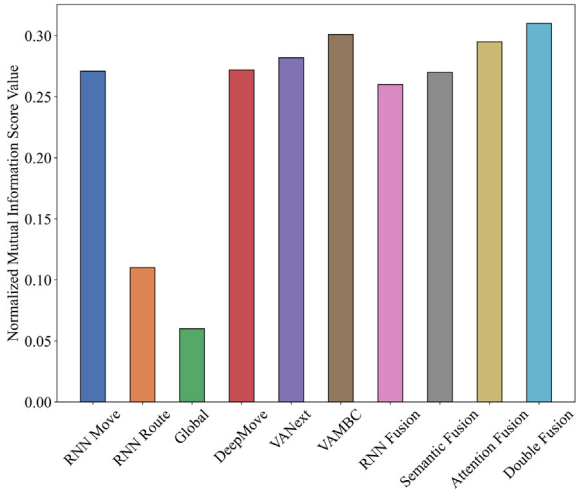
By analysing the results of the experimental evaluation, we present the effectiveness of each modal and fusion module. In the experimental step, we formulated some baseline algorithms by removing some components of DouFu. Generally, the *movement* feature is the primary responsibility for generating representations. Furthermore, the fusion of different modal inputs far exceeds that of the individual modals.

RNN Fusion only leverages *movement* and *route* features with RNNs. The *route* feature contributes to the improvement of the performance of the original model, which indicates that the simple *route* feature sequence contains additional information that helps generate a more effective representation.

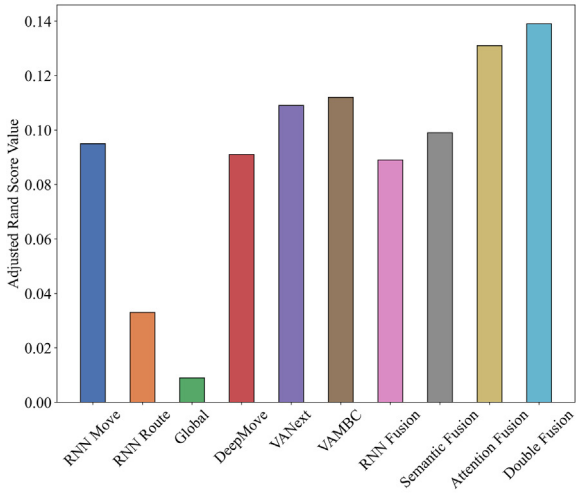
Semantic Fusion combines the three modal features without the attention mechanism. The results show that despite the poor



(a) Davies-Bouldin Index Metric



(b) Normalized Mutual Information Score Metric



(c) Adjusted Rand Score Metric

Fig. 6. Evaluation of trajectory clustering task. Our proposed method outperforms baseline algorithms.

performance of *global* features, a mixture of features considerably improves the performance. Therefore, multiple modality data can independently profile the trajectory information in different aspects to create a representation that is as complete as possible.

Attention Fusion applies the transformer module to process the time series features, including the *movement* and *route* features. The results indicate that the attention mechanism contributes to performance improvement. The attention module can extract more information from the time series feature with positional encoding than a traditional RNN.

Double Fusion uses attention weight fusion to combine the *movement* attention and *route* attention prior to feature fusion. The attention fusion module coordinates the *movement* feature and *route* features, which contributes to more effective representation generation.

Sensitivity analysis of the loss weight adjustment factors

Essentially, the error of the final fusion embedding that is used in the experimental evaluation must dominate in the training process. Therefore, the adjustment factors have to be much less than 1. In the sensitivity analysis, we performed a classification task using specific settings for α and β. As Table 4 shows, the loss weight adjustment factors α and β have an important impact on the performance of the model. In DouFu, the *route* attention weight is used to coordinate the *movement* feature. Therefore, the output of attention fusion contains the characteristics of the two features. Adjustment factor α should be larger than β to emphasise the importance of the combined feature.

6. Conclusion

In this study, we propose DouFu, a joint representation learning method for driving trajectories. DouFu applies multimodal learning and attention fusion to the trajectory and other data in order to generate comprehensive embeddings that are capable of capturing the internal characteristics. Empirically, we evaluate the models, and DouFu shows a better performance than the others on the classification and clustering tasks. The results show that in the absence of cameras or other sensor data, our approach can provide effective representations for trajectories using only trajectory and basic map information, demonstrating that route selection preferences and mobility intentions matter in trajectory pattern mining and user analysis.

