Urban Traffic Identification by Comparing Machine Learning Algorithms

Boris A. Medina Salgado[1], Jhon Jairo Feria Diaz[1], Sandra Rojas Sevilla[2]

[1]Faculty of Engineering, Universidad de Sucre, Sincelejo 700001, Colombia.
[2]Faculty of Education and Sciences, Universidad de Sucre, Sincelejo 700001, Colombia.
Email: boris.medina@unisucre.edu.co

Abstract

The Internet of Things (IoT) applied to intelligent transport systems has become a key element for understanding the way traffic flow behaves in cities, which helps in decision-making to improve the management of the transport system by monitoring and analyzing network traffic in real time, all with the aim of daily benefiting users of the city's road infrastructure. Traffic volume estimation in real time, with high effectiveness, may help mobility management and improve traffic flow. Moreover, machine-learning algorithms have shown effectiveness in various scientific fields and have provided a significant platform for achieving intelligent applications. Therefore, we applied various machine learning algorithms to classify the vehicular traffic status in the traffic network of two cities with more than 2 million inhabitants. It was first necessary to establish, from the attributes provided by the datasets, the object class from the LOS (Level of Services) thresholds proposed by the National Academies of Sciences, Engineering, and Medicine, for the basic segments of highways in an urban area. We then selected the attributes of interest using the Recursive Feature Elimination Method (RFE) to reduce the dimensionality of the data, and applied the DT, RF, ET, KNN, and MLP algorithms to train and classify the level of vehicular congestion, defining various volumes of training and validation data. The results show the high effectiveness of the algorithms, highlighting the MLP algorithm as the one that provides the highest effectiveness on average for the evaluated datasets, with a mean precision of 99.5%.

Keywords: Machine Learning, Urban Traffic Flow, Vehicular congestion, Internet of Things.

1. Introduction

Vehicular traffic congestion has become an important factor that affects people's daily commutes and the sustainable development of cities. In 2017, the cost of traffic jams for American drivers were estimated at $179 billion in terms of time and fuel consumption [1].

Latin America registers an accelerated increase in the vehicle fleet, which raises the jams problem and puts pressure on the rise of scarce public resources to expand the road system [2].

According to the Tomtom Traffic Index, cities such as Bogotá, Lima, and Mexico City showed congestion levels exceeding 50 % extra time on each trip in 2019 [3].

Measuring and analyzing traffic flow effectively can alleviate the spread of traffic congestion, provide information that enables decision-making and implementation of measures by the local administration, which could help reduce fuel costs and air pollution [4], and positively impact the sustainability of cities.

Within the context of smart cities, Intelligent Transportation Systems (ITS) have emerged as an efficient option to help improve traffic conditions in cities [5]. ITS merges road infrastructure with IoT, incorporating electronic sensors for information capture, data transmission technologies, and intelligent control systems to improve mobility management and offer value-added services to drivers and public transport users [6]. With the rise in information volume that can currently be obtained and the application of intelligent computational techniques, a large part of the study is focused on analyzing mobility information. Intelligent computational techniques have recently been used in tasks such as classification, prediction, pattern recognition, clustering, and others [7], and are increasingly popular for data analysis in ITS [8] because of their high capacity to capture complex non-linear relationships, typical of traffic behavior in urban areas.

In this study, we implemented various machine learning techniques to classify the congestion level in urban vehicular traffic networks using two datasets that widely differ in the number of electronic detectors used to capture traffic data. Some supervised machine learning algorithms were applied to the data, and the results were compared and evaluated using various validation techniques with the aim of identifying the algorithm that shows the best accuracy results.

The remainder of this paper is organized as follows. Section 2 discusses previous related studies that allow us to identify the opportunity area. In Section 3 a characteristic description of the datasets implemented in the experiment is presented, and the data preparation process for its subsequent classification is presented, which describes the classifiers used. In Section 4, the algorithms used in our methodology are assessed and compared using different validation criteria, establishing and describing the results obtained in the test data from two datasets that differ in the volume of nodes or data capture stations.

Finally, we present our conclusions in section 5.

2. State of the Art

IoT has had a significant impact on intelligent transport systems to support the improvement of interoperability and mobility management in smart cities. Knowing the traffic state or congestion level on certain routes in the city can help redistribute traffic through decision making, select alternative routes that alleviate the traffic load, and improve transport efficiency [9]. Several literature reviews have experimented with and proposed intelligent computational techniques in communication with the IoT for processing and analyzing high information volumes to classify traffic status.

