A Simulation Study on Automated Transport Mode Detection in Near-Real Time using a Neural Network

Rahul Deb Das
rahuld@student.unimelb.edu.au

Nicole Ronald
nicole.ronald@unimelb.edu.au

Stephan Winter
winter@unimelb.edu.au

Department of Infrastructure Engineering
The University of Melbourne, Australia

Abstract

Detecting transport modes in near-real time is important for various context-aware location based services and understanding urban dynamics. In this paper we present a simulated study on detecting transport modes in near-real time using a neural network. We have shown how detection accuracy will vary with different temporal window sizes and different combination of modes. Since in urban environment transport modes move slowly due to traffic, considering movement attributes or kinematics alone for mode detection is not sufficient. That is why we investigated how spatial information can improve mode detection accuracy. The model has achieved 82%-95% accuracy using different simulation designs and proves its efficacy over other detection models.

1 Introduction

Transport mode detection from trajectories has seen growing interest in research over last few years for its importance in various domains such as context-aware computing, location based services, understanding urban dynamics, travel demand surveys, traffic monitoring, and travel behaviour analysis. Traditionally travel modes have been surveyed in questionnaires, enabling also to capture additional knowledge including purpose of trip. Travel surveys, however, are burdensome, erroneous if made from memory, of low spatial detail, and reach only small sampling rates. Automation should overcome all these issues.

Since the late 1990's, due to advancements in positioning and navigation technology, GPS started being used as a mean to collect travel data and assess its reliability and future possibilities (Wolf, 2000). Eventually, the use of GPS has increased as it has become more precise, portable and ubiquitous. Nowadays people themselves can track their movement trajectories using GPS and potentially other sensors on-board their smart phones (Periera et al., 2013).

Most of the research on travel mode detection is based on rigid velocity based model. However, a velocity based approach is not always sufficient. For example, low speed conditions, which are nowadays typical in urban traffic due to traffic at capacity, or bad weather, produce mode ambiguities. In low speed traffic conditions, the speed of a bus is similar to a car or bicycle. Therefore, there is a need to consider various non-kinematic attributes along with movement attributes (kinematics) in order to detect different transport modes.

In: B. Veenendaal and A. Kealy (Eds.): Research@Locate'15, Brisbane, Australia, 10-12 March 2015, published at http://ceur-ws.org

Existing transport mode detection research is mostly offline. That means modes are detected once a trip is completed from historical trajectories. Existing methods use the entire trip record in the form of trajectory and then separates the trajectory based on walking based segments into number of meaningful parts that correspond to respective transport modes.

The hypothesis behind this research is that a neural network based model can adjust well in real time with varying movement behaviour and overcomes the mode ambiguity under low speed conditions.

In detecting transport modes, movement characteristics derived from trajectories of the users are the raw data source. In this paper we will concentrate on trajectories of a single sensor, as provided by GPS enabled smart phones. Such GPS trajectories are unlabelled and come in raw format. Other sensors in the phone are neglected for the time being, but can easily be included in the model. A classifier is required that can detect the various transport modes used along each trip in real to near-real time. In this paper, a neural network based classifier has been tested. Our contributions are as follows-

1) We developed a simulated near-real time transport mode detection model based on a multi-layer perceptron neural network.

2) Earlier approaches to neural network based transport mode detection are mostly offline and did not use any spatial information. In this research we show how spatial information can increase the detection accuracy.

3) Selecting a proper temporal window for detecting transport modes in near-real time is critical and context dependent. In this paper we investigate how detection accuracy varies with different temporal window sizes, which helps in selecting a proper window size based on accuracy requirement.

In this paper we also evaluate the performance measure of a multi-layer perceptron neural network in order to detect transport modes in near-real time. A real time model can detect the transport mode epoch by epoch basis (such as second by second). In this research, we simulated queries within short temporal window to detect a given mode instead of second by second basis. Hence, we call this model a near-real time mode detection approach.

Detecting transport mode in near-real time is comparatively an emerging research area. In this paper we have developed a basic but intuitive near-real time mode detection model using a supervised learning approach. Real time mode detection can be useful for a number of applications. Applications include various context-aware location based services where the context could be a given transport mode. A petrol pump can distribute an electronic discounted coupon within its neighbourhood to all the private cars only. Detecting transport modes in real time can also help developing various context-aware mobile applications that can sense the modality and act accordingly. One instant could be developing a mode-dependent auto-answering service on smart-phones. If the mobile senses the owner is in driving mode then the auto-answer can automatically be enabled and helps driver to concentrate on the road rather than receiving any incoming call. Thereby this can help in reducing distractions on the road in order to reduce road accidents. This approach can also be helpful for urban planners or emergency service providers who want to know people's mode choice at a given route or in a given region at a given time window for modeling travel demand or various spatio-temporal events.

The paper is organized as follows. Section 2 discusses related works in transport mode detection from various perspectives. Section 3 discusses some of the basic terminologies and methodology. Section 4 demonstrates data preparation and experimentation. Section 5 shows the experimental results. Section 6 presents the discussion of these results, and Section 7 concludes the paper.