In future work, some issues that need to be addressed. First, our model encodes the spatial location in terms of the relative position of the bounding box without considering position correlation. We plan to utilise specific representation learning methods to represent the spatial location of objects. Second, while processing the time series features, a simple transformer encoder and decoder were applied. A cross-modal transformer or a more complex model can be adopted to improve the performance of our model. Third, in this study, we aim to build a multimodal model with inexpensive spatial–temporal context data instead of applying additional sensor data such as cameras and driving recorders. Thus, as a multimodal model, various spatial–temporal context data can be exploited to produce a more efficient representation. Traffic condition data can be used to generate detailed time-varying road segment features and to convert raw trajectories into road segment sequences based on the corresponding time. Furthermore, satellite image data can also contribute to the construction of spatial–temporal context features.

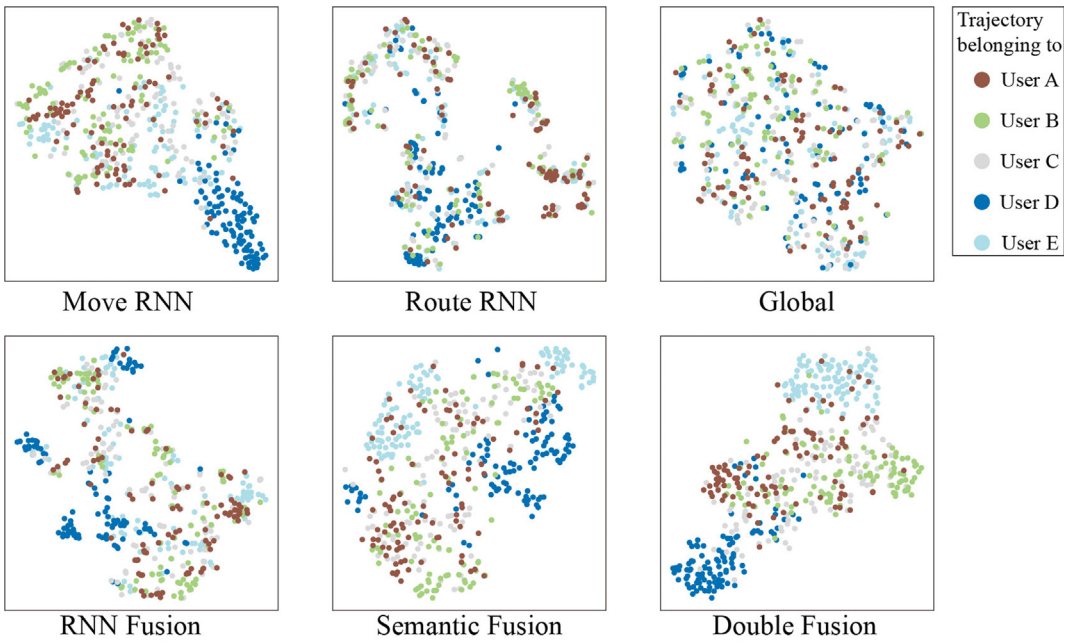


Fig. 7. TSNE visualisation of embeddings from selected users. The clustering of homochromatic dots and the separation between heterochromatic dots indicate the performance of trajectory representation learning.

Table 4
The result of the sensitivity analysis of loss weight adjustment factors. A larger adjustment factor α can improve the representation performance.

α	β	Linear regression		Ridge regression		Naive Bayes		SVM		KNN		MLP		Decision tree	
		ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1
0.001	0.001	41.02	40.68	40.02	34.55	38.12	34.98	30.08	29.07	32.76	30.22	39.44	35.24	23.02	23.24
0.001	0.005	40.72	40.08	39.97	34.02	37.98	34.05	29.32	28.13	32.33	30.12	38.98	35.12	22.87	22.98
0.005	0.001	43.81	42.03	41.45	35.33	39.01	35.85	31.10	29.47	34.41	32.34	40.56	36.85	24.47	24.61
0.005	0.005	40.33	39.91	39.58	33.42	37.55	33.58	29.55	27.42	32.01	30.72	38.24	34.78	21.85	22.09
0.010	0.010	38.02	37.89	37.42	32.48	36.84	33.12	28.08	26.88	31.62	30.14	37.79	33.56	22.05	22.33

