Convolutional LSTM based transportation mode learning from raw GPS trajectories

Article *in* IET Intelligent Transport Systems · February 2020

DOI: 10.1049/iet-its.2019.0017

6 authors, including:

Asif Nawaz
Nanjing University of Aeronautics & Astronautics
8 PUBLICATIONS **15** CITATIONS

Senzhang Wang
Nanjing University of Aeronautics & Astronautics
107 PUBLICATIONS **770** CITATIONS

Yasir Hussain
Nanjing University of Aeronautics & Astronautics
16 PUBLICATIONS **23** CITATIONS

Izhar Ahmed Khan
Nanjing University of Aeronautics & Astronautics
11 PUBLICATIONS **35** CITATIONS

IET Intelligent Transport Systems

Research Article

Convolutional LSTM based transportation mode learning from raw GPS trajectories

ISSN 1751-956X
Received on 8th January 2019
Revised 27th September 2019
Accepted on 24th February 2020
doi: 10.1049/iet-its.2019.0017
www.ietdl.org

Asif Nawaz[1], Huang Zhiqiu[1,2,3] ✉, Wang Senzhang[1], Yasir Hussain[1], Izhar Khan[1], Zaheer Khan[1]

[1]*College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing 211106, People's Republic of China*

[2]*Key Laboratory of Safety-Critical Software, NUAA, Ministry of Industry and Information Technology, Nanjing 211106, People's Republic of China*

[3]*Collaborative Innovation Centre of Novel Software Technology and Industrialization, Nanjing 210093, People's Republic of China*

✉ *E-mail: zqhuang@nuaa.edu.cn*

Abstract: With the advancement of location acquisition technologies, a large amount of raw global positioning system (GPS) trajectory data is produced by many moving devices. Learning transportation modes from the GPS trajectory data is an important problem in the domain of trajectory data mining. Traditional supervised learning-based approaches rely heavily on data preprocessing and feature engineering, which require domain expertise and are time consuming. The authors propose a deep learning-based convolutional long short term memory (LSTM) model for transportation mode learning, in which the convolution neural network is first used to extract deep high-level features and then LSTM is used to learn the sequential patterns in the data that uses both GPS and weather features, thus making the full use of spatiotemporal operations. The authors have also analysed the impact of the geospatial region on human mobility. Experiments conducted on the Microsoft Geolife data set fused with the weather data set show that their model achieves the state-of-the-art results. The authors compare the performance of their model with the benchmark models, which shows the superiority of their model having 3% improvement in accuracy using only GPS features, and the accuracy is further improved by 4 and 7% on including the impact of geospatial region and weather attributes, respectively.

1 Introduction

Advances in navigation and location acquisition technologies such as global positioning system (GPS), global system for mobile communications (GSM) or wireless networks, together with the development of location-based services, enable the smart devices to produce a massive amount of spatiotemporal data [1]. The use of this massive GPS data produced by location acquisition technologies is an important part of smart city and intelligent transportation system, and produced many opportunities for researchers in different application domains, such as traffic congestion estimation [2, 3], travel route planning [4], behaviour analysis [5], transportation monitoring [3, 6], point of interest recommendation [7], inference of taxi status [8], identifying travel trips and activities [9] or destination prediction [10, 11] etc. Transportation mode inference is the fundamental key research problem in the transportation domain, which aims to identify the sequence of transportation modes in a trip like a walk or bus from the trajectory data generated by the users through multiple sensors including GPS, GSM or accelerometer sensors etc.

The identification of transportation modes plays a vital role in providing task-centric support [12]. The service providers can send personalised advertisements and real-time traffic information according to the passenger's transportation mode and location. The end users can utilise the information provided by these services to reduce their cost and travel time [13]. In the past, the mobility information was collected through different techniques, either from telephone surveys or questionnaires. Such methods were slow and inconvenient and also incur huge costs for collection and analysis [14]. The collection of wearable sensors like gyroscope and accelerometer devices was among the techniques used for the collection of fine-grained information [15–17], but on the other side this is not feasible to carry many sensors all the time. Radio signals and wireless sensor technologies although incur less cost but provide coarse-grained information, they do not provide sufficient information for the classification of transportation modes [18]. This requires convenient and cost-effective technologies such as GPS, which is anonymously produced by many smart devices, and does not incur any human cost. These devices also able to get fine-grained information that results in better prediction of classification problems.

Over the past few decades, many studies have focused on using statistical and machine learning techniques to infer transportation modes from the GPS trajectory data. The authors in [12–14, 19–22] used conventional supervised learning-based approaches. These techniques require domain knowledge and human expertise to extract hand-crafted features from the GPS trajectory data for the classification purpose. These techniques are shallow in nature and can only learn up to a certain extent. To avoid such shortcomings, few recent studies exploited deep learning methods to automatically learn deep features for the inference of transportation modes from the GPS data [23–26]. Endo *et al*. [23] used a fully connected network to learn temporal deep features and classification of transportation modes, but its performance is undesirable in terms of accuracy. Wang *et al*. [24] made use of the convolution neural network (CNN) model, with a combination of deep features and hand-crafted features and used a fully connected network for the classification task. Few of these features are redundant and suffer from the problem of collinearity. Dabiri and Heaslip [25] addressed the problem in previous deep learning studies by Endo *et al*. [23] and Wang *et al*. [24] and proposed the CNN-based model to identify transportation modes. Dabiri and Heaslip [25] used the basic kinematic and behavioural features to classify modes based on derived deep features from the basic input features. The study did not consider contextual information, which is important for this problem. Liu and Lee [26] used the long short term memory (LSTM) network for the same purpose. LSTM network does not consider spatial dependencies and addresses the problem only based on temporal dependencies in the sequence of observations. To overcome the limitations in existing works, we propose to explain the convolutional LSTM (ConvLSTM) architecture [27] to infer the transportation modes of the users in their trips from raw GPS trajectory data. ConvLSTM architectures are quite powerful as they learn the relationship between time and space. Conventional supervised learning-based approaches are highly dependent on human expertise and domain knowledge.

Studies on deep learning methodologies used the basic models like fully connected, CNN or LSTM independently. ConvLSTM architecture uses CNN that extracts the deep features from the trajectory data, combined with LSTM network used to interpret the features and support sequence prediction task [28], thus it makes full use of spatiotemporal operations.

