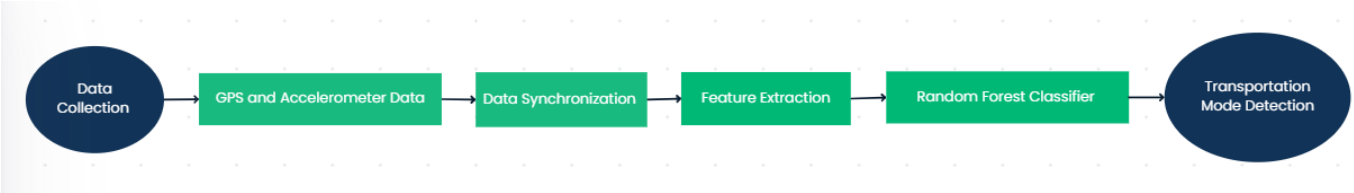
Transport Mode Classification Using Smartphone Sensor Data

1. Overview

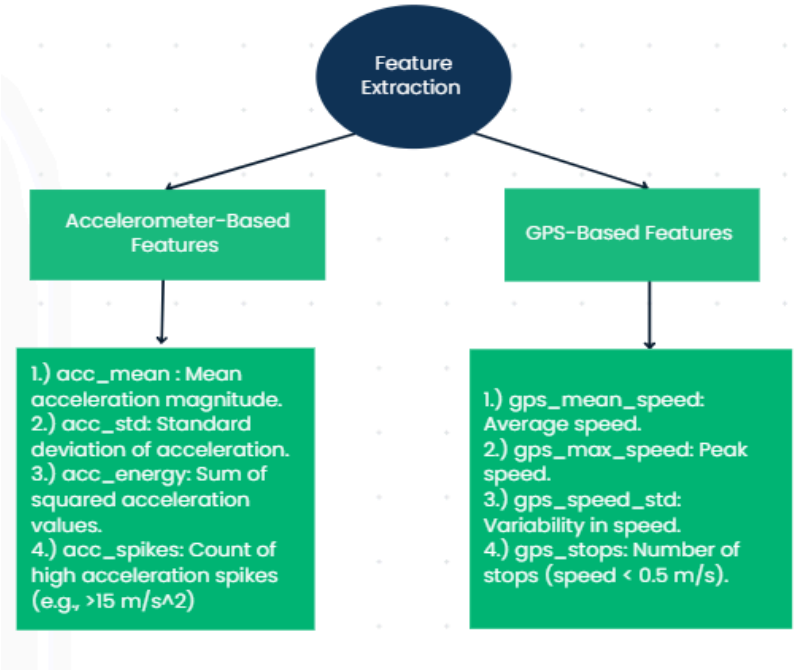
The project demonstrates the classification of transportation modes- walking, bus, train and driving which is based on data collection from smartphone sensors. In-built sensors from the smartphones were used through a sensorlog app to collect the data for this project. These sensors have been geared up to different fields in data mining and data analytics (Souza & Rajamohan, 2017). To enhance the decision-making, it is essential to obtain accurate information about the chosen transportation mode by the user in order to formulate effective strategies in transportation planning (Xiao et al., 2012, Liang et al., 2019). The aim of this project is to build a reliable classifier to detect these modes which contributes to the expanding field of sensor-based mobility inference. It can be titled originally as "Noise, Sensors and Walking" because it focuses on detecting motion in sensor data. The project is then extended to include different modes of transportation to enhance its scope. The data provided a deep understanding of motion patterns which then required the classification system to study that motion patterns for inference. A robust classification method Random Forest which can be used to detect the transportation mode to increase the prediction ability and model clarity(Cheng et al., 2019). This project was inspired and informed by (Giri et al., 2022) study, which demonstrated the application of random forest models to predict transportation modes using GPS, accelerometer, and heart rate data, and emphasized methodological considerations including the use of participant-level cross-validation and post-processing smoothing.



2. Methodology

The project includes the classifier which was developed to distinguish between multiple transportation types. It used the accelerometer and global positioning system (GPS) data simultaneously to collect both motion intensity and spatial behavior. The pipeline was created to synchronize the data, feature extraction, windowing and classification. It is essential to maintain the scalability of the transport mode classification system. A full pipeline was designed to work through the entire process on its own to make the predictions based on the sensor input data. The

data may have some missing values because it was collected by using GPS and accelerometer sensors which were operated at different frequencies. So, it is vital to synchronize the data to align these streams to make it capable of extracting the features. If the data is used to train the model without synchronization process then, it will perform poorly. The feature extraction is important for machine learning models to get trained because they need well-organized and meaningful data to work on. After that, windowing is done to create the manageable units from the time-series data to analyze the behavioral patterns over time. Windows were chosen to capture short-term patterns of movement while filtering out random noise. Speed, acceleration and motion energy like features are extracted to use to train the model for inference.

