International Journal for Traffic and Transport Engineering, 2021, 11(2): 184 - 198
DOI: http://dx.doi.org/10.7708/ijtte2021.11(2).01 *UDC: 656.021*

TRAFFIC VOLUMES PREDICTION USING BIG DATA ANALYTICS METHODS

Slađana Janković[1], Ana Uzelac[2], Stefan Zdravković[3], Dušan Mladenović[4], Snežana Mladenović[5], Ivana Andrijanić[6]

[1,2,3,4,5] *University of Belgrade, Faculty of Transport and Traffic Engineering, Vojvode Stepe 305, 11000 Belgrade, Serbia*

[6] *Public Enterprise "Roads of Serbia", Bulevar kralja Petra I 28a, Novi Sad, Serbia*

Received 30 November 2020; accepted 20 January 2021

Abstract: The use of various advanced traffic data collection systems on one hand, and the development of Big Data technologies for the storage and processing of large amounts of data on the other hand, have enabled the application of various non-parametric methods for traffic volume prediction. In this research, the possibilities of application of supervised machine learning, as a method of Big Data analytics, with the aim to predict various indicators of the traffic volume were investigated. The research was conducted through two case studies. In both studies, for training and testing predictive models, traffic data generated by selected automatic traffic counters on the roads in the Republic of Serbia, in the period from 2011 to 2018, were used. Prediction models were trained, tested and applied using Weka software tool. The most basic data preparation was performed using macros for MS Excel written in VBA (Visual Basic for Applications). In the first case study, the goal was to predict the total volume of traffic by days, on selected sections of state roads in the Republic of Serbia. The datasets used for training and testing of machine learning models in the first case study were prepared using MS Access database, and the prediction results were presented using Excel Pivot Charts. In the second case study, we selected one counting point and performed prediction of the hourly vehicle flow, by directions and in total for both directions. The preparation of data sets, as well as the visualization of the results of the Big Data analysis in the second case study, was performed using programs written in the Python programming language. On the prepared data sets, using Weka software tool, different regression prediction models were trained and tested in both case studies. In the first case study, the best results were received by models based on regression decision trees, while in the second study, models based on Lazy IBk, Random Forest, Random Committee and Random Tree algorithms were among best. In each of the case studies, the best prediction model was selected by comparing model performance measures, such as: correlation coefficient, mean absolute error, and square root of mean square error. The model based on the M5P algorithm has shown the best performance in the first study, while the Lazy IBk algorithm gave the best results in the second study. Using the best predictive models, the prediction of daily or hourly traffic for 2020 was made at selected traffic counting points. Supervised machine learning has proven to be an effective method in predicting the volume of traffic flow.

Keywords: Big Data analytics, machine learning, Python, Weka, traffic prediction.

[1] Corresponding author: s.jankovic@sf.bg.ac.rs

1. Introduction

Since the early 1980s, researchers have been applying various parametric and non-parametric methods to predict the traffic volume (the volume of traffic flow) (Xu, Kong and Liu, 2013). If the available dataset is either small and/or data allocation function is already known, parametric methods are recommended, while in the case of a large amount of data and an unknown data allocation function, non-parametric methods are recommended. Various traffic flow data collection systems and Big Data technologies for storage and processing of collected data have enabled the expansion of the application of non-parametric methods in this area. In this paper, the possibilities of applying one of the non - parametric prediction methods (supervised machine learning) for predicting the traffic volume are investigated. Traffic volume is defined as the number of vehicles crossing a section or over a point of road per unit time at any selected period (Parvathi and Akki, 2017).

Having in mind the amount of available data, the nature of the defined problem and the technique chosen to solve it, this problem can be classified as a problem of Big Data analytics. The research was conducted through two case studies. In the first study, a prediction of the daily traffic volume at selected locations on state roads in the Republic of Serbia, for 2020, was made. In the second study, the volume of the traffic flow per hour was predicted at one selected location, for the whole 2020 year. Various machine learning regression models were created, trained and tested using Weka software tool, and the models that have shown the best performance were used to predict the daily or hourly volume of the traffic flow.

The second section of the paper presents the machine learning algorithms that have given the best results in this research and were used to predict the volume of the traffic flow. In the third section we have described the methodology we used in order to conduct case studies, while the fourth section presents the most significant results obtained in the case studies. The last section contains concluding remarks.

2. Literature Review

Traffic flow forecasting has become one of the main tasks in the field of smart transport systems (Lippi *et al.,* 2013). Statistical methods, artificial intelligence, and data mining techniques have been progressively used in the recent years with the aim to analyze data in the road traffic and to predict future traffic indicators (Aqib *et al.,* 2019). Previous researches indicate that there is no single technology that is capable of analyzing large datasets only by itself. Therefore, depending on the data structure and its volume, it is necessary to apply the appropriate technology in order to get the best insight from the collected data. The smooth traffic flow is influenced by various factors that can be categorized into following four groups: patterns of activities that affect daily traffic, anomalies of activity patterns, weather conditions, and time of holidays and vacations (Xie *et al.,* 2020). While selecting appropriate methods for data

processing, these factors should be taken into consideration as they are found to be very important for the traffic prediction.

