Building Quantile Regression Models for Predicting Traffic Flow

Kirill Smelyakov [1], Olha Klochko[1], Zoia Dudar[1]

[1] *Kharkiv National University of Radio Electronics, 14 Nauky Ave., Kharkiv, 61166, Ukraine*

Abstract

Traffic is one of the most important aspects of managing cities and transportation infrastructure. Fast and accurate traffic forecasting can help solve various transportation-related problems, such as congestion, increased air pollution, and improved road safety. In this paper, we investigate the use of quantile regression and its modifications such as KNN Quantile Regression, Random Forest Quantile Regression, Gradient Boosting Quantile Regression, and XGBoost Quantile Regression for traffic intervals prediction using Uber data on traffic in Kyiv in January 2020. Results showed the Gradient Boosting Quantile Regression model appeared to perform the best. But others KNN and Random Forest algorithms work well for lower quantiles and XGBoost work the best for the median. The findings of this paper is that it can be used to improve traffic forecasting, which is an important task for traffic management authorities, logistics and transportation companies, and other stakeholders.

Keywords

Traffic flow, quantile regression, speed prediction, machine learning

1. Introduction

Almost all cities in the world face serious congestion problems. Excessive traffic flow leads to the paralysis of the urban transportation system on a daily basis, which creates great inconvenience and a negative impact on people's travel. Different countries are actively taking appropriate measures, i.e. redirecting traffic, limiting the number or expanding the scale of the road network, but these measures may have little effect [1].

Intelligent transport systems are used to manage traffic flows, allowing real-time data collection and processing of information about the road network, including traffic speed, number of vehicles for a certain period, traffic density, road network occupancy, and public transport schedules.

There are several reasons for the need to regulate urban traffic flows in Ukrainian cities [2]: increasing urbanization, growing congestion on the road network, poor quality of public transport services, inconvenient routes, long travel times, etc. These problems are especially acute in the largest cities and encourage citizens to increasingly choose a car for daily correspondence, which in turn increases delays, travel time, and leads to environmental pollution [3].

The relevance of this work is to find tools for managing and monitoring these processes in cities. According to the developing but still insufficient scientific literature, which focuses on how the dynamism of intelligent transport systems affects urban innovation and how traffic management tools can be activated to obtain optimal results, it is important to analyze urban transport systems as a dynamic whole.

The aim of the paper is to research the efficiency using quantile regression models for predicting traffic flow based on historical data on example of average speed of cars per hour on a particular road segment, to evaluate the accuracy of the prediction and describe the applicability to improve the road traffic system

COLINS-2023: 7th International Conference on Computational Linguistics and Intelligent Systems, April 20–21, 2023, Kharkiv, Ukraine

EMAIL: kyrylo.smelyakov@nure.ua (K. Smelyakov); olha.klochko.cpe@nure.ua (O.Klochko), zoia.dudar@nure.ua (Z. Dudar).
ORCID: 0000-0001-9938-5489 (K. Smelyakov); 0009-0008-5355-4222 (O.Klochko), 0000-0001-5728-9253 (Z. Dudar).

CEUR Workshop Proceedings (CEUR-WS.org)

2. Related works

Traffic forecasting is an important task in the field of transportation logistics and road traffic management. Research in the field of traffic prediction uses machine learning methods, in particular quantile regression methods.

The authors of work [4] describe internal and external, static and dynamic factors affecting traffic conditions. Internal factors:

- Driving behavior (dynamic);
- Vehicle information (static);
- Vehicle condition (dynamic).

External factors:

- Traffic flow condition (dynamic);
- Weather conditions (dynamic);
- Traffic rules and regulations (static);
- Traffic signals and events (dynamic).

Dynamic factors are known to change over time, so they are more difficult to model than static factors. Thus, in forecasting, historical and current information on dynamic factors should usually be considered together. Finally, this section analyzes the main factors affecting various forecasts. The first is classified in terms of the vehicle, which represents the internal factors of the vehicle and the external factors of the environment, as shown in Figure 1 and Figure 2.

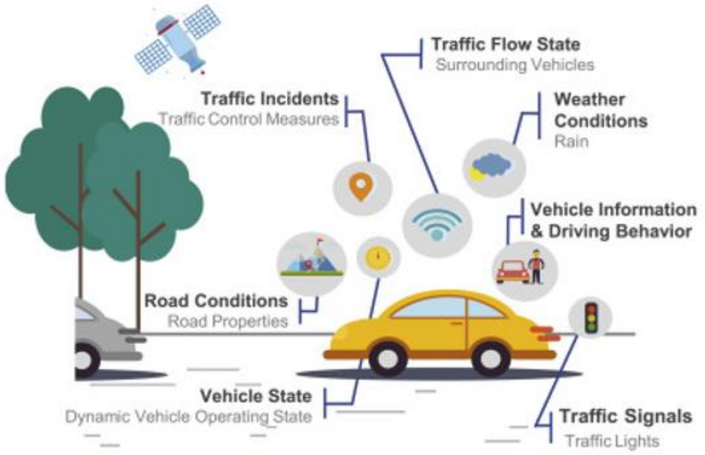


Figure 1: Vehicle speed prediction [4]

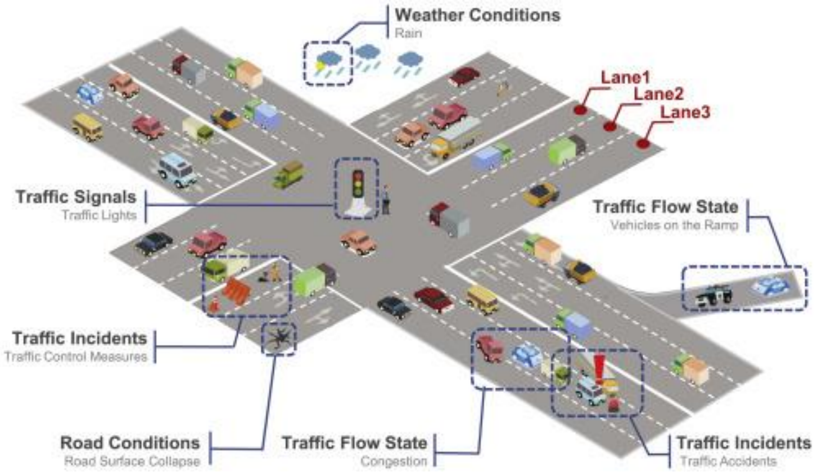


Figure 2: Flow speed prediction [4]

Most studies on estimating the traffic flow of an entire road network are based on one or more road network properties, and the results may not be promising [5-7], and evaluating the efficiency of network transfer or tuning parameters of an intelligent system were seen in recent researchers [8,9]. There is a way to combine five topological indicators and road length to estimate traffic flow based on a multiple regression approach [10]. Six measures are used to estimate traffic flow: road length, proximity, intermediate, degree, page rank, and clustering coefficient [5]. It is worth noting that each measure requires a different correlation for different types of traffic data.

Big data methods are used in a wide range of fields and industries, including: e-commerce, healthcare, transportation, energy, government, education [11,12]. It drives innovation and improve efficiency in many different industries, leading to significant advancements in technology and business practices.

