# Predicting Transportation Modes of GPS Trajectories Using Feature Engineering and Noise Removal

Mohammad Etemad[1(✉)], Amílcar Soares Júnior[1], and Stan Matwin[1,2]

[1] Institute for Big Data Analytics, Dalhousie University, Halifax, Canada
etemad@dal.ca

[2] Institute for Computer Science, Polish Academy of Sciences, Warsaw, Poland

**Abstract.** Understanding transportation mode from GPS (Global Positioning System) traces is an essential topic in the data mobility domain. In this paper, a framework is proposed to predict transportation modes. This framework follows a sequence of five steps: (i) data preparation, where GPS points are grouped in trajectory samples; (ii) point features generation; (iii) trajectory features extraction; (iv) noise removal; (v) normalization. We show that the extraction of the new point features: bearing rate, the rate of rate of change of the bearing rate and the global and local trajectory features, like medians and percentiles enables many classifiers to achieve high accuracy (96.5%) and f1 (96.3%) scores. We also show that the noise removal task affects the performance of all the models tested. Finally, the empirical tests where we compare this work against state-of-art transportation mode prediction strategies show that our framework is competitive and outperforms most of them.

**Keywords:** Feature engineering · Noise removal
Trajectory classification

## 1 Introduction

Research on trajectory analysis is a mature area since positioning devices are now used to track people, vehicles, vessels, and animals. In the case of trajectory data, the object's movement is represented as a discrete collection of spatiotemporal points.

A domain where trajectories are frequently analyzed is the prediction of transportation modes from users, which is essential for cities and people to reduce travel time and traffic congestion. Transportation mode estimation involves two steps [11]: (i) extraction of segments of the same transportation modes; and (ii) classification of transportation modes for each segment. For the first step, several segmentation algorithms have been proposed in the past years and include temporal-based [8], cost function-based [5] and semantic-based methods [7]. For the second step, which is the focus of this work, the classification (or prediction) of the transportation modes is performed by creating domain expert features for


E. Bagheri and J. C. K. Cheung (Eds.): Canadian AI 2018, LNAI 10832, pp. 259–264, 2018.
https://doi.org/10.1007/978-3-319-89656-4_24

supervised classification (e.g., the distance between consecutive points, velocities, acceleration, and bearing).

We classify the research in transportation modes prediction regarding the type of features in two branches: (i) domain expert features; and (ii) learned features. From raw GPS data points (e.g., latitude, longitude and time) it is possible to calculate many attributes regarding the moving object's movement. Examples include distance traveled between points, estimated speed, bearing, acceleration, etc. For segments of trajectories, it is possible to extract mean, median, minimum, maximum, standard deviations, etc., of point-wise features. These are examples of domain expert features employed to predict transportation modes. Examples of works that apply domain expert features include [6,11].

In this work, we also explore the effects of noise removal in the prediction of transportation modes. Dealing with noise in trajectories is essential because GPS recorder devices are not accurate in the moving object's positioning due to many reasons like satellite geometry, signal blockage, atmospheric conditions, and receiver design features/quality. By removing GPS noise, it is expected that the derived features from the trajectories are more likely to represent the standard pattern of a transportation mode.

Noise-perturbed GPS data influences the quality of the domain expert features, e.g. distance traveled, speed or acceleration are susceptible to errors. It is important to point out that these errors may impact the distributions of values, where statistics like the mean, in trajectory segments of transportation modes. This uncertainty of data can lead a classifier to create models that are not able to accurately predict a transportation mode from a trajectory. Thus, the works in transportation mode prediction are classified regarding the (i) presence or (ii) absence of noise removal strategies. An example of work in the transportation mode prediction that does not deal with noise removal is [11]. In others, like [1,2,4,9,10], noise is removed. This paper applies domain expert features and noise removal to predict transportation are as follows: (i) we introduce new point and trajectory features; (ii) we propose a framework composed of 5 steps for transportation mode prediction; (iii) we compare the proposed approach with state-of-art strategies and show that our results are competitive.

## 2 A Framework for Transportation Mode Prediction

In this section, we present the sequence of steps used in this work to predict transportation modes (Fig. 1). This framework has five steps and is described in detail below.

