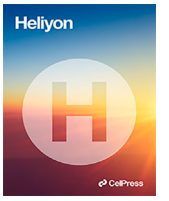
Heliyon 10 (2024) e35572

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon



Research article

Travel-mode inference based on GPS-trajectory data through multi-scale mixed attention mechanism

Xiaohui Pei [a,b], Xianjun Yang [b,*], Tao Wang [b], Zenghui Ding [b], Yang Xu [b], Lin Jia [b], Yining Sun [b]

[a] University of Science and Technology of China, No. 96, JinZhai Road Baohe District, Hefei, 230026, Anhui, China
[b] Hefei Institutes of Physical Science, Chinese Academy of Sciences, No. 350, Shushanhu Road, Hefei, 230031, Anhui, China

ARTICLE INFO

Dataset link: https://github.com/peixiaoh/Travel-Mode-Inference

Keywords:
Travel-mode inference
GPS-trajectory data
Multi-scale convolution
Attention mechanism

ABSTRACT

Identifying travel modes is essential for modern urban transportation planning and management. Recent advancements in data collection, especially those involving Global Positioning System (GPS) technology, offer promising opportunities for rapidly and accurately inferring users' travel modes. This study presents an innovative method for inferring travel modes from GPS trajectory data. The method utilizes multi-scale convolutional techniques to capture and analyze both temporal and spatial information of the data, thereby revealing the underlying spatiotemporal relationships inherent in user movement and behavior patterns. In addition, an attention mechanism is integrated into the model to enable autonomous learning. This mechanism enhances the model's capacity to identify and emphasize key information across different time periods and spatial locations, thus improving the accuracy of travel mode inference. Evaluation on the open-source GPS trajectory dataset, GeoLife, demonstrates that the proposed method attained an accuracy of 83.3%. This result highlights the effectiveness of the method, demonstrating that the model can more accurately understand and predict user travel modes through the integration of multi-scale convolutional technologies and attention mechanisms.

1. Introduction

Identifying travel modes is essential for modern urban transportation planning and management [1]. Presently, cities face significant challenges including escalating traffic congestion, frequent traffic accidents and inadequate transportation systems. These challenges disrupt citizens' daily lives and impact urban landscapes' sustainability and economic competitiveness. Hence, understanding the distribution and trends of various travel modes has become imperative for urban planners, as it can help them make informed decisions to enhance the city's transportation infrastructure, mitigate congestion issues, and improve transportation efficiency. For example, if travel mode inference indicates that residents in a specific area prefer bicycles, city planners can prioritize constructing bicycle lanes, thus fostering sustainability, reducing air pollution, and enhancing residents' quality of life. Moreover, travel mode inference assists planners in identifying the timing and locations for essential infrastructure such as new transportation hubs, bus stops, bike rental stations, and walking trails. This detailed information ensures that the city's transportation system operates more efficiently, diminishes traffic accidents, and offers more convenient travel alternatives. However, traditional travel mode data collec-

* Corresponding author.
E-mail address: xjyang@iim.ac.cn (X. Yang).

https://doi.org/10.1016/j.heliyon.2024.e35572
Received 28 December 2023; Received in revised form 31 July 2024; Accepted 31 July 2024

Available online 5 August 2024

tion methods often rely on face-to-face or online surveys [2], frequently yielding low response rates. Furthermore, due to limitations in individual memory and subjective perceptions, the provided information may be incomplete or inaccurate. Consequently, these methods are time-consuming and ineffective, potentially leading decision-makers to make suboptimal planning decisions.

In recent decades, advanced data collection methods, particularly civilian and smartphone devices equipped with GPS [3], have accumulated a wealth of GPS trajectory data, opening new avenues for travel mode inference. These datasets record the movement trajectories of individuals at different times and locations, typically comprising latitude, longitude, timestamps and other relevant information. Despite the increasing availability and scale of GPS trajectory data, accurately discerning travel modes within it remains a significant challenge. This challenge stems from the myriad of travel modes encountered in real-world scenarios, covering walking, cycling, driving, public transportation, etc. Individuals often transition between multiple travel modes at different times and locations, so it is challenging to infer travel modes accurately. Furthermore, GPS trajectory data is prone to various sources of noise, such as positioning errors, signal interference, and data missing, which can lead to the misidentification of travel modes [4]. Moreover, different travel modes may exhibit similar trajectory patterns; for example, walking and cycling may show geographical resemblances despite their disparate modes of transportation.

In recent years, researchers have increasingly focused on utilizing GPS trajectory data to infer users' travel modes. They are mainly divided into two categories: traditional feature engineering and the other is deep learning. Traditional feature engineering methods involve two core steps. First, features are manually extracted based on domain knowledge. These characteristics include but are not limited to motion and displacement characteristics, such as velocity, acceleration, and frequency domain characteristics, such as spectrum diffusion [5–7]. These features are calculated through descriptive statistics, offer insights into various aspects and modes of trajectories. Following feature extraction, a supervised learning method is used to build a machine-learning model utilizing the extracted features. Common machine-learning algorithms utilized include decision trees, random forests, and support vector machine [8,9]. However, traditional feature engineering requires specialized domain knowledge and is susceptible to subjective feature selection, as well as vulnerability to varying traffic and environmental conditions. For example, a common feature engineering method involves using the maximum speed in GPS trajectory data as a decision feature. Nevertheless, it is worth pointing out that during traffic congestion, maximum speeds may appear across different travel modes, such as car, bicycle and walking modes, leading to inaccuracies and ambiguities in feature interpretation. Alternatively, another domain expert might choose to utilize the first three speeds and accelerations from the user's GPS trajectory data to compensate for the lack of traffic situation information.

The other category is deep learning methods, which typically utilize deep learning frameworks such as convolutional neural network (CNN) to automatically extract features from GPS trajectory data [6,10]. The advantage of this method is that it does not require manual feature extraction, but allows the neural network to automatically learn and extract information from GPS trajectories, thereby circumventing the potential interference of subjective biases. For example, Endo et al. [11] used a fully connected network to learn information about temporal depth features and transportation mode classification. While Wang et al. [12] adopted a CNN model that combined deep features with hand-designed features, and used a fully connected network for classification tasks. Additionally, Dabiri and Heaslip [13] used basic kinematic and behavioral features combined with deep features derived from basic input features to classify modes.