The basic GPS attributes do not help to identify the dynamic traffic patterns in a complex city environment. Some researchers take the benefit of using geographic information system (GIS) data with the GPS data [29, 30] and outperform the GPS-based models, but the GIS data is not always available for all locations and may change with the passage of time in the case of any modification in the city road network. Simple motion and behavioural features may help to identify the modes of transport in normal traffic conditions, but these features may not help to identify the dynamic traffic patterns in the case of abnormal traffic conditions. For example, in real life, we can differentiate between transportation modes based on speed, velocity and bearings, but in bad weather conditions, where traffic congestions are imminent [6], it is difficult to differentiate between transportation modes based on kinematic and behavioural features. Therefore, we need additional weather features to uniquely identify the patterns among different transportation modes. Mobility patterns of the users are heavily dependent on weather conditions [31]. Based on this fact, we make use of the weather data which is easily available for all the locations and helps to overcome the limitations of GPS only attributes.

In this paper, we propose a ConvLSTM model that blends the concepts of both CNN and LSTM into a single model. Before these CNN and LSTM were the most effective deep neural network (DNN) architectures to predict human mobility, both of these methods have their own capabilities and limitations. ConvLSTM is another DNN based model which is capable of learning hundreds of features automatically from the small number of input features using operations in hidden layers, thus avoiding the problem of human expertise to extract hand-crafted features based on their experience. Existing studies used only kinematic features like speed, velocity, jerk, bearing rate etc., but regions of observations were unknown to model, and this is important because traffic behaviour varies from one region to another. We also include the impact of the region that is important to distinguish the transportation patterns at different regions in geographical space, thus making full use of the spatiotemporal capability of the model. Weather-based features are also included in the model with GPS features, which help to further enhance the performance of the model. To the best of our knowledge, this is the first work that studies the effects of weather on human mobility using the GPS data, and none of the existing works used ConvLSTM architecture for the identification of transportation modes and even to model the GPS data. The main challenge in this study is to represent the input point level features in the form that is acceptable to the ConvLSTM architecture and learns the deep features and predicts the transportation modes with higher accuracy. We classify four distinct transportation modes from the raw GPS data, i.e. walk, bike, bus and car that have sufficient availability of data in a dense region to train deep neural networks, giving a reliable estimation of transportation modes.

The remaining sections of this paper are organised as follows. Section 2 discussed the brief overview of related work with their limitations. Section 3 discussed the design of the model and used to classify transportation modes. In Section 4, we discussed the results of our experiments and evaluated our approach in comparison with existing approaches. Section 5 is the conclusion, in which we summarise our work.

2 Related work

Traditionally, traffic monitoring and evaluation have relied heavily on a variety of road sensors [3]. Some studies like [16, 32–34] use data from a different set of sensors such as GPS, accelerometer, gyroscope or magnetometer etc. The performance of these studies has produced significant results, but it has a big limitation that it is not easy to carry multiple sensors all the time. Few studies [20, 29, 35, 36] make use of the GIS data with GPS sensor data, but the GIS information is not always available for all places [14], and the structure of city changes from time to time. This results in the change of GIS maps and in result need to re-train the models every time the update is recorded in the GIS data. So based on these constraints, such approaches are not practical in real life [18]. The GPS data is anonymously produced by many different devices such as vehicle tracking systems, mobile phones etc., therefore, using the GPS data for mobility inference is more practical and non-trivial. Few studies like [19, 25, 37, 38] use only the GPS data for inference task, but at the same time only using the GPS data is not enough, because there are some hidden patterns that cannot be identified from GPS attributes as human mobility is highly correlated with the climate conditions [31, 39] and the weather data can be easily integrated with the GPS data. Therefore, we use weather attributes in addition to the GPS data, because using the weather data does not incur any cost and is easily available for all locations.

Mining of the GPS trajectories data has become an important research topic during the past few years [40]. One of the key research problems in GPS trajectory data mining is the identification of transportation modes of a user. Several different methods of transportation mode identification have been proposed in the past decade using statistical techniques and classical machine learning algorithms [13, 19, 21, 41, 42]. Existing studies extract a different set of features including motion and behavioural features using the mathematical and statistical knowledge. These features are then passed into machine learning models like K-nearest neighbour, support vector machine (SVM), hidden Markov model, decision tree, random forest to classify different types of transportation modes. All these techniques have certain limitations. First these techniques rely heavily on hand-crafted features that require human expertise and domain knowledge. In addition, the accuracy of machine learning algorithms is highly dependent on the accuracy of the extracted features. Second, the classical machine learning models are a type of shallow models that can only learn the patterns in small data sets and considered to be unsatisfactory in big data scenarios [43]. As the traffic data were blowing up and produced anonymously by many smart devices at a large scale, and entered into the era big data, therefore our conventional machine learning algorithms are not able to perform well in such a big data set [44]. These problems can be avoided if such features are learned automatically without requiring human expertise and domain expertise, and the deep network can be trained well on a big data set.

In recent years the deep learning methodology is being used in many scientific problems ranging from biomedical, speech recognition, computer vision etc. and the achieved state-of-the-art results [45]. In recent years, the deep learning methodologies are also used to solve transportation problems, but the use of deep learning is still very limited in this domain [43]. Deep neural nets represent the data at multiple levels and use non-linear functions to transform the data from one level to the next higher level. Using these transformations at different levels, the models are able to learn complex functions and structures from the input data [45]. Deep learning models such as CNN and LSTM have shown the ability to automatically learn features from raw GPS data, and achieved the best results. With the recent success of deep learning, some researchers have tried to apply deep learning techniques to solve transportation mode inference problems. Gonzalez *et al*. [22] used the hand-crafted statistical features such as average and maximum speeds and accelerations, and just used a multi-layer perceptron model as the classifier to classify three types of transportation modes such as walk, bus and car. Endo *et al*. [23] used a feed-forward DNN to automatically extract high-level features. Their main idea is to transform a raw GPS trajectory into a 2D image structure as the input in which the time duration of a GPS point is considered as the pixel value, and used decision trees to classify the modes. The authors in [24, 25] used the CNN architecture with different approaches of data modelling to learn the transportation modes from the GPS trajectory data. Liu and Lee [26] used the LSTM network for the identification of transportation modes, by exploiting the long-term dependencies in time-series trajectory data. The limitation in their studies is that these studies do not take full advantage of spatial features, and do not consider locality information, which is important to learn transportation pattern, because traffic patterns vary from one region to another. Although Dabiri and Heaslip [25] claimed the use of spatiotemporal features, like velocity that is based on both distance and time, but this is spatiotemporal only in a minor sense, because these features did not consider the exact location or the region, which is also important for knowing the traffic behaviour e.g. the speed of the bus or car is different in downtown areas of the city as compared to the speeds of vehicles in the countryside. Simple feed-forward neural networks are complex and need to learn millions of parameters that require heavy computing resources [45]. Both CNN and LSTM have different capabilities. CNN is good in learning deep features, while LSTM is good to capture long-term dependencies in sequencing data [45]. We used the ConvLSTM model in our study to get the benefit of both these architectures in a single combined architecture, and to make full use of the spatiotemporal information.