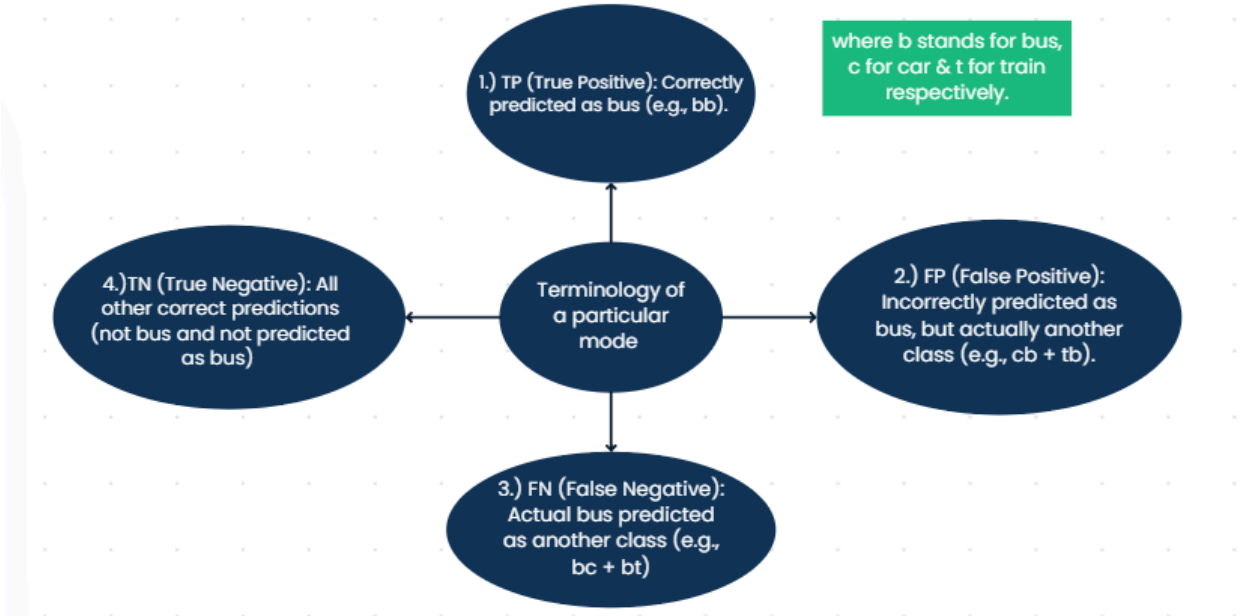


The mobile app called “SensorLog” was used by each team member who was responsible for one or more modes of transport. There are many sensors available in this app. But, for this project, an accelerometer and gps were used to record. The project could be extended by using the other sensors like gyroscope, gravity, orientation and compass. The gyroscope and orientation data was recorded during the initial phrase but later dropped. Because it was highly insensitive to deal with due to unintentional hand movements. The system used two types of input files: Accelerometer.csv which records three-dimensional acceleration(x,y,z) along with time stamps and Location.csv which provides GPS data including latitude, longitude and speed with time stamps. Data was segmented into 30-second windows. The preprocessed data included the magnitude of acceleration. Both accelerometer and GPS windows are then aligned to ensure feature extraction. To generate the meaningful statistical mean, standard deviation and median of acceleration magnitude are used in accelerometer based feature types. GPS based mean, max, min speed, stop counts (speed < 0.5 m/s), variance in location displacement and distance covered per window features are used which allows the classifier to distinguish between transport modes

effectively. A Random Forest Classifier is selected because it performs well with mixed data types and noisy features (Giri et al., 2022). It is also resistant to overfitting compared to models like K-Nearest Neighbors. It doesn't require heavy preprocessing or parameter tuning. It provides feature importance metrics, helping in explainability. And, its ability to handle both linear and non-linear relationships between features. The data collected on bus, train and car however, walking was not collected. So, the model was trained on car, train and bus data. And, the residual classification strategy was used for walking. If the model has the probability less than 0.6 then it sets the default mode as walking. Ultimately, this allows the classifier to focus on learning the transport classes with stronger, more consistent patterns while still accounting for ambiguous or low-activity segments. With scikit.learn's RandomForestClassifier, we can define 'balanced' class_weights, which automatically assigns weights to underrepresented classes such as trains in our case. Other models like Naive Bayes or KNN either performed poorly on GPS-derived features. Labeled feature dataset was used to perform an 80/20 train-test split and trained the model with 100 trees. The performance of the classes were then calculated by using the results from the confusion matrix. The table was made to show the precision, recall, F1-score and support to define the performance class-wise. To calculate overall accuracy:

Accuracy = Correctly Predicted as (bus + train + car) / Total # of instances of modes

The classification report was also made to analyze the performance of the model by calculating the values by using the following formulas and guidance from the chart.