Although, current deep learning methods have made notable progress in solving the problem of subjective bias caused by manual feature extraction in traditional methods, few studies have paid attention to the user's short-term movement behavior and long-term movement trends inherent in GPS trajectory data, resulting in incomplete feature extraction. Furthermore, GPS trajectory data contains a large amount of information, but only a fraction of it is relevant to the user's movement mode. To solve these problems, this study proposes an innovative method for inferring travel modes from GPS trajectory data. This method first uses multi-scale convolution techniques to capture and analyze both temporal and spatial information of the data to reveal the underlying spatiotemporal relationships inherent in user movement and behavior patterns. Subsequently, an attention mechanism is used to automatically identify and emphasize relevant information across different time periods and spatial locations. This approach aims to improve the model's ability to concentrate on significant features while reducing its susceptibility to external noise. Experimental results on the open-source GeoLife trajectory dataset show a significant improvement in the model's accuracy in inferring travel modes. The main contributions of this study are summarized as follows:

(1) The proposed travel mode inference methodology utilizes multi-scale convolution to leverage its benefits in comprehending spatiotemporal correlations within user movement and behavior trends, thereby enhancing the accuracy of travel mode inference.
(2) Incorporation of an attention mechanism enables autonomous acquisition and prioritization of relevant information across various time epochs and spatial locations. This integration promotes the model's adaptive response to different contextual scenarios, strengthens focus on critical features, and enhances the model's flexibility and generalization capabilities.

The rest of the paper is as follows: In Section 2, we present related work. Section 3 provides a detailed introduction to the framework, including residual networks, multi-scale convolutions, attention mechanisms, and the CNN configuration adopted for our application. Section 4 evaluates our proposed CNN architecture on the GeoLife trajectory dataset and compares our results with classic machine-learning algorithms and previous studies. Section 5 discusses the implications, limitations, and future work. Finally, Section 6 concludes the paper.

2. Related work

Deep learning methods, grounded in neural networks, have made significant strides in domains such as image processing and natural language processing. In terms of image processing, deep learning technologies have demonstrated capabilities ranging from traditional image recognition and classification to more complex tasks such as object detection and segmentation. Similarly, in the field of natural language processing, these technologies leverage robust neural network architectures to tackle a spectrum of tasks, including text classification, sentiment analysis, machine translation and dialogue systems. Advancements in these technologies not only improve machines' ability to understand language, but also bring greater efficiency and accuracy to applications such as information retrieval and intelligent customer service. Beyond image processing and natural language processing, deep learning technologies are widely applied across various other domains. One of them is the processing and analysis of GPS trajectory data [6,10].

In past research, the processing of GPS trajectory data usually focused on simple feature extraction and pattern recognition, while ignoring the more detailed and complex user behavior information [11,12]. With the continuous development of deep learning technology, researchers have begun to realize that GPS trajectories contain more information, including users' short-term movement behaviors and long-term movement trends [13]. This information is pivotal for comprehending users' travel patterns, behavioral habits, and preferences. Multi-scale convolution, serving as an effective method for feature extraction, holds significant promise in this domain. By performing convolution operations on input data across different scales, multi-scale convolution can capture feature information at different scales in the data [14,15]. Furthermore, GPS signals often contain a large amount of information redundancy, with only a small part being related to the user's movement pattern. To improve the model's processing effect on trajectory data, researchers have introduced an attention mechanism to enable the neural network to focus on the most relevant parts when processing input data, thereby boosting the model's attention to important information, and improving the model robustness and accuracy [16,17]. Despite the attention mechanism has shown significant advantages in improving the model's attention to important information, it introduces additional parameters and computational burden, resulting in an increase in the complexity of the model [18,19]. This complexity increase may cause the model training and inference process to become more time-consuming, and may limit the scope of the model's application in resource-constrained environments.

This study combines multi-scale convolution and lightweight efficient channel attention (ECA) [20] to build a travel mode inference model based on GPS trajectory data. During the processing of GPS trajectory data, this model can more accurately infer travel modes by fully utilizing the temporal and spatial features in the data and focusing on the most relevant information. In contrast to traditional methods, this model exhibits superior accuracy and robustness, and can be better applied to various practical scenarios, such as intelligent traffic management, urban planning and other fields. The following section provides a detailed description of our methodology.

3. Method

3.1. Framework

This study uses a deep residual network (ResNet) [21] as the backbone to construct a travel mode inference model that integrates multi-scale convolution and attention mechanisms. The model consists of 1 convolutional layer, 3 residual networks with different convolution kernel sizes and 1 fully connected layer. The overall structure can be seen in Fig. 1. First, we use convolutional layers to perform feature transformation on GPS trajectory data and then perform multi-scale feature extraction through residual networks with three different convolution kernel sizes. To enhance the residual module's attention to critical features and reduce its sensitivity to noise, an ECA [20] module is embedded after each residual module. Finally, the multi-scale features extracted by the residual network are fused to form a deep feature representation. The fused features are processed through the global average pooling (GAP) layer to obtain a feature vector containing spatial information. Finally, the corresponding travel mode inference results are output through the classification layer with the SoftMax function. A detailed introduction to each part is given below.

3.2. Deep residual network

In traditional neural networks, the phenomenon known as the "vanishing gradient" often occurs as the number of network layers increases, resulting in the training process being extremely difficult. To address this issue, He et al. [21] introduced ResNet, which employs a concept termed "residual learning". Unlike traditional neural networks, ResNet adds the previous layer's output to the current layer's input to learn the residual between the previous layer's output and the current input. The typical ResNet structure is shown in Fig. 2. The advantage is that this structure allows the gradient to be more efficiently passed to the first few layers of the network during back propagation, thereby effectively reducing the gradient vanishing problem. Specifically, the skip connection can maintain the stability of information flow during training and avoid gradient attenuation caused by the network being too deep. In addition, through multiple stacking, the residual module can gradually learn more complex and high-level feature representations, thereby improving the classification performance of the model. Therefore, we chose to use ResNet as the backbone framework in this study.