In this work, we define a trajectory as a sequence of GPS points that belongs to the same transportation mode. In the first (step 1), we group the raw GPS points by *userid*, *day* and transportation mode to create trajectory samples. We discard trajectory samples with less than 10 GPS points because these examples may affect our model since trajectories with low quality may be created.

In this work, we calculate some point features (step 2) that were used previously in literature [11]: distance, speed, acceleration, jerk [1], and bearing.

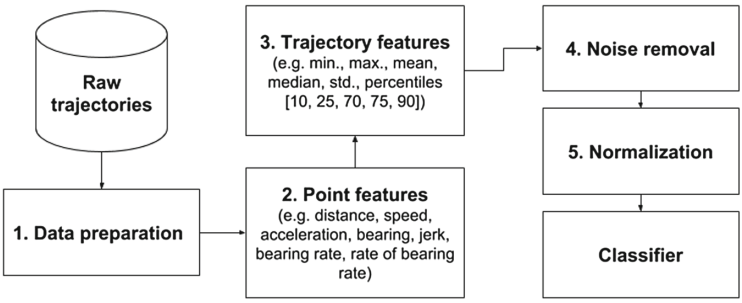


**Fig. 1.** The steps of the proposed framework to predict transportation modes

Two new features are introduced in this work, named bearing rate, and the rate of bearing rate. They are detailed as follows. The bearing rate was computed using Eq. 1, where $B_i$ and $B_{i+1}$ are the bearing values in points $i$ and $i+1$, and $\Delta t$ is the time difference.

$$B_{rate(i+1)} = (B_{i+1} - B_i)/\Delta t \tag{1}$$

Some moving objects tend to change the bearing more often because they commute in a straightforward route. This behavior can be captured by using the rate of the bearing rate. This feature is calculated using Eq. 2.

$$Br_{rate(i+1)} = (B_{rate(i+1)} - B_{rate(i)})/\Delta t \tag{2}$$

After calculating all the point features for each trajectory, we extract some statistical attributes referred to as trajectory features (step 3). Trajectory features are divided into two different types: (i) global trajectory features, which summarize information regarding the whole trajectory in a single value; and (ii) local trajectory features, which describe a local part of the trajectory. In this work, we extracted global features like the Minimum, Maximum, Mean, Median, and Standard Deviation values of each trajectory point feature to feed our classifier. The local trajectory features extracted in this work was the percentiles of every point feature. Five different percentiles were extracted (10, 25, 50, 75, and 90) and were used in the models tested in this work. In summary, we compute 70 trajectory features (10 statistical measures including five global and five local features calculated for 7 point features) for each transportation mode example.

In step 4, the framework deals with noise in the data. In this work, we used a simple method called median filter to create a mask. The method is described in Algorithm 1 ($threshold = 3$) and it removes the noise based on *speedmean* (i.e. the average speed of a trajectory) attribute since a human can classify the transportation mode mostly by knowing the mean speed of a trajectory.

Finally, we normalized the features (step 5) using the Min-Max normalization method, since this method preserves the relationship between the values to transform features to the same range and improve the quality of classification process [3].

**Data**: Speed mean of trajectories
**Result**: mask vector to remove the noisy trajectories
$difference \longleftarrow |speedmean_{Trajectory} - median(speedmean)|$;
$median\_difference \longleftarrow median(difference)$;
**if** *median_difference* $== 0$ **then**
  $indicator \longleftarrow 0$;
**else**
  $indicator \longleftarrow difference/median\_difference$;
**end**
**return** $indicator > threshold$;

**Algorithm 1.** Mask the noisy samples to remove from dataset using median

## 3 Experiments

In this section, we detail the experiments performed in this work to validate our framework. The data used in this work is the GeoLife GPS dataset, that was collected by Microsoft Research Asia from April 2007 to October 2011 [11]. The dataset has a 5,504,363 number of records labeled by eleven transportation modes: taxi (4.41%); car (9.40%); train (10.19%); subway (5.68%); walk (29.35%); airplane (0.16%); boat (0.06%); bike (17.34%); run (0.03%); motorcycle (0.006%); and bus (23.33%).

