Chapter 1
Data-Driven Approach for Modeling the Mixed Traffic Conditions Using Supervised Machine Learning

Narayana Raju, Shriniwas Arkatkar, Gaurang Joshi, and Constantinos Antoniou

Abstract The article describes modeling vehicular movements using supervised machine learning algorithms with trajectory data from heterogeneous non-lane-based traffic conditions. The trajectory data on the mid-block road section of around 540 m is used in the study. Supervised machine learning algorithms are employed to model the vehicular positions. A set of parameters were identified for modeling the longitudinal and lateral positions. With the set of parameters, the algorithm's potentiality for mimicking vehicular positions is evaluated. It was identified that supervised machine learning algorithms would model the vehicles' positions with accuracy in the range of 20–60 mean absolute percentage error. The k-NN algorithm was marginally edging past all algorithms and acted as a promising candidate for modeling vehicular positions.

Keywords Extended trajectory data · Machine learning · Data driven

1 Background

Examining driving behavior on a given road section is one of the complex phenomena. Additionally, it is one of the demanding elements in understanding the road network performance, particularly from a road safety and efficiency point of view. Since its inception, different behavior models have explained vehicular behavior. Under lane-based traffic conditions (prevailing in the USA), through NGSIM datasets [1], extended vehicular trajectory over the road space (say the length of about 600–800 m) turns out to be a prime data source in understanding the driving behavior throughout

N. Raju
Transportation, Planning, T.U. Delft, 2628 CD Delft, The Netherlands
e-mail: S.S.N.Raju@tudelft.nl

S. Arkatkar (✉) · G. Joshi
Department of Civil Engineering, SVNIT, Surat 395007, Gujarat, India

C. Antoniou
Technical University of Munich, Munich, Germany
e-mail: c.antoniou@tum.de

J. Shah et al. (eds.), *Intelligent Infrastructure in Transportation and Management*,
Studies in Infrastructure and Control, https://doi.org/10.1007/978-981-16-6936-1_1

the world. Numerous studies [2] are reported using this extended data, for modeling the driving behavior from homogeneous traffic conditions. However, under non-lane-based heterogeneous traffic conditions, the driving behavior has not been explored much due to the absence of this extended vehicular trajectory data.

Further, due to the variation in vehicle classes, even the automated image processing tools are reported to have failed in tracking the vehicular position over road segments. Nevertheless, in this direction, very few studies [3, 4] reported having used trajectory data for reasonable trap lengths in the range of 100–250 m developed using a semi-automated image processing tool. Nonetheless, modeling the driver's behavior comprehensively, even under heterogeneous traffic conditions, warrants a high-quality extended trajectory dataset, almost like NGSIM is a substantial research gap, particularly under heterogeneous traffic conditions. In addressing this research gap, it can be noted that with advancements in technology, there is an availability of high computational tools out of which, supervised machine learning [5] falls in that category and is proven to be one of the powerful data-driven tools in predicting the trained observations' responses. With this motivation, the supervised machine learning algorithms' competency for replicating the vehicular positions under heterogeneous non-lane-based traffic conditions is explored in the present research work.

2 Research Methodology

In addressing the research gaps in the literature, the research work is performed in three parts as trajectory data development, training the supervised machine learning algorithms followed by evaluation of algorithms. For better readability, the flow of the work is presented in Fig. 1 below. Supervised machine algorithms were employed to model the vehicular positions considering their potentiality and robustness in the data predictions. Next, the given subject vehicle's behavioral instincts were related to the surrounding vehicle's actions. Based on this, using correlation analysis, the influencing parameters were identified, and the supervised machine learning algorithms were trained. Finally, the trained algorithms were validated with different techniques. Based on the algorithm, the positions of the subject vehicles are predicted over the road space. In these lines, the error in terms of MAE is evaluated for longitudinal/lateral instant velocities and positions.

