Transportation Mode Detection Using Crowdsourced Smartphone Data

Pramith Nirmal
University of Moratuwa
Sri Lanka
pramith.16@cse.mrt.ac.lk

Ishan Disanayaka
University of Moratuwa
Sri Lanka
ishan.16@cse.mrt.ac.lk

Dilantha Haputhanthri
University of Moratuwa
Sri Lanka
dilantha.15@cse.mrt.ac.lk

Adeesha Wijayasiri
University of Moratuwa
Sri Lanka
adeeshaw@cse.mrt.ac.lk

Abstract— **Smartphones have gained immense popularity among people and they incorporate several sophisticated capabilities like sensors that broaden the domain of usability. Consequently, smartphone-based crowdsourcing methods show importance as an efficient data acquisition strategy. GPS receiver and sensors like accelerometer and gyroscope can be used to gather data which helps identify the travel mode of the smartphone user. Numerous research projects have been conducted regarding the area of concern and this study uses GPS and Accelerometer data collected from smartphones using crowdsourced methods to develop a robust public transport mode detection approach using machine learning algorithms.**

I. INTRODUCTION

Transportation has become a major factor which drives the socio-economic growth of communities. The demand for safe and efficient transportation is expanding significantly due to population growth and urbanization. Analysis of transport-related data can advance the identification of unexplored characteristics and patterns that can optimize the utilization of the limited supply to meet this growing demand [1]. Smartphones can be used as an effective mobility data acquisition mode for transportation-related research because of the advanced inbuilt sensors and the extensive number of users. In 2020, the number of smartphone users all over the world is around 3.5 billion [2]. Many research projects have been conducted in recent years which involves smartphone-based data collection using inbuilt components such as accelerometer sensor, global positioning system (GPS) receiver, etc. Transport mode detection is one of the research areas conducted using smartphone-based mobility data. Several research projects have been conducted on transport mode detection using smartphones as the data collection mechanism and machine learning algorithms for the classification [3]–[5]. The benefits of transport mode detection are applicable for the users themselves and the authorities in optimizing transportation. Real-time location of public transit modes is one of the use cases of transport mode detection that is beneficial for the users to manage their transportation and the relevant transport authorities to further improve the transport schedule. There are several urban transport systems currently in commission around the world that utilize the use of installed GPS trackers in buses and trains to track them in real-time. But these facilities not often cover the whole transportation system and are only used in some long-running routes.

In this paper, we propose a smartphone-based crowdsourcing approach for real-time location of public transport modes. A mobile application was developed and used as the main data collection source and the solution for providing the user with real-time locations of detected public transit modes. Although several smartphone-based transportation mode detection approaches have been proposed in related work, none of them has attempted to provide a solution real-time location. Further, the Couchbase DB and the Sync gateway is used to transfer real-time data collected by the mobile application to the cloud in an effective way. Collected data are sent to the backend server where the learning model classifies the transport mode of that device. Those results are then used to update a database which stores the current mode of each device. Furthermore, the user can see the nearby devices that are classified as public transportation modes such as, buses and trains in the maps interface of the mobile application, utilizing the web sockets to communicate with the backend server and acquire real-time updates of the transportation modes.

To select the optimal learning algorithm and the segment size, we performed the classification process using several learning algorithms and segment sizes. Three feature sets were generated based on the segment sizes of 1, 2 and 3 minutes. Five different machine learning algorithms were employed in the classification module where two binary classifications and a multi-class classification scenario were carried out for each feature set to evaluate and select the optimal learning model.

The paper is organized as follows; section 2 explores the existing literature, section 3 describes the methodology including the dataset, theories, techniques and the system model, section 4 evaluates and analyses the results, section 5 presents the discussion, and section 6 concludes the paper.

II. BACKGROUND

Transport mode detection is a prevalent research area, which is categorized under activity recognition. Although a variety of sources and technologies have been involved in transport mode detection, the inbuilt GPS receiver, sensors such as accelerometer, gyroscope and magnetometer have facilitated the smartphones to be one of the most applicable data sources in transport mode detection. Several smartphone-based approaches have been proposed for classifying transport modes utilizing the technology and the extensive number of devices. Transport mode detection can be used to study the travel patterns of individuals and groups which can be used towards the optimization of transport facilities allocation and scheduling. Real-time detection of public transport modes can enhance the transit schedules and notify passengers with real-time locations. Apart from that, other advantages of transport mode detection include urban planning, domain-specific analysis, fleet management, etc.

A. GPS and Accelerometer

Global Positioning System (GPS) is a satellite-based radio navigation system which provides geolocation and time information to GPS receivers. Smartphones have GPS receiver chips which triangulate the signals coming from nearby satellites to accurately derive the location and time with an accuracy around 5m. GPS measures are also used to derive the speed of the device. Smartphone GPS nowadays has become an essential utility for people and smartphone-based innovative solutions and research. Smartphone GPS location data are used as a main data collection source in transport mode detection. Smartphone inbuilt accelerometer sensor provides axis-based motion-sensing measurements. Accelerations along 3 physical axes X, Y and Z, with respect to the smartphone orientation are measured in m/s^2. Accelerometer readings are often used activity recognition tasks and it also consumes less power compared to other smartphone sensors. Transport mode detection is one of the prevalent research areas where accelerometer data are used.