In the literature, we observed different sub-selections of these classes for evaluating transportation mode prediction strategies; therefore, we decided to select different target subsets for comparing our result with other papers.

To evaluate the performance of classifiers in this work we used the Accuracy and the F1 measure. In all our experiments, we used a 10-fold cross-validation strategy and computed a paired t-test to verify if the difference in the means were statistically different. We executed our framework with different classifiers such as Decision Tree (DT) (with *maxdepth* equals five), Random Forest (RF) (with 50 trees estimators), Neural Network (NN), Naive Bayes (NB), and Quadratic Discriminant Analysis (QDA). In all cases, the random forest surpasses all the other classifiers in both accuracy and f1.

Subsequently, we compared the RF using all the steps of our framework against the results of five papers. It is important to point out that all these papers reported their accuracy values on the Geolife dataset. Table 1 shows a side-by-side comparison between some related works and the results of our framework. Our work does not surpass Jiang's et al. accuracy [4] but outperforms all the others. It is important to highlight that the complexity and high training time of the RNN model used in his work may not be worth the 1.42% difference in accuracy.

Finally, we evaluated the effects of noise removal performed by our framework. We established as a baseline the performance of our framework using the data to train classifiers with noise and without noise (clean). Table 2 shows the mean of the f1 values obtained by 10-fold cross-validation for the different group of classes. We can observe in Table 2 that for all classifiers and different

**Table 1.** Comparison of accuracy and f1 measure of proposed model against related works

| Related work | | Proposed model | |
|---|---|---|---|
| Reference: classes used in the experiments | acc | acc | f1 |
| Dabiri et al. [1]: walk, bike, bus, driving, and train | 84.8% | **93.35%** | 93.22% |
| Jiang et al. [4]: bike, car, walk, and bus | **97.9%** | 96.45% | 96.31% |
| Xiao et al. [9]: walk, bus & taxi, bike, car, subway, and train | 90.77% | **93.19%** | 92.81% |
| Zheng et al. [11]: walk, driving, bus, and bike | 76.2% | **93.61%** | 93.51% |
| Endo et al. [2]: walk, car, taxi, bike, subway, bus, and train | 83.2% | **90.20%** | 89.95% |

**Table 2.** F1 measures to classifiers for different class groups.

| Reference | DT | | RF | | NN | | NB | | QDA | |
|---|---|---|---|---|---|---|---|---|---|---|
| | With noise | Clean | With noise | Clean | With noise | Clean | With noise | Clean | With noise | Clean |
| Dabiri et al. [1] | 85.56 | 92.31 | 88.07 | 93.22 | 85.18 | 89.87 | 63.30 | 82.91 | 54.76 | 79.83 |
| Jiang et al. [4] | 88.26 | 95.47 | 91.56 | 96.31 | 88.63 | 94.11 | 65.68 | 85.19 | 54.70 | 82.55 |
| Xiao et al. [9] | 84.38 | 89.79 | 88.75 | 92.81 | 82.93 | 89.01 | 51.40 | 70.03 | 47.81 | 71.45 |
| Zheng et al. [11] | 85.62 | 91.92 | 88.72 | 93.51 | 85.76 | 91.33 | 64.61 | 84.22 | 51.33 | 79.48 |
| Endo et al. [2] | 79.53 | 82.09 | 85.57 | 89.95 | 79.33 | 85.70 | 57.31 | 72.68 | 49.13 | 72.30 |

**Table 3.** Accuracy to classifiers for different class groups.

| Class group | DT | | RF | | NN | | NB | | QDA | |
|---|---|---|---|---|---|---|---|---|---|---|
| | With noise | Clean | With noise | Clean | With noise | Clean | With noise | Clean | With noise | Clean |
| Dabiri et al. [1] | 85.54 | 92.36 | 88.47 | 93.35 | 85.54 | 90.13 | 63.56 | 83.28 | 53.65 | 79.76 |
| Jiang et al. [4] | 88.41 | 95.54 | 91.91 | 96.45 | 88.80 | 94.21 | 63.70 | 84.31 | 53.03 | 82.07 |
| Xiao et al. [9] | 85.01 | 89.96 | 89.33 | 93.19 | 83.61 | 89.43 | 51.96 | 69.90 | 46.59 | 70.99 |
| Zheng et al. [11] | 85.77 | 92.13 | 89.09 | 93.61 | 86.10 | 91.45 | 64.36 | 84.53 | 50.85 | 79.50 |
| Endo et al. [2] | 80.25 | 83.61 | 86.36 | 90.20 | 80.27 | 86.28 | 56.66 | 73.27 | 47.92 | 71.60 |

subgroups of classes, performance gains ranging from 2.56 (Decision Tree, using classes of [2]) to 28.15 (QDA, using classes of [11]) in f1.