Ata, et al., (2021), propose a model to analyze urban traffic congestion using traffic and weather data from the City of Leeds Data Mill North database and weather data in England. The model includes a support vector machine (SVM) algorithm to classify vehicular congestion levels and, through IoT, announce the users about congestion occurrence in a particular city sector. The SVM classifier achieved 98.7% accuracy and proved to be more effective than methods proposed by other authors [10].

Bandaragoda, et al., (2019), employ unsupervised learning algorithms for traffic congestion detection and profiling. Their proposal extracts interesting features using the Incremental Knowledge Acquisition Self-Learning Algorithm (IKASL) from approximately 190 million travel path records obtained from the Bluetooth traffic monitoring system installed in Melbourne, Australia [11].

Mondal and Rehena, (2019), propose a machine learning algorithm in order to classify vehicular congestion level through an artificial neural network model based on information on speed and vehicular density from stationary sensors placed in various road sections to classify them among three congestion levels [12].

Cheng, et al., (2020), use an unsupervised clustering classifier to determine the urban traffic congestion level, using the Fuzzy C-Means Clustering Algorithm. The model includes speed, flow, and occupancy violations as well as an attribute that determines the grade of the road structure. To do this, the authors took data from 28 loop detectors located on a segment of the Shanghai North-South Elevated Expressway. The discussion shows the superiority of the proposed algorithm by more than 5% compared to other machine learning methods, such as support vector machines (SVM), decision tree, or k-nearest neighbor (k-NN) [13].

Impedovo, et al., (2019), compare three machine learning classifiers: k-NN, SVM, and Random Forest to establish the state of traffic on roads from data collected through video sequences that capture vehicular traffic information using computer vision techniques. The highest performing algorithm was Random Forest with 0.84, beating the others by more than three points [14].

Tišljarić, et al., (2020), employ unsupervised learning techniques to estimate traffic congestion at the city scale on the main roads of Zagreb, Croatia, based on data delivered by around 4200 vehicles equipped with tracking devices that capture traffic heading information every 100 meters in terms of geographic coordinates. Enlistment of the data was performed using speed transition matrices to represent traffic data. The classification results indicate average accuracies greater than 91% when using hierarchical clustering to identify the three congestion levels [15].

Multiple authors have experimented with algorithms and techniques to determine the volume of traffic and classify the congestion level in urban road networks and have suggested that machine learning models have the potential to serve as a reference for publishing accurate information on the traffic state and preventing congestion and traffic risk [16–18].

2.1. Area of Opportunity

Most classification results that use machine learning techniques to estimate vehicular traffic flow depend on the fit of their parameters and the dataset size in terms of record volume, selected attributes, and data percentage for their training and validation processes [19]. Furthermore, it has been found in the literature that IoT traffic data analyzed with machine learning tools employ various urban environments such as intersections [20], highways [13] or suburban roads to evaluate the performance of classifiers.

Our work seeks to evaluate the behavior and effectiveness of different machine learning algorithms to classify and compare the state of traffic on highways and highway intersections within the city in dataset that provided by 3900 traffic measurement stations. Furthermore, we will look for the best effectiveness by evaluating the classifiers with different validation criteria and traffic information according to the four seasons of the year.

3. Experimental Framework

3.1. Dataset

For the proposed experiment, we used a dataset provided by the Performance Measurement System of the California Department of Transportation (CALTRANS PeMS), which provides vehicular traffic information collected in real time with more than 40,000 detectors that cover the road system in the main metropolitan areas of the state of California.

The selected CALTRANS PeMS dataset corresponds to the information provided on speed, occupancy, headway, intensity and direction of vehicle flow as well as metadata of date, time, detector ID and number of lanes. The collected data is sampled every 5 minutes, between July 2019 and December 2020 from around 3923 detectors located within District 4.

3.2. Data preparation

From the referenced described, to provide the object class for carrying out the classification task, an attribute called 'Level of Service' (LOS) is added, a measure that identifies the congestion or saturation level of a road section at a given time.

For each time interval, the LOS is established from the average vehicle density on a road segment [21]. The density for each time interval is established as the relationship between the intensity of the traffic flow and the speed in a segment of the highway.