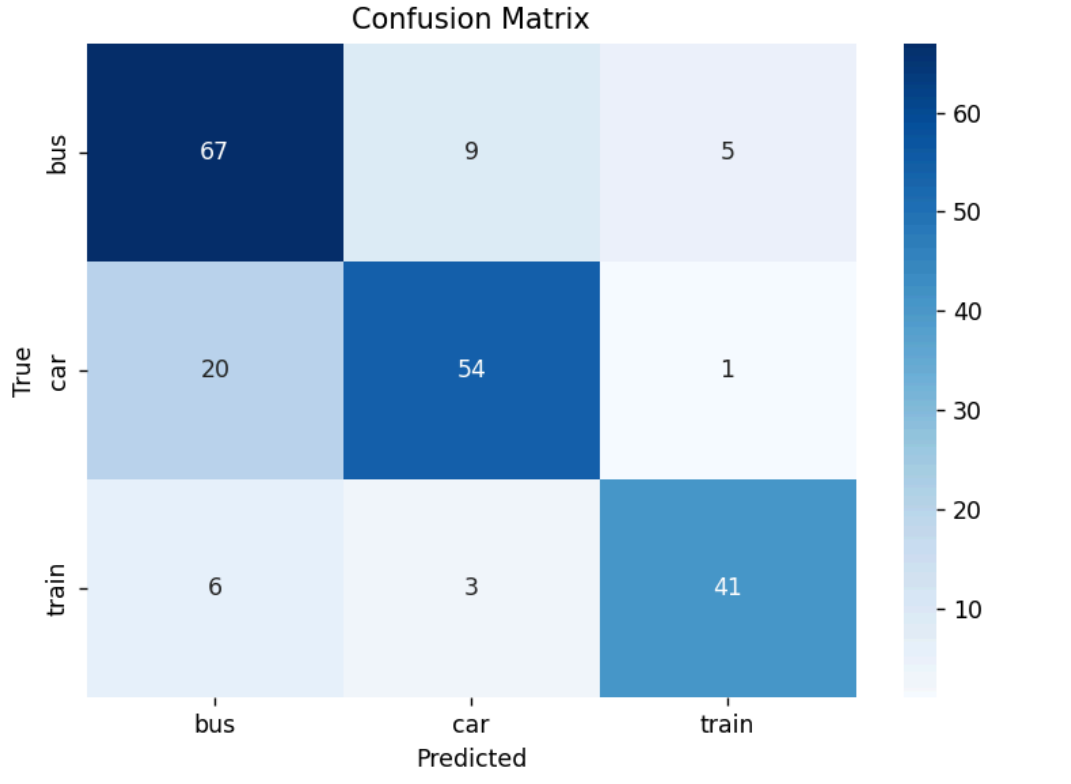


Formulas for a particular mode:

- **Precision = TP / (TP + FP);** it measures how many predicted modes were actually correct.
- **Recall = TP / (TP +FN);** it measures how many actual positives were correctly predicted.
- **F1-Score = 2* (Precision*Recall) / (Precision + Recall);** it balances the trade-off between the two.
- **Support = TP+FN;** # of actual occurrences of the mode in the dataset.
  These values were used to make the classification report for this model.

3. Results

The evaluation was done by using the classification report (Precision, Recall, F1 score) and Confusion Matrix. Visualization helped to detect the confusion between similar modes like car and bus. Walking was accurately detected through the fallback mechanism. The model showed the performance supported by chosen features and balanced training. High accuracy was found in distinguishing between motorized modes. However, walking was captured by using the fallback logic. With the confusion matrix, overlaps were revealed between car and bus. The statistical features from GPS and accelerometer provided enough discriminative power to separate modes despite the variations of real-world. The below confusion matrix was used to calculate the precision, recall, F1-score and support of the modes of transportation to define the performance.