B. Learning algorithms

Decision tree algorithm considers every possible outcome of a decision through means of categorizing the data in each step for regression and classification. It creates a tree according to a specific algorithm which is a supportive tool used to simplify a given set of complex data. The decision tree consists of nodes and branches based on a rule. The nodes illustrate that a decision has been made whilst the branches that spread to the left or right from the nodes show that the data is further being categorized. Decision trees are simple to understand, easy to interpret and robust against skewed distributions but a small change can alter the results drastically. One more problem with the decision tree is that they can overfit easily.

Random forest is one of the ensemble algorithms that build multiple models and generate the overall result by combining the results of these models. During the classification process, it develops a collection of decision trees from randomized subsets of the training data and the result is generated by combining the results of each decision tree. Having multiple decision trees mitigate the quality of the prediction by reducing the noise and other biases of the classification process. Nonetheless, the development of multiple decision trees can delay the classification.

The SVM classification algorithm is one of the methods that can perform both classification and regression. It can capture complex relationships without going into difficult transformations. SVM constructs a set of hyperplanes in high-dimensional space to separate categories of examples and according to these separated categories, we can find obvious differences of each category and classify unknown examples into specific groups more accurately. A large functional margin of the hyperplane which in return lowers the generalization error can provide good separation.

XGB algorithm is one of the implementations of the gradient boosted trees algorithm. It is a supervised learning algorithm and it achieves accurate prediction of the target variable by combining the estimates of the several weaker models. Based on the difference between the predicted and target outputs, that algorithm minimizes the L1 and L2 regularized objective function with a convex loss function. In the training process that adds new trees iteratively and errors of the prior trees combined with current iteration to make the final prediction.

Neural networks are built imitating the human nervous system. That consists of an input layer, one or more hidden layers and an output layer which can perform classification, clustering, prediction, etc. The weighted sums of the inputs are fed into each node of the first hidden layer, and the activation function decides the output of each node. If multiple layers exist, weighted sums of the outputs of the nodes in the first hidden layer become the inputs to the next hidden layer and the final output is generated by the output layer. Despite the neural network's ability to build complex relationships and to avoid issues such as the curse of dimensionality, their “black box” nature hinders the interpretability of the system.

C. Related work

Related literature on smartphone-based transport mode detection has made significant strides [6], focusing on different perspectives and measures. Most of the conducted research projects are based on classifying transport modes such as car, bus, bike, and train along with some going further to detect human activities like walking and running. Also, several smartphone sensors such as GPS, accelerometer, gyroscope, and magnetometer have been employed for the data collection.

In a research done by Xia et al. [3], they proposed a transport mode detection approach for detecting modes such as walking, bicycling, motorized transport and further distinguish whether the vehicle is stationary or not. This proposed approach did not incorporate any re-orientation mechanism for deriving three-axis acceleration readings but rather used a combined Euclidian value of three-axis accelerations. Discrete Fast Fourier Transform (DFFT) was used for the feature extraction of the acceleration data. Sample entropy and statistical features such as minimum, maximum, mean, and standard deviations of speed were also extracted. The classification process was carried out using an optimized support vector machine model which produced an accuracy of 96.31%. They further utilized Ant Colony Optimization (ACO) in feature extraction to reduce the feature dimension by determining key features for the transport mode detection. Data collection was carried out using GPS, accelerometer, and gyroscope sensors with a data collection frequency of 50Hz and a sampling window of five seconds. In the context of transport mode detection, selecting a higher data collection frequency appear to be reasonable, but for a real-time crowdsourcing-based solution a higher frequency is not effective because of the high data and power consumption.

Shin et al. [2] have proposed an algorithmic approach for distinguishing transport modes such as train, tram, car, bus and walking. Smartphone accelerometer readings and network-based location-sensing are used for data collection. They utilized a sampling window of five seconds. Although both the data collection and classification are done within the mobile application, the proposed application has been proven to be capable of optimizing the battery usage which is important in the context of crowdsourcing data collection. The Euclidean length of 3 axis accelerations is used for mitigating the orientation issue. A customized rule-based algorithm has been developed for the classification which produced an accuracy of 82%.

Castrogiovanni et al. [5] have proposed a transportation mode detection approach based on classifying public and private transport modes, using a smartphone-based data collection process focusing on the optimization of power consumption. They utilized smartphone GPS, accelerometer, and Android activity recognition API for the data acquisition with the frequency of 5Hz which is comparably lower than that in other literature. To mitigate the orientation issue, they have used the magnitude of the resultant reading after merging them. Norms of auto correlogram and norms of the spectrum are some of the acceleration-based feature extraction methods used and along with some descriptive static measures like minimum, maximum. Some data from the activity recognition API such as travel duration, number of tilting activities was also employed. Decision tree, Support Vector Machine (SVM), random forest and naïve Bayes algorithms were employed for the classification and the random forest classifier produced the best accuracy of 90%. They also utilized a map-matching algorithm to predict the likelihood of being in a public transport mode based on GPS observations.

In Shafique et al. [7], a transport mode detection approach has been proposed for the transport modes bicycle, car, bus, train, and subway, evaluating the classification accuracy with varying smartphone data collection frequencies. GPS and accelerometer readings were acquired as smartphone data and the disorientation issue in accelerometer data was solved using Euclidean length based resultant acceleration calculation as in several related works. The maximum, average and skewness with moving window were extracted as features. A random forest model was used for the classification which produced an accuracy of 99.96% after evaluating different data collection frequencies. During the evaluation of classification accuracy with the varying frequencies, the accuracy was observed to decrease with the decrease in the frequency. But lower data frequencies give higher efficiency in data and power consumption. This research has revealed the trade-off between accuracy and efficiency which is an important insight when selecting an optimal data collection frequency for transport mode detection.