Traffic sensors and counters generate a large amount of traffic data, which can be processed with the aim to receive significant information to support and improve traffic control. Kong *et al.* (2019) developed a new approach in the traffic forecasting based on machine learning techniques. They believe that the size of data received in the traffic is suitable for applying this technique. In a study conducted by Aqib *et al.* (2019), a traffic prediction model based on the application of the LSTM machine learning algorithm was developed. In addition to the widespread use of time series in traffic estimations (Salamanis *et al.,* 2015), there are various approaches in the traffic prediction based on machine learning algorithms such as SVR, Random Forest, and neural networks (Bratsas *et al.,* 2020).

Machine learning algorithms united under the name of decision trees have a significant role in this research. Decision trees are one of the most popular classes of supervised machine learning algorithms, which are most commonly used for classification, but can also be used for regression analysis. The prediction model, based on any algorithm from this class, predicts the output value based on the input values of several parameters. Input values, as well as output values, can be either categorical or continuous. According to the type of the output value, decision trees are divided into classification and regression trees. In the case of classification trees, the output variable is categorical, while in the case of regression trees the output variable is continuous. Decision trees are popular because they offer systematic structure that is understandable to humans. Decision tree is based on discrete objective functions approximation in which the learning function is represented in the form of a tree, where each node in the tree is related to some attribute of the instance, branches that exit the node have different values for that attribute, and leaves correspond to the values of the objective function. The instances of the observed phenomenon are described by the values of their attributes. The classification is performed starting from the root, then going down to the branches that corresponds to the value of the tested attribute of the instance we are classifying, and when the leaf is reached, the class is assigned to the instance. The leaves of the tree are the nodes of the response, while the others are called the nodes of the decision. In decision trees, each internal node divides the input data into two or more subspaces according to certain discrete functions of the input values of the attributes. The basic algorithm for making a decision tree is several decades old and was developed by John Ross Quinlan (Quinlan, 1986). The first version of the algorithm was known as ID3 (Iterative Dichotomiser 3), while later versions of the algorithm removed some of the limitations of the original algorithm, and improved classification performance. Quinlan (1992) created the M5 algorithm for learning the decision tree where the dependent variable is continuous. This algorithm allows each decision tree leaf to be a single linear regression function. Wang and Witten (1996) improved the M5

algorithm and the CART (Classification and Regression Trees) system (Breiman *et al.,* 1984), and called the improved algorithm the M5' tree model. The M5' tree model was implemented in the Weka software tool called M5P (Witten *et al.,* 2017) and will be used to predict the volume of daily traffic flow, in the first case study conducted in this research.

In another case study of this research, the k-Nearest Neighbors (k-NN) algorithm was used to predict the volume of the hourly traffic flow. It is implemented in the Weka software tool under the name Lazy IBk. It is a nonparametric method of pattern recognition, used for classification and regression (Niu, Zhao and Zhang, 2013). In both cases, the input consists of k closest test examples. The output depends on whether k-NN was used for classification or regression. In the k-NN classification, the output is a member of the class. An object is classified by the votes of the majority of its neighbors, so that the object is arranged in a class most often among its k nearest neighbors (k is a positive integer). If k = 1, the object is simply assigned to the class of the nearest neighbor. In k-LV regression, the output represents the value of the object. This value represents the average value of its k nearest neighbors (Saadatfar *et al.,* 2020).

3. Methodology

Two case studies were conducted as part of the research. Both studies were conducted following the same methodology, and through the subsequent activities: data collection, data preparation, Big Data analysis and discussion of analysis results.

The method of predictive analysis, based on machine learning, was chosen for the Big Data analysis of traffic data. Training, validation and testing of machine learning models were performed in the data mining software tool Weka (Waikato Environment for Knowledge Analysis). This software is a set of machine learning algorithms used to detect patterns in data.

3.1. Data Acquisition

In both case studies, datasets derived from data generated by selected automatic traffic counters on state roads of the I category in Serbia, during the period from 1.1.2011. to 31.12.2018, were used. Selected traffic counters (21 in total) are located on the following roads:

- Road number 1, IA category (state border with Hungary: border crossing Horgoš - Novi Sad - Beograd - Niš - Vranje – stete border with Northern Macedonia: border crossing Preševo);
- Road number 22, IB category (Beograd - Ljig - Gornji Milanovac - Preljina - Kraljevo - Raška - Novi Pazar - Ribariće – stete border with Montenegro: border crossing Mehov Krš);
- Road number 23, IB category (Pojate - Kruševac - Kraljevo - Preljina - Čačak - Požega - Užice - Čajetina - Nova Varoš - Prijepolje – stete border with Montenegro: border crossing Gostun) and
- Road number 46, IB category (Ravni Gaj - Knić - Mrčajevci).