To some extent, the depth of a network is positively correlated with its learning ability. However, there is a trade-off between the depth of the network and the need for training data, and a network that is too deep may lead to model overfitting. Therefore, each of the three residual networks designed in this study only contains three residual blocks. At the same time, in these three residual

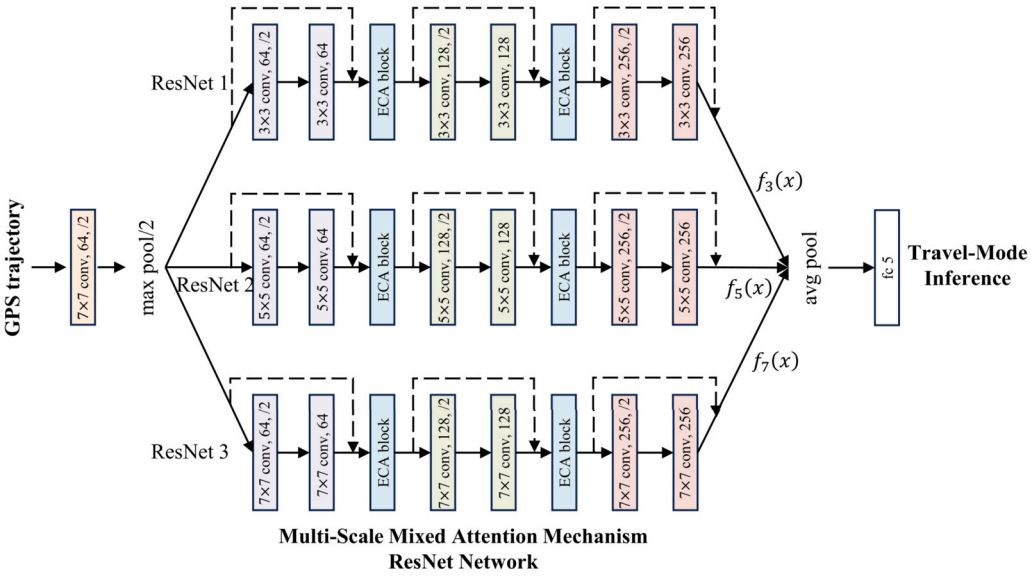


Fig. 1. Overall network structure.

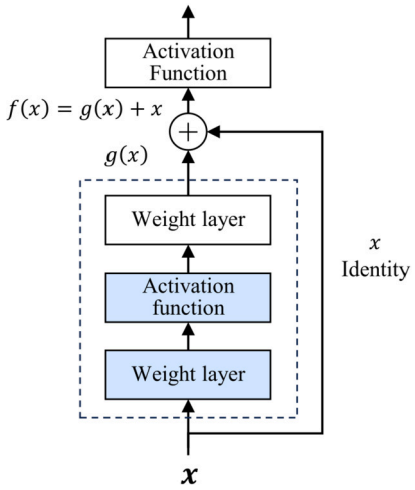


Fig. 2. ResNet structure.

networks, except for the different convolution kernel sizes in the residual block, the depth of the convolution kernel at the same position is the same. The depth of the convolution kernel of the first residual block is 64, the convolution kernel depth of the second residual block is 128, and the convolution kernel depth of the third residual block is 256. The stride size of each residual block is set to 2, the convolution kernel is filled in the same way, and the activation function uses ReLU. Furthermore, each residual block consists of a convolutional layer, a batch normalization layer, and an activation layer. An ECA module is introduced between the first residual block and the second residual block, and between the second residual block and the third residual block to enhance the model's attention to critical features, and reduce sensitivity to noise. The primary purpose of applying the batch normalization layer in the residual block is to further deal with the gradient disappearance or gradient explosion problems that may occur during network training, thereby improving the stability and convergence speed.

3.3. Multi-scale convolution

In CNN, the size of the convolution kernel greatly affects the network's perception and feature extraction capabilities. Small-sized convolution kernels usually have small perceptual areas, which enables them to effectively capture local features and tiny details in the input data. This is useful for many tasks such as edge detection and texture analysis in image recognition [22,23]. However, a potential problem with small-sized convolution kernels is that they cannot fully capture the global characteristics of the input data, which may lead to the loss of contextual information. On the contrary, large-size convolution kernels have a larger perceptual area and can cover more input data, thereby helping to capture global features and contextual information. In the inference of user

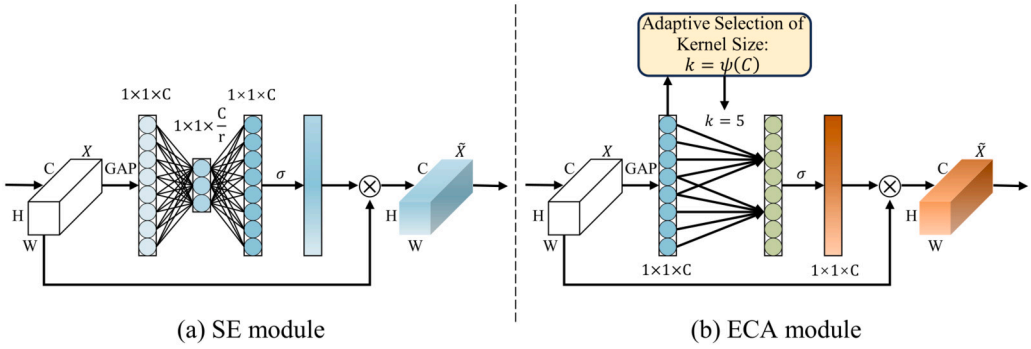


Fig. 3. The basic structure of the ECA module. (a) SE module; (b) ECA module.

travel mode based on GPS trajectory data, the user's GPS trajectory data may include instantaneous movement behaviors, such as local features such as parking and turning, as well as long-term movement trends, such as the entire travel route. To effectively capture these features of different scales and improve the accuracy of travel mode inference, this study built a multi-scale fusion framework. In this multi-scale fusion framework, three convolution kernel residual networks of different sizes are used to extract features. The dimensions of these convolution kernels are 3×1, 5×1 and 7×1, respectively. Each convolution kernel can capture information at different scales. Each residual network consists of 3 residual blocks, and each residual block contains the same pooling layer and batch normalization layer, which are used to extract features of the same scale. Finally, the features of convolutional residual networks as different scales are fused to generate a multi-scale fusion feature. This feature contains information extracted from convolution kernels of different scales, considering global features and contextual information while retaining local details. For user travel mode inference based on GPS trajectory data, this feature enables the model to consider both instantaneous movement behavior and long-term movement trends, thereby improving the accuracy and robustness of the model.

3.4. ECA attention mechanism

GPS trajectory data usually contain a large amount of redundant information, which interferes with effective focus on critical features while increasing the model's sensitivity to noise. To deal with this problem, this study introduces the ECA attention mechanism to learn and select the information that should be focused on at time steps or position points. The ECA attention mechanism is based on the Squeeze-and-Excitation (SE) [24] module, but unlike the SE module, ECA can effectively achieve cross-channel information without reducing the channel dimension. Fig. 3(a) and 3(b) show the structural differences between the SE and ECA modules. Assume that an input image $x \in V^{H \times W \times C}$ is given, and one-dimensional feature information $y \in V^{1 \times 1 \times C}$ is obtained through the GAP operation. Then, set a parameter k, interact with each channel and its adjacent k channels for cross-channel local information, and obtain the weight $w \in V^{1 \times 1 \times C}$. Finally, perform a dot product operation on these weights and the corresponding dimensions of the input features to obtain the output feature map $p' \in V^{1 \times 1 \times C}$.