Fang et al. [8] have proposed a transport mode detection approach using smartphone sensor data based on deep neural network (DNN). Accelerometer, magnetometer, and gyroscope sensor data were used to classify transport modes such as bike, car, bus, and metros while also identifying states such as still, walking, and running. Average, standard deviation, highest FFT value, and the ratio between the highest and the second-highest FFT value were extracted as features for the classification where several learning algorithms were employed including DNN model which produced the best accuracy of 95%. DNN was trained and evaluated with several hyperparameter configurations to optimize the classification. Besides, the proposed approach has investigated different system parameters from feature and model perspectives to address practical issues including latency and model size.

Another deep learning based human activity and transport mode detection research [6] was proposed, utilizing smartphone accelerometer, gyroscope, magnetometer, and phone's current orientation for classifying both human activities like running, walking and transport modes like bikes, cars, bus and train. A Data collection frequency of 100Hz was used which comparably higher than other related work. The Euclidean length method was applied to mitigate the orientation error of magnetometer, gyroscope, and accelerometer reading. Subsampling and windowing were employed in feature extraction where both time and frequency domain features were used. Fast Fourier transform (FFT) was applied to obtain frequency domain features such as energy, entropy, and power spectral density. In the time domain, statistical features such as mean, variance, Pearson correlation between axes, covariance, skewness, kurtosis, and quartiles were extracted from the sensor data. Several windows sizes, 20s, 40s and 60s were chosen for testing, where the largest window size produced the best results. Machine learning models such as SVM, Random Forest, Multilayer perceptron neural network, k-nearest neighbor (KNN), and decision tree, were used with bagging and boosting optimizations. Besides, deep learning algorithms with techniques such as convolutional, long-short-term memory (LSTM), and multimodal spectro-temporal ResNet, were employed. However, machine learning and deep learning ensembled approach produced the best accuracy of 97.2% after smoothing results from the classification model using a Hidden Markov model to account for temporal dependencies of the activities.

Jahangiri et al. [9] proposed a transportation mode recognition approach utilizing machine learning techniques and mobile phone sensor data. This study has classified transportation modes such as car, bicycle, and bus and human activities such as running and walking. Smartphone accelerometer, gyroscope, and rotation vector sensors were used for data collection with a frequency of 25Hz. Euclidean resultant method was employed for preventing disorientation of accelerometer and gyroscope measures. Statistical features such as mean, variance, min, max, interquartile range, energy and spectral entropy were extracted. Several methods were employed such as analysis of variance (ANOVA) and correlation-based feature selection (CFS) for feature selection and K-fold cross-validation and out-of-bag error for model selection and validation. KNN, SVM, decision tree, and random forest were used for the classification and F-Score, Youden's index, and Discriminant Power were applied to assess model performances on the individual modes. However, the best accuracy of 95.1% was produced by the random forest model.

Table I summarizes the related work in the context of data sources, learning models and the best accuracies recorded for each research.

TABLE I. SUMMARY OF RELATED STUDIES ON TRANSPORT MODE DETECTION

Description of the related study	Transportation modes	Data sources	Learning models	Best accuracy
Using Smartphone Sensors to Detect Transportation Modes [3]	Walk, Bicycle, Motorized transport, Stationary	GPS, Accelerometer, Gyroscope	SVM	96.31%
Urban Sensing: Using Smartphones for Transportation Mode Classification [4]	Train, Tram, Car, Bus, Walking	Accelerometer, Network-based location sensing	Rule-based algorithm	82.05%

Description of the related study	Transportation modes	Data sources	Learning models	Best accuracy
Smartphone Data Classification Technique for detecting the Usage of Public or Private Transportation Modes [5]	Public, Private transportation	Accelerometer, GPS, Android activity recognition API	Decision tree, SVM, Random forest, Naïve Bayes	90%
Travel Mode Detection with Varying Smartphone Data Collection Frequencies [7]	Walk, Bicycle, Car, Bus, Train, Subway	GPS, Accelerometer	Random forest	99.96%
Learning Transportation Modes From Smartphone Sensors Based on Deep Neural Network [8]	Still, Walk, Run, Bike, Motorcycle, Car, Bus, Metro	Accelerometer, Magnetometer, Gyroscope	DNN, SVM, KNN, Decision tree	95%
Classical and deep learning methods for recognizing human activities and modes of transportation with smartphone sensors [10]	Still, Walk, Run, Bike, Car, Bus, Train, Subway	Accelerometer, Magnetometer, Gyroscope, Phone's current orientation	Random forest, MLP, SVM, KNN, Decision tree, DNN	97.2%
Applying Machine Learning Techniques to Transportation Mode Recognition Using Mobile Phone Sensor Data [9]	Car, Bicycle, Bus, Walking, Running	Accelerometer, Gyroscope, Rotation vector sensors	KNN, SVM, Decision tree, Random forest	95.1%

III. METHODOLOGY

An efficient mobile application can make crowdsourcing an applicable option to acquire data transportation mode detection using smartphone sensor data. Further, the Couchbase DB and the Sync gateway could be used to transfer real-time data collected by the mobile application to the cloud in an effective way. Collected data are sent to the backend server where the learning model classifies the transport mode of that device. Those results are then used to update a database which stores the current mode of each device. Furthermore, the user can see the nearby devices that are classified as public transportation modes such as, buses and trains. The google maps interface of the mobile application is used for the visualization, utilizing the web sockets to communicate with the backend server and acquire real-time updates of the transportation modes.