The geographical locations of the traffic counters, that generated data used in the case studies, are shown as blue circles in Figure 1.

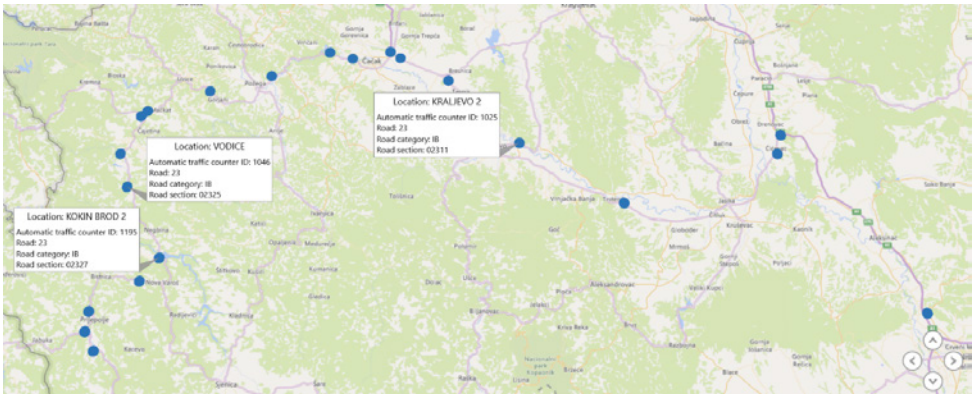

Fig. 1.

Geographic Locations of Automatic Traffic Counters that Generated Data used in our Case Studies

3.2. Data Preparation

In the first part of this research, data preparation was performed with the aim to obtain files that are suitable for Big Data analysis in the selected software tool. We transformed source files to the favorable files suitable for applying machine learning algorithms using Excel macros written in VBA (Visual Basic for Applications) and programs written in the Python programming language. The source files were XLS (Microsoft Excel Spreadsheets) files, and the files used directly in the Weka software tool were CSV (Comma-Separated Values) and ARFF (Attribute-Relation File Format) files.

3.3. Big Data Analytics Method

Since we were equipped with the labeled dataset, in the research we have used methods of supervised machine learning. Building each of the machine learning model consisted of the following phases:

1. Defining the goal of the model;
2. Choosing dependent variables (label, class), i.e. the dataset attribute which value we want to predict using the machine learning model. Different traffic volume indicators expressed in the number of vehicles per unit time were selected as dependent variables in this study;
3. Selecting relevant attributes (features) of a dataset;
4. Selecting supervised machine learning algorithm, according to the nature of labels and attributes (Jain, Murty and Flynn, 1999). As the target variables were numerical, we have used regression algorithms such as: linear regression, regression tree, neural networks, etc.;
5. Datasets (training and test) preprocessing that fulfills requirements of the selected algorithm;
6. Model tuning – setting hyperparameters that are specific for each type of the machine learning algorithm;

7. Model training – application of the selected machine learning algorithm on the training dataset in order to obtain model parameters;

Model evaluating using cross-validation. Cross-validation is a method for getting a reliable estimate of model performance using only training data. Witten et al. (2017) proposed several alternative measures that can be used to evaluate the success of numeric prediction: mean-squared error - Eq. (1), mean-absolute error - Eq. (2), root mean-squared error - Eq. (3), relative-squared error - Eq. (4), root relative-squared error - Eq. (5), relative-absolute error - Eq. (6) and correlation coefficient - Eq. (7). These metrics were used to evaluate the machine learning models in this research. The total number of instances for testing is n; the projected values on test instances are p_1, p_2, ..., p_n; real values are a_1, a_2, ..., a_n; $\overline{p}$ and $\overline{a}$ are mean values of projected or actual values.

$$Mean-squared\ error = \frac{(p_1-a_1)^2+\cdots+(p_n-a_n)^2}{n} \quad (1)$$

$$Mean-absolute\ error = \frac{|p_1-a_1|+\cdots+|p_n-a_n|}{n} \quad (2)$$

$$Root\ mean-squared\ error = \sqrt{\frac{(p_1-a_1)^2+\cdots+(p_n-a_n)^2}{n}} \quad (3)$$

$$Relative-squared\ error = \frac{(p_1-a_1)^2+\cdots+(p_n-a_n)^2}{(a_1-\bar{a})^2+\cdots+(a_n-\bar{a})^2} \quad (4)$$

$$Root\ relative-squared\ error = \sqrt{\frac{(p_1-a_1)^2+\cdots+(p_n-a_n)^2}{(a_1-\bar{a})^2+\cdots+(a_n-\bar{a})^2}} \quad (5)$$

$$Relative-absolute\ error = \frac{|p_1-a_1|+\cdots+|p_n-a_n|}{|a_1-\bar{a}|+\cdots+|a_n-\bar{a}|} \quad (6)$$

$$Correlation\ coefficient = \frac{S_{PA}}{\sqrt{S_P S_A}} \quad (7)$$

where:

$$S_{PA} = \frac{\sum_{i=1}^{n}(p_i-\bar{p})(a_i-\bar{a})}{n-1} \quad S_P = \frac{\sum_{i=1}^{n}(p_i-\bar{p})^2}{n-1} \quad S_A = \frac{\sum_{i=1}^{n}(a_i-\bar{a})^2}{n-1} \quad (8)$$