The ECA module can not only effectively learn the attention channel feature layer information, but also reduce the complexity of the model. The calculation formula for channel weight information is as follows:

$$w_i = \sigma\left[\sum_{j=1}^{k} w_i^j y_i^j\right], y_i \in \Omega_i^k \tag{1}$$

where y_i represents the channel feature information after pooling, w_i represents the channel weight value, k represents the range of local cross-channel interaction, σ represents the activation function, Ω_i^k represents the $k-th$ domain channel of y_i^j. ECA achieves local information interaction between fast channels by maintaining the dimensionality without reducing the dimension and adaptively selecting the size of the one-dimensional convolution kernel, allowing the neural network to focus effectively on key features in a wide range of areas. This study introduces an ECA module after each residual block to remove redundant information in GPS signal data and enhance the model's focus on critical features.

4. Experiments and results

4.1. Dataset

This study uses the open-source GPS trajectory dataset GeoLife [25] provided by Microsoft Research Asia to validate the proposed method. The dataset covers GPS trajectory data collected by 182 volunteers over five years from April 2007 to August 2012. Each trajectory in this dataset is represented as a sequence of GPS records, with 69 users providing travel mode information. While the dataset contains annotations for various transportation modes, this study focuses on ground transportation modes. Following the user guide related to the dataset [25], we merged the two labels of taxi and car into "driving", while rail-based transportation modes (such as subway, train and railway) were all classified as "train". Therefore, we consider five transportation modes: walking, cycling,

Table 1
Number of samples and average values of four motion features under different transportation modes.

Transportation Modes	Speed (m/s)	Acceleration (m/s^2)	Jerk (m/s^3)	Load-Bearing Rate (degree)	Count
Walking	0.835	0.01	-0.015	2.221	7842
Cycling	1.828	0.03	-0.038	0.035	3960
Bus	3.736	-0.009	-0.009	0.905	5184
Driving	6.703	0.026	-0.019	2.707	3273
Train	10.577	0.011	-0.019	1.466	3311
Total					23570

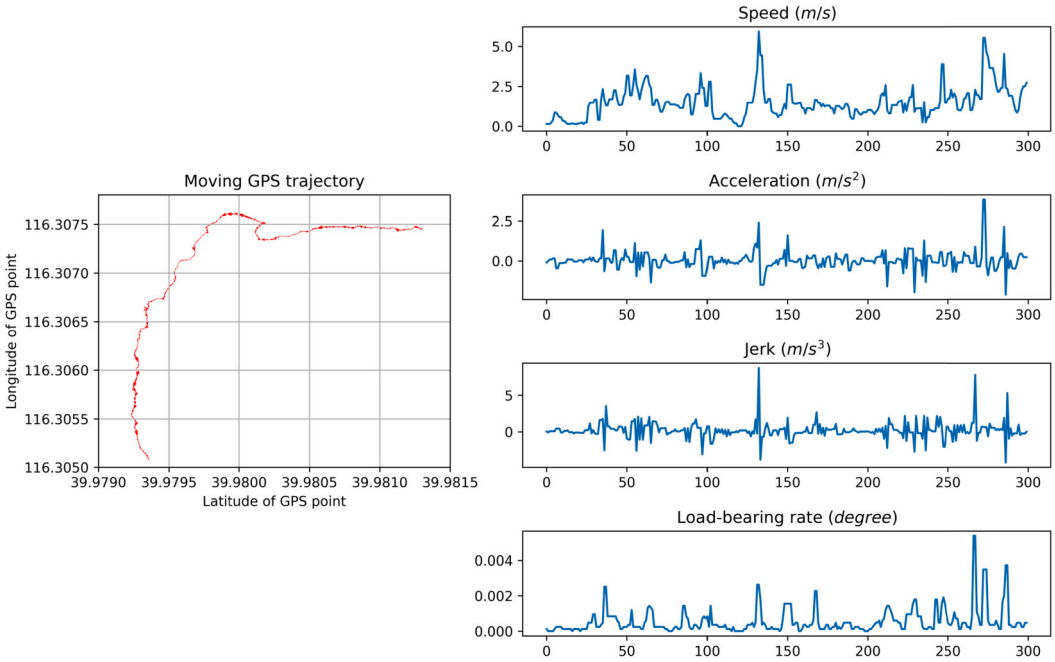


Fig. 4. Example of four motion features extracted from GPS trajectories.

bus, driving and train. After aligning each user's tag file with their corresponding GPS trajectory file, the GPS trajectory is segmented based on changes in transportation mode (i.e., each segment of the GPS trajectory exclusively contains one transportation label), and segments of the GPS trajectory exceeding 30 minutes are further partitioned into different trips, that is, a GPS trajectory segment longer than 30 minutes is subdivided into multiple sub-segments, each lasting less than or equal to 30 minutes.

To obtain the motion characteristics of these GPS trajectories, we followed the preprocessing method used in previous research [13] and extracted four motion signals: speed, acceleration, jerk, and load-bearing rate. Due to factors such as GPS positioning errors, signal interference, and data missing, the extracted motion features contain noise. To address this, we used the Savitzky-Golay filter [26] for denoising before further processing. Subsequently, the motion signals related to each trajectory segment are superimposed to construct a four-channel structure, where each channel represents speed, acceleration, jerk, and load-bearing rate. Table 1 shows the number of samples and average values of the four motion signals under different transportation modes. Fig. 4 shows an example of four motion signals extracted from GPS trajectories. To effectively evaluate this method, we divided the data set into a training set, a validation set, and a test set, of which the test set accounted for 20%, and the training set and validation set accounted for 80%.

4.2. Experimental setup

The experimental platform configuration for this study is as follows: the operating system used is Ubuntu 20.04.6 LTS. The hardware configuration includes a 12th Gen Intel(R) Core (TM) i9-12900K processor (3.9GHz, multi-core architecture), 64GB of RAM, and two NVIDIA GeForce RTX 3080 GPUs, each with 24GB of VRAM. The software employed includes the Python 3.9 programming language and the PyTorch 2.0.1 deep learning framework. During the model training phase, we use the GPU in the research platform for fast training. In terms of hyperparameter settings, we take into account previous experience and common default values. We chose the Adam optimization algorithm to minimize the cross-entropy loss and set the learning rate to 0.001. The Adam optimization algorithm combines the characteristics of momentum and adaptive learning rate, and is usually able to quickly converge to satisfactory results during training. Setting the learning rate to 0.001 is a common initial value and is suitable for a wide range of deep learning

Table 2
Performance comparison with our method and traditional machine learning.