2 Related work

Nowadays, smart-phones come with GPS enabled facilities. Since smart-phones are carried by the users almost everywhere and all the time, hence, this positioning facility can be utilized in order to collect trajectories without any external intervention. Once a GPS trajectory has been collected or is in the process of being collected, those trajectories or part there-of can be used for transport mode detection in real time or post-processing mode. In this regard, existing work is mostly based on post-processing of the trajectories or detecting modes offline. Existing literature shows a wide variety of post-processing algorithms and classifiers. Some of the approaches used the classification technique directly without segmenting the GPS trajectories (Byon et al., 2007; Dodge et al., 2009; Reddy et al., 2010). At the same time there are approaches applying segmentation of the entire trajectory into meaningful parts, corresponding to different modes, before classification (Mountain and Raper, 2001; Tsui and Shalaby, 2006; Schussler and Axhaussen, 2009; Zheng et al., 2010; Biljecki et al., 2012; Hemminki et al., 2013).

Segmentation is done based on those points that show high probability of mode change. Mountain and Raper used change in speed and direction for segmentation in their work (Mountain and Raper, 2001). However, this approach creates ambiguities in certain cases where the vehicles move slowly and constrained to specific roads or the rail or tram networks. Liao et al. used proximity to potential change points, such as bus stops or train stops for offline mode detection (Liao et al., 2007). However, GPS accuracy greatly varies in urban environments, depending on the number of satellites in view, time of the day and season, atmospheric conditions and surrounding sources of multipath effects. Other research used change in peaks of acceleration curves in order to segment the trajectory (Hemminki et al., 2013). However this approach also suffers from low ambiguity resolution, typically in low speed condition such as during bad weather or traffic congestion. Another common and intuitive way for segmenting the trajectory is based on detecting walking segments. The rationale behind this approach is the observation that a person generally walks between using two modes of transport. This approach has achieved promising results for segmentation (Tsui and Shalaby, 2006; Zheng et al., 2010; Biljecki et al., 2012). However this approach also fails when there is a quick mode change or walking is negligible.

There have been a number of different algorithms for mode classification used so far. Zheng and colleagues used a decision tree, Bayesian Net, Conditional Random Field (CRF) and Support Vector Machine (SVM) in their work with 75% reported accuracy (Zheng et al., 2010). Gonzalez et al. used neural networks with 91% accuracy (Gonzalez et al., 2010). Some works are solely based on statistical measures (Patterson et al., 2003).

As far as the input parameters or indicators are concerned, prior work mostly concentrated on velocity attributes (Bohte et al., 2008; Schussler and Axhausen, 2009). But in low speed condition velocity and acceleration are not sufficient to resolve the ambiguities. So, more recently, research has incorporated additional movement attributes including heading rate change and stop rates (Zheng et al., 2010). Vibration data has also been tested as an additional attribute with promising results (Ohashi et al., 2013). However, in order to achieve better accuracy and account for GPS signal loss others have used inertial localization and navigation sensors such as accelerometers, along with GPS sensors (Reddy et al., 2010; Hemminki et al., 2013; Ohashi et al., 2013).

Byon and colleagues used GPS trajectories collected by GPS loggers to study detection accuracy in real time. However there focus was mainly on how accuracy varies with different sampling frequencies (Byon et al., 2009). They achieved high detection accuracy at 20 min temporal window. However they observed mainly four modes auto, walk, car, bus. Although Byon and colleagues developed two neural network models, one route specific and another one a universal model, they did not explore how spatial knowledge can help in detecting different modes. Also their approach is limited by their use of GPS loggers: they used instantaneous speed, acceleration, number of satellites in view for a given transport mode to train their classifier. Number of satellites in view depends on particular transport mode. Such as GPS device inside a bus is obstructed by the metallic body and ceiling and vertical windows limiting the number of satellites in view. Whereas a car would have wider front windshield that would allow stronger and multiple GPS signals. However when using smart-phones for detecting modes, instantaneous acceleration, number of satellites in view or horizontal dilution of precision values may not be available.

Gonzalez and colleagues developed a neural network based mode detection model with a core focus on how to reduce streaming of movement data. Earlier work used a static and fixed data transmission procedure but that suffered from high financial costs associated with data transmission as well as computational overhead and storage issues. Gonzalez and colleagues proposed a novel critical point (CP) algorithm to transmit only the relevant GPS points during the trip (Gonzalez et al., 2010).

Since movement states are uncertain and imprecise there are a couple of mode detection appraoch using fuzzy logic (Tsui and Shalaby, 2006; Biljecki et al., 2010). A fuzzy approach with three criteria and five to ten modes has been tested with an accuracy of more than 90% (Schussler and Axhausen, 2009; Biljecki et al., 2010). However these approaches are rule-based and involve fuzzy antecedents and fuzzy consequents (Zadeh, 1965; Mamdani and Assilian, 1975). This approach cannot adapt with different movement behaviour in real time. Since fuzzy logic based models are developed based on expert knowledge with predefined premise and consequents hence they are not scalable with new parameters and thus pose scalability and flexibility issues. In this paper we present a neural network based model that can learn in real time. A neural network based model is flexible and scalable.

3 Theory

In this section we presented some basic definitions, concepts and methodology used in this research.

3.1 Raw Trajectory

A raw trajectory is a set of spatio-temporal points arranged in a chronological order. This can be mathematically expressed as

$Tr = \{P_i\}: P_i = (x_i, y_i, z_i, t_i)\ ;\ i \in [0, N]\ ; \forall\ i : (t_i < t_{i+1})$ …………………………………………… (1)

3.2 Segment

Any connected part of a raw trajectory with a specific semantics is a segment. For example, if a part of trajectory is extracted with a given annotated mode, then that is a modal segment. Similarly, if certain part(s) of a trajectory is extracted over a given time period that part would be a temporal segment.