3 Model design

The GPS trajectory data is the sequence of GPS points ordered with their timestamps. The first step is to divide the user's trajectories into trips if the distance between two consecutive GPS points exceeds a certain time duration threshold value. In the next step, input features are computed for all GPS points. Preprocessing of GPS trajectory is also required to remove the outlier points caused by the device error or erroneously labelled by the user. Inspired from the study [25] all GPS points belonging to certain transportation modes that exceed that maximum allowable threshold velocities and acceleration are removed from the trajectory data. As the input to deep neural network required fixed-length input, all user's trajectories are divided into fixed-length segments. The size of the last segment in the trip may be less than the desired length, so we pad these segments with zero values to make its length equal to the desired length. In the following subsections, we explain the design of our methodology. We start with the preliminary subsection, giving a brief overview of baseline models. This subsection also elaborates in detail the concepts of the ConvLSTM model. Second part of this section will discuss the input features used in our model. Last, we discuss the description of the model architecture used for the identification of transportation modes.

3.1 Preliminaries

Following is the description of models that we consider as a baseline for our model.

3.1.1 Multilayer perceptron: The multilayer perceptron is the simplest deep neural network model, also called a fully connected feed-forward neural network. It can be regarded as a special logistic regression classifier and this special logistic regression classifier performs a non-linear transformation on the input of the sample. Non-linear transformations are performed to map the input samples to a space in which the samples are linearly separable using the set of hidden layers.

3.1.2 Convolution neural network: CNN or ConvNet were initially developed for image recognition tasks. The model learns the internal representation of a 2D image for the classification task [45]. The convolution layer utilises the correlation between spatially connected points by strengthening the local connection pattern between adjacent neurons. The same process can be harnessed to learn one-dimensional sequence data. It is composed of two sets of layers, convolution layer and pooling layer. The input to convolution layers is transformed into a feature map through the set of weights called filters. Each feature map corresponds to different filters.

3.1.3 Long short term memory: LSTM network is a kind of recurrent neural network (RNN), which is able to learn sequential dependencies in sequence prediction tasks [27]. The basic version of RNN suffers from vanishing and exploding gradient problems, so in result cannot model long-term dependencies. LSTM is an extension of basic RNN architecture that includes the memory cell to accumulate the state information. Its purpose is to overcome the vanishing and exploding gradient problem to learn long-term dependencies. The key equations of LSTM are as follows:

$$\begin{aligned} i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i)\\ f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f)\\ c_t &= f_t \circ c_{t-1} + i_t \circ \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c)\\ o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co} \circ c_t + b_o)\\ h_t &= o_t \circ \tanh(c_t) \end{aligned} \tag{1}$$

where i_t is the input gate at time t, f_t is the forget gate, c_t is the cell state, o_t is the output gate and h_t is the hidden state at time t. These gates control what to feed in the next cell, what to forget and what to output. Similarly, i_{t-1} is the input gate, f_{t-1} is the forget gate, c_{t-1} is the cell state, o_{t-1} is the output gate and h_{t-1} is the hidden state at time $t-1$. σ and tanh are the sigmoid and hyperbolic tangent functions, respectively. W and b are the weight and bias, respectively.

3.1.4 Convolutional LSTM: Fully connected LSTM networks are powerful in handling temporal dependencies, but it suffers from redundancy in spatial data. The drawback of the LSTM network is that it can only handle one-dimensional data by unfolding all the dimensions to a single dimension, as a result it only keeps temporal dependencies and loses spatial dependencies [27]. The core functionality of ConvLSTM can be seen as the convolution layer embedded within the LSTM. The ConvLSTM involves the convolution operation in both input-to-state and state-to-state transitions, instead of a matrix multiplication in LSTM. It can also determine the future state of the cell through current input and the past states of its neighbours, thus the model is able to capture temporal dependencies. Therefore, the ConvLSTM can take benefit of CNN by handling spatial dependencies, and also takes the benefit of LSTM by handling temporal dependencies. Fig. 1 is the basic representation of ConvLSTM, both input and inner representations are two dimensional in the case of one channel, thus enabling the model to encode both the temporal and spatial correlations [27]. The key equations of ConvLSTM are given below:

$$\begin{aligned} i_t &= \sigma(W_{xi}X_t + W_{hi}H_{t-1} + W_{ci} \circ C_{t-1} + b_i)\\ f_t &= \sigma(W_{xf}X_t + W_{hf}H_{t-1} + W_{cf} \circ C_{t-1} + b_f)\\ C_t &= f_t \circ C_{t-1} + i_t \circ \tanh(W_{xc}X_t + W_{hc}H_{t-1} + b_c)\\ o_t &= \sigma(W_xoX_t + W_hoH(t-1) + W_co \circ C_t + b_o)\\ H_t &= o_t \circ \tanh(C_t) \end{aligned} \tag{2}$$

where * is the convolutional operator and the rest of the notations are the same as in (1). In order to have the same dimension for input and the states, padding is required before applying the convolutional operator. In ConvLSTM networks, the feature maps produced using CNN activations are taken as input to the LSTM network. As a result, it performs both the spatial and temporal operations to make the best use of spatiotemporal information.

3.2 Input features

In this subsection, we describe two types of features that are used for the classification of transportation modes: GPS-based features and weather features.

3.2.1 GPS-based features: Inspired from the study [25], we used the following point level features given two consecutive GPS points as p_1 and p_2:

$$V_{p1} = \frac{\text{Vincenty}(p_1, p_2)}{\Delta t} \quad (3)$$

$$A_{p1} = \frac{V_{p2} - V_{p1}}{\Delta t} \quad (4)$$

$$J_{p1} = \frac{A_{p2} - A_{p1}}{\Delta t} \quad (5)$$

where V_{p1}, A_{p1}, J_{p1} are the velocities, acceleration and jerk of point p_1, computed in (3)–(5). Δt is the time difference between points p_1 and p_2. The bearing rate is calculated from the given set of equations:

$$\begin{aligned} y &= \sin[p_2(\text{long}) - p_1(\text{long})]\cos[p_2(\text{lat})], \\ x &= \cos[p_1(\text{lat})]\sin[p_2(\text{lat})] - \sin[p_1(\text{lat})]\cos[[p_2(\text{lat})] \\ &\quad \cos[[p_2(\text{long}) - p_1(\text{long})], \\ \text{Bearing}_{p1} &= \tan^{-1}(y, x), \\ \text{BR}_{p1} &= |\text{Bearing}_{p1} - \text{Bearing}_{p2}| \end{aligned} \quad (6)$$

In addition to these features, we also introduced the region index by dividing the whole geographical region into two-dimensional grid of rows and columns. All latitude and longitude values are associated with their respective row and column numbers in the grid. For simplicity purpose, the two-dimensional grid values are converted to a single dimension attribute using (7), as given below:

$$\text{gridIndex} = (\text{row} - 1) \cdot \text{cols} + \text{col} \quad (7)$$

where row and col are the specific row and column number of GPS points, and cols are the total number of columns in the 2D grid.