The application of the KNN algorithm in short-term urban traffic forecasting. The KNN algorithm has good performance in dealing with sudden changes and non-linearity of urban traffic flow due to its non-parametric regression characteristics [13,14]. However, the long execution time of the KNN forecasting system leads to a decrease in forecasting efficiency. To solve this problem, the two-stage search algorithm proposed in this paper finds and identifies the best decision input set from historical data using two similarity measures. Experimental results show that this method effectively improves the prediction performance of the system under the condition of guaranteeing the accuracy of the original prediction. The ideas presented here can be further explored with additional data, such as weather conditions or emergencies, more complex urban topologies, and different types of forecasting methods [15].

Work [16-18] show quantile regression to predict traffic based on smartphone data. They compared different quantile regression methods, including nearest neighbors, random forests, and gradient boosting, and found that the gradient boosting method gave the best results. As well as a number of statistical methods to predict the 5th, 10th, 25th, 50th, 75th, and 90th percentile of traffic speeds. As a result of comparing the models, the authors concluded that the nearest neighbor method and random forests showed the best performance for traffic prediction using quantile regression.

Also several quantile regression methods were compared to predict traffic speed on a highway [19,20]. The authors compared different methods, such as the nearest neighbor method, random forests, gradient boosting, and XGBoost. They compared the results with several other congestion prediction methods and concluded that the random forest method is effective in predicting quantile values of road congestion. According to the results of the study, the XGBoost method showed the best performance for predicting speed quantiles on the highway.

The quantile regression method can be combined or combined with other methods to improve forecast accuracy, so this article [21] describes an algorithm for short-term nonparametric probabilistic quantile regression forecasting that incorporates the advantages of a hybrid neural network and quantile regression.

Approaching the quantile regression problem [22,23] from a multitasking perspective solves the unpleasant problem of overlapping quantiles, while greatly outperforming current quantile regression methods. Work say that jointly modeling the mean and several conditional quantiles leads to improved predictions of conditional expectation due to the additional information and regularization effects caused by the added quantiles.

Also in the literature there are studies using artificial neural networks [24], like long short term memory. Describes the state of the lack of traffic speed data and proposes a method for predicting traffic speed based on measuring traffic flow in the previous and later moment states. The performance of five prediction models was compared: KNN, support vector regression (SVR), classification trees, exactly long short term memory (LSTM) and back propagation (BP) [25]. The method works on the basis of the LSTM model and achieves the best result.

In general, many studies use quantile regression methods to predict traffic speeds and traffic congestion. Different methods are used, such as the nearest neighbor method, random forests, gradient boosting, and XGBoost. Each of these methods has its own advantages and disadvantages, so the choice of method depends on the specific task and the amount of data, i.e. searching in Big Data Warehouses [26].

The effect of the dataset on evaluating urban traffic prediction was analyzed as well and experimental results show that the predictive effect of the multiscale model is much better than that of the single-scale prediction and fully reflects the data set, adding more information is of greater research value [27]. These resources may provide further insights and perspectives on the use of data science and machine learning techniques for predicting and analyzing transportation patterns and trends.

Consequently, research in quantile regression for traffic prediction is ongoing, and allows for the development of increasingly accurate and efficient methods to solve this important problem.

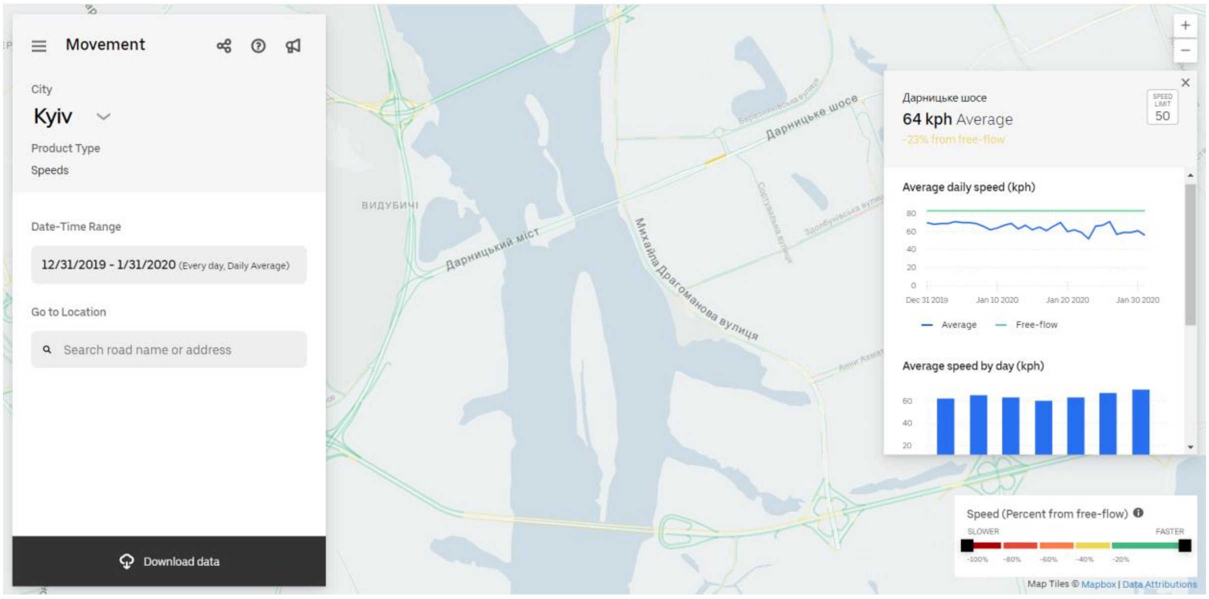
3. Method and materials

Consider the dataset, machine learning methods and metrics that was used in further experiments.

3.1. Dataset Description

As mentioned earlier, traffic data is collected by many organizations involved in transportation, logistics, and mapping services. However, due to certain restrictions, such data is usually not publicly available. Most of the traffic data is provided by taxi services. Also, up-to-date data was needed, as most open datasets store information on traffic speeds up to 2012. Since it was decided to use Kyiv data to build the model, it was decided to search for the necessary information on the resources of well-known taxi services.

There are several large taxi services in Kyiv, one of the largest is Uber [28]. An important fact is that in 2018, the company launched the Uber Movement resource, which provides access to data on the speed of taxi movement of this service over time. It contains data from January 2018 to March 2020. The data is divided into sets, each of which contains information about the average taxi speed on a segment of the region's road for each hour of each day of a particular month. The data includes only those observations for which there is data on at least 5 unique trips on the segment in question at the time point in question (Figure 3).



City
Kyiv

Product Type
Speeds

Date-Time Range
12/31/2019 - 1/31/2020 (Every day, Daily Average)

Дарницьке шосе
64 kph Average
-23% from free-flow
SPEED LIMIT 50

Figure 3: Uber Movement Speeds Web Exploration Tool for Kyiv [28]

It includes the following fields:

- year - year of observation;
- month - the number of the observation month (from 1, which corresponds to January, to 12, which corresponds to December);
- day - day of observation (from 1 to 31);
- hour - hour of observation in local time (from 0 to 23);
- utc_timestamp - date and time of observation in UTC (Coordinated Universal Time) format;
- osm_way_id - OpenStreetMap road identifier for the corresponding segment;

- osm_start_node_id - the corresponding OpenStreetMap node identifier for the start of the segment;
- osm_end_node_id - the corresponding OpenStreetMap node ID for the end of the segment;
- speed_kph_mean - the average speed of Uber vehicles on the corresponding road segment in km/h;
- speed_kph_stddev - standard deviation of the speed on the corresponding road segment in km/h.

The road segment is fully defined by the OpenStreetMap road identifier, as well as the start and end node identifiers in OpenStreetMap. This data can be used to get information about the name of the street where the segment is located, as well as its location. Uber Movement also provides this data, but as a separate set.