3.3 Model architecture

The architecture of a multi-layer feed forward back propagation neural network is explained as follows-

A multi-layer feed forward neural network consists of mainly three layers- a) input layer, b) hidden layer, c) output layer. These layers contain one or more than one nodes or neurons. Input nodes are connected to hidden nodes and hidden nodes are connected to output nodes. But nodes of the same layer have to be disjoint and they cannot be connected to each other. Input layer is responsible to get input signals from the external world typically in the form of movement attributes (kinematics) or spatial attributes (non-kinematics) in the context of transport mode detection.

3.4 Training

Neural network can learn online and adapt well with given instances. However before using a neural network, it has to be trained to map a given set of inputs to a given output class. The training typically starts from input layer as soon as input stimuli are fed in. Nodes in each layer receive input signal from the preceding layer and send an output signal to the nodes in immediate succeeding layer. Each node multiplies the input signal with a previously established weight, adds a threshold, converts into an output signal through an activation function and sends it to the other nodes in the succeeding layer. The hidden layer is not directly connected to real world. This is the most important layer that processes the information and creates categorizing features for classification which sends signal to the output layer to categorize a given set of feature vectors. Once the output signal produces a response it is evaluated with the actual response. The difference between the predicted response and desired response is the error term of the neural network which is then back propagated to the model in order to adjust the weight and threshold values iteratively. This iterative process goes on in a cyclic way until a prescribed number of cycles (epochs) or a desired error level is achieved during training phase.

The rate at which a neural network learns can be adjusted by changing certain parameters called learning rate (LR) and momentum (M). These parameters control the change in weight and their persistence throughout total number of epochs.

3.5 Near-real time simulation

In order to detect transport modes in near-real time, queries will be fetched to a central server with kinematic and non-kinematic information. In this research we used a set of historical trajectories for near-real time simulation purpose. In order to train a neural network model small temporal segments have been extracted from the trajectories. Kinematic and non-kinematic attributes are then estimated over that temporal window in order to capture various movement behaviour of the given mode within that time period. Since a transport mode can exhibit different movement behaviour at different instant hence there is a need to train the classifier with movement behaviours for each mode at different instant of time over different trajectories. In order to extract movement behaviour of each transport mode, temporal segment over a given temporal window of a given mode segment has been extracted at regular interval of time.

3.6 Mode segmentation

Each trajectory can be expressed as a set of modal segments. This can be expressed as

$T=\{SM_j\}$ ………………………………………………………………………………………… (2)

Where, $j \in [1, N]$; N= total number of modes used by the user over the trajectory

Each modal segment can be expressed as $SM_j= \{P_{ij}, M_{ij}\}$……………………………………… (3)

where i= i^{th} spatio-temporal index; j= j^{th} modal segment index; SM_j= j^{th} segment of the trajectory; P_{ij}= i^{th} patio-temporal point in j^{th} segment; M_{ij}= mode for i^{th} point in j^{th} segment.

A modal segment can be divided into a number of overlapping temporal segments. That can be mathematically expressed as -

$SM_j=\{TW_{kj}\}^t : k \in [1, M]$ ………………………………………………………………..….. (4)

Where, TW_{kj}= k^{th} temporal segment of j modal segment over time window t; M= total number of temporal segments over a given modal segment.

3.7 Temporal segmentation

Once a trajectory is segmented into number of modal segments then each modal segment is segmented in number of temporal segments overlapping by (n-1) spatio-temporal points, where n is the number of spatio-temporal points in a given temporal segment. The overlap is chosen as n-1 in order to capture diverse movement behaviour of the given mode at a finer granularity.

Each temporal segment can be expressed as $TW_{kj}^{t}=\{P_{ijk}, M_{ijk}\}$ ……………………………. (5)

Where, TW_{kj}^{t}= k^{th} temporal segment of 't' time length in j^{th} modal segment
P_{ijk}= i^{th} spatio-temporal point in k^{th} temporal segment of j^{th} modal segment
M_{ijk}= Annotated mode in k^{th} temporal segment of j^{th} modal segment

3.8 Kinematics and spatial information

In this research eight kinematic attributes are estimated using Euclidean functions in space-time domain such as average speed, average acceleration, variance of speed, variance of acceleration, maximum speed, maximum acceleration, minimum speed and minimum acceleration. In order to understand how a given mode behaves spatially with respect to different spatial objects (route network or POI), eight spatial relevance measures have been considered (see Table 2). Spatial relevance with respect to different spatial objects is calculated based on spatial proximity of spatio-temporal points to the given spatial object or a part thereof.

3.9 POI relevance estimation

In order to estimate POI relevance (proximity to bus stop, train stop, traffic signal or car wash or parking lot) a density-based clustering kernel is ran over each temporal window. Then POI relevance over a given temporal segment is estimated as

$POIRel_{c}=POIRel_{c-1}+ s*(n/N)$ ……………………………………………………….…….. (6)
Where, $POIRel_{c}$= Relevance measure of a given POI over cluster 'c' over temporal window [t1, t2]
$POIRel_{c-1}$= Relevance measure of a given POI over cluster 'c-1' over temporal window [t1, t2]
s= scaling factor (s=10 in this case)
n= number of elements in the cluster falling in the search radius of the given POI
N= total number of the elements in the cluster

3.10 Instance formation

In order to train and test the model using N-fold cross-validation, instances are created in the form of feature vectors which include kinematics and spatial attributes estimated over each temporal segment and fed into the model. A flowchart is given to show the workflow in Figure 1.