CRediT authorship contribution statement

Han Wang: Conceptualization, Methodology, Writing – original draft. **Zhou Huang:** Supervision, Project administration, Writing – review & editing. **Xiao Zhou:** Investigation, Formal analysis, Writing – review & editing. **Ganmin Yin:** Data curation, Methodology, Validation. **Yi Bao:** Conceptualization, Validation, Writing – review & editing. **Yi Zhang:** Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

We acknowledge the financial support from the National Natural Science Foundation of China (42271471, 42201454, 41971331, 41830645), and China Postdoctoral Science Foundation (2022M710193). We also appreciate the detailed comments from the Editor and the anonymous reviewers.

References

[1] Y. Zheng, Trajectory data mining: an overview, ACM Trans. Intell. Syst. Technol. 6 (3) (2015) 1–41.
[2] X. Zhou, H. Wang, Z. Huang, Y. Bao, G. Zhou, Y. Liu, Identifying spatiotemporal characteristics and driving factors for road traffic CO2 emissions, Sci. Total Environ. 834 (2022) 155270.
[3] P. Wang, Y. Fu, J. Zhang, P. Wang, Y. Zheng, C. Aggarwal, You are how you drive: Peer and temporal-aware representation learning for driving behavior analysis, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 2457–2466.
[4] H. Cao, F. Xu, J. Sankaranarayanan, Y. Li, H. Samet, Habit2vec: Trajectory semantic embedding for living pattern recognition in population, IEEE Trans. Mob. Comput. 19 (5) (2019) 1096–1108.
[5] Q. Liu, S. Wu, L. Wang, T. Tan, Predicting the next location: A recurrent model with spatial and temporal contexts, in: AAAI, 2016.
[6] Y. Bao, Z. Huang, L. Li, Y. Wang, Y. Liu, A BiLSTM-CNN model for predicting users' next locations based on geotagged social media, Int. J. Geogr. Inf. Sci. 35 (4) (2021) 639–660.
[7] L. Wan, H. Wang, Y. Hong, R. Li, W. Chen, Z. Huang, Itourspot: a context-aware framework for next POI recommendation in location-based social networks, Int. J. Digit. Earth 15 (1) (2022) 1614–1636.
[8] X. Sun, Z. Huang, X. Peng, Y. Chen, Y. Liu, Building a model-based personalised recommendation approach for tourist attractions from geotagged social media data, Int. J. Digit. Earth 12 (6) (2019) 661–678.
[9] F. Zhou, Q. Gao, G. Trajcevski, K. Zhang, T. Zhong, F. Zhang, Trajectory-user linking via variational AutoEncoder, in: IJCAI, 2018, pp. 3212–3218.
[10] Y. Bengio, A. Courville, P. Vincent, Representation learning: A review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828.
[11] W. Dong, J. Li, R. Yao, C. Li, T. Yuan, L. Wang, Characterizing driving styles with deep learning, 2016, arXiv preprint arXiv:1607.03611.