3 Study Area

In the present work, a mid-block road section on Dumas Road in Surat, India, without any intersection and free from higher side-friction, is selected for a trap length of about 600 m and width 10.5 m (3 lanes of each 3.5 m). At about 400 m, a foot over bridge is located across the carriageway for pedestrians crossing in the extended

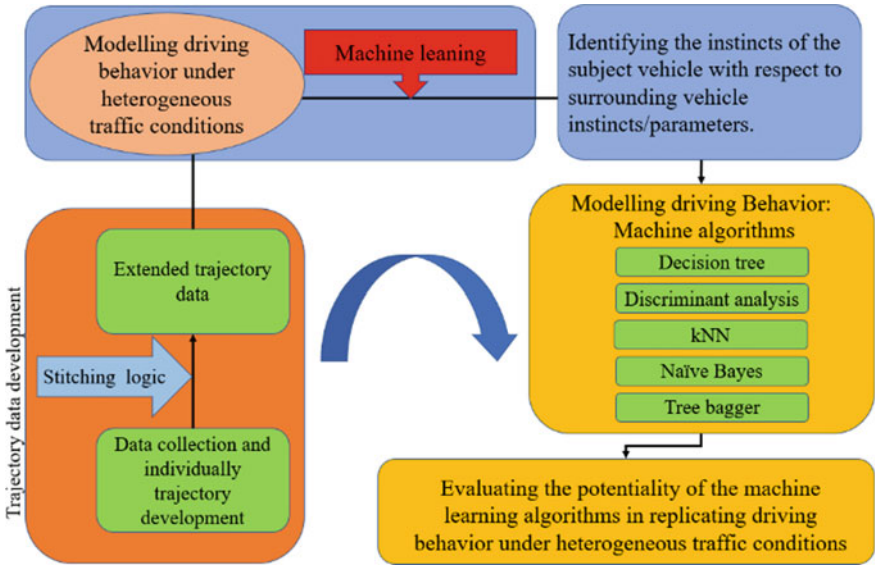

Fig. 1 Flow of work

study section. The foot over bridge as a vantage point, four cameras are installed and aimed at four uninterrupted road sections with trap segments of 230 m, 120 m, 100 m, and 75 m. This covers an entire extended study stretch of 535 m for developing a high-quality trajectory using continuously captured data on vehicular movement over space and time.

By employing an open-source image processing tool [6], trajectory data were developed separately for segments with an update of 0.5 s in which the vehicles are tracked using a computer mouse pointer with 0.5 s update interval over the segments for better accuracy. On this basis, around trajectory data at two-flow levels were extracted for 20 min having a traffic volume of 706 and 891 vehicles. The vehicles were tracked over the study sections as Flow-1 and Flow-2, respectively. The traffic volume comprises five categories of vehicles, namely Motorized three-wheeler, Motorized two-wheeler, Car, Truck, and Light Commercial Vehicle (LCV). The developed trajectory data for each of the individual sections was then decided to be stitched to obtain an extended trajectory data using a suitable algorithm, coded using MATLAB. The complete details of trajectory data can be found in Paul et al. [7] and Raju et al. [8].

4 Modeling Vehicular Positions

The present study is focused on modeling vehicular positions using supervised machine learning algorithms [9]. Supervised machine learning involves constructive training algorithms, learning the data responses, and making predictions. The algorithms are trained in such a way to identify the data patterns to match the field outcomes. The predictive potentiality of the algorithms can be improved by training with more observations over more substantial ranges.

In the present study, six machine learning algorithms are selected to model vehicular positions of vehicles. For this purpose, two sets of trajectory data are used to train the algorithms at two different traffic-flow levels. A 10 min data was selected out of 20 min for training algorithms from each of the flows. The remaining 10 min trajectory data were tested to validate the modeled vehicular positions separately for each machine learning model. It is a well-known fact that a particular subject vehicle's behavior is traditionally modeled concerning the surrounding vehicle actions (can be leading, trailing, and adjacent vehicles). Based on this premise, eight possible combinations of surrounding vehicles are considered in the present study for a given subject vehicle, as shown in Fig. 2. It depicts the surrounding vehicle nomenclature as trailing and adjacent, and their relative side based on their position concerning the subject vehicle as a leader. For identifying the surrounding vehicles, a section with the 60 m front and 40 m behind [10] longitudinally and laterally a distance of 5.5 m from the subject vehicle center, and next close vehicles will be considered in this range. Based on the positions of the surrounding vehicles, the independent