3.11 Performance measure

In order to evaluate the performance of the model, we used N-fold cross-validation. Since in N-fold cross-validation all the feature space is used using N-1 as training and 1 set of feature vectors as test thus it can capture the state behaviour at a fine granularity. However in hold-back type training, the accuracy of the model depends on the percentage of training instances that can represent all the details and characteristic behaviour of the entire population. Since in real time mode detection instances may vary with a temporal window size and modal movement behaviour, hence performance measure of a N-fold cross-validation strategy has been presented in this research (see experiment and result section), assuming an iterative N-fold cross-validation over growing time can dynamically improve the model in near-real time.

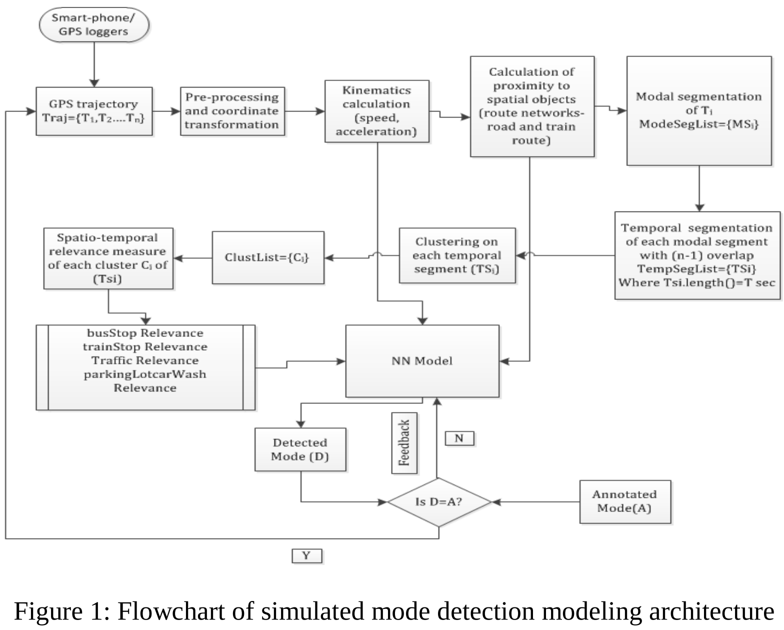


Figure 1: Flowchart of simulated mode detection modeling architecture

4 Data preparation and experimental setup

In order to evaluate our hypothesis Microsoft's GeoLife datset has been used in this experiment (Zheng et al., 2008a; Zheng et al., 2008b; Zheng et al., 2010). The dataset mainly covers Beijing CBD and its surrounding suburb. The dataset was collected by smart-phones and GPS loggers in the form of GPS trajectories. Different sampling intervals ranging from 2-5 seconds have been used in this dataset. Users provided their trajectories and ground truth separately. This dataset contains various transport modes such as car, walk, bus, taxi, train, subway and bike. However, no accuracy measure such as horizontal dilution of precision (HDOP) is provided in this dataset. The dataset also suffers from semantic gaps due to both technical reasons such as urban canyons, indoor environments and misreporting such as time gap between annotating two different mode segments by a user. During pre-processing stage some of the GPS points are found to be outside the study area. There were also some inconsistencies in the annotations, such as walking over unreasonable longer duration or with unreasonable speed. There are also semantic gaps during signal loss and missing annotations in the dataset. The portions of the dataset containing such semantic inconsistencies are discarded. However a future work can use signal loss and contextual information to detect certain modes such as train in a subway.

Figure 2: Beijing GeoLife GPS dataset overlaid on the road network (in blue colour) and rail network (in red colour)

In a data filtration stage 2.5 m/s has been set as walking speed threshold (Minetti, 2000), and some of the trajectories are discarded. In this experiment 264 trajectories have been used including training and test trajectories (Fig. 2).

Earlier works used HDOP value and can easily filter noise points (Byon et al., 2009; Gonzalez et al., 2010), but in this dataset we do not have any information that can provide positional accuracy or confidence level for each GPS fix. Hence, we setup two different experimental designs. One with filtered GPS data points where walking speed more than 2.5 m/s have been removed. Another setup was used without any filtering walking speeds. In both cases, raw velocity values are smoothed using an inverse distance weightage (IDW) smoothing kernel.

However technically a segment can also be viewed as a trajectory if it is treated discretely for further analysis. In order to detect transport modes in near-real time, we used a portion of a historical GPS dataset with transport mode annotated to train the mode detection model. Then we generated queries randomly at different instant of the trajectory and fetch the queries to the model to detect the transport mode as if the queries are coming in near-real time. A multi-layer perceptron (MLP) neural network has been realized in this research in order to detect various transport modes. The reason neural network has been investigated in this research because neural network is flexible, easily scalable and most importantly, it can learn online and adapt well.