The deep features are derived by the model from basic input features using multiple types of operations using hidden layers in the model. The derived deep features are segment level features, like mean, variance, standard deviation and many other statistical attributes that are based on the input variable. There are other important auxiliary segment level features that are necessary for the classification of transportation modes, but cannot be derived from basic input features. For example, the features like time of day, day of the week cannot be derived using our point level input features, because input point level features like distance, velocity do not have any relationship with time of the day or day of the week. We merge those auxiliary features with deep features before classifying transportation modes. The point level features fluctuate between all GPS points like distance, velocity etc., we feed these features as the input to the ConvLSTM model to derive the deep features. The features remain the same throughout all GPS points of the fixed-length segments and these features are treated as auxiliary features which bypass the ConvLSTM layers. These auxiliary features integrate with deep features directly to reduce the redundancy and model complexity as shown in Fig. 2. We use two segment level auxiliary features, day of the week and time slice. These hand-crafted auxiliary features are merged with deep features learned by the ConvLSTM model before classification using fully connected layers. Time slice is categorised into two different values, busy and idle based on rush hours, and the time when people go to office and schools or the time when people come back home are considered rush hours.

3.2.2 Weather features: The user's decision to take the transportation mode is also dependent on weather conditions [31]. For example, the user decides to take a walk in good weather, but in extreme weather conditions like in the hot weather, the user avoids taking walk mode. Traffic patterns are also dependent on weather conditions, which are different in normal and abnormal weather conditions. In the rainy weather, where traffic congestions are very common, the velocity of car mode may become as slow as the walk mode at the busiest locations of the city. Combining the weather attributes with deep features, the model can be able to correctly predict the transportation modes. Weather attributes that we will be using are temperature, visibility, wind speed and dew point, which determine the overall climate of the environment. Weather attributes are treated as auxiliary features in the architecture.

3.3 Model description

Fig. 2 explains the proposed ConvLSTM-based architecture for the identification of transportation mode from the GPS trajectory data. The input to the architecture is in the form of samples, time steps, channels. The sample is the total number of trajectory segments in the training data, the time step is the number of observations in a single segment and channels are the number of basic input features.

The input to this model is a segment with basic point level features. Five-point level features shown in (3)–(7) are fed as input to the ConvLSTM layer. Features from the weather data and two GPS features, day of the week and time slice, are taken as auxiliary features. These auxiliary features at a later stage merge with the deep features produced by ConvLSTM layers. Each input segment in our data set is 2D for the given sample, and the first dimension is the length of the time step and the second dimension is the number of channels. The length of the time step is considered to be the total number of GPS points in a segment and the number of input features is taken as the number of channels. The input to the simple LSTM model is 2D with dimensions of time steps and channels, and the input to CNN is a 3D tensor with dimensions of row, columns and channels. In contrast to LSTM and CNN dimensions, the input to the ConvLSTM model requires a 4D tensor with the dimension of time steps, rows, columns and channels. To transform the shape of the input sample into the form that is acceptable to ConvLSTM architecture, the two-step process is used. At first step, we divide the whole segment into multiple sub-segments of equal sizes, and these number of sub-segments are represented as time steps, whereas the length of each sub-segment has become a one-dimensional row vector. Each one-dimensional row vector or a sub-segment in the next step is further divided into two equal sub-