The LOS thresholds established in our study, provided in Table 2, corresponded to levels established by the highway capacity manual for the basic segments of highways in an urban area of the National Academies of Sciences, Engineering, and Medicine [21].

Table 1. Statistical description of attributes, PEMS dataset

	Lanes	Ocupation	Velocity	Intensity
count	5141183	5141183	5141183	5141183
mean	3.99	0.057	63.52	827.35
std	0.91	0.043	7.68	447.23
25%	3	0.032	62.33	475
50%	4	0.050	65.23	776
75%	5	0.070	67.56	1142

Table 2. Level of Service (LOS), based on traffic flow density

Nivel de servicio	Densidad (Veh/mi/ln)
A	< 11
B	> 11 - 18
C	> 18 - 26
D	> 26 - 35
E	> 35 - 45
F	> 45

The geographical locations of the different detectors are shown in Figure 1, which also highlights the areas with the highest vehicle volume by using a heat map measured by the attribute 'Occupancy'. Occupancy is a measurement captured and delivered by each of the electronic detectors and indicates the number of vehicles that cover a portion or section of the road. The service level classified with the letter 'A' corresponds to fluid and decongested vehicle flow, and gradually, the service level classified with the letter 'F' corresponds to a level of severe congestion that can cause long queues in critical accesses. For the dataset, the distribution of the object class determined by the Level of Service (LOS), is shown in Figure 2.

The dataset used in this study were initially cleaned and pre-processed. In the first instance, missing and/or null data were detected and filled in by applying the average of the data for their respective attributes. Table 1 provides a statistical description of the datasets used and consists of count, mean, standard deviation, 25th percentile, median and 75th percentile of the columns that attribute numerical records. These parameters help identify and understand any modeling-related errors or biases generated from these variables.

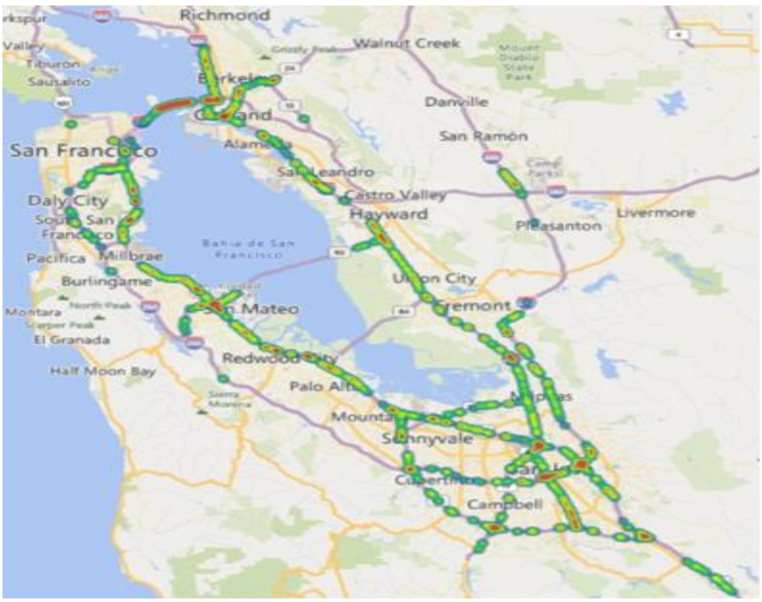

Figure 1. Network of vehicle flow detectors of Dataset1 (PeMS)