3.2. Machine learning methods

There are various regression types. Regression models aim to fit a target variable that is expressed as a numerical vector. Nevertheless, statisticians have increasingly developed sophisticated regression techniques. Quantile regression (QR) is a procedure for estimating the parameters of a linear relationship between explanatory variables and a given level of the quantile of the variable being explained [29, 30].

Unlike ordinary least squares, quantile regression is a non-parametric method. This allows you to get more information: regression parameters for any quantiles of the distribution of the dependent variable. In addition, such a model is much less sensitive to outliers in the data and to violations of the assumptions about the nature of the distributions.

Quantile regression is a regression that intentionally introduces a bias into the result. Instead of looking for the mean of the predicted variable, quantile regression aims to find the median and any other quantiles (which are sometimes called percentiles). The classic and most straightforward prediction is that based on mean values: the respective over- and under-prediction weights must be equal, otherwise the prediction becomes biased (more accurately, biased relative to the mean value).

The first refinement of this approach is the median prediction: the corresponding over- and under-prediction frequencies must be equal, otherwise the prediction becomes biased relative to the median. At this point, we shifted the notion of unbiased predictions from equal weights to equal probability. This shift is not obvious, but it can make a huge numerical difference in some situations. The median value represents the threshold value where the distribution breaks down with a 50/50 probability. However, it is possible to consider other frequency ratios as well. For example, we can consider ratios of 80/20, 90/10, and any other, as long as their total value is 100%.

Quantiles are a generalization of the median value to any percentage expression. For τ, whose value is between 0 and 1, the quantile regression Q(τ) represents the threshold value at which the probability of a value below the threshold is equal to τ [31].

Strict mathematical definition of quantile according to the following: if Y is a random variable with a distribution function F(y) or a distribution density f(y), then the quantile q_τ of order $\tau \in [1,0]$ of a one-dimensional distribution is the value y_τ of the random variable Y for which the distribution function takes the value τ or there is a "jump" with a value less than τ to a value greater than τ. For continuous distributed, the quantile of order τ, where the number $\tau \in [1,0]$, is defined as the solution of the equation:

$$F(q) = \int_{-\infty}^{q} f(y)\, dy = \tau, \tag{1}$$

K-nearest neighbors (KNN) is a nonparametric regression tool that attempts to estimate the conditional mean for a new observation, x_0, by identifying the k points of observed data that are closest to the new observation for which a prediction is needed. The response values for these nearest observations are then averaged together [14-16]. The k-nearest neighbor predictions are more formally computed by using the following equation:

$$\hat{Y} = \frac{1}{k} \sum_{x_i \in N_k(x_0)} y_i, \tag{2}$$

where $N_k(x_0)$ is the neighborhood of x_0 defined by the k closest points xi in the training data. Since most observed data will likely have just one or no observations at a candidate x_0, the observed response values for the closest neighbors serve as an approximate for the conditional distribution Y|x = x_0. Thus, averaging across these observed values is an estimate of the conditional mean at x_0.

A regression tree, like KNN, is a nonparametric prediction method that approximates the conditional mean by using available data close in proximity to the point one wishes to predict. For continuous predictors, regression trees split the predictor space into high dimensional rectangles rather than using neighbors.

The random forest (RF) model is a highly valuable and applied nonparametric form of regression. The trees provide a natural way to automatically approximate f(X) without doing a lot of thinking about what the true from of f(X) looks like. Its bagging nature lends itself to better prediction accuracies than a regression tree and also allows for categorical predictors to be incorporated where other nonparametric tools, like KNN, do not.

XGBoost is a machine learning algorithm based on a decision search tree and using a gradient binning framework. In prediction tasks that use unstructured data (such as images or text), the artificial neural network outperforms all other algorithms or frameworks. But when it comes to structured or tabular data of small size, algorithms based on a decision tree take precedence [32,33].

XGBoost and Gradient Boosting Machines (GBM) are ensembles of tree methods that use the principle of boosting weak learners (most commonly, the binary decision tree algorithm) using a gradient descent architecture [33]. In turn, XGBoost is an improvement of the GBM framework through system optimization and algorithm refinement.

3.3. Machine learning metrics

To assess the prediction accuracy of the chosen models, in this paper, we use four statistical scores: Mean Absolute Percent Error (MAPE), Mean Absolute Error (MAE), Mean Square Error(MSE), Root Mean Square error (RMSE) [34]. They are calculated as follows

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|, \tag{3}$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y}_i|}{y_i}, \tag{4}$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2, \tag{5}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}, \tag{6}$$

where N – number of data points, y_i – observed value, $\hat{y}_i$ – predicted value.

4. Experiment

In general, the first step is to collect data. This can be data on some traffic characteristics. This can be done with the help of special devices that can be installed either outside the vehicle, such as radars, or inside, such as GPS trackers.

Recognition systems are often used to add data from surveillance cameras to determine traffic on the roads using computer vision technology. The data can be collected at a single point or part of the road, or at a set of observation points or road sections.

For performing experiment was used service Uber Movement that provide data in the public domain, in particular for academic purposes [28].

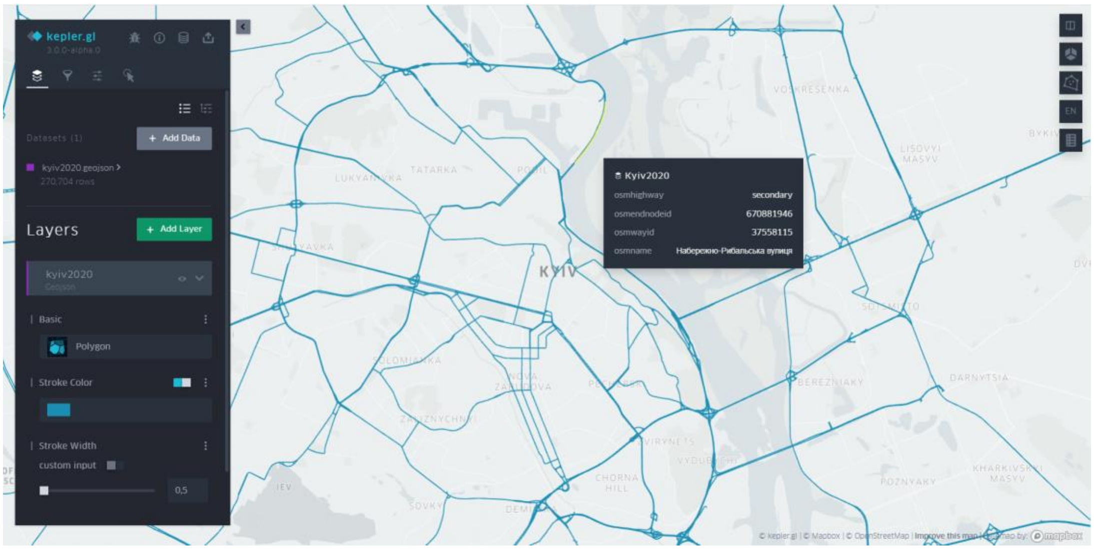
However, there is one non-obvious aspect in this case. On the one hand, the speed of a taxi generally reflects the speed of the traffic in which the vehicle is moving. However, sometimes this is not the case, in particular, there is a study that shows that taxis move somewhat slower than the traffic, which logically implies that taxis slow down traffic.