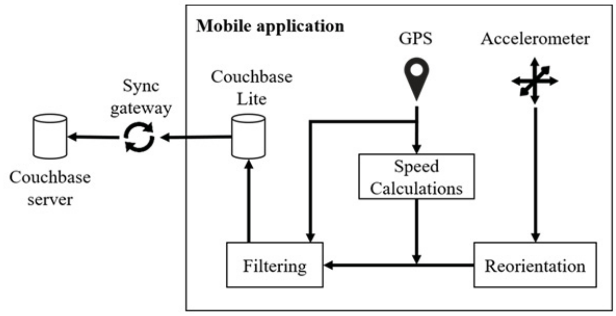

Fig. 1. Architecture of the mobile application

A. Mobile application

The mobile application utilizes smartphone sensors and other native functionalities in the mobile device, because of this Android is selected as the development platform. Besides, Android supports several external libraries that are essential for mobile application [11]. Fig. 1 illustrates the architecture of the mobile application. The mobile application is designed to implement both data collection and pre-processing. The accelerometer and GPS sensors are used to gather 3-axis acceleration readings and GPS coordinates. Acceleration data are then pre-processed using the Nericell reorientation mechanism [12] and GPS coordinates are used to calculate the speed of the mobile device. Then, both raw and processed accelerometer data and speed data are filtered to remove unnecessary data when the user does not move. These filtered data are stored in Couchbase lite database on the device storage, which then synced with the remote database. If the device is offline, the data are stored in the local database until a network connection is available to sync the local database with the remote database server. Furthermore, the locations of the devices that are classified as public transit modes are received through a web socket and visualized in the mobile application in a graphical view on a google maps layer

1) Reorientation mechanism

In general, the accelerometer embedded in a mobile device is in an arbitrary orientation with respect to the direction of the vehicle movement and the earth. Furthermore, the orientation of the device can change over time depending on how the user keeps it. This makes the axial system and the readings of the accelerometer inconsistent. In an instance of the readings, if the z-axis is directed in the original x-axis direction which is to the front rather than down, the horizontal acceleration can be misinterpreted as vertical acceleration in the readings in that instance. Because of this inconsistency, the reorientation of the accelerometer readings is important to mitigate the disorientation before the accelerometer measurements are used. In the proposed approach, the Nericell reorientation mechanism is used to address this issue.

In the Nericell reorientation mechanism, the system use acceleration vectors to calculate the Euler angles for the orientation of the mobile device. The acceleration vectors are then converted to the stable position of the device using the calculated Euler angles. Fig. 2 illustrates the tri-axis system of

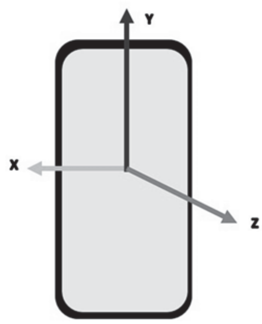

Fig. 2. Tri-axis system of the mobile device

the device after the Nericell reorientation.

2) GPS speed calculation

Speed is calculated based on consecutive GPS locations. The distance between two locations was calculated using the

Haversine formula (1) which provides the circular distance between two points on a spherical surface. According to the haversine formula the distance between two coordinates d,

$$d = 2R.\arcsin(\sqrt{(\sin2(\Delta\Phi/2) + \cos(\Phi1)\ \cos(\Phi2)\ \sin2(\Delta\lambda/2))}) \quad (1)$$

R denotes the mean radius of the earth, $\Phi1$ and $\Phi2$ denote latitudes of the coordinates. $\Delta\Phi$ and $\Delta\lambda$ denote the difference of latitudes and the difference of longitudes, respectively and d is the distance between the two coordinates. Then the device speed is calculated using this distance and the time difference between the two coordinates. Furthermore, the calculated speed is used to identify the state of the device which can be then used to reduce the data collection and transmission when the device is stationary. This is an added functionality to increase the efficiency of the mobile application in the context of the data and power consumption.

3) Visualization of transportation modes

The locations of the nearby devices that are detected as public transport modes are visualized to the user through the map view of the mobile application utilizing WebSocket communication. The lightweight WebSocket inside the mobile application that communicates with the main WebSocket handler in the backend API is used to acquire nearby device locations and real-time location updates of those devices from the backend server.

4) Data collection

The mobile application collects location-based data of mobile device including the timestamp, data generated by device sensors, reoriented 3-axis acceleration readings of the accelerometer, the speed calculated using the GPS coordinates. The mobile application generates a data instance for each second (1Hz frequency) with the timestamp, location coordinates, moving speed and the 3-axis acceleration readings which are then stored as JSON objects in the Couchbase lite database on the device storage. Couchbase is a NoSQL document-based interactive database which highly supports data synchronization with mobile applications. A separate API provided by Couchbase called sync gateway is used to sync the local database with the remote database server, storing the local data in the remote database. Couchbase lite local database in the mobile application can also maintain offline synchronization with the main database via the sync gateway. This WebSocket based inbuilt mechanism of the Couchbase server increases the efficiency of the data collection. For this study, we were able to collect around 285 minutes of bus data, 197 minutes of train data and 164 minutes of other vehicle data.