Mode	Precision	Recall	F1-Score	Support
Bus	0.72	0.83	0.77	81
Car	0.82	0.72	0.77	75
Train	0.87	0.82	0.85	50
Accuracy			**0.79**	**206**
Macro Avg.	0.80	0.79	0.79	206
Weighted Avg.	0.79	0.79	0.79	206

The model performs strongly on the train class due to its consistent movement pattern and fewer stops. However, Bus and Car had similar performance with slightly lower precision due to overlapping similar motion characteristics (e.g., frequent stops, variable speed). The overall accuracy is 79% which demonstrates good generalization across all three transport modes. Also, precision and recall of modes are well-balanced and the system will perform efficiently in real-world scenarios. It is required to add new features to improve the performance of cars and buses.

4. Limitations

This project comes with some limitations and unavoidable inaccuracies. Uneven distribution of the dataset of transportation modes, firstly, is apparent. For instance, train data was underrepresented compared to car and bus data. This unbalance leads to biased learning where the model performs better on overrepresented classes like trains. Secondly, data was collected by a small group of three people which reduces the model's exposure to real-world variability. For example, the car data was only mostly collected by one person, with the same car. This exposes the model to predict based on his driving tendencies. Additionally, GPS data can be inaccurate or missing, especially in areas of poor satellite. This affected features like speed and stop count which can lead to incorrect classification. The gyroscope, orientation and magnetometer data were excluded due to noise and inconsistency. It reduced the system's ability to detect turns, heading shifts or orientation shifts which might be valuable for distinguishing modes which showed the similar pattern. Lastly, the choice of a 30-second window was a practical decision but it can smooth over short and meaningful events. It makes it harder to

classify transitional behaviors or sudden pattern shifts, while longer windows might capture stable patterns but risk blending different modes.

5. Future Improvements

With the limited time that we had, we only are able to include three modes of transport: bus, car, and train. And thus, the model is specialized to classify only these modes. Theoretically, expanding to other modes of transport such as cycling, scooters, motorcycles, etc are very feasible. In fact, doing this might enhance the stability of the model by adding some more features of modes to detect the transportation patterns efficiently. Other sensors, such as gyroscope, barometers, microphone and heart sensors can be included to increase model accuracy. These sensors are ultimately excluded due to the amount of noise data the sensors produce, making it very unreliable, and also demanding to record in the first place. However, given enough time, by integrating deep learning techniques and tools, we might have been able to exclude noise data from these sensors efficiently. The model works on the preprocessed data, it can be further improved to work in real-time by implementing the on-device inference system by using libraries such as TensorFlow. Collection of more data will also increase the performance in different seasons and cities. The equal distribution of data for modes of transport is important instead of weights assigned for better results to reduce variability.

6. Conclusion

This project's scope is to detect a user's transportation mode by using accelerometer and location (GPS) data from a standard-issue smartphone. The pipeline was designed for synchronization, feature extraction, windowing and then finally the model is designed for classification using Random Forest Classifier to optimize class differentiation, ultimately, transportation mode detection. The model showed 79% accuracy in test sets, with Train mode of transport dominating high accuracy predictions at 87% accuracy, outperforming bus and car. This is most likely due to cars and buses having a myriad of overlapping features which reduced the accuracy of the model while predicting them. Nevertheless, buses and cars still have passable accuracies, 72% and 82% respectively. The model can be improved further with the support of more training and testing datasets, as well as additional sensors that we do not have access to, or do not have the means to properly utilize.

References

[1] S. Giri, R. Brondeel, T. El Aarbaoui, and B. Chaix, “Application of machine learning to predict transport modes from GPS, accelerometer, and heart rate data,” *International Journal of Health Geographics*, vol. 21, no. 19, 2022. [Online]. Available: https://doi.org/10.1186/s12942-022-00319-y.

[2] Cheng, L., Chen, X., De Vos, J., Lai, X., & Witlox, F. (2019). Applying a random forest method approach to model travel mode choice behavior. Travel Behaviour & Society, 14, 1–10. doi:10.1016/j.tbs.2018.09.002.

[3] Tang, Q., & Cheng, H. (2024). Feature pyramid biLSTM: Using smartphone sensors for transportation mode detection. Transportation Research Interdisciplinary Perspectives, 26(101181), 101181. doi:10.1016/j.trip.2024.101181.

[4] Souza, W., & Rajamohan, K. (2017). Human Activity Recognition Using Accelerometer and Gyroscope Sensors. International Journal of Engineering and Technology, 9, 1171–1179. doi:10.21817/ijet/2017/v9i2/170902134.