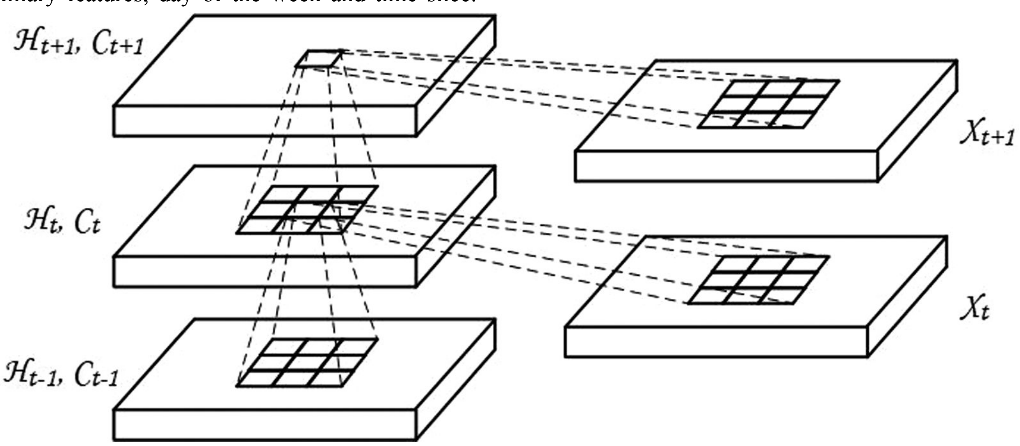

Fig. 1 *Inner structure of ConvLSTM [27]*

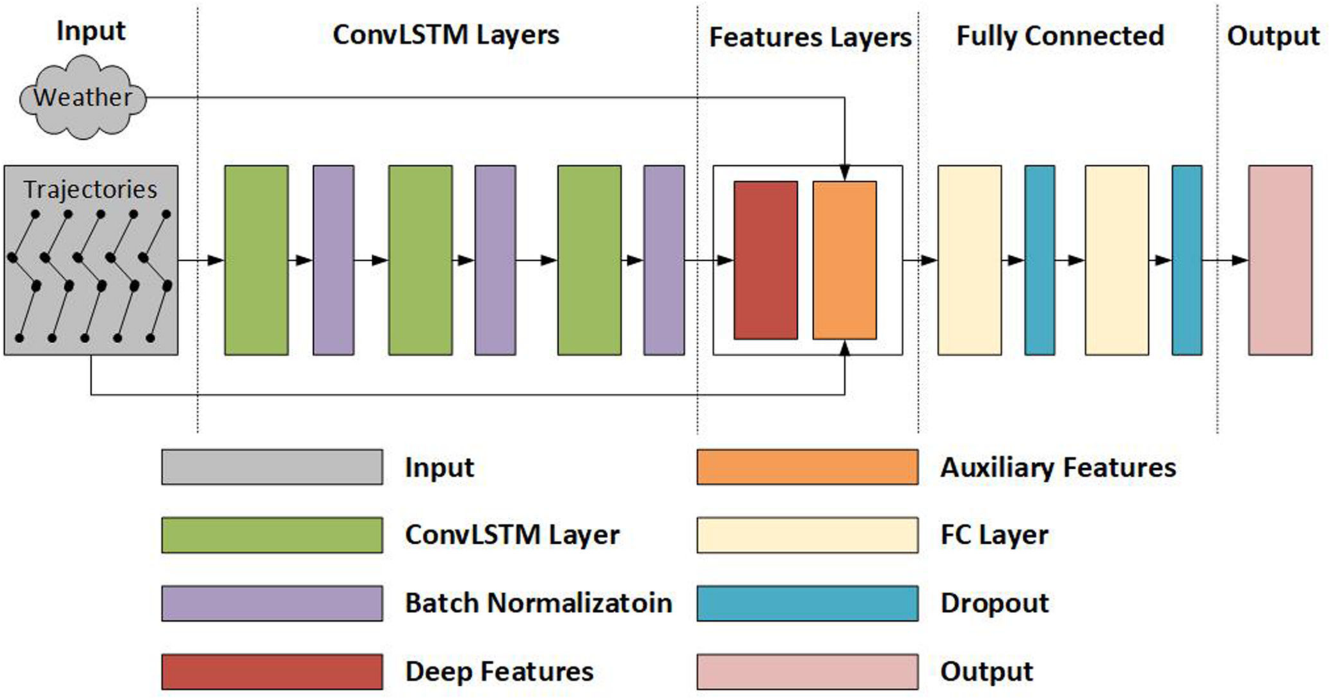

Fig. 2 *Architecture of ConvLSTM model for transportation mode identification*

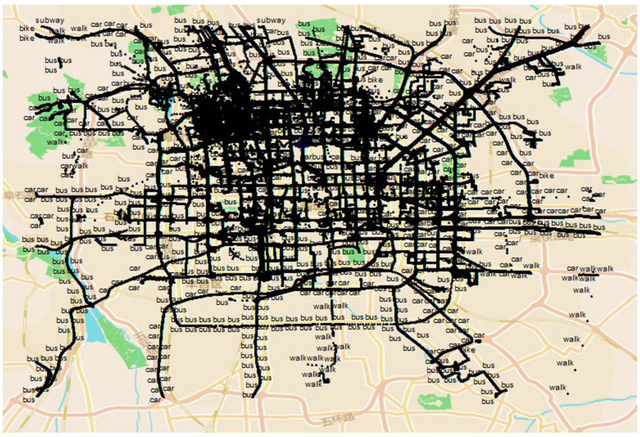

Fig. 3 *Microsoft geolife data set in Beijing*

parts of rows and columns. In the second step, we further divide each sub-segment into rows and columns to make a 4D tensor, which will be taken as an input to the ConvLSTM model. The ConvLSTM layer performs convolutions to produce feature maps that are interpreted by LSTM to output deep features. The same padding is applied on feature maps, so that the data that flows between ConvLSTM cells keeps the same dimensions for both input and states before applying the convolution operator. The batch normalisation layer is used for the normalisation of mini-batch input. Flatten layer transforms the deep level features into one-dimensional vector in order to merge with one-dimensional auxiliary features before applying a fully connected network as shown in the features layer part of Fig. 2. Drop out layers are used with the fully connected layers to reduce the overfitting in the training data. The output layer at the end uses the Softmax classifier, which predicts the transportation modes based on class membership probabilities.

4 Experiments

In this section, we explain the results of our experiments performed during our study. We start with the description of GPS and weather data sets used in our study. Next, we discuss the configurations used for our model. Finally, we will explain the results obtained by our experiments. For all these experiments, we used python version 3.7, scikit learn version 0.20.0 and deep learning library keras v.2.2.4 using tensor flow at the backend. We used ArcMap v.10.3 for analysing the GPS data.

4.1 Data set

For our experiments, we used Microsoft geolife data set v1.3 [19, 46, 47], which was collected by 182 users over a period of 5 years from April 2007 to August 2012. Out of these 182 users, 73 users labelled their trajectories. More than 90% of the GPS trajectories were logged in dense representation with the sampling rate of 1–5 s or at a distance of 5–10 m between two consecutive points. This variation in the sampling rate between 1 and 5 s or between 5 and 10 m does not affect the classification performance, because the user's behaviour does not frequently change in such a short duration and distance. However, on very few occasions, the behavioural features like the bearing rate may not be calculated precisely, but this loss in precision does not affect the classification performance of transportation modes. The data set is collected in 30 cities of China, and almost 70% of data is recorded in Beijing city, which has a complex road traffic network, and the data is recorded with dense representation as shown in Fig. 3. The distribution of different transportation modes in perspective of GPS points is shown in Fig. 4. It can be seen that the first five transportation modes from left to right have more than 200,000 GPS points, which is a reliable number to train the neural network. Based on these statistics, we only consider these modes in our experiments, because the performance and reliability of deep neural networks are dependent on the scale of the data [44]. Subway although has sufficient data available, but due to the signal issues in underground modes, these points are not recorded accurately. Taxi and car both have similar behaviour, so we treat them as a single-mode car. In addition to the GPS data, we also used the weather data set [48], and the global weather data is recorded with an interval of 3 h, from the year 1974 to present. We considered four transportation modes such as walk, bike, bus and car; however, the same model can work for more transportation modes subject to the availability of data.

There are a total of 9654 single-mode segments identified from the geolife data set. The distribution of a total number of trajectory samples for all transportation modes is given in Table 1.

4.2 Experimental setup

Inspired from the study [12], the GPS track of the user is divided into trips if the time duration between two consecutive points is

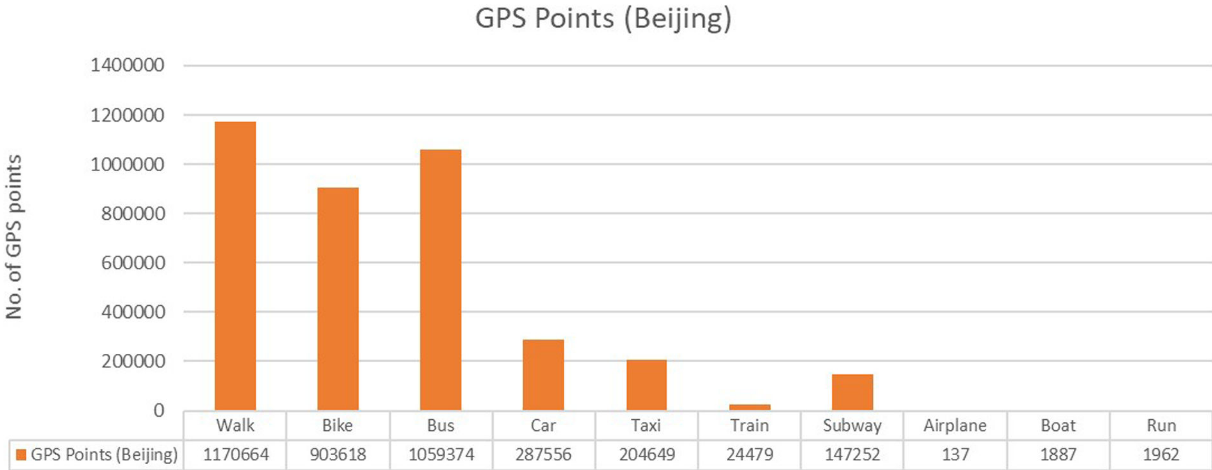

Fig. 4 *GPS data distribution for different classification modes*

Table 1 Transportation mode wise distribution of trajectories

Transportation modes	Number of segments
walk	2440
bike	2748
bus	3282
car	1184

Table 2 Features evaluation set

Features set	Features
basic features	speed, acceleration, jerk, bearing rate, time of the day, day of the week
enhanced features	regional index combined with basic features
all features	weather features combined with enhanced features