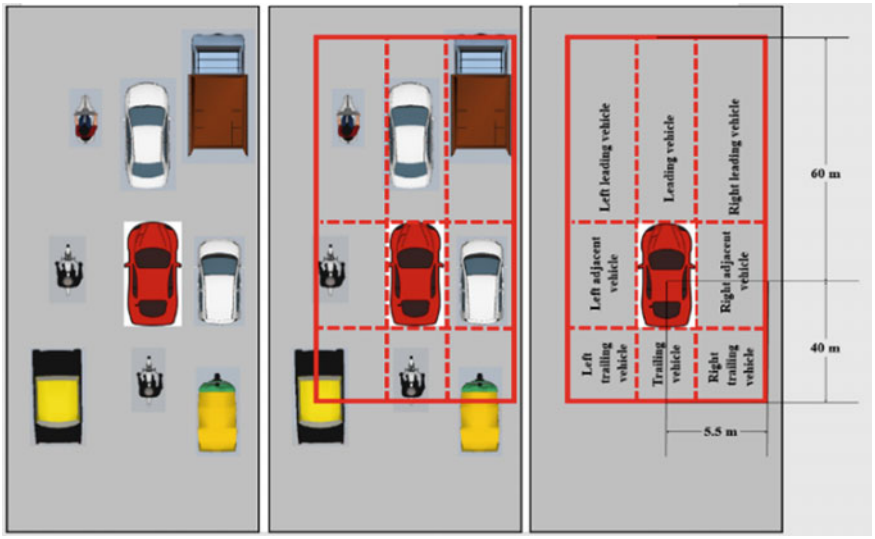

Fig. 2 Schematic diagram explaining the identification of surrounding vehicles

Table 1 Significant parameters for modeling driver actions

Dependent parameter	Independent parameter	Dependent parameter	Independent parameter
Instant longitudinal velocity (m/s)	Leader present (0/1)	Instant lateral velocity (m/s)	Leader present (0/1)
	Vehicle category of leader		Left adjacent clearance
	Vehicle category of subject vehicle		Right adjacent clearance
	Longitudinal distance (m)		Vehicle category of subject vehicle
	Velocity of leader (m/s)		Longitudinal distance (m) (m)
	Longitudinal distance from left leader (m)		Longitudinal distance from left leader (m)
	Longitudinal distance from right leader (m)		Longitudinal distance from right leader (m)

parameters that can impact the subject-vehicle behavior are recognized, and correlation analysis was performed considering responding variables to be accelerations, instant velocities, and Spearman correlation [11], and the identified parameters for modeling vehicular positions are reported in Table 1.

5 Machine Learning

In the present work, to improve the precision of algorithm training, the dependent variables, instantaneous longitudinal and lateral velocities, are rounded off to 0.5 and 0.01 m/s. Due to this, the variable classes decrease, and the data correlation patterns will be smooth. Based on this scheme, the first 10 min from a given flow level is engaged to train the algorithms, and the other 20 min data to validate the trained algorithms in replicating the vehicular positions and hence the driver behavior. Further, in the following sections, the logic behind the machine learning algorithms is explained briefly.

5.1 Decision Trees

The decision tree algorithm [12] is a projecting model, where the inputs form the branches and the outputs take the leaf forms, the dependent variable is filtered through subsets. Further, by means of recursive partitioning, the trained data is paired with an observed target value. By following this Top-Down Induction of Decision Trees,

the decision tree mechanism will be developed. Using this, the independent variables are filtered over a series of conditions for the target variable.