For performing experiment was downloaded dataset the Uber Movement Speed Data from Kyiv in January 2020 and appropriate .geojson file with the OpenStreetMap data.

The data of the main streets of the central part of Kyiv were selected for the study. In a set with segment meta-information, the importance of a street is determined by the osmhighway parameter.

In particular, the values trunk, primary, and secondary denote the main roads, and trunk_link, primary_link, and secondary_link denote the main connections between streets that do not have their own name, such as exits from overpasses or overpasses.

Roads that fit this description are shown in Figure 4. These are arterial streets bounded on one side by the so-called "small ring road" and on the other side by the Dnipro River.



Kyiv2020
osmhighway secondary
osmendnodeid 670881946
osmwayid 37558115
osmname Набережно-Рибальська вулиця

Figure 4: Segments of main streets in the central part of Kyiv [35]

The main programming tool in this work is the Python language for several reasons. Firstly, it is quite easy to use because it has an intuitive syntax. Thus, it is widely used by professionals at every level of their science/engineering career. Secondly, it has a large selection of libraries and frameworks for ML models. And the last reason is that it is a modern programming language that can be easily integrated with other languages if necessary. Python 3.10 and the Jupyter Notebook were used to simulate the prediction algorithm. Libraries for data analysis and visualization - NumPy, Pandas, Sklearn, matplotlib, seaborn.

This can be due to sensor failures or, for example, the absence of cars on a given road segment at a given time in the case of data from taxi services. For correct operation, it is necessary to pre-process the data under study, in particular, to restore the missing values in some way. Secondly, it is also important for researchers to pay attention to how the traffic data on the road network is organized. Given their origin, they usually contain temporal and spatial dependencies of varying complexity. In particular, they are characterized by seasonality in time, for example, both daily and weekly seasonality. Neighboring road segments also affect the traffic of the target segment, which indicates obvious spatial dependencies in this kind of data. If these aspects are ignored, it is difficult to build an adequate forecasting system. So, it was removed unnecessary fields such as the year and month, as the dataset contains data for only one year and month, as well as the id of the road segments, start and end segments, since they were duplicated.

The collected data is often not perfect and cannot be immediately used. They may contain gaps or unnecessary information that will overburden the algorithm, and as a result, the algorithm will not give an accurate prediction. Various reasons cause gaps in the collected telecommunications data: system problems, packet loss, interference, etc. Other data-related issues include sensor measurement errors, emissions, and gaps. For the experiment, data were taken from one Umanskaya street, which contains 412 examples, where from 10 to 14 measurements were made for each day. The chosen target variable is the traffic in one particular cell. Characteristics are selected for work - time (hour), day (day). As quantiles was chosen 0.1, 0.25, 0.5, 0.75, and 0.9.

We started by preprocessing the data and splitting it into training 70% and testing 30% sets. We then implemented the KNN Quantile Regression algorithm, Random Forest Quantile Regression algorithm, Gradient Boosting Quantile Regression, and XGBoost Quantile Regression algorithm using sklearn library and xgb. In the experiment, the parameter settings of all models are shown in Table 1.

Table 1
Algorithms' parameter description

Algorithm	Parameter description
K-nearest neighbors	neighbors = 5
Random Forest	n_estimartors = 100, max_depth = 10, random_state = 42
Gradient Boosting	n_estimartors = 100, max_depth = 10, random_state = 42, loss='quantile', alpha=q, min_samples_leaf=9, learning_rate=0.01
XGBoost	n_estimartors = 100, max_depth = 10, random_state = 42, alpha=q, learning_rate=0.01

We also defined a function to compute the quantile losses and plot the predicted versus actual values. The quality of forecasting can be assessed by the indicator included MAE, MSE, MAPE and RMSE from library sklearn.metrics.

Machine learning requires a lot of RAM. To speed up access to it, you need a processor that supports four channels, not two as in conventional custom solutions. To perform machine learning efficiently, it is important to consider the number of cores and the memory size of the graphics card. Since deep learning is a lot of linear functions, a lot of simple operations occurring at the same time, graphics processors are better suited for it. The fact is that they are designed for a lot of parallel calculations, while CPUs are designed for sequential ones. The wall training time of the model is highly dependent on processing power, was taken with an Intel Core i7 processor and 8 GB of RAM, which was used to test the wall training time with the given parameters.

Overall, the comparison provided insights into the strengths and weaknesses of each algorithm, and the results could be used to select the most appropriate algorithm.

5. Results

As a result of the experiment, we got the following plot of actual and predicted data with KNN Quantile Regression in Figure 4.

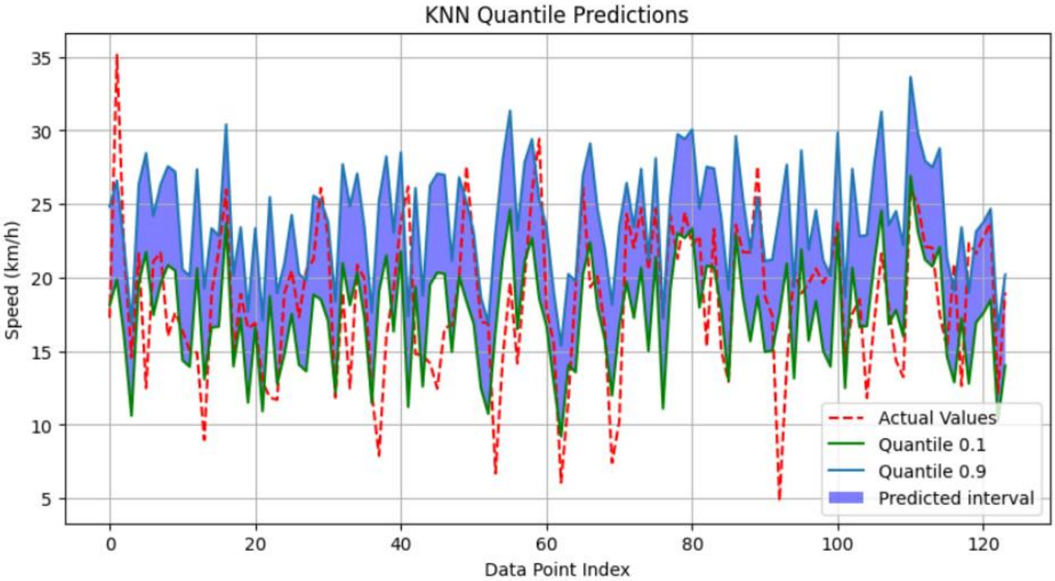


Figure 4: KNN Quantile Regression plot with predicted values

Table 2 shows more detailed information about predicted values with KNN Quantile Regression for all quantiles.

Table 2

KNN Quantile Regression predicted values

Actual Value	Quantile 0.1	Quantile 0.25	Quantile 0.5	Quantile 0.75	Quantile 0.9
17.273	18.111	19.1784	20.9654	23.0662	24.8344
35.127	25.1621	24.3558	22.7592	24.8254	26.5936
21.543	15.9387	16.8786	18.4326	20.6951	22.1128
14.589	10.6245	11.5644	13.1184	15.3809	16.7986
21.638	19.6254	20.6928	22.4798	24.5806	26.3488
12.496	21.739	22.8064	24.5934	26.6942	28.4624
21.107	17.4632	18.5306	20.3176	22.4184	24.1866
21.700	19.609	20.6764	22.4634	24.5642	26.3324

Table 3 shows error metrics for predicted values with KNN Quantile Regression for all quantiles.