Table 3 Confusion matrix of ConvLSTM using basic features

ConvLSTM		Actual, %				Precision, %
		Walk	Bike	Bus	Car	
predict, %	walk	80.96	7.10	9.08	8.01	77.75
	bike	8.52	79.93	10.24	4.81	80.11
	bus	8.25	11.39	77.51	9.61	79.18
	car	2.26	1.58	3.17	77.56	79.87
recall, %		80.96	79.93	77.51	77.56	**79.15**

Bold values indicate the accuracies of the model.

>20 min. A trip is further divided into fixed-size segments, and the length of each segment is considered to be the median length for all the segments, each with a length of 200 GPS points [25]. We divided the geographical space into a 2D grid with 144 columns and 120 rows, where each grid cell is of size 250 m^2. The value of the time slice is either set to be idle or busy based on rush hours. The value of time slice is set to be busy if the time of day is between 7 and 10 h or in between 16 and 21 h, otherwise this is set as idle [21]. We design our model starting with the shallow network using 32 filters in the first layer and double the number of filters in each next layer of ConvLSTM. The filter size is set to 1×3 for a one-dimensional row vector. The full-length segment of 200 GPS points is transformed into 8 time steps and 25 columns for a 1 row vector. For a multi-class classification problem, the network is optimised using the cross-entropy loss function for computation of error in the output layer, and Adam version of stochastic gradient descent to update the parameters and adjusting the learning rate. We used the rectified linear unit activation function in all the layers to introduce non-linearity in the model, except for the last fully connected layer, in which we used the softmax activation function to compute the probabilities of all transportation modes. We used three ConvLSTM layers followed by two fully connected layers. All models are trained at 100 epochs. We used L2 regularisations, i.e. 0.01 and the probability of dropout is set to 0.5 to avoid overfitting. Parameters of all deep models are tuned through a set of different configurations during our experiments. In our experiments, we used a 70% training data set and 30% test data, and the same training and test set are used to have a fair comparison among all methods. The training time is acceptable for our proposed model ConvLSTM using both GPS and weather features. Our proposed model takes around 34 min for 100 epochs using the system Intel Xeon(R) CPU E5-2620 @ 2.10 GHz. Once the model is trained, the prediction time is almost the same for all the machine learning and deep learning models.

4.3 Features evaluations

In this section, we compare the performance of three different sets of features as shown in Table 2 using the proposed ConvLSTM model. The first set contains only GPS-based features named as basic features, the second set of features added one additional feature of region id with the basic features to analyse the impact of the region on learning transportation modes. The third set of features is a combined set of all GPS and weather-based features.

Tables 3–5 show the precision, recall and accuracy for all transportation modes using the ConvLSTM architecture on all three sets of features.

Table 4 Confusion matrix of ConvLSTM using enhanced features

ConvLSTM		Actual, %				Precision, %
		Walk	Bike	Bus	Car	
predict, %	walk	83.09	6.65	8.87	8.01	78.79
	bike	7.86	81.06	9.61	3.85	81.61
	bus	7.19	11.16	78.35	7.37	80.82
	car	1.86	1.13	3.17	80.77	82.35
recall, %		83.09	81.06	78.35	80.77	**80.67**

Bold values indicate the accuracies of the model.

Table 5 Confusion matrix of ConvLSTM using all features

ConvLSTM		Actual, %				Precision, %
		Walk	Bike	Bus	Car	
predict, %	walk	86.68	5.30	8.02	5.77	82.20
	bike	6.79	83.99	8.13	3.52	84.28
	bus	4.93	9.58	81.10	6.09	84.49
	car	1.60	1.13	2.74	84.61	84.61
recall, %		86.68	83.99	81.10	84.61	**83.81**

Bold values indicate the accuracies of the model.

Table 6 Deep neural net architectures comparison on transportation modes identification

Models	Avg. precision, %	Avg. recall, %	Avg. FScore, %	Accuracy, %
SVM (conventional features)	66.52	65.05	65.67	66.21
RF (conventional features)	72.22	71.66	71.89	72.17
MLP (basic features)	60.29	56.81	57.06	57.12
LSTM (basic features)	73.92	74.03	73.90	74.21
CNN (basic features)	77.77	76.25	76.90	76.73
ConvLSTM (basic features)	79.23	78.99	79.09	79.15
ConvLSTM (enhanced features)	80.90	80.82	80.83	80.67
ConvLSTM (all features)	83.89	84.10	83.97	83.81

It can be observed by comparing Tables 3 and 4 that the performance of ConvLSTM has slightly improved by adding one additional feature of region id. As the behaviour of transportation mode varies from one region to another, therefore the ConvLSTM is able to learn the distinct mobility pattern region wise. It can also be observed by comparing Tables 4 and 5 that the performance of ConvLSTM has further improved by >3%, by adding the auxiliary weather features with the GPS features. This performance can be further improved by incorporating more weather features like rain data, which is not available in MIDAS data set at this time [48]. These additional features can help to identify more distinct patterns in transportation modes. Although the values of rain attribute are missing in a weather table, the rain feature has, however, some direct relationship with other attributes like wind speed and dew point, because rain occurs at high dew point and lower wind speed, so the hidden layers in our model can identify such a hidden relationship through the hidden layers in the architecture.

4.4 Model evaluations

For the purpose of evaluation of our model, we compare the performance of the ConvLSTM network with the most widely used deep net models MLP, LSTM and CNN described in Section 3.1 as the benchmark models using a minimal set of features, named as basic features in our study. For comparing these architectures, we performed experiments using multiple configurations for each of these models. The comparison of best accuracies of these three architectures with similar configurations of three layers of the given architectures for feature extraction followed by two fully connected layers at the end for the purpose of classification. For MLP, we used three fully connected layers. In addition to that we also compare the performance with widely used and most successful conventional machine learning classification models, SVM and random forest (RF). As the conventional machine learning algorithm requires human engineered features for classification, we used the features from the study [19] including distance, maximum velocity and acceleration, average velocity, expectation and variance of velocity, heading change rate, stop rate and velocity change rate. In order to have a fair comparison, we used the same segments for training and test data for all these models. The accuracy of all these models is shown in Table 6.