8. Model testing - to predict the performance of a model on a new dataset, we need to assess its performance measures on a dataset that played no part in the formation of the model. This independent dataset is called the test dataset. Comparing test vs. training performance allows us to avoid overfitting. If the model performs very well on the training data but poorly on the test data, then it is overfit;
9. Selecting a winning model - model that has the best performance on the test dataset;
10. Labels prediction using the winning model.

4. Data Analysis

In this section we will describe the datasets that we have analyzed, all the phases present in model development, application of the machine learning models, as well as the results of predictive analysis, for each case study individually.

4.1. Case Study I

The dataset used in the case study contains data on the total number of vehicles registered by each selected traffic meter on a daily basis. The instances of the initial dataset are described by the following attributes: Counter ID, date, day of the week, and vehicle number. The aim of the case study was to create machine learning models on the available dataset to predict the volume of traffic flow by days, at each counting point, for the whole 2020 year. The attribute number of vehicles was chosen as the target (dependent) variable.

We transformed attribute that describes date into three new attributes: day, month, and year with the aim to capture their individual influences on the target attribute. Instances related to the period between 2011 and 2015 year were selected as a training dataset, while the instances relating to the period from 2016 to 2018 year were used to test trained machine learning models. The training dataset consisted of 37,616 instances, while the test dataset contained 22,818 instances.

The following seven machine learning algorithms were applied to the training dataset using Weka software tool: Linear Regression, Multilayer Perceptron, Lazy IBk (k-Nearest Neighbors), M5P, Random Forest, Random Tree, and REPTree. For model validation, 10-fold cross validation that is already implemented in Weka software was used. Models based on Linear Regression and Multilayer Perceptron algorithms were rejected as they have shown very low performance (a correlation coefficient of 0.2364 and 0.2256, respectively, was recorded). The performance of the remaining five prediction models is shown in Table 1. Based on the data shown in Table 1, it is easy to see that the models based on decision tree type algorithms (M5P, Random Forest, Random Tree and REPTree) are superior to the Lazy IBk algorithm -the closest neighbors).

Table 1

Performance of the Five Prediction Models that have shown the Best Results on the Training Dataset

Algorithm	Correlation Coefficient	Mean Absolute Error	Root Mean – Squared Error	Relative Absolute Error (%)	Root Relative – Squared Error (%)
Lazy IBk	0.6287	2615.4468	3092.3207	84.2493	78.0204
M5P	0.9778	505.5362	832.2666	16.2844	20.9984
Random Forest	0.9752	532.6613	877.7513	17.1582	22.1460
Random Tree	0.9578	666.1694	1153.0679	21.4588	29.0923
REPTree	0.9732	559.2707	912.5619	18.0153	23.0243

At this phase of the research, models based on decision trees have shown the best results in predicting the volume of the traffic flow. Since the target variable is numerical, these algorithms are regression decision trees. In order to determine which model is the best, it is necessary to perform a model evaluation. In order to obtain an objective assessment of the model performance, it was necessary to evaluate the model by calculating the metrics of its performance on a new, previously unknown set of data. Such dataset is called a test dataset. The performance of models that had the best performance in the previous phase (models based on decision trees), measured on the test dataset, is shown in Table 2. The first thing that can be observed from Table 2 is that the correlation coefficient has extremely high values for all predictive models. For each model, the correlation coefficient measured on the test dataset is slightly lower than the correlation coefficient measured on the training dataset. This means that none of the predictive model has a problem of over-adaptation. Comparing all calculated performance metrics, it is obvious that models based on the M5P, Random Forest, and REPTree algorithms are better than the model based on the Random Tree algorithm.

Table 2

Performance of chosen Predictive Models obtained using Test Dataset

Algorithm	Correlation Coefficient	Mean Absolute Error	Root Mean – Squared Error	Relative Absolute Error (%)	Root relative – Squared Error (%)
M5P	0.9756	1279.0156	1738.7847	35.0184	35.6941
Random Forest	0.9657	1371.4177	1863.3353	37.5482	38.2509
Random Tree	0.9488	1465.0600	2043.0803	40.1121	41.9407
REPTree	0.9717	1299.6219	1760.8794	35.5825	36.1477

In order to get better understanding and clearer comparation of the results received by three prediction models, the visualization of the predicted volume of daily traffic flows was performed by applying these three models on the test dataset. Figure 2 shows, as an example, the ratio of the actual daily number of vehicles registered by the counter marked 1025, in November 2016, and the projected number of vehicles obtained by applying these three prediction models (for the same counter for the same time period).