Methods	Macro Precision	Macro Recall	Macro F1-Score	Accuracy
Random Forest	80.2	77.0	78.3	80.4
Decision Tree	75.2	72.0	73.2	76.0
SVM	51.5	45.2	44.7	55.0
KNN	59.5	55.6	56.5	62.4
MLP	59.4	54.5	52.2	62.7
DNN	67.3	65.6	66.1	70.5
LeNet5	76.5	73.5	74.6	77.2
The proposed	84.0	80.9	82.2	83.3

Table 3
Performance comparison of our method with other work.

Work	Approach	Year	Accuracy
Dabiri et al. [10]	SECA	2019	77.2
Güvensan and Asci [27]	LSTM	2019	81.7
Yu et al. [28]	LSTM-based DNN	2021	82.1
Namdarpour et al. [29]	SVMs and NN	2021	83.4
Zhu et al. [30]	GRU-based DNN	2022	82.1
Zheng et al. [31]	STPC-Net	2022	80.7
Cardoso-Pereira et al. [32]	ML	2022	74.9
The proposed	MS-ECA-ResNet	-	83.3

tasks. Batch size determines the number of samples used for each parameter update. A larger batch size usually speeds up training and improves model stability. In this study, we set the Batch size to 64. Considering the small amount of data, we limit the number of iterations to 100. At the same time, to avoid overfitting, we adopt an early stopping strategy. Once the model's performance on the validation set no longer improves, we end training early to prevent further overfitting. In the model evaluation stage, we use macro precision, macro recall, macro F1-score and accuracy, as evaluation indicators to evaluate the overall performance of the proposed model in inferring different transportation modes.

4.3. Comparison with traditional machine learning methods

In this article, we first conduct a detailed comparison between this method and traditional machine learning methods, including five widely used machine learning algorithms (i.e. random forest, decision tree, SVM, KNN, MLP) and two basic deep learning models (DNN, LeNet5). For the five machine learning algorithms, we use features manually extracted from GPS trajectories in [13] as input, and for two basic deep learning models, we use the four motion signals derived from GPS trajectories as input. We use multiple evaluation indicators such as accuracy, macro precision, macro recall, and macro F1-score to evaluate the overall performance comprehensively. The results are detailed in Table 2. Among those methods, random forest performs best, with its macro precision, macro recall, macro F1-score, and accuracy reaching 80.2%, 77.0%, 78.3% and 80.4%, respectively. Followed by the LeNet5, its corresponding indicators reached 76.5%, 73.5%, 74.6% and 77.2%, respectively. Compared with those methods, the method proposed in this article shows apparent advantages. Its macro precision, macro recall, macro F1-score and accuracy have all been significantly improved. Compared with the best-performing random forest, they have increased by 3.8%, 3.9%, 3.9% and 2.9%, respectively. This shows that the method has significant performance advantages and greater potential in processing related tasks.

4.4. Comparison with existing work

To comprehensively confirm the effectiveness of this method, we conducted a literature review and selected some representative works for comparison with our work. It is noteworthy that, given the divergence in the evaluation indicators employed across these studies, this study utilizes the accuracy metric consistently applied in all referenced studies for comprehensive evaluation. Table 3 shows the comparison results of this method with existing in terms of accuracy. As can be seen from Table 3, compared with existing work, this research method achieves equivalent or better performance, and it does not require manual feature extraction, avoiding the fact that traditional feature engineering requires not only professional domain knowledge, but also involves subjective feature selection, as well as vulnerability to traffic and environmental conditions.

4.5. Ablation experiment

To ensure the effectiveness of the module design, we used a control variable approach by gradually adding different modules and evaluating the contribution of each module to the overall performance. As the evaluation baseline, we also report the result of integrating different modules on the based deep learning LeNet5. It is worth pointing out that, to ensure a fair evaluation, we used the same parameter configuration in these models except for added/removed modules. For the single-scale convolution model, we use a

Table 4
Ablation experiment.

Models	Macro Precision	Macro Recall	Macro F1-Score	Accuracy	FLOPs	Params
LeNet5	76.5	73.5	74.6	77.2	0.1466M	0.0461M
ECA-LeNet5	76.7	73.6	74.8	77.3	0.1470M	0.0461M
MS-LeNet5	76.7	74.0	75.1	77.5	0.3324M	0.1163M
MS-ECA-LeNet5	77.1	74.2	75.3	77.7	0.3335M	0.1163M
ResNet	80.9	77.9	79.1	80.8	6.3232M	0.4442M
ECA-ResNet	83.0	79.6	80.8	82.4	6.3291M	0.4442M
MS-ResNet	82.8	78.9	80.4	81.9	28.9229M	2.1151M
MS-ECA-ResNet (The proposed)	84.0	80.9	82.2	83.3	28.9405M	2.1151M

Table 5
Impact of different noise handling methods on performance.

Noise Handling Methods	Macro Precision	Macro Recall	Macro F1-Score	Accuracy
Not adopted	80.0	77.4	78.0	79.6
Mean filter	82.4	79.7	80.8	82.2
Median filter	82.5	78.7	79.9	81.8
Gaussian filter	83.5	80.1	81.3	82.9
Savitzky filter	84.0	80.9	82.2	83.3

convolution kernel of size 3. The settings of the hyperparameters followed the same scheme as the experimental setup. Table 4 lists the experimental results and the impact of different module integrations on computational complexity and memory requirements (i.e. FLOPs, Parameters). As can be seen from Table 4, the lightweight ECA module will not cause a significant increase in the number of parameters and FLOPs, effectively reducing the complexity of the model. Meanwhile, compared with basic and intermediate networks, the network integrating all modules (i.e. MS-ECA- LeNet5 and MS-ECA-ResNet) has been significantly improved. Specifically, after using multi-scale convolution, the performance has been improved, especially under our framework with ResNet as the backbone (MS-ResNet), its performance on macro precision, macro recall, macro F1-score and accuracy has been improved by 1.9%, 1.0%, 1.3% and 1.1%, respectively. This confirms that this module effectively captures the spatiotemporal correlation of user motion behavior and movement trends by incorporating receptive fields of varying sizes, which in turn enhances the model's classification performance. ECA, as an attention mechanism, can learn and select the information that should be focused on at time steps or position points. After combining this module with a multi-scale convolution network, the performance can be further improved. For example, compared with MS-ResNet, the macro precision, macro recall, macro F1-score and accuracy of MS-ECA-ResNet increase from 82.8%, 78.9%, 80.4%, 81.9% to 84.0%, 80.9%, 82.2%, 83.3%. These results suggest that each module contributes positively to the model's performance.