We used average precision, average recall, average FScore and accuracy for all transportation modes as our evaluation metric. In Table 6, the performances of different classification methods are compared with the ConvLSTM architecture using basic features, and then the performance of ConvLSTM architecture is further compared using different set of features mentioned in Table 2. It can be seen from Table 6 that the ConvLSTM architecture produced better classification results as compared to other models in terms of all evaluation metrics, and almost achieved 3% improvement in accuracy as compared to the best benchmark model. We further analysed the impact of the region and weather on classifying transportation modes using the ConvLSTM, which results in further improvement of 1 and 3% in the accuracy of correctly identified transportation modes respectively.

5 Conclusion

We have surveyed the literature on detecting the mobility prediction and found certain limitations that we addressed to improve the prediction accuracies of transportation modes on raw GPS data. Our study focused on detecting the modes of transport using the GPS sensor data with the weather data. Our study also analysed the impact of the region on identifying the transportation modes. Existing studies solved this problem using different types of deep net architectures, but none of them are able to address the spatiotemporal data. We proposed the ConvLSTM architecture, which is able to address both the spatial and temporal dependencies to achieve full capability of spatiotemporal attributes of time-series GPS data. Our study has achieved the highest performance as compared to existing models. In our study, we consider four transportation modes; however, the same model will work for more transportation modes subject to availability of data. The accuracy of the model has improved using only few weather features; however, the accuracy can be further improved by incorporating more weather features for identifying more distinct patterns.

6 References

[1] Zhang, J., Zheng, Y., Qi, D., *et al.*: 'DNN-based prediction model for spatio-temporal data'. Proc. of the 24th ACM SIGSPATIAL Int. Conf. on Advances in Geographic Information Systems – GIS '16, New York, NY, USA, September 2016, pp. 1–4

[2] Wang, S., He, L., Stenneth, L., *et al.*: 'Estimating urban traffic congestions with multi-sourced data'. Proc. – IEEE Int. Conf. on Mobile Data Management, Porto, Protugal, 2016

[3] Wang, S., Zhang, X., Li, F., *et al.*: 'Efficient traffic estimation with multi-sourced data by parallel coupled hidden Markov model', *IEEE Trans. Intell. Transp. Syst.*, 2018, **20**, (8), pp. 3010–3023, https://ieeexplore.ieee.org/xpl/tocresult.jsp?isnumber=8782673

[4] Xu, J., Li, C., Wang, S., *et al.*: 'DTRP: a flexible deep framework for travel route planning'. Int. Conf. on Web Information Systems Engineering, Puschino, Russia, 2017, pp. 359–375

[5] Khosroshahi, A., Ohn-bar, E., Trivedi, M.M.: 'Surround vehicles trajectory analysis with recurrent neural networks'. IEEE 19th Int. Conf. on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 2016, pp. 2267–2272

[6] Wang, S., Zhang, X., Cao, J., *et al.*: 'Computing urban traffic congestions by incorporating sparse GPS probe data and social media data', *ACM Trans. Inf. Syst.*, 2017, **35**, (4), pp. 1–30

[7] Liu, Y., Seah, H.S.: 'Points of interest recommendation from GPS trajectories', *Int. J. Geogr. Inf. Sci.*, 2015, **29**, (6), pp. 953–979

[8] Tang, J., Jiang, H., Li, Z., *et al.*: 'A two-layer model for taxi customer searching behaviors using GPS trajectory data', *IEEE Trans. Intell. Transp. Syst.*, 2016, **17**, (11), pp. 3318–3324

[9] Fang, Z., Jian-yu, L., Jin-jun, T., *et al.*: 'Identifying activities and trips with GPS data', *IET Intell. Transp. Syst.*, 2018, **12**, (8), pp. 884–890

[10] Wang, L., Zhong, Y., Ma, W.: 'GPS-data-driven dynamic destination prediction for on-demand one-way carsharing system', *IET Intell. Transp. Syst.*, 2018, **12**, (10), pp. 1291–1299

[11] Besse, P.C., Guillouet, B., Loubes, J.-M.M., *et al.*: 'Destination prediction by trajectory distribution-based model', *IEEE Trans. Intell. Transp. Syst.*, 2018, **19**, (8), pp. 2470–2481

[12] Zheng, Y., Liu, L., Wang, L., *et al.*: 'Learning transportation mode from raw GPS data for geographic applications on the web'. Proc. of the 17th Int. Conf. on World Wide Web – WWW '08, New York, NY, USA, 2008, number 49, pp. 247–256

[13] Zheng, Y., Chen, Y., Li, Q., *et al.*: 'Understanding transportation modes based on GPS data for web applications', *ACM Trans. Web*, 2010, **4**, (1), pp. 1–36

[14] Xiao, Z., Wang, Y., Fu, K., *et al.*: 'Identifying different transportation modes from trajectory data using tree-based ensemble classifiers', *ISPRS. Int. J. Geoinf.*, 2017, **6**, (3), p. 22

[15] Jahangiri, A., Rakha, H.A.: 'Applying machine learning techniques to transportation mode recognition using mobile phone sensor data', *IEEE Trans. Intell. Transp. Syst.*, 2015, **16**, (5), pp. 2406–2417

[16] Fang, S.H., Fei, Y.X., Xu, Z., *et al.*: 'Learning transportation modes from smartphone sensors based on deep neural network', *IEEE Sens. J.*, 2017, **17**, (18), pp. 6111–6118

[17] Ballı, S., Arif Sağbaş, E.: 'Diagnosis of transportation modes on mobile phone using logistic regression classification', *IET Intell. Transp. Syst.*, 2018, **12**, (2), pp. 142–151

[18] Zheng, Y.: 'Trajectory data mining', *ACM Trans. Intell. Syst. Technol.*, 2015, **6**, (3), pp. 1–41

[19] Zheng, Y., Li, Q., Chen, Y., *et al.*: 'Understanding mobility based on GPS data'. Proc. of the 10th Int. Conf. on Ubiquitous computing – UbiComp '08, ACM New York, NY, USA, 2008, number 49, pp. 312–321

[20] Bolbol, A., Cheng, T., Tsapakis, I., *et al.*: 'Inferring hybrid transportation modes from sparse GPS data using a moving window SVM classification', *Comput. Environ. Urban Syst.*, 2012, **36**, (6), pp. 526–537