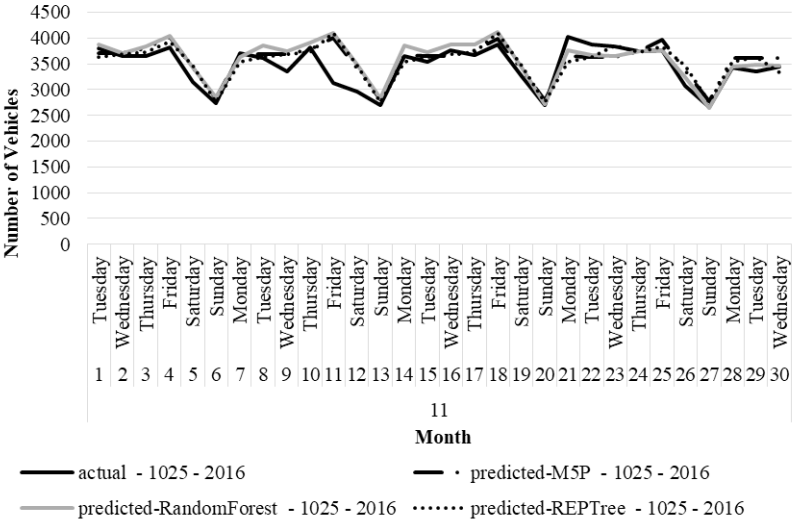

Fig. 2.

Actual and Predicted Daily Vehicle Flows Recorded by Counter 1025 in November 2016

Figure 3 shows the actual and projected daily vehicle flows registered by counter 1195 during July, August, September, and October 2016, and Figure 4 shows the total monthly actual and projected vehicle flows by days of the week, registered by the same counter during the same time period. The geographical locations of the counters marked 1025 and 1195, as well as details of these counters, can be seen in Figure 1. In Figures 2, 3 and 4 it can be seen that the lines representing the predictions using the M5P and REPTree algorithms (red and yellow lines) almost overlap and deviate slightly

less from the line of actual values, compared to the line that represents prediction using the Random Forest algorithm (gray line). In addition, the performance of these two models are slightly better than the performance of the model based on the Random Forest algorithm (Table 2). Finally, by comparing the performance of the model based on the M5P algorithm and the model based on the REPTree algorithm, according to the all performance measures, the M5P algorithm has a slight advantage (Table 2) over others. Thus, the model based on the M5P algorithm was chosen as the best one with the aim to be used for traffic volume prediction in the future.

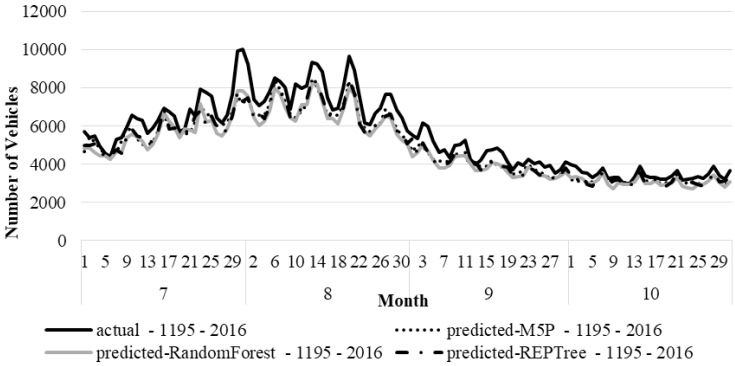

Fig. 3.

Actual and Predicted Daily Vehicle Flows Registered by 1195 Counter during July, August, September and October 2016

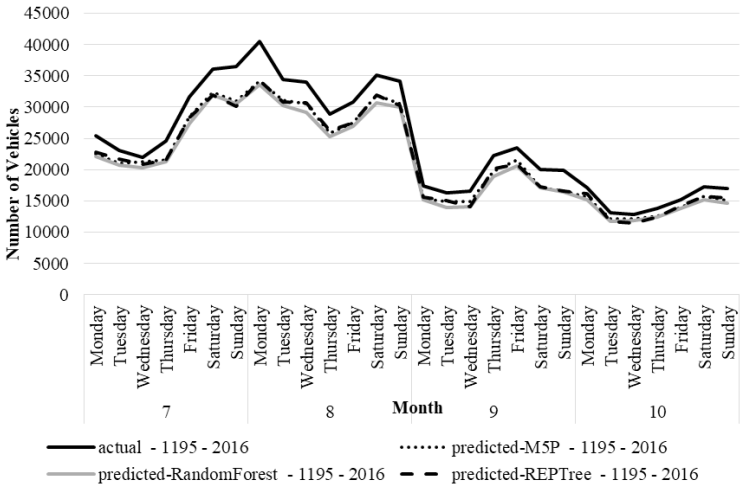

Fig. 4.