B. Back-end implementation

The mobile application generates large scale real-time data that needs to be stored, analyzed, and then efficiently accessed for visualization purposes. This requires an efficient and scalable backend platform. We designed a microservices-based architecture for the backend.

Fig. 3 illustrates the backend microservice architecture. Data stored in the Couchbase lite database in mobile device storage are sent to the Couchbase database in the backend through the sync gateway. The learning model, which is hosted on a microservice, then acquires data instances from the Couchbase database through the Couchbase service. Then, the data are segmented, and the required features are extracted for the classification. Using the predictions of the classification, the learning model updates the SQL database which stores the detected transport modes and the latest locations of the devices, through the SQL database service. For the visualization, the mobile application requests the data from the backend API which then access the SQL database through SQL database service to get a list of relevant devices that are identified as public transportation modes, where the latest location is nearby to the location of the device that requested the data. Then the mobile application establishes a WebSocket connection with the backend API which acquires real-time location data of the devices in the list which was obtained from the SQL database. This devices list is scheduled to be updated by the mobile application as the learning model updates the SQL database.

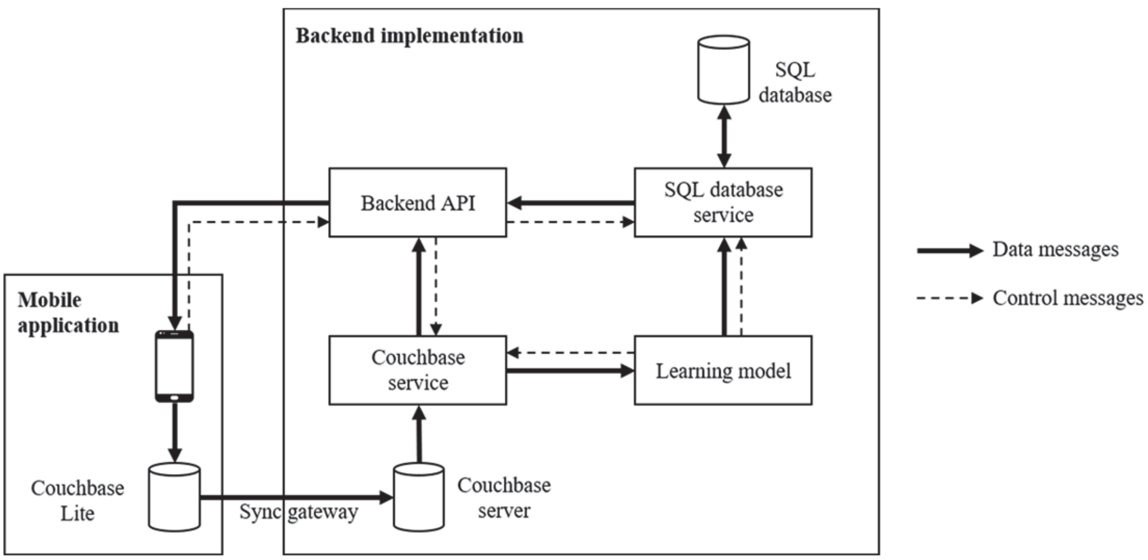

Fig. 3. Architecture of the mobile application

C. Transport mode detection

The proposed approach for the transport mode detection is designed with three main modules, namely segmentation, feature extraction and classification. To identify the optimal segment size, the data sent from the mobile application were segmented and three feature sets were generated based on the segment sizes of 1 minute, 2 minutes, and 3 minutes. In the feature extraction module, we have extracted statistical features such as mean, standard deviation, quartiles, inter-quartile range and then only the highest correlated features were selected for the classification reducing the dimension of the feature set. Five different machine learning algorithms were employed in the classification module where two binary classification and multi-class classification were carried out for each feature set. Cross-validation was used for the evaluation of the models to identify a better learning model and the classification approach.

1) Materials

Each data instance sent from mobile application consists of following a unique ID for each mobile user, longitude and latitude coordinates of the current location, speed derived from previous and current GPS locations, horizontal acceleration which is perpendicular to moving direction, acceleration along the direction of movement, acceleration along the vertical axis, and the current timestamp in milliseconds.

From these entries, speed and acceleration values are used to generate features for the classification. The deviceID is used to identify the mobile devices and once a device is identified as a public transport mode, the corresponding deviceID is used to update the device state in the SQL database.

2) Feature extraction

The processed data sent from the mobile application were segmented to create three feature sets based on the segment size. The data sent from the mobile application including the calculated speed, and reoriented acceleration readings which have a frequency of 1Hz were segmented to 1 minute, 2 minutes, and 3 minutes segments. Then the feature extraction and classification were performed on those feature sets separately to evaluate the performance of the models on segment size. For each segment, the mean and the standard deviation were calculated as the features. Also, the quartiles (Q1, Q2, Q3) of speed, interquartile range (IQR) of speed and the count of speed above 50 kmph. Then the transport mode that was used at the time of data collection was added as the label for each segment. Finally, the selected features are as follows, the mean and standard deviation of speed, x-acceleration, y-acceleration, and z-acceleration, quartiles of speed, Q1, Q2, Q3, and IQR of speed and count of speed values greater than 50 kmph.