[21] Zhu, Q., Zhu, M., Li, M., *et al.*: 'Identifying transportation modes from raw GPS data'. Int. Conf. of Pioneering Computer Scientists, Engineers and Educators, Springer, Singapore, August 2016, vol. 623, pp. 395–409

[22] Gonzalez, P.A., Weinstein, J.S., Barbeau, S.J., *et al.*: 'Automating mode detection for travel behaviour analysis by using global positioning systems-enabled mobile phones and neural networks', *IET Intell. Transp. Syst.*, 2010, **4**, (1), p. 37

[23] Endo, Y., Toda, H., Nishida, K., *et al.*: 'Deep feature extraction from trajectories for transportation mode estimation'. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Springer, Cham, Switzerland, April 2016 (LNAI, **9652**), pp. 54–66

[24] Wang, H., Liu, G.J., Duan, J., *et al.*: 'Detecting transportation modes using deep neural network', *IEICE Trans. Inf. Syst.*, 2017, **E100D**, (5), pp. 1132–1135

[25] Dabiri, S., Heaslip, K.: 'Inferring transportation modes from GPS trajectories using a convolutional neural network', *Transp. Res. C, Emerg. Technol.*, 2018, **86**, pp. 360–371

[26] Liu, H., Lee, I.: 'End-to-end trajectory transportation mode classification using Bi-LSTM recurrent neural network'. Proc. of the 2017 12th Int. Conf. on Intelligent Systems and Knowledge Engineering, ISKE 2017, Nanjing, People's Republic of China, 2018, vol. 2018-Janua, pp. 1–5

[27] Shi, X., Chen, Z., Wang, H., *et al.*: 'Convolutional LSTM network: a machine learning approach for precipitation nowcasting'. 29th Annual Conf. on Neural Information Processing Systems (NIPS), Montreal, Quebec, Canada, 2015, vol. 28, pp. 802–810

[28] Zapata-Impata, B.S., Gil, P., Torres, F.: 'Learning spatio temporal tactile features with a convLSTM for the direction of slip detection', *Sensors (Switzerland)*, 2019, **19**, (3), pp. 1–16

[29] Stenneth, L., Wolfson, O., Yu, P.S., *et al.*: 'Transportation mode detection using mobile phones and GIS information'. Proc. of the 19th ACM SIGSPATIAL Int. Conf. on Advances in Geographic Information Systems – GIS '11, Chicago, IL, USA, 2011, pp. 54–63

[30] Gong, H., Chen, C., Bialostozky, E., *et al.*: 'A GPS/GIS method for travel mode detection in New York City', *Comput. Environ. Urban Syst.*, 2012, **36**, (2), pp. 131–139

[31] Brum-Bastos, V.S., Long, J.A., Demšar, U.: 'Weather effects on human mobility: a study using multi-channel sequence analysis', *Comput. Environ. Urban Syst.*, 2018, **71**, pp. 131–152

[32] Wang, B., Gao, L., Juan, Z.: 'Travel mode detection using GPS data and socioeconomic attributes based on a random forest classifier', *IEEE Trans. Intell. Transp. Syst.*, 2018, **19**, (5), pp. 1547–1558

[33] Bantis, T., Haworth, J.: 'Who you are is how you travel: a framework for transportation mode detection using individual and environmental characteristics', *Transp. Res. C*, 2017, **80**, pp. 286–309

[34] Asci, G.: 'A novel input set for LSTM-based transport mode detection'. 2019 IEEE Int. Conf. on Pervasive Computing and Communications Workshops (PerCom Workshops), Kyoto, Japan, 2019, pp. 107–112

[35] Liang, J., Zhu, Q., Zhu, M., *et al.*: 'An enhanced transportation mode detection method based on GPS data'. International Conference of Pioneering Computer Scientists, Engineers and Educators, Springer, Singapore, September 2017, pp. 605–620

[36] Kasahara, H., Iiyama, M., Minoh, M.: 'Transportation mode inference using environmental constraints'. Proc. of the 11th Int. Conf. on Ubiquitous Information Management and Communication, IMCOM 2017, Beppu, Japan, 2017, pp. 1–8

[37] Lin, M., Hsu, W.J., Lee, Z.Q.: 'Detecting modes of transport from unlabelled positioning sensor data', *J. Location Based Services*, 2013, **7**, (4), pp. 272–290

[38] Rudloff, C., Ray, M.: 'Detecting travel modes and profiling commuter habits solely based on GPS data', Transp. Res. Board 89th Annu. Meeting, Washington, DC, USA, 2010, pp. 1–15

[39] Ma, L., Xiong, H., Wang, Z., *et al.*: 'Impact of weather conditions on middle school students ' commute mode choices: empirical findings from Beijing, China', *Transp. Res. D, Transp. Environ.*, 2019, **68**, pp. 39–51

[40] Yang, X., Stewart, K., Tang, L., *et al.*: 'A review of GPS trajectories classification based on transportation mode', *Sensors (Switzerland)*, 2018, **18**, (11), pp. 1–20

[41] Xiao, G., Juan, Z., Zhang, C.: 'Travel mode detection based on GPS track data and Bayesian networks', *Comput. Environ. Urban Syst.*, 2015, **54**, pp. 14–22

[42] Das, R.D., Winter, S.: 'A fuzzy logic based transport mode detection framework in urban environment', *J. Intell. Transp. Syst., Technol, Plan. Oper.*, 2018, **22**, (6), pp. 478–489

[43] Nguyen, H., Kieu, L.-M., Wen, T., *et al.*: 'Deep learning methods in transportation domain: a review', *IET Intell. Transp. Syst.*, 2018, **12**, (9), pp. 998–1004

[44] Lv, Y., Duan, Y., Kang, W., *et al.*: 'Traffic flow prediction with big data: a deep learning approach', *IEEE Trans. Intell. Transp. Syst.*, 2015, **16**, (2), pp. 865–873

[45] Lecun, Y., Bengio, Y., Hinton, G.: 'Deep learning', *Nature*, 2015, **521**, pp. 436–444

[46] Zheng, Y., Zhang, L., Xie, X., *et al.*: 'Mining interesting locations and travel sequences from GPS trajectories'. Proc. of the 18th int. Conf. on World Wide Web – WWW '09, New York, NY, USA, 2009, pp. 791–800

[47] Zheng, Y., Xie, X., Ma, W.: 'Geolife: a collaborative social networking service among user, location and trajectory', *IEEE Data Eng. Bull.*, 2010, **33**, (2), pp. 32–40

[48] Met Office: 'MIDAS: global weather observation data. NCAS British atmospheric data centre', 2006. Avtailable at https://catalogue.ceda.ac.uk/uuid/0ec59f09b3158829a059fe70b17de951