[12] W. Dong, T. Yuan, K. Yang, C. Li, S. Zhang, Autoencoder regularized network for driving style representation learning, in: IJCAI, 2017.
[13] T. Kieu, B. Yang, C. Guo, C.S. Jensen, Distinguishing trajectories from different drivers using incompletely labeled trajectories, in: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, 2018, pp. 863–872.
[14] S. Liu, Y. Liu, L.M. Ni, J. Fan, M. Li, Towards mobility-based clustering, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2010, pp. 919–928.
[15] X. Li, K. Zhao, G. Cong, C.S. Jensen, W. Wei, Deep representation learning for trajectory similarity computation, in: 2018 IEEE 34th International Conference on Data Engineering, ICDE, 2018, pp. 617–628.
[16] X. Gong, Z. Huang, Y. Wang, L. Wu, Y. Liu, High-performance spatiotemporal trajectory matching across heterogeneous data sources, Future Gener. Comput. Syst. 105 (2020) 148–161.
[17] L. Wu, L. Yang, Z. Huang, Y. Wang, Y. Chai, X. Peng, Y. Liu, Inferring demographics from human trajectories and geographical context, Comput. Environ. Urban Syst. 77 (2019) 101368.
[18] J. Letchner, J. Krumm, E. Horvitz, Trip router with individualized preferences (trip): Incorporating personalization into route planning, in: AAAI, 2006, pp. 1795–1800.
[19] H. Ren, M. Pan, Y. Li, X. Zhou, J. Luo, ST-SiameseNet: Spatio-temporal siamese networks for human mobility signature identification, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020, pp. 1306–1315.
[20] Q. Gao, F. Zhou, K. Zhang, G. Trajcevski, X. Luo, F. Zhang, Identifying human mobility via trajectory embeddings, in: IJCAI, Vol. 17, 2017, pp. 1689–1695.
[21] J.J.-C. Ying, W.-C. Lee, T.-C. Weng, V.S. Tseng, Semantic trajectory mining for location prediction, in: Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 2011, pp. 34–43.
[22] J.J.-C. Ying, W.-C. Lee, V.S. Tseng, Mining geographic-temporal-semantic patterns in trajectories for location prediction, ACM Trans. Intell. Syst. Technol. 5 (1) (2014) 1–33.
[23] T.-Y. Fu, W.-C. Lee, TremBR: Exploring road networks for trajectory representation learning, ACM Trans. Intell. Syst. Technol. 11 (1) (2020) 1–25.
[24] F. Zhou, Y. Dai, Q. Gao, P. Wang, T. Zhong, Self-supervised human mobility learning for next location prediction and trajectory classification, Knowl.-Based Syst. 228 (2021) 107214.
[25] M. Siami, M. Naderpour, J. Lu, A mobile telematics pattern recognition framework for driving behavior extraction, IEEE Trans. Intell. Transp. Syst. 22 (3) (2020) 1459–1472.
[26] M. Yue, Y.-Y. Chiang, C. Shahabi, VAMBC: A variational approach for mobility behavior clustering, in: Machine Learning and Knowledge Discovery in Databases. Applied Data Science Track, Springer International Publishing, Cham, 2021, pp. 453–469.
[27] M. Tabatabaie, S. He, X. Yang, Reinforced feature extraction and multi-resolution learning for driver mobility fingerprint identification, in: Proceedings of the 29th International Conference on Advances in Geographic Information Systems, 2021, pp. 69–80.
[28] T. Baltruaitis, C. Ahuja, L.-P. Morency, Multimodal machine learning: A survey and taxonomy, IEEE Trans. Pattern Anal. Mach. Intell. 41 (2) (2018) 423–443.
[29] M. Wllmer, A. Metallinou, F. Eyben, B. Schuller, S. Narayanan, Context-sensitive multimodal emotion recognition from speech and facial expression using bidirectional lstm modeling, in: Proc. INTERSPEECH 2010, Makuhari, Japan, 2010, pp. 2362–2365.
[30] S.E. Kahou, X. Bouthillier, P. Lamblin, C. Gulcehre, V. Michalski, K. Konda, S. Jean, P. Froumenty, Y. Dauphin, N. Boulanger-Lewandowski, R. Chandias Ferrari, M. Mirza, D. Warde-Farley, A. Courville, P. Vincent, R. Memisevic, C. Pal, Y. Bengio, EmoNets: Multimodal deep learning approaches for emotion recognition in video, J. Multimodal User Interfaces 10 (2) (2016) 99–111.
[31] R. Xu, C. Xiong, W. Chen, J. Corso, Jointly modeling deep video and compositional text to bridge vision and language in a unified framework, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2015, p. 29.