Significantly unique patterns for different transport modes were observed in the mean and standard deviation of accelerations data. Fig. 4 illustrates the mean and standard deviation plots for the x-axis acceleration which recorded as the highest correlated features among the extracted features. The mean value distribution of x-axis acceleration for train data segments shows significant fluctuations compared to the respective distribution of bus and other data segments. Also, the standard deviation distribution of x-axis acceleration for non-public transport mode data segments behave differently compared to the public transport (bus, train) data segments. Such identifications have given valuable insights for selecting features for machine learning models that optimize the classification process.

3) Classification

The classification of the proposed approach was carried out employing several learning algorithms for all three feature sets which containing thirteen features.

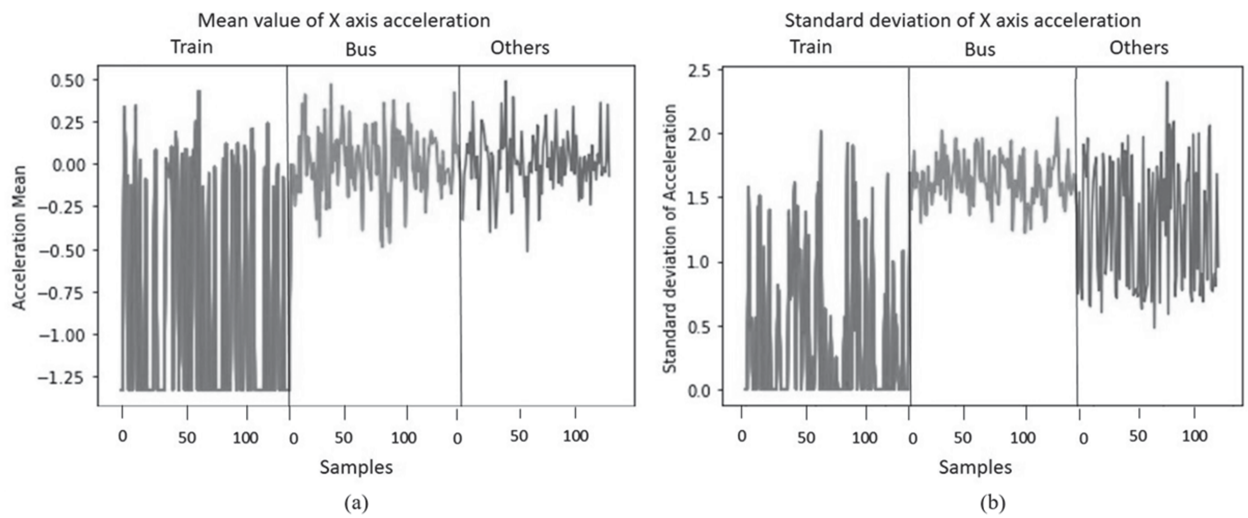

Fig. 4. Mean (a) and Standard deviation (b) on x-axis acceleration

For each feature set and learning model, we consider two binary classification scenarios and a multiclass classification scenario to classify busses, trains, and other transportation modes. All the five learning algorithms were used separately with each of these feature sets, resulting in a total of 15 models. The performances for each segment size were compared with each other making a total of 45 learning models. The learning algorithms used are namely XGBoost, random forest, decision tree, SVM, and neural networks.

Considering the results of the recent literature based on transportation mode detection, random forest [7], [9], decision trees [8], [10] and the SVM classifier [3], [5] were employed in the classification. We also used a multilayer perceptron neural network where sigmoid activation was used in all the nodes for binary classification and SoftMax activation was used in all the nodes for multiclass classification. XGBoost is a scalable implementation of the gradient boosting learning algorithm, which was also utilized in the proposed classification approach, dynamically changing the hyper-parameters to achieve higher performance.

4) Results and Evaluation

For the evaluation of the classification results, we used the cross-validation which is a useful evaluation technique for classification algorithms when it comes to handling datasets [13]. In k-fold cross-validation, the dataset is divided into k partitions of approximately equal sizes. The evaluation performs iteratively, for k iterations. During each iteration, one partition is used for testing while the rest ones are used for training. The overall accuracy is calculated by averaging all the accuracies, obtained for each iteration. This accuracy is used as the primary performance metric of the evaluation of learning models. Furthermore, we have used precision and recall.

In this research, there are three main classification scenarios: binary classification for bus detection, train detection binary classification and multiclass classification for predicting bus, trains, and others. Further, each classification was evaluated with three features sets with segment sizes of 1 minute, 2 minutes and 3 minutes.

TABLE II. PERFORMANCE OF THE BINARY CLASSIFICATION FOR BUS DETECTION

Learning algorithm	Accuracy			Precision			Recall		
	1 min	*2 min*	*3 min*	*1 min*	*2 min*	*3 min*	*1 min*	*2 min*	*3 min*
XGBoost	0.87	0.90	0.90	0.89	0.91	0.92	0.91	0.94	0.94
Random forest	0.84	0.86	0.86	0.87	0.86	0.91	0.89	0.92	0.90
Decision Tree	0.81	0.84	0.89	0.85	0.85	0.83	0.87	0.93	0.90
SVM	0.86	0.85	0.89	0.88	0.87	0.91	0.89	0.90	0.94
MLP	0.86	0.86	0.85	0.86	0.90	0.90	0.88	0.84	0.80

Table II compares the performance of the learning models used in the binary classification for bus detection. XGBoost classifier achieved the highest accuracy, precision and recall with all three feature sets. The XGBoost classifier with the 3-minute feature set produced the highest performance, achieving the highest accuracy, precision and recall among the 15 binary classification models for bus detection.