5.2 Discriminant Analysis

Discriminant analysis (DA) [13] is a simplification of Fisher's linear discriminant for characterizing two or more classes of target data outcomes. Discriminant analysis is a mixture of principal component analysis [14] and factor analysis. Let $\overrightarrow{x}$ be the sets of independent classes, and y the dependent outcome. Initially, DA accepts the conditional probability $p(\overrightarrow{x}/y=0)$ and $p(\overrightarrow{x}/y=1)$ will go after normal distribution having mean and variances $\mu_{0,}$ Σ_0 and $\mu_{1,}$ Σ_1 individually. Based on the Bayes optimal, the threshold T is stipulated to categorize the data and is given as in Eq. (1):

$$(\vec{x}-\vec{\mu}_0)^{\boldsymbol{T}}-\sum_{0}^{-1}(\vec{x}-\vec{\mu}_0)+\boldsymbol{ln}|\Sigma_0|-(\vec{x}-\vec{\mu}_1)^{\boldsymbol{T}}-\sum_{1}^{-1}(\vec{x}-\vec{\mu}_1)+\boldsymbol{ln}|\Sigma_1|>\boldsymbol{T} \tag{1}$$

Further, DA assumes equal variances ($\Sigma_0=\Sigma_1=\Sigma$), which results in a decrease in terms of Eq. (1). Based on the threshold value T, the observation will be categorized, and the outcome will be predicted.

5.3 k-Nearest Neighbors Classifier (k-NN)

k-NN [15] assumes a pattern from the data for classifying. K-NN generally assumes the Euclidean distance measure for marking the neighbors. Based on the optimal number of neighbors, the nearest neighbors will be identified, and the target outcomes will be mapped. With the help of a weighted average, the dependent variable will be projected as the mean of possible results.

5.4 Naïve Bayes Classinaïve

Naive Bayes [16] assigns class labels to cases, for categorizing the series of vectors to draw the label sets from the limited datasets. Naïve Bayes employed Bayes' theorem which is given as in Eq. (2) for the data classification:

$$P\left(\frac{h}{d}\right)=\frac{P\left(\frac{d}{h}\right)*p(h)}{P(d)} \tag{2}$$

for a hypothesis h and data sample d, P(h|d) is the probability. P(d|h) is the probability of data d given that hypothesis h was true. The probability of hypothesis h being true is given as P(h) and P(d) is the probability of the data. The hypothesis with maximum probability is generally chosen and termed as maximum posteriori (MAP)**,** which is given as in Eq. (3):

$$MAP(h) = max\left(\frac{P(d/h)*P(h)}{P(d)}\right) \tag{3}$$

For the revealed hypothesis with the highest probability, the probability of each class P(h) is back-calculated for the class having maximum probability and is predicted as the output for that certain dataset.

5.5 Tree Bagger (Random Forests)

Random decision forest theory [17] was initially proposed by Tin Kam Ho, with the help of the subspace method, in which he used stochastic discrimination [18] proposed by Eugene Kleinberg. Tree bagger works by constructing the compilation of decision trees for forecasting the variable. For example, $X = x_1, \ldots, x_n$ are the observations for the variables $Y = y_1, \ldots, y_n$, by bagging repeatedly (B times) from the random samples with a substitute of the training set which fits trees to these samples: For $b = 1, \ldots, B$; by the random sampling with n trained examinations, the variables are selected for training as X_b, Y_b.

6 Validation of Trained Algorithms

The machine learning algorithms are trained with instantaneous longitudinal velocities and lateral velocities as dependent variables. To understand the performance of the algorithms in imitating traffic behavior, with the help of trajectory data other than the trained datasets, the instantaneous longitudinal and lateral velocities are predicted. The vehicles' positions are computed over the entire road space based on the longitudinal velocities and lateral velocities. Evaluating the error in predictions, Mean Absolute Error (MAE) is calculated for velocities which is reported in Table 2. Similarly, the predicted outcomes are compared with the observed instant velocities, as shown in Fig. 3a for some sample points. From the analysis, it was observed that in the case of instantaneous velocities, the MAE is in the limit of 4.5–10.65 m/s (longitudinal) and 0.41–0.68 m/s (lateral). The results show that the k-NN algorithm performs better, followed by decision trees, discriminant analysis, and Tree bagger.