Table 3
KNN Quantile Regression error metrics

Error Metric	Quantile 0.1	Quantile 0.25	Quantile 0.5	Quantile 0.75	Quantile 0.9
MAE	4.0871	3.882	3.996	4.8788	6.032
MSE	25.1621	24.3558	27.592	39.4032	55.0289
MAPE	0.2573	0.2672	0.2817	0.3524	0.4285
RMSE	5.0162	4.9352	5.2528	6.2772	7.4181

Then, we got the following plot of actual and predicted data with Random Forest Quantile Regression in Figure 5.

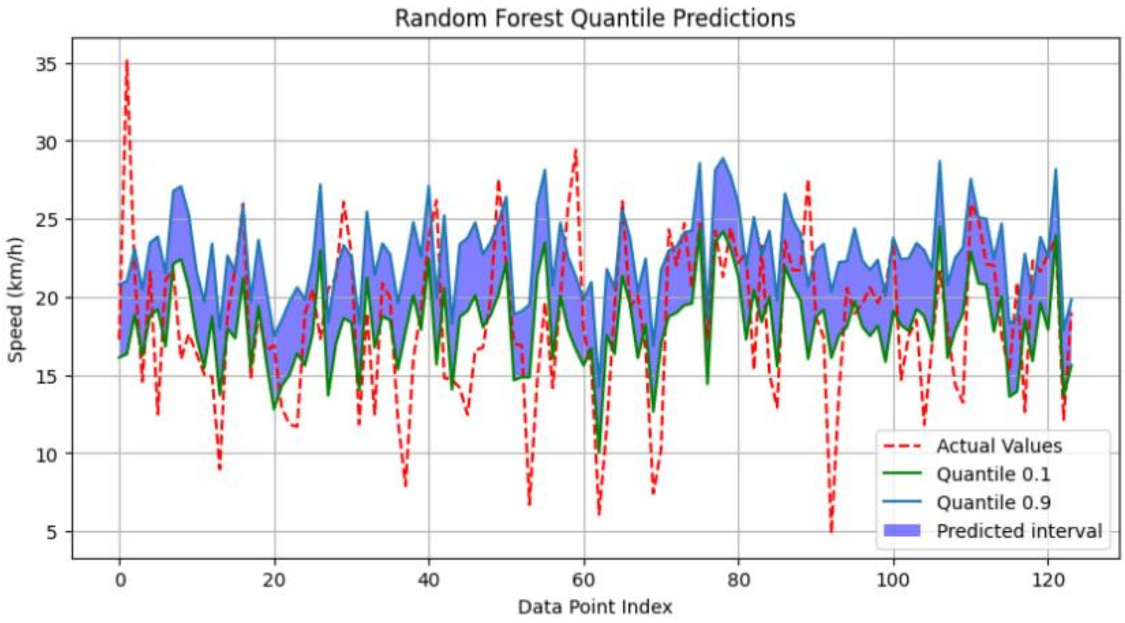


Figure 5: Random Forest Quantile Regression plot with predicted values

Table 4 shows more detailed information about predicted values with Random Forest Quantile Regression for all quantiles.

Table 4
Random Forest Quantile Regression predicted values

Actual Value	Quantile 0.1	Quantile 0.25	Quantile 0.5	Quantile 0.75	Quantile 0.9

17.273	16.1325	16.8245	17.8081	18.8774	20.7978
35.127	16.3438	17.0359	18.0194	19.0887	21.0092
21.543	18.8605	19.3505	20.0911	21.7207	23.0817
14.589	16.2318	16.7219	17.4624	19.092	20.453
21.638	18.8078	19.4999	20.4834	21.5527	23.4731
12.496	19.1998	19.8918	20.8754	21.9447	23.8651
21.107	16.8387	17.5307	18.5143	19.5836	21.504
21.700	22.1192	22.8112	23.7948	24.8641	26.7845

Table 5 shows error metrics for predicted values with Random Forest Quantile Regression for all quantiles.

Table 5
Random Forest Quantile Regression error metrics

Error Metric	Quantile 0.1	Quantile 0.25	Quantile 0.5	Quantile 0.75	Quantile 0.9
MAE	3.6162	3.5917	3.6817	4.1329	5.0252
MSE	21.6478	21.5349	22.665	27.7673	38.359
MAPE	0.2311	0.2374	0.253	0.294	0.3597
RMSE	4.6527	4.6406	4.7608	5.2695	6.1935

Then, we got the following plot of actual and predicted data with Gradient Boosting Quantile Regression in Figure 6.

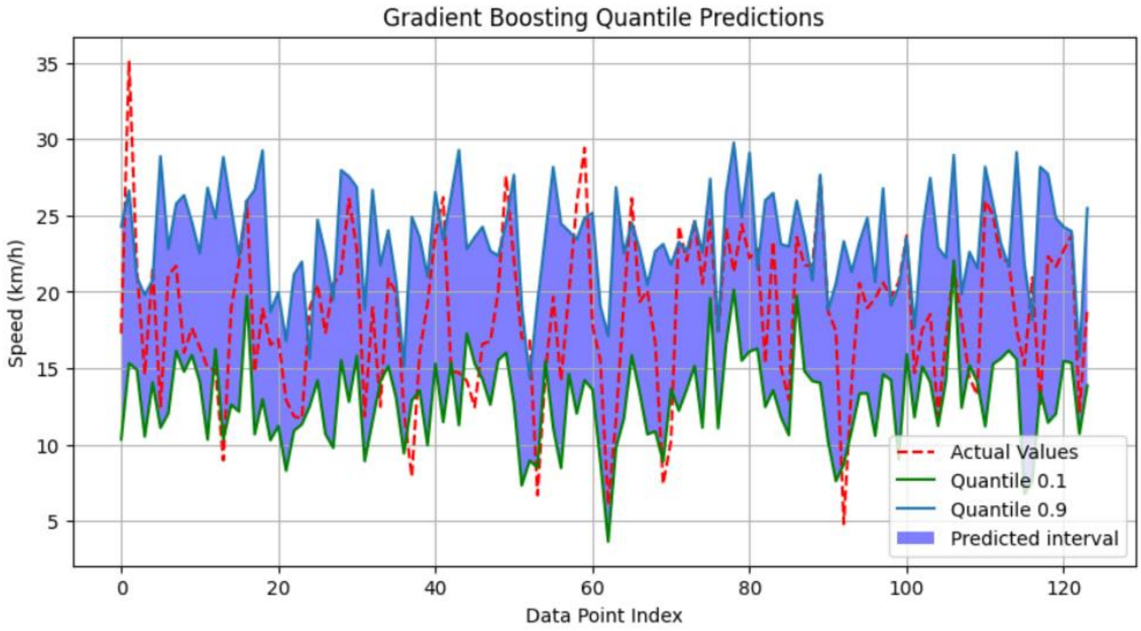


Figure 6: Gradient Boosting Quantile Regression plot with predicted values

Table 6 shows more detailed information about predicted values with Gradient Boosting Quantile Regression for all quantiles.

Table 6
Gradient Boosting Quantile Regression predicted values

Actual Value	Quantile 0.1	Quantile 0.25	Quantile 0.5	Quantile 0.75	Quantile 0.9
17.273	10.359	12.585	18.5325	22.0517	24.2792
35.127	15.3296	14.631	18.6229	22.2287	26.6235

21.543	14.905	17.0811	18.1323	20.237	20.966
14.589	10.5382	11.6224	15.6755	21.0406	19.8034
21.638	14.0761	18.0598	17.3835	19.6437	20.7779
12.496	11.1083	18.6569	19.9422	25.1292	28.874
21.107	12.0426	13.8044	16.119	21.7462	22.792
21.700	16.134	21.0012	22.8791	24.779	25.7685

Table 7 shows error metrics for predicted values with Gradient Boosting Quantile Regression for all quantiles.