TABLE III. PERFORMANCE OF THE BINARY CLASSIFICATION FOR TRAIN DETECTION

Learning algorithm	Accuracy			Precision			Recall		
	1 min	*2 min*	*3 min*	*1 min*	*2 min*	*3 min*	*1 min*	*2 min*	*3 min*
XGBoost	0.92	0.94	0.92	0.91	0.92	0.91	0.84	0.87	0.82
Random forest	0.93	0.93	0.94	0.93	0.94	0.92	0.84	0.89	0.89
Decision Tree	0.88	0.87	0.92	0.93	0.76	0.81	0.78	0.74	0.88
SVM	0.78	0.61	0.79	0.72	0.64	0.71	0.54	0.64	0.64
MLP	0.87	0.85	0.87	0.85	0.84	0.93	0.87	0.87	0.85

Table III shows the results of the learning models that were used in the binary classification for train detection. XGBoost classifier achieved the highest accuracy in both the 1 minute and 2 minutes feature sets. In the 3 minutes feature set, Random forest classifiers achieved the highest accuracy which is also greater than the highest accuracies achieved using 1 minute and 2 minutes features sets. The highest precision and the recall were produced by the random forest classifier with the 2 minutes segment.

TABLE IV. PERFORMANCE OF THE MULTI-CLASS CLASSIFICATION

Learning algorithm	Accuracy			Precision			Recall		
	1 min	*2 min*	*3 min*	*1 min*	*2 min*	*3 min*	*1 min*	*2 min*	*3 min*
XGBoost	0.85	0.86	0.86	0.85	0.91	0.80	0.82	0.91	0.83
Random forest	0.83	0.83	0.84	0.86	0.85	0.84	0.81	0.86	0.86
Decision Tree	0.75	0.74	0.78	0.79	0.80	0.84	0.72	0.78	0.88
SVM	0.79	0.76	0.77	0.87	0.77	0.84	0.83	0.75	0.88
MLP	0.81	0.80	0.84	0.81	0.86	0.88	0.83	0.76	0.82

Table IV presents the performance of the learning models in the multi-class classification scenario. XGB classifier achieved the highest accuracy in all the features sets. The highest precision and the recall were produced by the XGBoost classifier for the 2 minutes feature set.

These results revealed that the best learning algorithm for the multi-class classification was XGBoost.

IV. DISCUSSION

In this paper, a crowdsourcing based transport mode detection approach was proposed for classifying two public transport modes(buses and trains) from other transport modes. Advance technology of smartphones has unveiled paths to develop convenient solutions, especially in mobility data analysis. In a smartphone-based approach involving mobility data collection, several facts need to be considered. In the proposed mobile application, both the accelerometer sensor and the GPS receiver are used for the data collection and the mobile application needs to transfer data to a remote server frequently. Therefore, the data collection and the transfer only happen when the user is moving since only the mobility data is needed. Also, the appropriate frequencies were selected, optimizing the data and the power consumption. As related literature shows high data collection frequencies often improve the performance of the classification, but it leads to high battery consumption of the device thereby decreasing the power efficiency.

Approaches proposed in [1] and [6] have used high frequencies of 50Hz and 60Hz respectively, which are not suitable for a power-efficient solution for a crowdsourced method. As suggested in [4], this trade-off between the classification performance and the power efficiency must be highly considered when developing a crowdsourcing based approach. Besides, high data collection frequencies consume more mobile data to upload data to the server which demotivates the user contribution. Table V compares the data gathering frequencies among the related studies and the proposed method. The proposed mobile application holds the minimum data transfer frequency of 1Hz compared to the related work.

TABLE V. COMPARISON OF DATA COLLECTION FREQUENCIES OF RELATED STUDIES AND THE PROPOSED METHOD

Title	Frequency (Hz)
Classical and deep learning methods for recognizing human activities and modes of transportation with smartphone sensors [10]	100
Using Smartphone Sensors to Detect Transportation Modes [3]	50
Applying machine learning techniques to Transport Mode Detection Using Mobile Phone Sensor Data [9]	25
Travel Mode Detection with Varying Smartphone Data Collection Frequencies [7]	10
Smartphone Data Classification Technique for Detecting the Usage of Public or Private Transportation Modes [5]	5
Proposed approach	1

To demonstrate the power efficiency, the battery drainage of the mobile application was recorded with and without using the application. To measure battery drainages, a third-party battery monitoring application called GSam Battery Monitor was used. In normal use, common mobile applications including music player, video player and Facebook were kept running. Over a period of 1 hour, only 4% was reduced additionally for the mobile app. At the same time, considering the individual batter consumption, the mobile application is in the same class as other regular applications in a smartphone.