4.6. Exploration of the impact of key factors on model performance

Impact of noise handling methods During the GPS data collection process, due to various reasons such as GPS positioning errors, signal interference, and data missing, there will be errors and missing parts of the GPS trajectory data, resulting in a certain amount of noise when extracting motion signals. To effectively deal with these noises, it is necessary to introduce signal processing methods to preprocess the data. In this section, we will analyze the impact of different denoising methods on performance before and after denoising. Table 5 displays the result of different noise handling methods on performance. As can be seen from Table 5, compared to several methods that use denoising, the method without denoising has the worst performance. This is because the noise contained in GPS trajectory data will affect the accuracy and stability of the data. Among several denoising methods, the Savitzky filter has the best filtering effect, followed by the Gaussian filter. Therefore, in this study, we use the Savitzky filter to preprocess the data.

Impact of different attention mechanisms The attention mechanism plays an important role in acquiring and prioritizing relevant information at different time epochs and spatial locations to improve the feature presentation ability of the network. To assess the effectiveness of the attention module employed in this study, we embedded three common attention modules [20,24,33] into the ResNet network to analyze their impact on performance. The relevant experimental results are shown in Table 6. As can be seen from Table 6, ECA has the most significant impact on model performance improvement, followed by SE. The performance improvement of CBAM is relatively small, but overall, they all show a certain improvement. Compared with the basic model without the attention mechanism, the ECA used in this study increased the macro-precision, recall, F1 score, and accuracy by 1.2%, 2%, 1.8% and 1.4%, respectively. This shows that introducing the attention mechanism can, to a certain extent, enable the model to autonomously learn and select the information that should be focused on at time steps or positions, allowing the model to adapt to different contextual situations automatically.

The generalization ability of the model To evaluate the generalization of the method, we employed 10-fold cross-validation. This technique divides the dataset into 10 different subsets. In each iteration, one subset is used as the test set, while the remaining 9

Table 6
The impact of different attention modules on performance.

Attention Mechanisms	Macro Precision	Macro Recall	Macro F1-Score	Accuracy
Not adopted	82.8	78.9	80.4	81.9
SE	83.7	79.6	81.2	82.6
CBAM	83.2	80.0	81.3	82.4
ECA	84.0	80.9	82.2	83.3

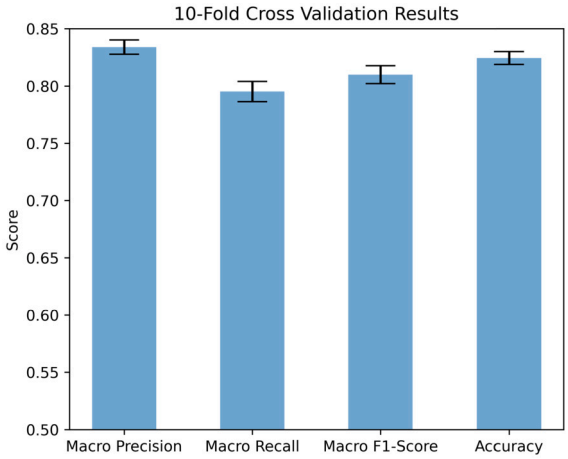


Fig. 5. Experimental results of our method on 10-fold cross-validation.

subsets are used for training. This process is repeated 10 times, with each subset used as the test set once. The final result is obtained by averaging the evaluation scores from all 10 iterations, which helps mitigate possible biases caused by uneven data distribution. The experimental results are shown in Fig. 5. The average macro precision, macro recall, macro F1-score and accuracy of our method in 10-fold cross-validation are 83.4%±0.6%, 79.9%±0.7%, 81.2%±0.6% and 82.6%±0.6% respectively. These indicators show that our method can obtain relatively robust results under different data set partitioning situations and has good generalization ability.

5. Discussion

5.1. Interpretation of findings

This study proposes a travel mode inference method based on a multi-scale mixed attention mechanism, which significantly improves the accuracy and robustness of GPS trajectory data analysis. Compared with traditional methods, this method extracts spatiotemporal information through multi-scale convolution, and combines the attention mechanism to automatically identify and prioritize relevant information at different time periods and spatial coordinates, thereby alleviating the subjective bias and incomplete feature extraction caused by manual feature extraction. This method shows obvious advantages in multiple evaluation indicators, especially in accuracy, macro-precision, macro-recall rate and macro-F1 score, which significantly outperforms traditional machine learning methods and basic deep learning models (Table 2 and Table 3). Compared with the best performing random forest, this method shows a significant performance improvement, which highlights its significant advantages and potential in handling related tasks. In addition, this method achieves comparable or better accuracy than existing work without the need for manual feature extraction, avoiding the dependence of traditional feature engineering on domain knowledge, the limitations of subjective feature selection, and sensitivity to traffic and environmental conditions. This provides novel insights and methods for future travel mode inference research.

The effectiveness of the network module design is verified by ablation experiments (Table 4), which proves the positive impact of each module on the model performance, especially after the introduction of the attention mechanism, the model performance is further improved. This shows that the attention mechanism can autonomously learn and prioritize relevant information in different time periods and spatial coordinates, allowing the model to adapt to various scenarios. In addition, the study also explored the impact of key factors on model performance, including the choice of noise processing methods and attention mechanisms (Table 5 and Table 6). The experimental results show that appropriate noise processing techniques and attention mechanisms can significantly improve the performance and robustness of the model, which provides an important reference for further improving the model in the future.

In general, the results of this study align with some previous findings in travel mode inference, while also introducing some differences and innovations. For instance, Jin et al. [34] utilized GPS trajectory data, combining the Viterbi algorithm and the hidden Markov model (HMM) to infer travel modes, achieving moderate success. However, their approach depended on manual

feature extraction. In contrast, our study automatically extracts spatiotemporal features using multi-scale convolution and attention mechanisms, thereby overcoming the limitations of manual feature extraction and enhancing the model's accuracy and robustness. Similarly, Weng et al. [35] proposed a travel mode identification method based on a Bayesian neural network. While they accounted for spatiotemporal information to some extent, their feature selection was still subjective and incomplete. Our study addresses these issues by integrating multi-scale convolution and attention mechanisms, resulting in higher accuracy and robustness. Furthermore, this study reveals some inconsistencies with prior research. For instance, Zhu et al. [36] asserted that noise processing has minimal impact on travel mode inference model performance. However, our experimental results demonstrate that appropriate noise processing techniques significantly enhance model performance and robustness, underscoring the importance of noise processing in this context.