Table 7
Gradient Boosting Quantile Regression error metrics

Error Metric	Quantile 0.1	Quantile 0.25	Quantile 0.5	Quantile 0.75	Quantile 0.9
MAE	6.0531	4.7557	3.7397	4.183	5.5759
MSE	50.7854	34.1377	21.4317	21.2053	51.6991
MAPE	0.3127	0.2589	0.234	0.299	0.4146
RMSE	7.1264	5.8427	4.6924	5.3109	7.1902

Then, we got the following plot of actual and predicted data with XGBoost Quantile Regression in Figure 7.

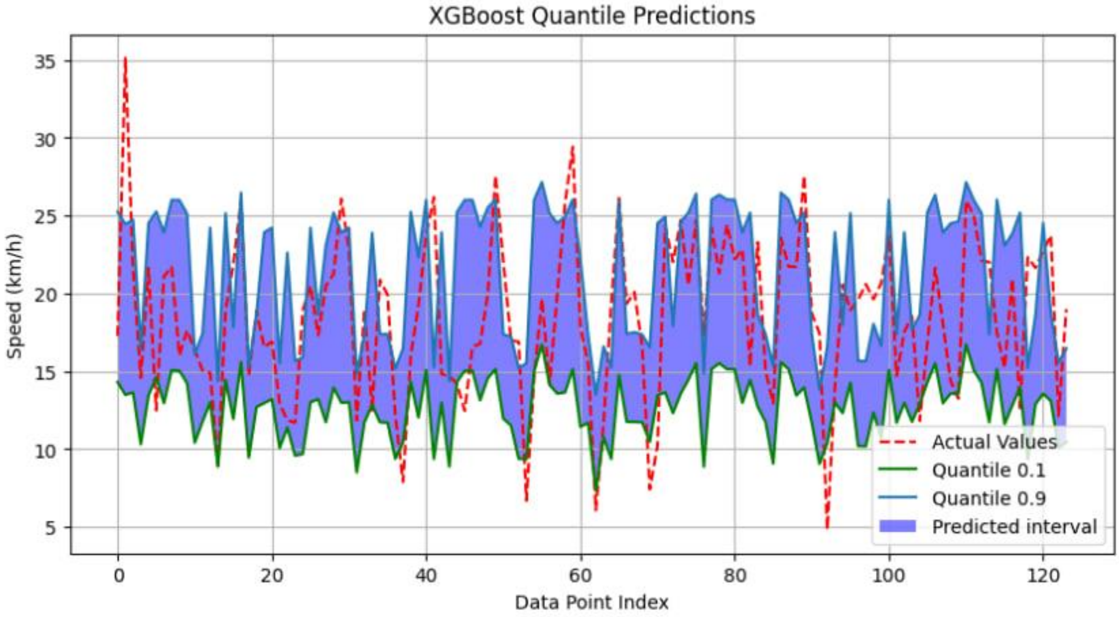


Figure 7: XGBoost Quantile Regression plot with predicted values

Table 8 shows more detailed information about predicted values with XGBoost Quantile Regression for all quantiles.

Table 8
XGBoost Quantile Regression predicted values

Actual Value	Quantile 0.1	Quantile 0.25	Quantile 0.5	Quantile 0.75	Quantile 0.9
17.273	14.308	16.4747	19.3855	22.0762	25.2414
35.127	13.489	15.6672	18.6009	21.3046	24.4588
21.543	13.641	15.8255	18.8474	21.5444	24.7337
14.589	10.3246	11.0647	12.4532	14.6278	16.404
21.638	13.4552	15.6713	18.6286	21.3149	24.5216

12.496	14.6191	16.7533	19.6139	22.203	25.263
21.107	12.9573	15.1261	18.068	20.7625	23.9235
21.700	15.0792	17.2347	20.1586	22.8382	26.0072

Table 9 shows error metrics for predicted values with XGBoost Quantile Regression for all quantiles.

Table 9
XGBoost Quantile Regression error metrics

Error Metric	Quantile 0.1	Quantile 0.25	Quantile 0.5	Quantile 0.75	Quantile 0.9
MAE	6.2693	5.2484	4.3637	3.9445	4.8255
MSE	54.6029	39.5698	27.3035	24.1575	35.6935
MAPE	0.3189	0.2762	0.2518	0.2583	0.3302
RMSE	7.3894	6.2905	5.2253	4.915	5.9744

The training time for all algorithm of regressions can be seen at the Table 10.

Table 10
Algorithms' training time

Algorithm	Time, s
K-nearest neighbors	12.5098
Random Forest	43.4547
Gradient Boosting	13.4066
XGBoost	11.4844

6. Discussions

Before discussion of results, it is worth mentioning some of the assumptions and limitations of this work. Assumptions for this study may include the following:

- The study was conducted on data taken from the Uber Movement service, which contains data on the average speed of taxi traffic on a certain road segment for each hour of each day of a particular month.
- The study was conducted using machine learning methods, in particular quantile regression methods such as KNN, Random Forest, Gradient Boosting, XGBoost, and others.
- The study used qualitative metrics such as MSE, RMSE, MAE, MAPE to clearly assess the effectiveness of the regression methods under consideration.

Limitations include the following points:

- The study was conducted on a limited data set, which may affect the overall representativeness of the study.
- For the regression methods under consideration, additional parameters and hyperparameters may be required and need to be optimized to obtain better forecasting results.
- Some factors affecting traffic may be difficult to measure or unavailable for data collection (unpredictable changes in traffic, e.g. due to accidents or weather conditions), which may result in insufficient accuracy of regression models.

The training time of the model is an important parameter, as it was established by the results of experiments that the fastest algorithms for this task are XGBoost and KNN.

We observed that KNN and Random Forest algorithm performed relatively well for lower quantiles, but its performance degraded for higher quantiles (Figure 8). The table 2 provides actual values and predicted values for 5 quantiles for the KNN model. It appears that the predicted values are generally higher than the actual values, and the difference between the predicted and actual values increases as the quantile level increases.

They work well for lower quantiles, then they are able to model complex nonlinear data and fill large amounts of training data. However, for higher quantiles, when the data are smaller and the values for the data cut-offs, these methods may be less efficient. For such tradeoffs, there may be better methods that require more complex models and make fewer assumptions about the distribution of the data, such as gradient boosting.

Gradient Boosting and XGBoost perfomed bad enough for lower quantiles and the smallest loss near the median – 0.5 quantile. The MAE, MSE, and RMSE decrease as the quantile increases, indicating that the model is performing better at higher quantiles. However, the MAPE increases as the quantile increases, indicating that the relative error of the model is higher at higher quantiles. Because methods are based on sequentially adding weak models to the ensemble in order to improve predictive abilities. They are commonly used to reduce MSE of the prediction, which is a standard metric in many regression problems. However, in quantile regression, where the target variables are quantiles, MSE may not be a suitable predictor.