The inbuilt smartphone accelerometer sensor is tri-axial and provides three acceleration readings for each of the axes, oriented respect to the smartphone. In a crowdsourcing approach, acquiring these readings is not practical, especially in mobility data collection since the orientation of the smartphone can change frequently. In the related work, this issue was mitigated by merging the three readings, trading off the disorientation issue for the characteristics of the three-axis accelerometer data. In the proposed approach, a re-orientation mechanism is employed to overcome the disorientation issue, while preserving the characteristics of the three-axis readings

V. CONCLUSION

Transportation is a complex system, affecting the socio-economic growth of a territory. Because of the large number of transportation modes, transportation mode detection can improve the behavior of the vehicle as well as the behavior of the passengers. Designing an effective solution to classify transportation modes, based on acquired crowdsourced data can enhance the scheduling and resource allocation of the transportation system. For crowdsourcing data, a mobile application that utilizes the device sensors such as accelerometers and GPS sensors were designed and developed. The accelerometer data were pre-processed in the mobile application using the Nericell reorientation mechanism and the speed data was calculated using the GPS sensor data. These data were then stored in a remote Couchbase database in the backend through the sync gateway. One of the unique features of the mobile application is the visualization of the nearby devices that are detected as buses and trains via a map view.

To select the optimal learning algorithm and the segment size, first, the data stored in the backend database were segmented to 1 minute, 2 minutes and 3 minutes segments generating three datasets. We extracted statistical features from each dataset, creating three separate feature sets for the classification. Five different machine learning algorithms were used for classification and two binary classifications and a multi-class classification scenario were performed. Models were then evaluated using cross-validation, listing the performance of the models using accuracy, precision, recall and as the metrics. The performance of the models was observed to be increased with the segment size. The highest accuracy for the multi-class classification was produced by the XGBoost classifier using both 2 minutes and 3 minutes features sets. However, considering the precision and the recall 2 minutes feature set was selected for the learning model.

This research is the first step towards the detection of transport modes using smartphone data. Based on the results that we have achieved, with the continued data collection and enhancement of the classification process, this approach has the potential for expansion to detect more transport modes such as taxies, personal vehicles and even human activities such as walking and running revealing the importance in transportation mode detection.

ACKNOWLEDGEMENT

This research was supported by the Accelerating Higher Education Expansion and Development (AHEAD) Operation of the Ministry of Higher Education, Sri Lanka funded by the World Bank. Also, we acknowledge the support received from the Senate Research Committee Grant, University of Moratuwa, Sri Lanka.

REFERENCES

[1] M. N. Mouchili, S. Aljawarneh, and W. Tchouati, “Smart city data analysis,” in *ACM International Conference Proceeding Series*, 2018, pp. 1–6.

[2] S O’Dea, “Smartphone users worldwide 2020 | Statista,” *Statista*, 2020. [Online]. Available: https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/. [Accessed: 19-Oct-2020].

[3] H. Xia, Y. Qiao, J. Jian, and Y. Chang, “Using smart phone sensors to detect transportation modes,” *Sensors (Switzerland)*, vol. 14, no. 11, pp. 20843–20865, Nov. 2014.

[4] D. Shin *et al.*, “Urban sensing: Using smartphones for transportation mode classification,” *Comput. Environ. Urban Syst.*, vol. 53, pp. 76–86, Sep. 2015.

[5] P. Castrogiovanni, E. Fadda, G. Perboli, and A. Rizzo, “Smartphone Data Classification Technique for Detecting the Usage of Public or Private Transportation Modes,” *IEEE Access*, vol. 8, pp. 58377–58391, 2020.

[6] J. Biancat, C. Brighenti, and A. Brighenti, “Review of Transportation Mode Detection techniques,” *ICST Trans. Ambient Syst.*, vol. 1, no. 4, p. e7, Oct. 2014.

[7] M. A. Shafique and E. Hato, “Travel mode detection with varying smartphone data collection frequencies,” *Sensors (Switzerland)*, vol. 16, no. 5, May 2016.

[8] S. H. Fang, Y. X. Fei, Z. Xu, and Y. Tsao, “Learning Transportation Modes from Smartphone Sensors Based on Deep Neural Network,” *IEEE Sens. J.*, vol. 17, no. 18, pp. 6111–6118, Sep. 2017.

[9] A. Jahangiri and H. A. Rakha, “Applying Machine Learning Techniques to Transportation Mode Recognition Using Mobile Phone Sensor Data,” *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2406–2417, Oct. 2015.

[10] M. Gjoreski *et al.*, “Classical and deep learning methods for recognizing human activities and modes of transportation with smartphone sensors,” *Inf. Fusion*, vol. 62, pp. 47–62, Oct. 2020.

[11] M. Amarasinghe, S. Kottegoda, A. L. Arachchi, S. Muramudalige, H. M. N. D. Bandara, and A. Azeez, “Cloud-based driver monitoring and vehicle diagnostic with OBD2 telematics,” in *15th International Conference on Advances in ICT for Emerging Regions, ICTer 2015 - Conference Proceedings*, 2016, pp. 243–249.

[12] P. Mohan, V. N. Padmanabhan, and R. Ramjee, “Nericell: Rich monitoring of road and traffic conditions using mobile smartphones,” in *SenSys’08 - Proceedings of the 6th ACM Conference on Embedded Networked Sensor Systems*, 2008, pp. 323–336.

[13] M. Saltan and S. Terzi, “Modeling deflection basin using artificial neural networks with cross-validation technique in backcalculating flexible pavement layer moduli,” *Adv. Eng. Softw.*, vol. 39, no. 7, pp. 588–592, Jul. 2008.