Finally, Table 3 shows the mean of the accuracy values obtained by 10-fold cross-validation. For all classifiers and different subgroups of classes and classifiers, performance gains ranging from 3.36 (Decision Tree, using classes of [2]) to 29.04 (QDA, using classes of [4]) in accuracy were observed. The results presented in this section indicate that dealing with noise in transportation mode prediction is an important topic, and the lack of this step in the classification task decreases the performance of the classifiers.

## 4 Conclusions and Future Works

In this work, we propose a framework for transportation mode prediction using feature engineering and noise removal. The results showed that the newly engineered features (e.g., bearing rate, and rate of bearing rate) and the application of a noise removal technique improve the performance of all tested classifiers. We intend to extend this work in two directions: (i) test and evaluate different noise removal techniques like wavelet-based, MCMC and fast Fourier based denoising methods, and (ii) investigate the performance of trajectory segmentation algorithms and include this step in our framework.

**Acknowledgments.** The authors would like to thank NSERC (Natural Sciences and Engineering Research Council of Canada) for financial support.

## References

1. Dabiri, S., Heaslip, K.: Inferring transportation modes from GPS trajectories using a convolutional neural network. Transp. Res. Part C Emerg. Technol. **86**, 360–371 (2018)
2. Endo, Y., Toda, H., Nishida, K., Kawanobe, A.: Deep feature extraction from trajectories for transportation mode estimation. In: Bailey, J., Khan, L., Washio, T., Dobbie, G., Huang, J.Z., Wang, R. (eds.) PAKDD 2016. LNCS (LNAI), vol. 9652, pp. 54–66. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31750-2_5
3. Han, J., Pei, J., Kamber, M.: Data Mining: Concepts and Techniques. Elsevier, Amsterdam (2011)
4. Jiang, X., Souza, E.N., Pesaranghader, A., Hu, B., Silver, D.L., Matwin, S.: trajectorynet: an embedded GPS trajectory representation for point-based classification using recurrent neural networks. arXiv preprint arXiv:1705.02636 (2017)
5. Soares Júnior, A., Moreno, B.N., Times, V.C., Matwin, S., dos Anjos Formiga Cabral, L.: GRASP-UTS: an algorithm for unsupervised trajectory segmentation. Int. J. Geogr. Inf. Sci. **29**(1), 46–68 (2015)
6. Lin, M., Hsu, W.-J.: Mining GPS data for mobility patterns: a survey. Pervasive Mob. Comput. **12**, 1–16 (2014)
7. Spaccapietra, S., Parent, C., Damiani, M.L., Macedo, J.A., Porto, F., Vangenot, C.: A conceptual view on trajectories. Data Knowl. Eng. **65**(1), 126–146 (2008)
8. Stenneth, L., Wolfson, O., Yu, P.S., Xu, B.: Transportation mode detection using mobile phones and GIS information. In: Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS 2011, pp. 54–63. ACM, New York (2011)
9. Xiao, Z., Wang, Y., Fu, K., Fan, W.: Identifying different transportation modes from trajectory data using tree-based ensemble classifiers. ISPRS **6**(2), 57 (2017)
10. Yanyun, G., Fang, Z., Shaomeng, C., Haiyong, L.: A convolutional neural networks based transportation mode identification algorithm. In: 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN), pp. 1–7, September 2017
11. Zheng, Y., Li, Q., Chen, Y., Xie, X., Ma, W.-Y.: Understanding mobility based on GPS data. In: Proceedings of the 10th International Conference on Ubiquitous Computing, pp. 312–321. ACM (2008)