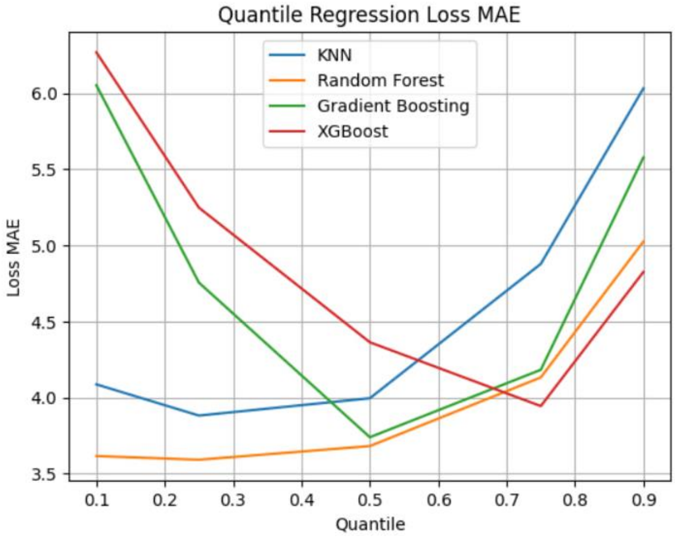


Figure 8: Comparison Quantile Regression Loss MAE

Gradient Boosting and XGBoost methods usually show good results for the median, as this metric is quite close to the MSE. However, for lower quantiles where predictions should be more conservative, these methods may be less effective. This may be due to the fact that gradient boosting and XGBoost methods use tree models, which usually tend to overfitting, that is, they can remove significant interactions between variables that are absent in the training data set. This may lead to less accurate predictions for lower quantiles, where the distribution of the data may be more complex and interactions between variables may be more important.

As we can see from the results, KNN and Random Forest show not bad results in accuracy but have narrow predicted interval and don’t cover all possible values.

KNN method shows a narrow interval in the quantile regression because it uses the most similar values from the training data set to predict the target variable. If the training data set is representative of the target variable, then the nearest neighbor method can provide reasonably accurate results for quantile regression. However, it may show less accurate results if the training data set is not representative of the target variable, as may be the case when the speed of cars depends on many other factors, such as weather conditions, traffic, day of the week, etc.

Random Forest method can also show a narrow interval in the quantile regression because this method is based on an ensemble of decision trees, which can be very flexible in modeling non-linear relationships between the dependent and independent variables. In addition, with the use of many trees in the ensemble, high prediction accuracy can be achieved. However, if the decision trees are too deep or the number of trees is too large, overtraining of the model may occur and its overall generalization ability may deteriorate. Therefore, it is important to carefully tune model hyperparameters such as tree depth and number of trees in the ensemble.

In quantile regression, a larger value of MSE indicates that the model has a higher dispersion of errors around the predicted quantile values. In other words, the model may be overestimating or underestimating the actual quantile values by a larger margin.

Since quantile regression is concerned with predicting specific quantiles of the target variable, a model with a larger MSE may be more appropriate if the goal is to identify extreme or outlier values of the target variable. This is because a larger MSE implies that the model is better able to capture the variability in the tails of the target variable's distribution. This is because quantile regression is concerned with modeling the entire conditional distribution of the response variable, rather than just its mean.

7. Conclusions

In this work, we explored the use of different quantile regression models for predicting speed based on Uber data in Kyiv, Ukraine during January 2020. We compared the performance of KNN quantile regression, Random Forest quantile regression, Gradient Boosting quantile regression, and XGBoost quantile regression, measuring errors and draw plots for each model.

Our results show that all four models performed well in predicting speed. We got knowledge that KNN and Random Forest algorithms work relatively well for lower quantiles but their effectiveness declines for higher quantiles. Gradient Boosting and XGBoost methods showed poor results for lower quantiles and the smallest losses near the median. KNN and Random Forest methods have a narrow prediction interval and do not cover all possible values. However, the Gradient Boosting quantile regression model appeared to perform the best, with the lowest overall mean absolute error and mean squared error.

Traditionally, mean regression models have been used for this purpose, but quantile regression provides a more comprehensive approach as it allows for the prediction of multiple quantiles, providing a fuller picture of the traffic flow distribution.

The results of this work can help to identify the most effective methods of traffic forecasting, which can reduce the time spent on forecasting and increase the accuracy of forecasts. In addition, the conclusions of this work can be used to develop new traffic forecasting algorithms that will be more efficient and accurate. The study shows that the proposed quantile regression models (KNN, random forest, gradient boosting, and XGBoost) outperform the traditional linear regression model in traffic flow prediction.

In the future, we plan to conduct research and compare the effectiveness of other machine learning methods that can be applied to traffic forecasting, such as neural networks. We need to consider the possibility of using different factors. For example, we can add data on weather, events in the city, road works and accidents, which will help identify key stress points in cities. We also want to expand our model to include last year or even previous years, hoping to identify seasonal patterns in urban mobility. In addition, it will be important to test the effectiveness of the developed models on real data and compare them with existing traffic forecasting systems to assess their potential usefulness and practical relevance.

8. References

[1] Khaled Shaaban, Mazen Elamin, Mohammed Alsoub, Intelligent Transportation Systems in a Developing Country: Benefits and Challenges of Implementation, Transportation Research Procedia, vol. 55, 2021, pp. 1373-1380, doi: 10.1016/j.trpro.2021.07.122.

[2] Kiev traffic index. URL: https://www.tomtom.com/en_gb/traffic-index/kiev-traffic.

[3] Daunoras J., Bagdonas V., Gargasas V. City transport monitoring and routes optimal management system. Transport. 2008. 23(2). p. 144-149.

[4] Zewei Zhou, Ziru Yang, Yuanjian Zhang, Yanjun Huang, Hong Chen, Zhuoping Yu, A comprehensive study of speed prediction in transportation system: From vehicle to traffic, iScience, vol. 25, Issue 3, 18 March 2022, doi: 10.1016/j.isci.2022.103909.

[5] L. Pun, P. Zhao and X. Liu, "A Multiple Regression Approach for Traffic Flow Estimation," in IEEE Access, vol. 7, pp. 35998-36009, 2019, doi: 10.1109/ACCESS.2019.2904645.

[6] G. Dai, C. Ma and X. Xu, "Short-Term Traffic Flow Prediction Method for Urban Road Sections Based on Space–Time Analysis and GRU," in IEEE Access, vol. 7, pp. 143025-143035, 2019, doi: 10.1109/ACCESS.2019.2941280.

[7] Y. Hou, P. Edara and C. Sun, "Traffic Flow Forecasting for Urban Work Zones," in IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 4, pp. 1761-1770, Aug. 2015, doi: 10.1109/TITS.2014.2371993.

[8] S. Bielievtsov, I. Ruban, K. Smelyakov and D. Sumtsov, “Network technology for transmission of visual information”, Selected Papers of the XVIII International Scientific and Practical Conference “Information Technologies and Security” (ITS 2018), Kyiv, Ukraine, November 27, 2018. In CEUR Workshop Proceedings, Vol-2318, 2018, pp. 160-175. https://ceur-ws.org/Vol-2318/.

[9] K. Smelyakov, P. Dmitry, M. Vitalii and C. Anastasiya, "Investigation of network infrastructure control parameters for effective intellectual analysis," 2018 14th International Conference on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Engineering (TCSET), Lviv-Slavske, Ukraine, 2018, pp. 983-986, doi: 10.1109/TCSET.2018.8336359.