5.2. Significance and implications

The findings of this study are of great significance to the field of travel mode inference. First, the multi-scale mixed attention mechanism introduced in this study effectively solves the subjectivity and incompleteness of manual feature extraction in traditional methods, and significantly improves the accuracy and robustness of the model. Second, compared with traditional machine learning methods, this method shows significant advantages in multiple evaluation indicators, especially in accuracy, macro precision, macro recall and macro F1-score, showing its great potential in handling related tasks.

In addition, this method provides comparable or better accuracy than existing methods without relying on manual feature extraction, avoiding the limitations of traditional feature engineering. This provides novel methods and ideas for future research, especially in the fields of smart cities, traffic management and personalized recommendations.

Finally, through the study of ablation experiments and noise processing methods, this study provides an important reference for further improving model performance and robustness. These findings not only verify the effectiveness of the research method, but also provide important reference for future research in related fields.

5.3. Limitations

Although this study has achieved some success in processing GPS trajectory data, there are still some limitations that deserve further exploration. First, the small dataset used may result in the model not being able to fully cover various scenarios and populations [37], which in turn affects its generalization ability in practice. Second, although the dataset has been annotated, there is still a certain degree of annotation error or bias [38], which may cause the model to learn the wrong patterns or regularities, reducing its accuracy and credibility.

In addition, this study used four motion signals as input features when processing GPS trajectory data, but it is not clear whether these features can fully express the user's motion behavior and environmental influences. There may be key features that are not considered or motion patterns that cannot be effectively captured, which may affect the performance of the model. For instance, Broach et al. [39] extracted the median speed, 95% percentile speed and 95% percentile acceleration as the motion features for inferring travel mode.

5.4. Future research directions

There are a large number of unlabeled GPS records in the current research data, which potentially contain rich information. However, how to effectively use this data to enhance the generalization and robustness of the inference model is a difficulty. Traditional supervised learning methods require a large amount of labeled data to train the model, but obtaining labeled data usually requires a lot of manpower and time costs, which limits the application scope and performance of the model.

As a method that can use labeled and unlabeled data for model training, semi-supervised learning can make up for the problem of insufficient labeled data to a certain extent [40]. For GPS trajectory data, labeled data may come from locations or trips actively marked by users, while unlabeled data may be users' daily travel records. Although these unlabeled data do not have clear labels, they contain valuable geographic information [41], such as users' common routes, stopovers, and travel habits. Through semi-supervised learning, these unlabeled data can be effectively used to enhance the ability of inference models to understand and predict user behavior.

Therefore, in future research, we will try to combine semi-supervised learning and GPS trajectory data analysis technology to achieve more accurate inference of user travel behavior and provide more accurate solutions for smart cities, traffic management, personalized recommendations and other fields.

6. Conclusion

The popularity of GPS technology has given it powerful practicality, helping to record personal movement trajectories across diverse temporal and spatial dimensions. This capability, in turn, provides a simplified and efficient conduit for inferring travel modes. This study presents a novel method for travel mode inference through GPS trajectory data. A distinctive characteristic of this approach is its integration of multiple techniques to accurately comprehend traveler behavior and trends, thereby enhancing the overall effectiveness of the model. By employing multi-scale convolution, the model can capture spatiotemporal correlations, strengthening the model's comprehension of user motion behaviors. Furthermore, the incorporation of an attention mechanism enables the model to autonomously assimilate and prioritize information pertinent to distinct temporal steps or spatial positions, thus refining its focus on salient features. Extensive experiments on a public dataset demonstrate the effectiveness of this method.

Ethics approval

The submitted manuscript is original and has not been published elsewhere in any form or language.

Funding

The work is supported by the Anhui Provincial Major Science and Technology Project (No. 202103a07020004, 202303a07020006-4, 202304a05020071), the Anhui Province Natural Science Foundation (No. 202204295107020004) and the National Natural Science Foundation of China (No. 62133004).

CRediT authorship contribution statement

Xiaohui Pei: Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis. **Xianjun Yang:** Writing – review & editing, Supervision, Project administration, Conceptualization. **Tao Wang:** Writing – review & editing, Conceptualization. **Zenghui Ding:** Writing – review & editing, Conceptualization. **Yang Xu:** Writing – review & editing, Conceptualization. **Lin Jia:** Writing – review & editing, Conceptualization. **Yining Sun:** Writing – review & editing, Conceptualization.

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

The data and code supporting our approach are available at https://github.com/peixiaoh/Travel-Mode-Inference.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used ChatGPT in order to improve language and readability. After using this tool, the authors reviewed and edited the content as needed and took full responsibility for the content of the publication.