In this research, five transport modes are considered: car, walk, bus, train and bike. Since car and taxi are difficult to distinguish especially in near-real time hence car and taxi are both grouped as car for time being. However in future car and taxi can be treated separately depending on the availability of contextual information. Similarly, train or light rail and subway are grouped as train. A multi-layer perceptron (MLP) neural network has been modelled using Weka, a Java based open source machine learning package. Since the time window is a critical factor in near-real time mode detection hence different temporal window size has been evaluated such as 120 sec, 180 sec, 240 sec, 300 sec and 600 sec based on subjective judgement. Experiments are also set up using only kinematic information and spatial information along with kinematics. For kinematic information eight movement attributes over a given temporal window have been considered since different modalities may exhibit different movement behaviour (Table 1). When kinematic attributes are used a 8-6-5 MLP was formed, and using spatial and kinematic attributes a 16-10-5 MLP model was used to detect different transport modes (Fig. 3). A popular approach to select number of hidden nodes can be calculated as the closest integer value of [(input nodes+target nodes)/2]. Hence we selected 6 hidden nodes when input nodes are 8 and output nodes are 5. Likewise, for 16 input nodes and 5 target nodes, number of hidden nodes are 10.

Table 1: Kinematic attributes

Attribute	Relevance
Average speed (avgSpeed)	Central tendency of a temporal segment in order to approximate a characteristic movement behaviour
Average acceleration (avgAccl)	
Variance of speed (varSpeed)	Spread of movement behaviour over the temporal window
Variance of acceleration (varAccl)	
Maximum speed (maxSpeed)	Upper bound of respective movement attributes within a given temporal window
Maximum acceleration (maxAccl)	
Minimum speed (minSpeed)	Lower bound of respective movement attributes within a given temporal window
Minimum acceleration (minAccl)	

For spatial information eight spatial attributes including proximity to route network and different POIs over a given temporal window have been considered (Table 2).

Table 2: Spatial attributes

Attribute	Relevance
Average road proximity (avgRoadProx)	Central tendency of proximity distribution over a given temporal window
Average railway proximity (avgTrainProx)	
Variance of road proximity (varRoadProx)	Spread of proximity distribution over a given temporal window
Variance of railway proximity (varTrainProx)	
Relevance score for bus stop (busRel)	Relevance measure of each relevant cluster from a given POI based on spatial proximity
Relevance score for train stop (trainRel)	
Relevance score for traffic stop (trafficRel)	
Relevance for parking lot and car wash (plcwRel)	

In order to study the performance of neural network through different training strategies, the model has been realized through N-fold cross-validation (where N=10). Table 3 shows number of instances used in N-fold cross-validation for different time window.

Table 3: Time window vs instances

Time window	Instances
120	15060
180	13735
240	12600
300	11488
480	9166
600	7835

5 Experimental results

The performance of the model has been evaluated on five modes against different temporal window size using filtered walking speeds in order to compare with the existing works that used positional uncertainty information. An experiment has also been carried out without filtering walking speeds assuming in real time people may run instead of walking during mode transfer or GPS positions can be subjected to various errors leading to walking speed greater than any threshold value.

The model was trained and tested using 10-fold cross-validation. In the first stage accuracy was tested against different temporal window sizes on trajectories where walking speeds are filtered. In the second stage accuracy was evaluated without filtering the trajectories in order to simulate real time mode detection. In both cases, the model shows that using spatial information can easily outperform the accuracy produced by only kinematics attributes. The reason behind this is that all transport modes may move slowly during traffic congestion or bad weather which leads to mode ambiguities. Figure 4 shows how mode ambiguity may arise using only acceleration measure. In this figure different modes may not be distinguished from their acceleration since they are clustered around the similar acceleration measures (Fig. 4).

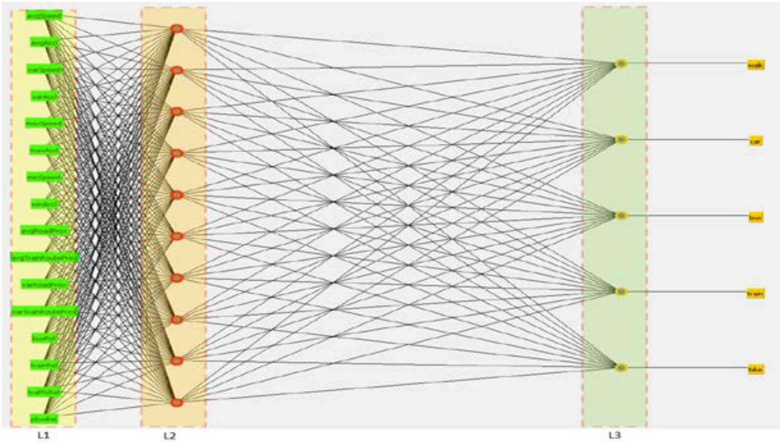


Figure 3: A 16-10-5 MLP neural network based mode detection model. L1 indicates input layer, L2 indicates hidden layer and L3 indicates output layer.

But when spatial relevance, in particular proximity to route network or given POI relevance, is considered the mode has been detected more accurately. In figure 5, the bus mode shows high bus stop relevance and hence bus mode is more prominent from other modes. However since walking can take place anywhere over the footpath near the bus stop hence, some of the walking instances have shown high bus stop relevance owing to false positives (Fig. 5). The rational is, a car can travel like a train with similar speed and acceleration but the underlying route network would be different and POI relevance will also vary accordingly.