[32] H. Yu, J. Wang, Z. Huang, Y. Yang, W. Xu, Video paragraph captioning using hierarchical recurrent neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4584–4593.
[33] T. Phan-Minh, E. Grigore, F. Boulton, O. Beijbom, E.M. Wolff, CoverNet: Multimodal behavior prediction using trajectory sets, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2020, pp. 14062–14071.
[34] H. Cui, V. Radosavljevic, F.-C. Chou, T.-H. Lin, T. Nguyen, T.-K. Huang, J. Schneider, N. Djuric, Multimodal trajectory predictions for autonomous driving using deep convolutional networks, in: 2019 International Conference on Robotics and Automation, ICRA, 2019, pp. 2090–2096.
[35] K. Chen, Y. Yu, P. Song, X. Tang, L. Cao, X. Tong, Find you if you drive: Inferring home locations for vehicles with surveillance camera data, Knowl.-Based Syst. 196 (2020) 105766.
[36] X. Huang, S.G. McGill, J.A. DeCastro, B. Williams, L. Fletcher, J. Leonard, G. Rosman, DiversityGAN: Diversity-aware vehicle motion prediction via latent semantic sampling, IEEE Robot. Autom. Lett. 5 (2020) 5089–5096.
[37] J. Feng, Y. Li, C. Zhang, F. Sun, F. Meng, A. Guo, D. Jin, Deepmove: Predicting human mobility with attentional recurrent networks, in: Proceedings of the 2018 World Wide Web Conference, 2018, pp. 1459–1468.
[38] Q. Gao, F. Zhou, G. Trajcevski, K. Zhang, T. Zhong, F. Zhang, Predicting human mobility via variational attention, in: The World Wide Web Conference, WWW '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 2750–2756.
[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, u. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.
[40] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: NAACL-HLT (1), 2019.
[41] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, Q.V. Le, Xlnet: Generalized autoregressive pretraining for language understanding, in: NeurIPS, 2019.
[42] H. Yao, X. Tang, H. Wei, G. Zheng, Z. Li, Revisiting spatial–temporal similarity: A deep learning framework for traffic prediction, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 5668–5675.
[43] B. Yang, Y. Kang, Y. Yuan, X. Huang, H. Li, ST-LBAGAN: Spatio-temporal learnable bidirectional attention generative adversarial networks for missing traffic data imputation, Knowl.-Based Syst. 215 (2021) 106705.
[44] Z. Yu, J. Yu, Y. Cui, D. Tao, Q. Tian, Deep modular co-attention networks for visual question answering, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 6274–6283.
[45] H. Tan, M. Bansal, Lxmert: Learning cross-modality encoder representations from transformers, 2019, arXiv preprint arXiv:1908.07490.
[46] G. Yin, Z. Huang, Y. Bao, H. Wang, L. Li, X. Ma, Y. Zhang, ConvGCN-RF: A hybrid learning model for commuting flow prediction considering geographical semantics and neighborhood effects, GeoInformatica (2022).
[47] P. Gong, B. Chen, X. Li, H. Liu, J. Wang, Y. Bai, J. Chen, X. Chen, L. Fang, S. Feng, Y. Feng, Y. Gong, H. Gu, H. Huang, X. Huang, H. Jiao, Y. Kang, G. Lei, A. Li, X. Li, X. Li, Y. Li, Z. Li, Z. Li, C. Liu, C. Liu, M. Liu, S. Liu, W. Mao, C. Miao, H. Ni, Q. Pan, S. Qi, Z. Ren, Z. Shan, S. Shen, M. Shi, Y. Song, M. Su, H. Ping Suen, B. Sun, F. Sun, J. Sun, L. Sun, W. Sun, T. Tian, X. Tong, Y. Tseng, Y. Tu, H. Wang, L. Wang, X. Wang, Z. Wang, T. Wu, Y. Xie, J. Yang, J. Yang, M. Yuan, W. Yue, H. Zeng, K. Zhang, N. Zhang, T. Zhang, Y. Zhang, F. Zhao, Y. Zheng, Q. Zhou, N. Clinton, Z. Zhu, B. Xu, Mapping essential urban land use categories in China (EULUC-China): preliminary results for 2018, Sci. Bull. 65 (3) (2020) 182–187.
[48] Z. Huang, H. Qi, C. Kang, Y. Su, Y. Liu, An ensemble learning approach for urban land use mapping based on remote sensing imagery and social sensing data, Remote Sens. 12 (19) (2020) 3254.
[49] Y. Feng, Z. Huang, Y. Wang, L. Wan, Y. Liu, Y. Zhang, X. Shan, An SOE-based learning framework using multisource big data for identifying urban functional zones, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 14 (2021) 7336–7348.
[50] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, in: ICLR, 2013.
[51] T.N. Kipf, M. Welling, Variational graph auto-encoders, 2016, arXiv preprint arXiv:1611.07308.