References

[1] K.E. Zannat, C.F. Choudhury, Emerging big data sources for public transport planning: a systematic review on current state of art and future research directions, J. Indian Inst. Sci. 99 (4) (2019) 601–619.
[2] N. Verzosa, S. Greaves, C. Ho, M. Davis, Stated willingness to participate in travel surveys: a cross-country and cross-methods comparison, Transportation 48 (2021) 1311–1327.
[3] Z. Xiao, Y. Chen, M. Alazab, H. Chen, Trajectory data acquisition via private car positioning based on tightly-coupled gps/obd integration in urban environments, IEEE Trans. Intell. Transp. Syst. 23 (7) (2021) 9680–9691.
[4] X. Lei, R. Wang, F. Fu, An adaptive method of attitude and position estimation during gps outages, Measurement 199 (2022) 111474.
[5] Y. Zheng, L. Liu, L. Wang, X. Xie, Learning transportation mode from raw gps data for geographic applications on the web, in: Proceedings of the 17th International Conference on World Wide Web, 2008, pp. 247–256.
[6] A. Bolbol, T. Cheng, I. Tsapakis, A spatio-temporal approach for identifying the sample size for transport mode detection from gps-based travel surveys: a case study of London's road network, Transp. Res., Part C, Emerg. Technol. 43 (2014) 176–187.
[7] B. Wang, L. Gao, Z. Juan, Travel mode detection using gps data and socioeconomic attributes based on a random forest classifier, IEEE Trans. Intell. Transp. Syst. 19 (5) (2017) 1547–1558.
[8] P. Nitsche, P. Widhalm, S. Breuss, N. Brändle, P. Maurer, Supporting large-scale travel surveys with smartphones–a practical approach, Transp. Res., Part C, Emerg. Technol. 43 (2014) 212–221.
[9] F. Zong, Y. Bai, X. Wang, Y. Yuan, Y. He, Identifying travel mode with GPS data using support vector machines and genetic algorithm, Information 6 (2) (2015) 212–227.
[10] S. Dabiri, C.-T. Lu, K. Heaslip, C.K. Reddy, Semi-supervised deep learning approach for transportation mode identification using gps trajectory data, IEEE Trans. Knowl. Data Eng. 32 (5) (2019) 1010–1023.
[11] Y. Endo, H. Toda, K. Nishida, A. Kawanobe, Deep feature extraction from trajectories for transportation mode estimation, in: Advances in Knowledge Discovery and Data Mining: 20th Pacific-Asia Conference, PAKDD 2016, Auckland, New Zealand, April 19-22, 2016, Proceedings, Part II 20, Springer, 2016, pp. 54–66.
[12] H. Wang, G. Liu, J. Duan, L. Zhang, Detecting transportation modes using deep neural network, IEICE Trans. Inf. Syst. 100 (5) (2017) 1132–1135.
[13] S. Dabiri, K. Heaslip, Inferring transportation modes from gps trajectories using a convolutional neural network, Transp. Res., Part C, Emerg. Technol. 86 (2018) 360–371.
[14] W. Chen, K. Shi, Multi-scale attention convolutional neural network for time series classification, Neural Netw. 136 (2021) 126–140.
[15] Y. Gao, X. Chen, A. Liu, D. Liang, L. Wu, R. Qian, H. Xie, Y. Zhang, Pediatric seizure prediction in scalp eeg using a multi-scale neural network with dilated convolutions, IEEE Journal of Translational Engineering in Health and Medicine 10 (2022) 1–9.
[16] L. Lin, W. Li, H. Bi, L. Qin, Vehicle trajectory prediction using lstms with spatial–temporal attention mechanisms, IEEE Intelligent Transportation Systems Magazine 14 (2) (2021) 197–208.
[17] R. Li, Y. Qin, J. Wang, H. Wang, Amgb: trajectory prediction using attention-based mechanism gcn-bilstm in iov, Pattern Recognit. Lett. 169 (2023) 17–27.
[18] Y. Yu, Y. Zhang, Z. Cheng, Z. Song, C. Tang, Mca: multidimensional collaborative attention in deep convolutional neural networks for image recognition, Eng. Appl. Artif. Intell. 126 (2023) 107079.
[19] D. Liu, N. Sheng, Y. Han, Y. Hou, B. Liu, J. Zhang, Q. Zhang, Scau-net: 3d self-calibrated attention u-net for brain tumor segmentation, Neural Comput. Appl. 35 (33) (2023) 23973–23985.

[20] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, Q. Hu, Eca-net: efficient channel attention for deep convolutional neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11534–11542.
[21] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[22] X. Wen, X. Li, C. Zhang, W. Han, E. Li, W. Liu, L. Zhang, Me-net: a multi-scale erosion network for crisp building edge detection from very high resolution remote sensing imagery, Remote Sens. 13 (19) (2021) 3826.
[23] X. Wei, B. Hu, T. Gao, J. Wang, B. Deng, Multi-scale convolutional neural network for texture recognition, Displays 75 (2022) 102324.
[24] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.
[25] Y. Zheng, H. Fu, X. Xie, W.-Y. Ma, Q. Li, Geolife GPS trajectory dataset - User Guide, geolife gps trajectories 1.1 Edition, https://www.microsoft.com/en-us/research/publication/geolife-gps-trajectory-dataset-user-guide/, July 2011.
[26] M. Massaoudi, S.S. Refaat, H. Abu-Rub, I. Chihi, F.S. Oueslati, Pls-cnn-bilstm: an end-to-end algorithm-based Savitzky–Golay smoothing and evolution strategy for load forecasting, Energies 13 (20) (2020) 5464.
[27] G. Asci, M.A. Guvensan, A novel input set for lstm-based transport mode detection, in: 2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), IEEE, 2019, pp. 107–112.
[28] J.J.Q. Yu, Travel mode identification with gps trajectories using wavelet transform and deep learning, IEEE Trans. Intell. Transp. Syst. 22 (2) (2020) 1093–1103.
[29] F. Namdarpour, M. Mesbah, A.H. Gandomi, B. Assemi, Using genetic programming on gps trajectories for travel mode detection, IET Intell. Transp. Syst. 16 (1) (2022) 99–113.
[30] Y. Zhu, Y. Liu, J. James, X. Yuan, Semi-supervised federated learning for travel mode identification from gps trajectories, IEEE Trans. Intell. Transp. Syst. 23 (3) (2021) 2380–2391.
[31] C. Zheng, C. Wang, X. Fan, J. Qi, X. Yan, Stpc-net: learn massive geo-sensory data as spatio-temporal point clouds, IEEE Trans. Intell. Transp. Syst. 23 (8) (2021) 11314–11324.
[32] I. Cardoso-Pereira, J.B. Borges, P.H. Barros, A.F. Loureiro, O.A. Rosso, H.S. Ramos, Leveraging the self-transition probability of ordinal patterns transition network for transportation mode identification based on gps data, Nonlinear Dyn. 107 (1) (2022) 889–908.
[33] S. Woo, J. Park, J.-Y. Lee, I.S. Kweon, Cbam: convolutional block attention module, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 3–19.
[34] Z. Jin, Y. Chen, C. Li, Z. Jin, Trip destination prediction based on hidden Markov model for multi-day global positioning system travel surveys, Transp. Res. Rec. 2677 (2) (2023) 577–587.
[35] P. Weng, S. Jia, X. Pei, Y. Yue, Bayes neural network with a novel pictorial feature for transportation mode recognition based on gps trajectories, in: CICTP 2021, 2021, pp. 1635–1645.
[36] Y. Zhu, C. Markos, J. James, Improving transportation mode identification with limited gps trajectories, in: 2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI), IEEE, 2021, pp. 655–660.
[37] H. Amiri, S. Ruan, J.-S. Kim, H. Jin, H. Kavak, A. Crooks, D. Pfoser, C. Wenk, A. Zufle, Massive trajectory data based on patterns of life, in: Proceedings of the 31st ACM International Conference on Advances in Geographic Information Systems, 2023, pp. 1–4.
[38] M. Etemad, A.S. Junior, S. Matwin, On feature selection and evaluation of transportation mode prediction strategies, preprint, arXiv:1808.03096, 2018.
[39] J. Broach, J. Dill, N. McNeil, Travel mode imputation using gps and accelerometer data from a multi-day travel survey, J. Transp. Geogr. 78 (2019) 194–204, https://doi.org/10.1016/j.jtrangeo.2019.06.001.
[40] J. Zhu, Semi-supervised learning: the case when unlabeled data is equally useful, in: Conference on Uncertainty in Artificial Intelligence, PMLR, 2020, pp. 709–718.
[41] C. Markos, J. James, Unsupervised deep learning for gps-based transportation mode identification, in: 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2020, pp. 1–6.