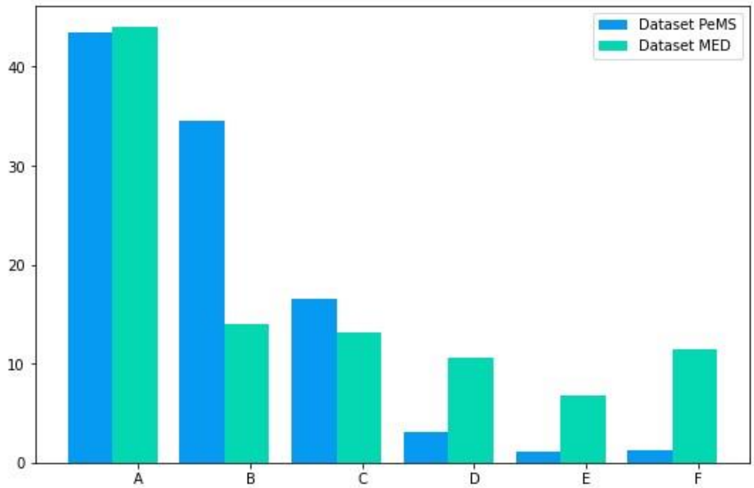

Figure 2. Object class distribution of Dataset

Subsequently, the numerical and categorical data were separated, and the numerical data were standardized, eliminating the mean and scaling the data so that its variance was equal to 1. The standard value for a sample x was calculated using the equation z = (x u) / s, where u is the average of the samples and s is the standard deviation. To enhance data quality, a procedure was implemented to identify and remove outliers - unusual values that deviated significantly from other observations. This process utilized the Interquartile Range (IQR) method, which measures the spread of the middle 50% of the dataset.

3.3. Classifiers

In this study, Decision Tree (DT), Random Forest (RF), Extra Decision Tree (ET), k-nearest neighbor (k-NN) classifiers, and a Multilayer Perceptron (MLP) Neural Network were used to assess the classification performance of the congestion level in the indicated urban areas. These classification techniques were carried out using the free software tool Scikit learning Python libraries version 3.9 on a Ryzen 7 5800H 3.2Ghz processor with 32GB RAM. A brief description of the methods used in the training is presented below.

3.3.1. Decision Trees algorithm (DT)

This algorithm builds a tree through nodes and branches using a recursive binary division approach. The root node first branches the population into homogeneous sets from the most significant input variable. The branches of a decision tree seek the optimal spacing [22]. If a node’s dataset belongs to the same class, it is considered a terminal node; however, if it belongs to several classes, the data are divided into smaller subsets based on a variable, and the process is repeated. The dataset is divided again according to the conditions formed from its variables until the last nodes of the scheme are formed, whose function is to indicate definitive classification. The successive division of the predictor space to obtain the best possible division of the node is generally performed using the algorithms of entropy, classification error, or the Gini index. This measures the probability that a record belongs to a node class. The Gini index measures the degree of purity of a node and selects the variable with the lowest weighted Gini, and is given by the equation 1 defined in [23]:

$$\mathrm{Gini} = 1 - \sum_{i=1}^{n} P_i^2 \tag{1}$$

where Pi is the probability of the i-th´ class.

Entropy is used to quantify the degree of uncertainty. A record with an uncertainty close to zero represents a class observation. Equation 2 represents the entropy measure:

$$H = -\sum_{i=1}^{n} P_i * \log_2 P_i \tag{2}$$

3.3.2. Random Forest algorithm (RF)

It is an ensemble algorithm that uses a set of decision trees combined with bagging for prediction. Each tree generates its own prediction from random subsets of samples obtained from the same population, and the predicted result is voted on for the final predictions. The most voted prediction result was the final prediction [24].

3.3.3. Extra trees algorithm (ET)

It is a variant of the decision tree, which takes decorrelation much further in each node, making it so that there is no dependency between the data being analyzed. Like RF, they consider random subsets of samples for each tree; nonetheless, the cutoff value that divides the node for each attribute is established by choosing the best among randomly generated cutoff values for each feature. This helps reduce variance [25].

3.3.4. k-NN algorithm

It is a non-parametric supervised learning classifier that selects a number k of labeled neighboring samples around a sample to be labeled and calculates their distances. The algorithm attributes a category to the sample by considering the most present label within the k-neighbors.

3.3.5. Multilayer Perceptron Algorithm (MLP)

An MLP is a neural network comprising layers of neurons. Each kth layer of neurons represents a set of neurons that receive the same input information, X_k, by means of connections or weights, W_k, and each neuron produces its own response, Z_k, where k, j =1, 2, ..., m and is the total number of neurons in the k-th layer of neurons. Each neuron in the network uses the same activation function to process the product of the input pattern and the connections associated with that input pattern. In certain cases, fixed inputs are considered, associated with a pattern b_k of values called a bias for each of the m neurons in the layer. If all the individual responses Z(k, j) are added in a pattern, $Z_k \in R^{mk}$, of the response, then the process carried out by the layer of neurons can be represented as Zk = fk(XkWk + bk), where $f_k(.)$ is the activation function that all the neurons of the kth layer use.