Total Monthly Actual and Predicted Vehicle Flows by Days of the Week Register by Counter 1195 During July, August, September and October 2016

Figure 5 shows part of the graphical presentation of the decision tree generated using Weka software tool, a prediction model based on the M5P algorithm, which was selected as the best one. Each leaf in this decision tree is a linear model by which the value of the target variable (number of vehicles) is determined during one specific passing through the decision tree, from root to leaf. The total number of leaves, i.e. linear models in this decision tree is 417.

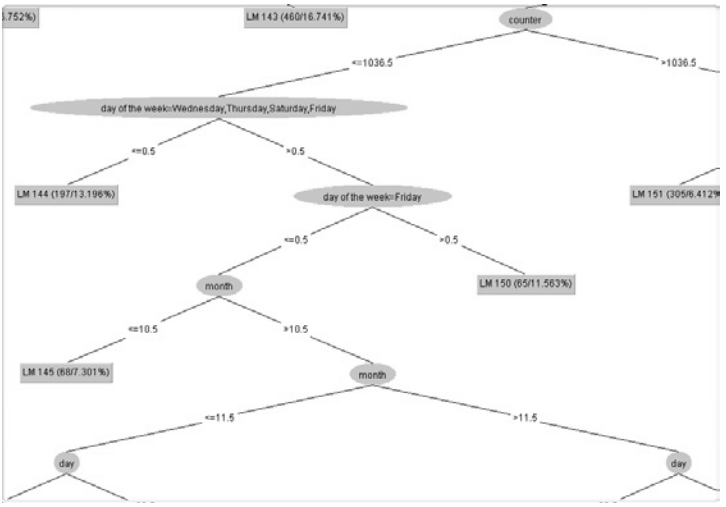

Fig. 5.
Part of the Decision Tree Model Based on the M5P Algorithm using Weka Software Tool

Actual daily vehicle flows for the following years: 2016, 2017, and 2018 and projected daily vehicle flows for the 2020 year at counter marked 1195 are presented in the Fig. 6. The projections for the 2020 year are obtained by applying a machine learning model based on the M5P algorithm.

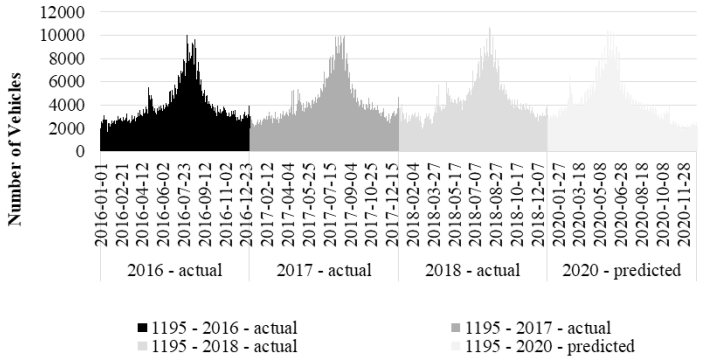

Fig. 6.
Actual Daily Vehicle Flows for the Years: 2016, 2017, and 2018 and Predicted Daily Vehicle Flow for the 2020 Year on Counter Number 1195

4.2. Case Study II

This case study was conducted on a dataset generated by an automatic traffic meter marked 1046. The selected meter (1046) is located on the road number 23, IB category (Pojate - Kruševac - Kraljevo - Preljina - Čačak - Požega - Užice - Čajetina - Nova Varos - Prijepolje - state border with Montenegro: border crossing Gostun), and is located in Vodice.

The dataset used in the case study contains data about vehicle flows, separately for each direction, registered every hour by the selected traffic counter. The counter was generating a set of data over a period of 8 years. The generated dataset consists of 140,256 instances, and each instance is described by the following attributes: date, day of the week, hour, direction, and the number of vehicles. The aim of the case study was to create machine learning models on the available dataset to predict the number of vehicles at the selected point, by directions, for each hour. The attribute number of vehicles was chosen as the target (dependent) variable. Instances related to the period 2011-2015 were used for model training, while the instances relating to the period from 2016-2018 were used to evaluate the developed machine learning model. Thus, the training dataset contained 87,648 instances, while the test dataset was consisted of 52,608 instances. Since the target variable is numerical, all regression algorithms for machine learning, which are available in the Weka software tool, were applied on the training dataset. The models were trained and evaluated by 10-fold cross-validation on the training set, and later the best of them were additionally evaluated at the test dataset. Among tested algorithms tested, only four had good results. The performance of the best four machine learning models, measured on a training dataset, using 10-fold cross-validation, is shown in Table 3.

Table 3

Performance of the Best Four Machine Learning Models, Measured on a Training Dataset

Algorithm	Correlation Coefficient	Mean Absolute Value	Mean Root Value
Lazy IBk	0.9424	13.4235	19.1172
Random Committee	0.9413	13.1908	19.1248
Random Forest	0.954	11.7609	16.9391
Random Tree	0.9259	14.9551	21.6507

For each model, the projected and actual number of vehicles per hour, individually for each direction, for the first 100 instances are shown (Figure 7). The visualization was done using the Python programming language. Actual values are represented using blue colour, while projected values are red. Due to the clarity of the chart, only the first 100 instances are shown. We can notice that all four models give quite similar predictions and that the differences in accuracy are rather minimal.