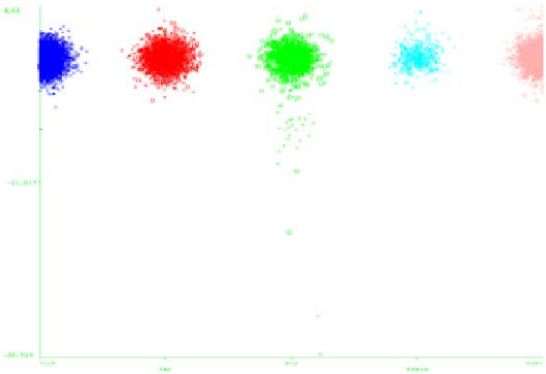


Figure 4: Mode ambiguities from similar acceleration distribution. Class map: {blue: walk; red: car; green: bus; indigo: train; brown: bike}. X-axis: modes; Y-axis: acceleration value

From the accuracy measures it is clear that there is a trade-off between temporal window size and mode detection accuracy. Selecting an optimal window size is context dependent. Overall 300 sec seem to an optimal window size for near-real time mode detection as the accuracy starts increasing gradually from this point and the accuracy measure is more than 82% for unfiltered trajectories and 86% for filtered trajectories (Fig. 6; Fig. 7).

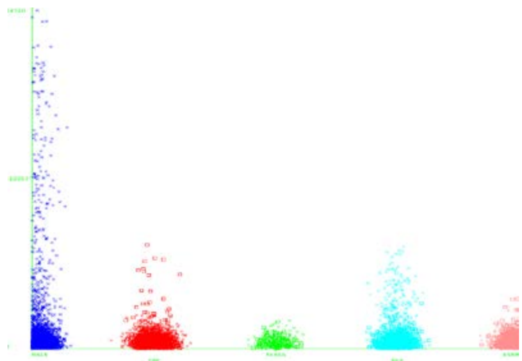
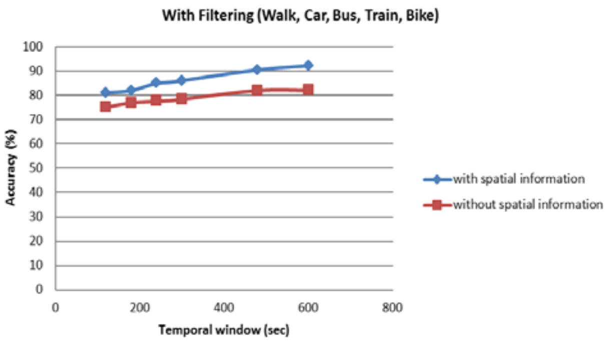


Figure 5: Bus stop relevance vs different modes. Class map: {blue: walk; red: car; green: train; indigo: bus; brown: bike}; X-axis: modes and Y-axis: bus relevance



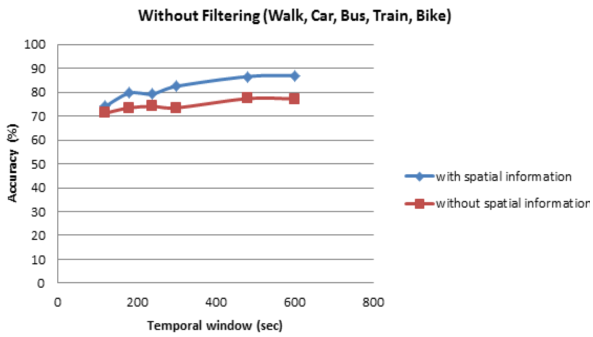


Figure 6: Accuracy measure with filtered walking speed

Figure 7: Accuracy without filtered walking speed

The methodology was also tested on car, walk and train assuming these modes will show quite distinct behaviour in terms spatial relevance as well as kinematic relevance. When spatial information was used accuracy reached 95% for filtered trajectories and 93% for unfiltered trajectories (Fig. 9). A spatial visualization is also presented to show the classification accuracy for three modes (train, car and walk). The diagonals are true positives and off-diagonals are false positives (Fig. 8). The figure shows the model can give a high accuracy and less type I and type II error for walking. However due to similar kinematic behaviour some of the car instances are mostly classified as walk owing to type I error. Likewise train instances are sometimes classified as walk and car.

In order to compare state-of-the-art approaches that used only kinematic information, another test was conducted within temporal window of 300 sec, on car, bus and walk modes. It was found there was a small difference in estimated accuracy by using only kinematics inputs, and kinematics and spatial inputs together. However using spatial information and kinematics, the accuracy is certainly more than that of using kinematics alone. The small difference of accuracy can be justified as the bus network has not been used in this research; only the road network was used. A car or bus both can travel on road network and hence it was not easy to distinguish between car and bus. But there is a good chance that car and bus can be easily distinguished by using a bus network. Using spatial information average accuracy for car, walk and bus was achieved 81.24 % whereas without spatial information the accuracy was 79.50 %.

We also compared the performance of our MLP neural network with some of the well-studied machine learning algorithms. The result shows a MLP neural network outperforms other approach. Interestingly the accuracy of a MLP neural network increases as the size of the time window increases whereas other approaches show saturation over growing time window. This clearly shows the ability of a MLP to learn and adapt well in near-real time as more instances come in with fine and varied state behaviour of different modes (Table 4).

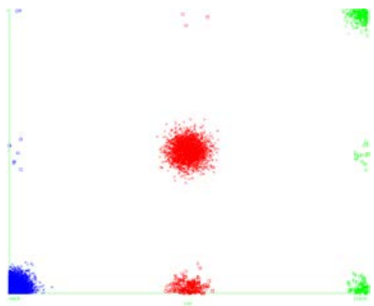


Figure 8: Classification accuracy visualization. Class map: {blue: walk; red: car; green: train}; X-axis indicates actual modes and Y-axis indicates predicted modes

6 Discussions

From the result it is evident that spatial information can improve mode detection accuracy significantly, especially in near-real time. In near-real time detecting different transport modes is challenging from their movement attributes only, as the queries are issued for a very short interval and different modes may have similar movement behaviour. Earlier literature did not consider minimum speed and acceleration as all of them are offline and based on segmenting the entire trajectory in each modal segment that normally starts with zero speed and zero acceleration (Gonzalez et al., 2010).