Table 3. Winter

	PeMS				
	60%	70%	80%	CrossVal	MR
DT	0.9987	0.9988	0.9989	0.9989	0.8798
RF	0.9974	0.9973	0.9975	0.9972	0.9112
ET	0.9961	0.9962	0.9964	0.9958	0.9143
KNN	0.9939	0.9941	0.9944	0.9936	0.8898
MLP	0.9971	0.9971	0.9968	0.9972	0.8891

Table 4. Spring

	PeMS				
	60%	70%	80%	CrossVal	MR
DT	0.9993	0.9994	0.9994	0.9993	0.9304
RF	0.9986	0.9986	0.9975	0.9988	0.9983
ET	0.9979	0.9982	0.9983	0.9976	0.9496
KNN	0.9957	0.9966	0.9956	0.9958	0.9320
MLP	0.9978	0.9980	0.9981	0.9979	0.9702

4. Results and discussions

In this section, we test our approach using mobility data from Caltrans PeMS for the dataset and vehicle mobility data. We recorded data at 15-minute intervals, for a year from December 21, 2019, to December 21, 2020. We performed four experiments using the seasons of the year as a reference (Subset1: Winter, Subset2: Spring, Subset3: Summer, Subset4: Autumn). We used 60%, 70%, and 80% of the data as the training set, and 40%, 30%, and 20% as the test set, respectively, a cross-validation criterion of 10 partitions, and a representative sample (MR) with a 95% trust level. Classification results were measured using an accuracy metric (Acc).

We measured the classification effectiveness of several algorithms: DT, RF, ET, KNN, and MLP. Tables 3, 4, 5, 6 show the classification effectiveness of each approach for the experimental datasets.

The machine learning algorithms were implemented with the Python library Scikitlearn (v3.6.9). The DT algorithm used in this library employs an optimized version of the CART (classification and regression Trees) algorithm. CART builds binary trees using the function and threshold that produces the greatest information gain at each node. For the DT algorithm, we use the Gini index as a criterion to split the tree node and find the possible class, which is equivalent to minimizing the registration loss between the true labels and the probabilistic predictions of the tree model for a given class. The k-NN algorithm was implemented under the parameters of the 10 nearest neighbors.

The MLP algorithm with 50 hidden layers and an ’Adam’ algorithm to optimize the weights at each level, and a RELU activation function present better results compared to reference algorithms. The ’Adam’ optimizer is an adaptive optimization method that requires low computational cost and works well on relatively large datasets (with thousands of training samples or more) [26] in terms of training and validation times. Its operation estimates the first and second moments of the gradients to update the network weights [27]. An important advantage of neural networks is that they can learn and generalize information. MLP is tolerant of missing values and can model complex relationships, such as nonlinear trends. In addition, it can support multiple inputs.

The results show that the amount of data or the percentage of training can affect the accuracy of the vehicle flow classification. In all the scenarios presented in Tables a slight improvement was observed when considering a percentage of 80% in the training data. The poorest results were obtained when we selected the training set from the representative sample formulated by [28], which has a lower percentage of records.

Table 6. Summer

	PeMS				
	60%	70%	80%	CrossVal	MR
DT	0.9990	0.9991	0.9991	0.9990	0.9235
RF	0.9980	0.9979	0.9980	0.9976	0.9212
ET	0.9965	0.9969	0.9971	0.9961	0.9151
KNN	0.9922	0.9945	0.9960	0.9970	0.8807
MLP	0.9944	0.9938	0.9944	0.9966	0.9480

Table 7. Autumn

	PeMS				
	60%	70%	80%	CrossVal	MR
DT	0.9991	0.9993	0.9994	0.9993	0.9357
RF	0.9980	0.9983	0.9984	0.9982	0.9441
ET	0.9975	0.9975	0.9975	0.9974	0.9426
KNN	0.9942	0.9946	0.9962	0.9947	0.9243
MLP	0.9976	0.9979	0.9980	0.9976	0.8708

The performance results indicate that the DTs reach the highest precision of the experiment, and it is observed that the DT has better classification results in the PEMS dataset, with values greater than 99.89% accuracy. It can be seen that the DTs are more amenable to the use of small amounts of data for training.