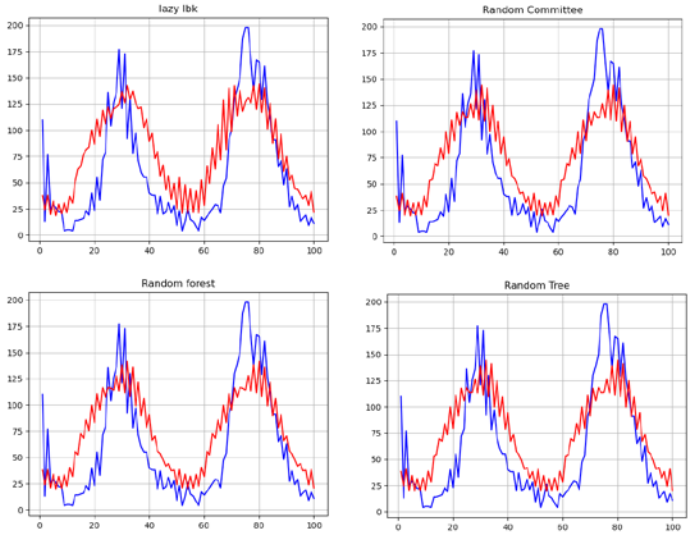

Fig. 7.
Actual and Projected Vehicle Flows by Hour, for Both Directions on a Counter Marked 1046 Generated Using Four Best Algorithms (First 100 Instances Related to the Training Dataset are Shown)

In order to determine the model that solves the previously appointed problem with the best results, we performed the evaluation of these four models. The evaluation has shown that the performances of these models are very similar. The Random Forest algorithm has the best correlation coefficient on the training dataset. In order to achieve the most objective assessment of the models performance, an evaluation was conducted on a new, previously unknown dataset - a test dataset. The evaluation was performed using the same metrics as previously.

The performance of the selected machine learning models, built on a test dataset, are shown in Table 4. The presented results show that the model based on the Lazy IBk algorithm has slightly better performance (has a higher correlation coefficient, but also lower mean absolute and mean square error) than the other models. That is why this model was chosen as the best.

Table 4
Performance of Chosen Machine Learning Models Build on the Test Dataset

Algorithm	Correlation Coefficient	Mean Absolute Error	Mean Root Error
Lazy IBk	0.727	31.7978	50.8012
Random Committee	0.7025	32.7818	52.3746
Random Forest	0.7018	32.8781	52.5327
Random Tree	0.7025	32.7818	52.3746

Using a model based on the Lazy IBk algorithm, a prediction on the number of vehicles by direction at the selected location, for each hour, for every day for the whole 2020 year was made. The first 100 projected values were visualized using the Python programming language and the visualisation is presented in the Figure 8.

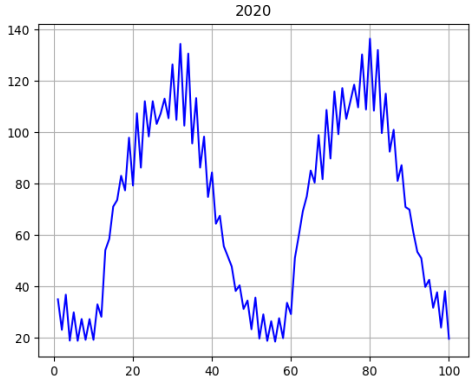

Fig. 8.
Projected Vehicle Flows by the Hour, for Both Directions for the Whole 2020 Year for the Counter 1046 (Presented First 100 Instances)

5. Conclusion

The focus of this paper is to investigate the possibility of applying different machine learning algorithms, as one of the techniques of Big Data analytics, with the aim to predict daily and hourly traffic volume. The research was conducted through two case studies. The first case study trained models for predicting daily traffic volume, based on the following regression algorithms: Linear Regression, Multilayer Perceptron, Lazy IBk (k-Nearest Neighbors), M5P, Random Forest, Random Tree, and REPTree. On the training dataset, models based on regression decision trees have shown significantly better performance than other models. Therefore, only these models were evaluated on the test dataset. Among the tested models, the best results were received by the model based on the M5P algorithm, so that the prediction of the daily traffic volume was performed using this model.

In another case study, traffic time prediction models based on regression algorithms were trained, using the following algorithms: Lazy IBk (k-Nearest Neighbors), Random Forest, Random Tree and Random Committee. On the training dataset, all algorithms had approximately similar performance, with the Random Forest algorithm being slightly better than the other three. We evaluated these models on the test dataset and used the same metrics. Again, these models have shown very similar performance, but this time the model based on the Lazy IBk algorithm showed slightly better results than the other models.