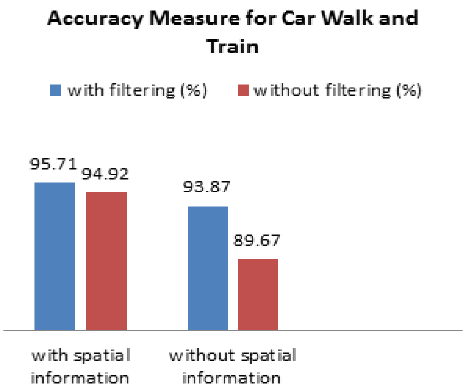


Figure 9: Accuracy measure for three modes

Table 4: Accuracy measures for different classifiers against different time window

Time window (sec)	SVM (%)	MLP Neural network (%)	Logistic regression (%)	RBF Network (%)
120	59.10	74.08	68.51	58.77
180	60.65	79.65	70.32	60.37
240	64.15	79.34	72.51	64.15
300	61.78	82.55	75.20	66.89
480	66.84	86.63	77.86	68.95
600	68.35	87.05	77.76	66.87

But in real time since a query can come at any time and the given mode need not to start or finish from stationary state within the given temporal window, hence minimum and maximum bounds, central tendency and variance of speed and acceleration distribution over the given time window have been considered. In order to supplement the mode detection accuracy various spatial information have been considered assuming the fact that each mode shows distinct spatial behaviour, such as bus and car will travel along the roadway; likewise train will travel on train network only. However, in this research the bus network of Beijing was not used, instead the entire road network consisting of the bus network and roads together was used. In order to distinguish different modes especially bike, car and bus different POI relevance has been considered such as bus stop for bus, traffic for bike, bus and car, car wash and parking lot for car. However, it has been observed that when the time window is short it cannot capture distinct behaviour and thus leads to ambiguities especially for car, bus, walk and bike that share similar movement and spatial relevance at some instants.

From individual classification accuracy for five modes it has been observed walking has the highest false positives as different modes can be slowed down and behave like walking. At the same time this can also give false negative as people can walk on the road, near the bus stop, traffic signal or parking lot and hence respective POI relevance may be higher for a walking segment. That gives a false impression of other modes corresponding to the respective spatial relevance.

There is also a trade-off between the temporal window and the accuracy that raises questions of selecting the proper time window for a given location based service. Say for emergency services, the amount of response time required is less than the time required by an urban planner or traffic engineer to understand travel demand from people's mode choice or location based context-aware advertisements. This also poses challenges of selecting a proper and optimal window size to detect transport modes as accurately as possible. However from this study it is evident that the window size bears an inverse relationship with the accuracy measure, assuming the mode is not changed within the temporal window.

Earlier work using neural network only performed offline analysis considering only three modes. The highest reported accuracy achieved was 91% (Gonzalez et al., 2010). However our approach capable of real time estimation that shows accuracy can reach up to 95% or more when using the road network along with other relevant information for three modes, and 87% when using five modes over 600 sec. However the accuracy depends on number of modes, their spatial relevance and the size of the temporal window. The accuracy also depends on the clustering algorithm used to estimate POI relevance over a given temporal window.

However, at this moment this approach is limited by detecting only a single mode within a given temporal window. But in real time there is a possibility that a quick transfer can take place from one mode to another mode followed by walking. This creates composite mode segments within a same temporal window which is difficult to detect using only GPS signals (Das et al., 2014). In order to distinguish two different modes with in a composite segment different inertial sensor information along with GPS is required that can distinctly detect presence of two different modal class within a given temporal segment from their characteristic kinematic signatures.

7 Conclusions

Detecting transport modes in near-real time is an emerging research area. This is particularly useful in various context-aware location based services and understanding urban dynamics in near-real time. In order to detect various transport modes Microsoft’s GeoLife dataset has been used in this research. In this research a simulated near-real time mode detection classification framework has been developed using a neural network based classifier. We have evaluated the performance of neural network in detecting various modes, since neural networks can adjust well with different input and output parameters online. Neural networks also offer flexibility and scalability in terms of learning ability and accommodating new information from the external world. In this paper, we particularly focused on how real time mode detection accuracy varies with varying temporal window size. This has been figured out within a small temporal window all the transport modes show similar kinematic behaviour. In order to detect different modes more accurately we used various spatial information such as route network information and POI information.

We tested our hypothesis on three sets of modes: two sets containing three modes and one set containing five modes). Our result shows incorporating spatial information can improve mode detection accuracy. We achieved accuracy 95% accuracy on three modes only and 93% accuracy on all five modes. The result also shows a MLP neural network can outperform other machine learning algorithms with growing temporal window size.