Table 2 Validation of machine learning algorithms

Flow type	Variables	Decision trees		Discriminant analysis	k-NN	Naive Bayes classifier	Tree bagger
		Classification	Regression				
Flow-1	Longitudinal velocity (m/s)	5.35	5.29	4.97	4.5	9.69	6.2
	Lateral velocity (m/s)	0.52	0.49	0.48	0.41	0.68	0.46
Flow-2	Longitudinal velocity (m/s)	5.98	5.75	6.12	4.99	10.65	7.54
	Lateral velocity (m/s)	0.57	0.59	0.46	0.45	0.68	0.46

Nevertheless, Naïve Bayes, due to the probabilistic formulation, considering a particular hypothesis, failed to replicate vehicle positions compared to other algorithms. From the analysis, it can be observed that over road space of 535 m, the supervised machine algorithms were able to imitate the vehicular behavior with the error of velocities with an MAE of 4.5–10 m/s (longitudinal velocities) and 0.4–1 m/s (lateral velocities).

7 Summary and Conclusions

From the study, it can be visualized that machine learning can be very much handy for modeling vehicular positions; given the possibility, the models can be implemented in traffic simulation tools instead of conventional models. Thus, the precision in replicating field conditions can be improved, lacking in heterogeneous non-lane-based traffic in the present context. It is observed that with supervised machine learning models, the vehicular positions under heterogeneous non-lane-based conditions can be replicated reasonably well. In the present study, among all models, k-NN is found to be the best model. With the help of the study methodology, the vehicular positions are modeled. Simultaneously, there is still scope for future studies to induce the machine learning models into the traffic simulation packages. This will facilitate and ease the process of traffic simulation. This will certainly benefit and can increase the accuracy of microscopic traffic modeling of heterogeneous traffic conditions.