[10] O. Lemeshko, M. Yevdokymenko, O. Yeremenko, A. M. Hailan, P. Segeč and J. Papán, "Design of the Fast ReRoute QoS Protection Scheme for Bandwidth and Probability of Packet Loss in Software-Defined WAN," 2019 IEEE 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM), Polyana, Ukraine, 2019, pp. 1-5, doi: 10.1109/CADSM.2019.8779321.

[11] Sharonova, N., Kyrychenko, I., Tereshchenko, G., “Application of big data methods in E-learning systems”, 2021 5th International Conference on Computational Linguistics and Intelligent Systems (COLINS-2021), 2021. – CEUR-WS, 2021, ISSN 16130073. - Volume 2870, PP. 1302-1311.

[12] Gruzdo, I., Kyrychenko, I., Tereshchenko, G., Shanidze, N., "Metrics applicable for evaluating software at the design stage," 2021 5th International Conference on Computational Linguistics and Intelligent Systems (COLINS-2021), 2021. – CEUR-WS, 2021, ISSN 16130073. - Volume 2870, PP. 916-936.

[13] Ken Chen, Shasha Zhao and Dengyin Zhang, Short-term Traffic Flow Prediction based on Data-Driven Knearest neighbour Nonparametric Regression, Journal of Physics: Conference Series, vol. 1213, Issue 5, 2019, doi: 10.1088/1742-6596/1213/5/052070

[14] Lun Zhang, Qiuchen Liu, Wenchen Yang, Nai Wei, Decun Dong, An Improved *K*-nearest Neighbor Model for Short-term Traffic Flow Prediction, Procedia - Social and Behavioral Sciences, vol. 96, 6 November 2013, pp. 653-662 doi: 10.1016/j.sbspro.2013.08.076.

[15] P. Dell'Acqua, F. Bellotti, R. Berta and A. De Gloria, "Time-Aware Multivariate Nearest Neighbor Regression Methods for Traffic Flow Prediction," in IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 6, pp. 3393-3402, Dec. 2015, doi: 10.1109/TITS.2015.2453116.

[16] Deekshetha, H.R., Shreyas Madhav, A.V., Tyagi, A.K. (2022). Traffic Prediction Using Machine Learning. In: Suma, V., Fernando, X., Du, KL., Wang, H. (eds) Evolutionary Computing and Mobile Sustainable Networks. Lecture Notes on Data Engineering and Communications Technologies, vol 116. Springer, Singapore, doi: 10.1007/978-981-16-9605-3_68.

[17] Amini, M. R., Feng, Y., Yang, Z., Kolmanovsky, I., & Sun, J. (2020). Long-Term Vehicle Speed Prediction via Historical Traffic Data Analysis for Improved Energy Efficiency of Connected Electric Vehicles. Transportation Research Record, 2674(11), 17–29, doi: 10.1177/0361198120941508

[18] Zahid M, Chen Y, Jamal A, Mamadou CZ. Freeway Short-Term Travel Speed Prediction Based on Data Collection Time-Horizons: A Fast Forest Quantile Regression Approach. Sustainability. 2020; 12(2):646, doi: 10.3390/su12020646.

[19] Faires, Jackson, "Prediction Intervals: The Effects and Identification of Sparse Regions for Nonparametric Regression Methods" (2021). Electronic Theses and Dissertations. 406. URL: https://scholarworks.sfasu.edu/cgi/viewcontent.cgi?article=1431&context=etds.

[20] Liu B, Zhang T, Hu W. Intelligent Traffic Flow Prediction and Analysis Based on Internet of Things and Big Data. Comput Intell Neurosci. 2022 Jun, doi: 10.1155/2022/6420799.

[21] Y. Yu, X. Han, M. Yang and J. Yang, "Probabilistic Prediction of Regional Wind Power Based on Spatiotemporal Quantile Regression," in IEEE Transactions on Industry Applications, vol. 56, no. 6, pp. 6117-6127, Nov.-Dec. 2020, doi: 10.1109/TIA.2020.2992945.

[22]F. Rodrigues and F. C. Pereira, "Beyond Expectation: Deep Joint Mean and Quantile Regression for Spatiotemporal Problems," in IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 12, pp. 5377-5389, Dec. 2020, doi: 10.1109/TNNLS.2020.2966745.

[23]Q. Meng, M. Mourshed and S. Wei, "Going Beyond the Mean: Distributional Degree-Day Base Temperatures for Building Energy Analytics Using Change Point Quantile Regression," in IEEE Access, vol. 6, pp. 39532-39540, 2018, doi: 10.1109/ACCESS.2018.2852478.

[24]K. Smelyakov, A. Chupryna, D. Sandrkin and M. Kolisnyk, "Search by Image Engine for Big Data Warehouse," 2020 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream), Vilnius, Lithuania, 2020, pp. 1-4, doi: 10.1109/eStream50540.2020.9108782.

[25]H. Ruan, B. Wu, B. Li, Z. Chen and W. Yun, "Expressway Exit Station Short-Term Traffic Flow Prediction With Split Traffic Flows According Originating Entry Stations," in IEEE Access, vol. 9, pp. 86285-86299, 2021, doi: 10.1109/ACCESS.2021.3087658.

[26]Gao, C. Zhou, J. Rong, Y. Wang and S. Liu, "Short-Term Traffic Speed Forecasting Using a Deep Learning Method Based on Multitemporal Traffic Flow Volume," in IEEE Access, vol. 10, pp. 82384-82395, 2022, doi: 10.1109/ACCESS.2022.3195353.

[27]Yue Hou, Jiaxing Chen, Sheng Wen, The effect of the dataset on evaluating urban traffic prediction, Alexandria Engineering Journal, vol. 60, Issue 1, February 2021, pp. 597-613, doi: 10.1016/j.aej.2020.09.038.

[28]Uber Movement. URL: https://movement.uber.com/?lang=en-US.

[29]Koenker, R. (2005). Quantile Regression (Econometric Society Monographs). Cambridge: Cambridge University Press. doi:10.1017/CBO9780511754098.

[30]Julien Salotti, Serge Fenet, Romain Billot, Nour-Eddin El Faouzi, Christine Solnon. Comparison of traffic forecasting methods in urban and suburban context. ICTAI 2018: IEEE 30th International Conference on Tools with Artificial Intelligence, Nov 2018, Volos, Greece. pp.846-853, doi:10.1109/IC-TAI.2018.00132.

[31]X. Yang, Y. Yuan and Z. Liu, "Short-Term Traffic Speed Prediction of Urban Road With Multi-Source Data," in IEEE Access, vol. 8, pp. 87541-87551, 2020, doi: 10.1109/ACCESS.2020.2992507.

[32]Yuan, H., Li, G. A Survey of Traffic Prediction: from Spatio-Temporal Data to Intelligent Transportation. Data Sci. Eng. 6, 63–85 (2021), doi: 10.1007/s41019-020-00151-z.

[33]O. Saidani, L. J. Menzli, A. Ksibi, N. Alturki and A. S. Alluhaidan, "Predicting Student Employability Through the Internship Context Using Gradient Boosting Models," in IEEE Access, vol. 10, pp. 46472-46489, 2022, doi: 10.1109/ACCESS.2022.3170421.

[34]N. Khan, F. U. M. Ullah, Afnan, A. Ullah, M. Y. Lee and S. W. Baik, "Batteries State of Health Estimation via Efficient Neural Networks With Multiple Channel Charging Profiles," in IEEE Access, vol. 9, pp. 7797-7813, 2021, doi: 10.1109/ACCESS.2020.3047732.

[35]Kegler. URL: https://kepler.gl/demo.