Future work will look into distinguishing a composite segment within a given temporal window where a quick transfer has occurred. In order to explore different modes within a temporal segment, different sensor signals such as accelerometer, proximity sensor, gyroscope information are required that can give characteristic movement behaviour of each modes at a very fine granularity. In this research while forming the clusters we only considered spatial relevance of each cluster with respect to given POI. We did not consider temporal relevance as temporal window may vary from as small as 120 sec to as high as 600 sec or more. During smaller temporal window, it is difficult to set a temporal relevance or dwell time. Future research will address spatio-temporal issues while developing potential clusters within a given temporal segment to calculate spatio-temporal relevance for each mode.

References

Biljecki, F., Ledoux, H., & Oosterom, P. V. (2012). Transportation mode-based segmentation and classification of movement trajectories. International Journal of Geographical Information Science, 17(2), 385-407.

Bohte, W., Maat, K., & Quak, W. (2008). A method for deriving trip destinations and modes for GPS-based travel surveys. In J. V. Schaick & S. V. d. Spek (Eds.), Urbanism on Track : IOS Press, pp. 129-145.

Byon, Y., Abdulhai, B., Shalaby, A. S. (2007). Impact of sampling rate of GPS enabled cell phones on mode detection and GIS map matching performance. Transportation Research Board Annual Meeting 7(1795), 21.

Byon, Y., Abdulhai, B., & Shalaby, A. (2009). Real-time transportation mode detection via tracking Global Positioning System mobile devices. Journal of Intelligent Transportation Systems: Technology, Planning, and Operations,, 13(4), 161-170.

Das., R. D., Ronald., N., & Winter., S. (2014). Clustering based transfer detection with fuzzy activity recognition from smart-phone GPS trajectories, 17th IEEE International Conference on Intelligent Transportation Systems, Qingdao, China.

Dodge, S., Weibel, R., & Forootan, E. (2009). Revealing the physics of movement: Comparing the similarity of movement characteristics of different types of moving objects. Computers, Environment and Urban Systems, 33(6), 419-434.

Gonzalez, P., Weinstein, J., Barbeau, S., Labrador, M., Winters, P., Georggi, N., & Perez, R. (2010). Automating mode detection for travel behaviour analysis by using Global Positioning Systems enabled mobile phones and neural networks. Intelligent Transport Systems, IET, 4(1), 37-49.

Hemminki, S., Nurmi, P., & Tarkoma, S. (2013). Accelerometer-based transportation mode detection on smartphones. 11th ACM Conference on Embedded Networked Sensor Systems, Rome, Italy.

Hemminki, S., Nurmi, P., & Tarkoma, S. (2013). Accelerometer-based transportation mode detection on smartphones. 11th ACM Conference on Embedded Networked Sensor Systems, Roma, Italy.

Liao, L., Fox, D., & Kautz, H. (2007). Extracting places and activities from GPS traces using hierarchical conditional random fields. International Journal of Robotics Research, 26(1), 119-134.

Mamdani, E.H. and Assilian, S, (1975). An experiment in linguistic synthesis with a fuzzy logic controller, International Journal of Man-Machine Studies, 7(1), 1-13.

Minetti, A. (2000). The three modes of terrestrial locomotion. Biomechanics and Biology of Movement B. M. Nigg, B. R. MacIntosh, and J. Mester, Eds., ed: Human Kinetics, pp. 67–78.

Mountain, D., & Raper, J. (2001). Modelling human spatio-temporal behaviour: a challenge for location-based services. GeoComputation, Brisbane, Australia.

Ohashi, H., Akiyama, T., Yamamoto, M., & Sato, A. (2013). Modality classification method based on the model of vibration generation while vehicles are running. 6th ACM SIGSPATIAL International Workshop on Computational Transportation Science, Orlando, USA.

Patterson, D., Liao, L., Fox, D., & Kautz, H. (2003). Inferring high-level behavior from low-level sensors. Lecture Notes in Computer Science, Berlin, Germany.

Pereira, F., Carrion, C., Zhao, F., Cottrill, C. D., Zegras, C., & Ben-Akiva, M. (2013). The Future Mobility Survey: Overview and preliminary evaluation. Proceedings of the Eastern Asia Society for Transportation Studies.

Reddy, S., Mun, M., Burke, J., Estrin, D., Hansen, M., & Srivastava, M. (2010). Using mobile phones to determine transportation modes. ACM Transactions on Sensor Networks, 6(2), 1-27.

Schuessler, N., & Axhausen, K. (2009). Processing GPS raw data without additional information. 88th Annual Conference of the Transportation Research Board, Washington D.C. USA.

Tsui, S. Y. A., & Shalaby, A. S. (2006). Enhanced system for link and mode identification for personal travel surveys based on Global Positioning Systems. Transportation Research Record, 1972, 38-45.

Wolf, J. (2000). Using GPS data loggers to replace travel diaries in the collection of travel data. PhD thesis, Georgia Institute of Technology, Atlanta, USA.

Zadeh, L. A. (1965). Fuzzy sets, Information and Control 8(3), 338-353.

Zheng, Y., Chen, Y., Li, Q., Xie, X., & Ma, W.-Y. (2010). Understanding transportation modes based on GPS data for Web applications. ACM Transactions on The Web, New York, USA.

Zheng, Y., Li, Q., Chen, Y., & Xie, X. (2008). Understanding mobility based on GPS data. ACM conference on Ubiquitous Computing (UbiComp 2008), Seoul, Korea. ACM Press: 312–321.

Zheng, Y., Liu, L.,Wang, L., & Xie, X. (2008). Learning transportation modes from raw GPS data for geographic application on the web, International conference on World Wide Web (WWW 2008), Beijing, China. ACM Press: 247-256.