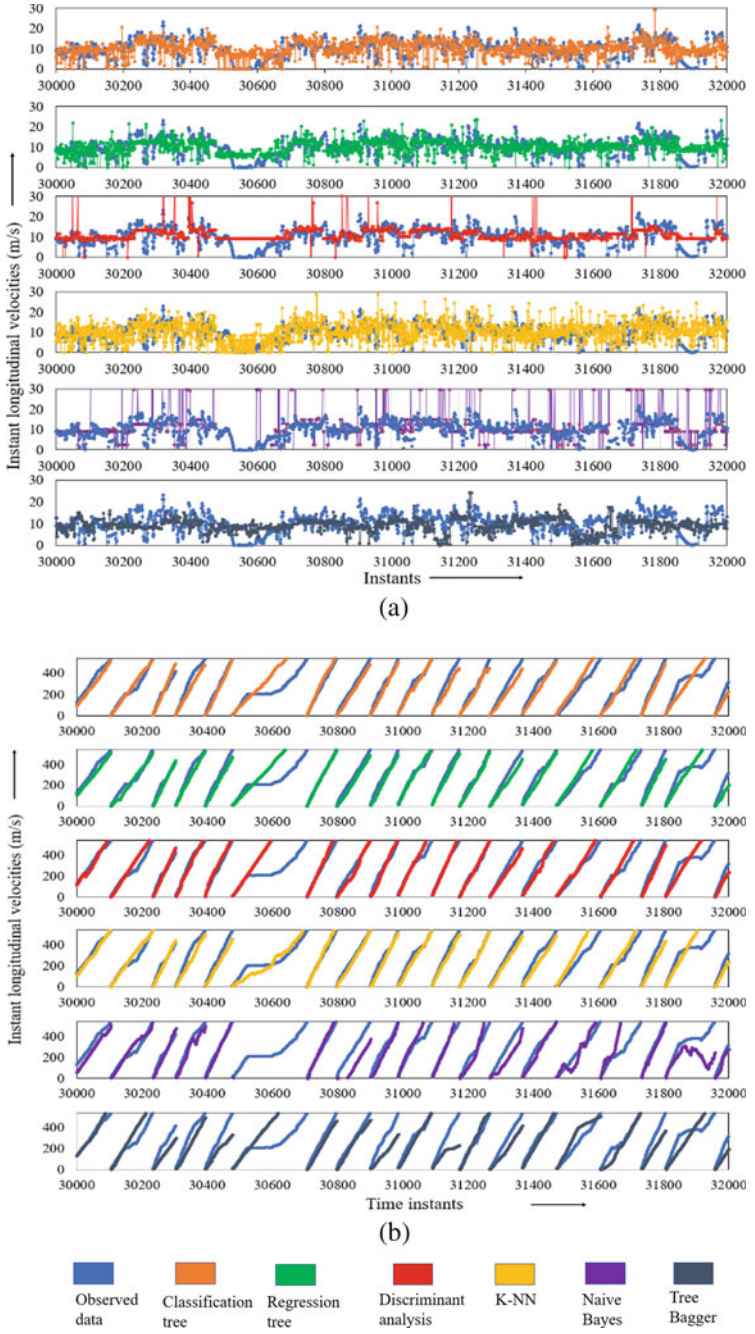

Fig. 3 Comparison of instant longitudinal speeds of vehicles one after another

References

1. NGSIM. Next Genaration Simultaion, FHWA [Internet] (2007). https://ops.fhwa.dot.gov/trafficanalysistools/ngsim.htm
2. Papathanasopoulou, V., Antoniou, C.: Towards data-driven car-following models. Transp. Res. Part C Emerg. Technol. **55**, 496–509 (2015)
3. Papathanasopoulou, V., Antoniou, C.: Flexible car–following models for mixed traffic and weak lane–discipline conditions. Eur. Transp. Res. Rev. (2018)
4. Raju, N., Arkatkar, S., Joshi, G.: Evaluating performance of selected vehicle following models using trajectory data under mixed traffic conditions. J. Intell. Transp. Syst. [Internet]. 1–18 (2019). Taylor & Francis. https://doi.org/10.1080/15472450.2019.1675522
5. Ng, A.: Supervised learning. Mach Learn. 1–30.
6. Vicraman, V., Ronald, C., Mathew, T., Rao, K.V.: Traffic Data Extractor [Internet]. IIT, Bombai (2014). http://www.civil.iitb.ac.in/tvm/tde2
7. Paul, G., Raju, N., Arkatkar, S., Easa, S.: Can segregating vehicles in mixed-traffic stream improve safety and throughput? implications using simulation. Transp. A Transp. Sci. (2020)
8. Raju, N., Arkatkar, S.S., Easa, S., Joshi, G.: Developing extended trajectory database for heterogeneous traffic like NGSIM database. Transp. Lett. Int. J. Transp. Res. (2021)
9. Dey, A., Learning, A.S.: Machine learning algorithms: a review. Int. J. Comput. Sci. Inf. Technol. **7**, 1174–1179 (2016)
10. Kim, D.H., Han, C.S., Lee, J.Y.: Sensor-based motion planning for path tracking and obstacle avoidance of robotic vehicles with nonholonomic constraints. Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci. **227**, 178–191 (2013)
11. Kendall, M.G.: Rank correlation methods. Rank Correl. Methods (1948)
12. Rokach, L., Maimon, O.: Decision tree. Data Min. Knowl. Discov. Handb. 165–192 (2005). http://www.ise.bgu.ac.il/faculty/liorr/hbchap9.pdf
13. Klecka, W.R.: Discriminant analysis. Analysis **19**, 71 (1980). http://srmo.sagepub.com/view/discriminant-analysis/SAGE.xml
14. Principal, P., Analysis, C.: Probabilistic principal component analysis and the EM algorithm (2007)
15. Takezawa, K.: Introduction to nonparametric regression. Introd. Nonparametr. Regres. (2005)
16. Bayes, N.: Naive Bayes classifier. Artic. Sources Contrib. 1–9 (2006). http://www.ic.unicamp.br/~rocha/teaching/2011s2/mc906/aulas/naive-bayes-classifier.pdf
17. Breiman, L.: Random forests. Mach. Learn. **45**, 1–35 (1999)
18. Kleinberg, E.M.: Stochastic discrimination. Ann. Math. Artif. Intell. **1**, 207–239 (1990)