In another case study, models based on regression algorithms such as Lazy IBk (k-Nearest Neighbors), Random Forest, Random Tree, and Random Committee were trained with the aim to predict the hourly volume of the traffic flow. On the training dataset, all algorithms had approximately

similar performance, while the Random Forest algorithm was slightly better than the other three. Then we performed model evaluation on test dataset using the same metrics. It has been shown that the models were approximately similar in performance, but this time the model based on the Lazy IBk algorithm has shown slightly better results than the others.

Algorithms that belong to the decision tree class have given good results in both our case studies. As the most significant advantage of decision trees, besides being applicable to both regression and classification problems, lays in high interpretability. The tree can be easily visualized, in order to analyze the prediction process. Another significant advantage of this class of algorithms is the simple preparation of data for model training and testing. These algorithms can work with attributes of almost all types: binary, nominal, numerical, date, as well as with missing values, so that standardization and normalization of data are not required. In addition to decision trees, the k-Nearest Neighbors algorithm have also shown good results in the traffic volume prediction. Both case studies have shown that machine learning can be effectively applied in traffic volume prediction.

Visualization of the actual and projected volume of daily traffic have shown irregularities on an annual level, while the graphical presentation of actual and projected hourly traffic showed irregularities on a daily basis. This indicates that in the future research that use the same datasets, it would be expedient to apply another machine learning technique - clustering. The expected results of the clustering would be clusters of counting points with some common characteristics of the daily vehicle flow, as well as clusters of time periods of the day with a similar traffic loads.

Acknowledgement

This work was partially supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia, within the project number 036012.

References

Aqib, M.; Mehmood, R.; Alzahrani, A.; Katib, I.; Albeshri, A.; Altowaijri, S.M. 2019. Smarter Traffic Prediction Using Big Data, In-Memory Computing, Deep Learning and GPUs, *Sensors* 19: 2206.

Bratsas, C.; Koupidis, K.; Salanova, J. M.; Giannakopoulos, K.; Kaloudis, A.; Aifadopoulou, G. 2020. A comparison of machine learning methods for the prediction of traffic speed in urban places, *Sustainability* 12(1): 1-15.

Breiman, L.; Friedman, J.; Olshen, R.; Stone, C. 1984. *Classification and Regression Trees*. Belmont, California: Wadsworth.

Jain, A. K.; Murty, M. N.; Flynn, P. 1999. Data clustering: a review, *ACM Comput Surveys* 31(3): 264–323.

Kong, F.; Li, J.; Jiang, B.; Zhang, T.; Song, H. 2019. Big data-driven machine learning-enabled traffic flow prediction, *Transactions on Emerging Telecommunications Technologies* 30(9): 1-13.

Lippi, M.; Bertini, M.; Frasconi, P. 2013. Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning, *IEEE Transactions on Intelligent Transportation Systems* 14(2): 871–882.

Niu, K.; Zhao, F.; Zhang, S. 2013. A Fast Classification Algorithm for Big Data Based on KNN, *Journal of Applied Sciences* 13(12): 2208-2212.

Parvathi, M. S.; Akki, B. 2017. Classified Traffic Volume Study at Ghatekesar Junction, *International Journal of Engineering and Techniques* 3(6): 420-435.

Quinlan, J.R. 1986. Induction of decision trees, *Machine Learning* 1: 81-106.

Quinlan, J.R. 1992. Learning with Continuous Classes. In *Proceedings of Australian Joint Conference on Artificial Intelligence,* Hobart, Australia, 343-348.

Saadatfar, H.; Khosravi, S.; Joloudari, J.H.; Mosavi, A.; Shamshirband, S. 2020. A New K-Nearest Neighbors Classifier for Big Data Based on Efficient Data Pruning, *Mathematics* 8: 286.

Salamanis, A.; Meladianos, P.; Kehagias, D.; Tzovaras, D. 2015. Evaluating the Effect of Time Series Segmentation on STARIMA-Based Traffic Prediction Model. In *Proceedings of the IEEE Conference on Intelligent Transportation Systems, ITSC*, 2225-2230.

Wang, Y.; Witten, I. H. 1996. Induction of model trees for predicting continuous classes. *Working Paper 96/23*. Hamilton, New Zealand: The University of Waikato.

Witten, I. H.; Frank, E.; Hall, M. A.; Pal, C. J. 2017. *Data Mining: Practical Machine Learning Tools and Techniques, (4th ed.)*. Burlington, USA: Morgan Kaufmann.

Xie, P.; Li, T.; Liu, J.; Du, S.; Yang, X.; Zhang, J. 2020. Urban flow prediction from spatiotemporal data using machine learning: A survey, *Information Fusion* 59: 1-12.

Xu, Y.; Kong, Q.; Liu, Y. 2013. Short-term traffic volume prediction using classification and regression trees. In *Proceedings of the 2013 IEEE Intelligent Vehicles Symposium (IV),* Gold Coast, Australia, 493-498.