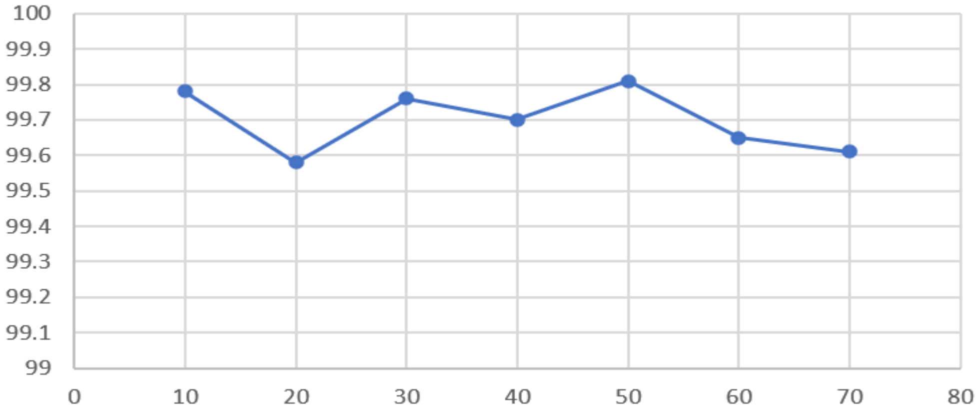

Figure 3. MLP accuracy in relation to number of hidden layers

One of the objectives of our study was to validate machine learning classifiers to determine the level of vehicular congestion in large-scale traffic networks. The results obtained were as follows:

The machine learning algorithms used in this study are efficient in classifying vehicular traffic flow in urban areas, considering that some ML algorithms are based on statistical decisions, instance decisions, and neural networks. By experimenting with the MLP algorithm trained by backpropagation and modifying its parameters, we validated that its effectiveness depends on the number of layers selected and the optimization algorithm for its training, as reported in the literature [29]. By varying the number of layers, it was possible to modify the effectiveness of the classifier, and the most accurate result is given with 50 hidden layers with an 'Adam' algorithm to optimize training. Figure 3 shows the classification results of the MLP algorithm obtained by varying the number of hidden layers. Nevertheless, when all features were considered, lower precision results were achieved. We also observed that the highest classification results were obtained in the spring dataset, a date that coincides with the mobility restrictions established in the cities considered caused by the Covid19 pandemic.

5. Conclusions and future work

A methodology was presented in this study for the classification of the state of traffic in an urban environment based on data collected by electronic detectors that capture information on flow, speed, and vehicular occupancy, in PEMS dataset. Machine learning algorithms such as decision trees, random forests, random extra trees, k-nearest neighbors, and multilayer neural networks have been considered and compared. Generally, the comparison indicates that the considered machine learning algorithms deliver satisfactory results in terms of effectiveness for traffic state classification. Nonetheless, the ability to learn and generalize information from neural networks allows satisfactory classification results that exceed the average of other approaches. The results show that in addition to the selected algorithm and the variation in its parameters, the percentage of training data affects the accuracy of vehicular traffic state classification. In future research,

we will explore the behavior of machine learning algorithms with attributes, including spatial dependencies, such as weather conditions and information from social networks. We also introduce hybrid algorithms that extract the most information of interest from the temporal and spatial features to predict the congestion level and vehicular traffic flow on city streets and roads.

Acknowledgments: This project was supported by CONAHCYT (Mexican government agency) through a scholarship with registration number 805075 (2021-2024), and the computational resources and technical assistance provided by Laboratorio Nacional de Supercomputo del Sureste de Mexico (LNS), through project 202201020N.

Conflicts of Interest: “The authors declare no conflicts of interest”.

WORKS CITED

Schrank, D., Eisele, B., Lomax, T. (2019). Urban mobility report 2019.

Estupiñan, N., Scorcia, H., Navas, C., Zegras, C., Rodríguez, D., Vergel-Tovar, E., Vasconcellos, E. (2018). Transporte y desarrollo en América Latina.

TomTom, T. (2019). Traffic congestion ranking: Tom tom traffic index.

Zhang, D., Kabuka, M. R. (2018). Combining weather condition data to predict traffic flow: a GRU-based deep learning approach. IET Intelligent Transport Systems, 12(7), 578-585.

Boukerche, A., Wang, J. (2020). Machine learning-based traffic prediction models for intelligent transportation systems. Computer Networks, 181, 107530.

Fantin Irudaya Raj, E., Appadurai, M. (2022). Internet of things-based smart transportation system for smart cities. In Intelligent Systems for Social Good: Theory and Practice (pp. 39-50). Singapore: Springer Nature Singapore.

Zantalis, F., Koulouras, G., Karabetsos, S., Kandris, D. (2019). A review of machine learning and IoT in smart transportation. Future Internet, 11(4), 94.

Bui, K. H. N., Oh, H., Yi, H. (2020). Traffic density classification using sound datasets: an empirical study on traffic flow at asymmetric roads. IEEE Access, 8, 125671-125679.

Montoya-Torres, J. R., Moreno, S., Guerrero, W. J., Mejía, G. (2021). Big data analytics and intelligent transportation systems. IFAC-PapersOnLine, 54(2), 216-220.

Ata, A., Khan, M. A., Abbas, S., Khan, M. S., Ahmad, G. (2021). Adaptive IoT empowered smart road traffic congestion control system using supervised machine learning algorithm. The Computer Journal, 64(11), 1672-1679.

Bandaragoda, T., De Silva, D., Kleyko, D., Osipov, E., Wiklund, U., Alahakoon, D. (2019, October). Trajectory clustering of road traffic in urban environments using incremental machine learning in combination with hyperdimensional computing. In 2019 IEEE intelligent transportation systems conference (ITSC) (pp. 1664-1670). IEEE.

Mondal, M. A., Rehena, Z. (2019, April). Identifying traffic congestion pattern using k-means clustering technique. In 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU) (pp. 1-5). IEEE.

Cheng, Z., Wang, W., Lu, J., Xing, X. (2020). Classifying the traffic state of urban expressways: A machine-learning approach. Transportation Research Part A: Policy and Practice, 137, 411-428.

Impedovo, D., Balducci, F., Dentamaro, V., Pirlo, G. (2019). Vehicular traffic congestion classification by visual features and deep learning approaches: a comparison. Sensors, 19(23), 5213.

Tišljaric´, L., Caric´, T., Abramovic´, B., Fratrovic´, T. (2020). Traffic state estimation and classification on citywide scale using speed transition matrices. Sustainability, 12(18), 7278..

Mandhare, P. A., Kharat, V., Patil, C. Y. (2018). Intelligent road traffic control system for traffic congestion: a perspective. International Journal of Computer Sciences and Engineering, 6(07), 2018.
Majumdar, S., Subhani, M. M., Roullier, B., Anjum, A., Zhu, R. (2021). Congestion prediction for smart sustainable cities using IoT and machine learning approaches. Sustainable Cities and Society, 64, 102500.
Bokaba, T., Doorsamy, W., Paul, B. S. (2022). Comparative study of machine learning classifiers for modelling road traffic accidents. Applied Sciences, 12(2), 828.
Medina-Salgado, B., Sanchez-DelaCruz, E., Pozos-Parra, P., Sierra, J. E. (2022). Urban traffic flow prediction techniques: A review. Sustainable Computing: Informatics and Systems, 35, 100739.
Ata, A., Khan, M. A., Abbas, S., Ahmad, G., Fatima, A. (2019). Modelling smart road traffic congestion control system using machine learning techniques. Neural Network World, 29(2), 99-110.
National Academies of Sciences, Engineering, and Medicine. (2022). Highway capacity manual 7th edition: A guide for multimodal mobility analysis.
Witten, D., James, G. (2013). An introduction to statistical learning with applications in R. springer publication.
Sharda, R., Delen, D., Turban, E. (2014). Business intelligence and analytics: systems for decision support. Pearson.
Zhang, C., Ma, Y. (Eds.). (2012). Ensemble machine learning: methods and applications. Springer Science & Business Media.
Rodrigo, J. A. (2017). Árboles de predicción: bagging, random forest, boosting y C 5.0. Estadística con R. p, 97.
Zhang, J., Hu, F., Li, L., Xu, X., Yang, Z., Chen, Y. (2019). An adaptive mechanism to achieve learning rate dynamically. Neural Computing and Applications, 31(10), 6685-6698.
Kingma, D. P., Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
Spiegel, M. R., Stephens, L. (1991). Teoría elemental del muestreo. Spiegel MR: Estadística. Madrid, Mc-Graw-Hill Interamericana, 186-207.
Ladic´, T., Mandekic´, A. (2021). Face mask classification using MLP Classifier. Ri-STEM-2021, 10(68), 77.