International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Traffic Prediction Based on Air Quality in IoT-Based Smart City Using Regression and Ensemble Techniques: Bagging and Stacking

Praveena Kumari M. K.[1], Manjaiah D. H.[2], Ashwini K. M.*[3]

Submitted:10/03/2024 **Revised**: 25/04/2024 **Accepted**: 02/05/2024

Abstract: Traffic forecast implies determining the volume and thickness of the traffic stream, typically for the reason of controlling vehicle development, decreasing congestion, and producing the ideal routes with the least amount of time or energy consumed. Accurate street traffic flow determination is among the foremost essential factors in smart cities. In this research, we utilized air quality data and ensemble regression methods to establish a predictive model for traffic patterns, recognizing the correlation between air pollution levels and congested traffic conditions. This study was conducted in two distinct stages. In the first phase, we compared the performance of 10 different regression models (Decision Tree, KNN, Cat Boost, Linear Regression, Lasso, Elastic Net, Kernel Ridge, Gradient Boost, XGB, and LGBM), and K-Nearest Neighbour gave the best result with RMSE 2.80 and Lasso gave the least performance with 5.28 RMSE. In the second phase, we developed models based on ensemble techniques: bagging and stacking. Depending on the performance of the regressors in the first phase, we attempted numerous permutations of distinctive models in bagging and stacking till we got the most excellent conceivable results. Finally, out of many arrangements, the Stacking Model with CatBoost, KNN, and Decision Tree as base learners and Lasso as meta learner performed better than KNN and Bagging Ensemble Regression models with RMSE 2.09.

***Keywords**: Regression, Air Quality, Ensemble, Stacking, Bagging, Smart City*

1. Introduction

As economic development accelerates and population density rises, the vehicle fleet has expanded significantly, resulting in a notable increase in air pollution levels and having a substantial impact on the quality of life for citizens. A primary concern arises from traffic congestion, particularly stemming from the rise in the volume of motor vehicles on the roads. The rate at which vehicles increase, the road is built at the same rate, which creates a significantly expanded congestion rate[1]. Traffic blockage encompasses a negative effect on traffic execution since it increments destination reaching time and air contamination. Subsequently identifying traffic blockage may be a key component in encouraging the advancement of productive Intelligent Transport Systems. The presence of a large number of vehicles within urban areas underscores the significance of traffic-related issues for the effective functioning of the city and the well-being of its inhabitants. Vehicle-generated air pollution compounds the challenges of urban congestion, contributing to elevated illness rates in densely populated metropolitan areas. Urban residents often face heightened risks due to the concentration of both mobile and stationary sources of air pollution (such as traffic flow, industrial activities, and energy generation) in and around cities.

[1] *Dept of Master of Computer Applications, NMAMIT Nitte (Deemed to be University), Nitte, India*
[2] *Department of Computer Science Mangalore University Mangalore, India*
[3] *Dept of Master of Computer Applications, NMAMIT Nitte (Deemed to be University), Nitte, India*
**Corresponding Author Email: ashwini.bhandary@nitte.edu.in*

Urbanization worldwide is swiftly advancing, with estimates suggesting that around two-thirds of the global population will inhabit urban regions by 2050 [2]. Traffic emissions are recognized as the primary contributors to air pollutants in various regions, encompassing substances like carbon monoxide, carbon dioxide, volatile organic compounds, hydrocarbons, nitrogen oxides, and particulate matter. The most recent figures give the impression that fine particulate matter or PM2.5, is accountable for almost four million impermanence around the world caused by cardiorespiratory circumstances like lung diseases, preterm births, and other disorders [3]. Enabling drivers to choose the most efficient route or adjust their departure times to avoid traffic congestion can significantly reduce their chances of getting stuck in traffic. Numerous studies have demonstrated the utility of road congestion data in predicting atmospheric pollutant levels.

Kumar K et al[4] examined six years of air quality data from 23 cities in India to evaluate and forecast air quality trends. Bekkar et al[5] developed various deep-learning models to predict the concentration of particulate matter (PM2.5) using air quality data collected from twelve locations by the Beijing Municipal Environmental Monitoring Centre. Their analyses did not incorporate traffic volume. Offering travelers information about the most efficient route to their destination improves their overall travel experience. Effective traffic management is integral to the concept of smart cities. The escalation of traffic congestion correlates with heightened air pollution levels, adversely affecting the sustainability of communities. Implementing intelligent congestion management schemes enables drivers to

circumvent congested routes, thereby mitigating air pollutant levels. Accurately predicting traffic congestion propagation is challenging due to the dynamic nature of road infrastructure. Intelligent transportation systems are crucial components of urban environments and are pertinent subjects within this domain. Our study findings indicate that atmospheric pollutants significantly influence traffic forecasting. Integrating pollution levels into traffic forecasting enhances its reliability. Prior research consistently utilized traffic data to forecast air quality, extrapolating pollutant proportions based on vehicular traffic volume.

To date, only a minority of researchers have undertaken studies aimed at refining traffic forecasting techniques, and among those who have, the majority have neglected to consider air pollution. Additionally, they have predominantly relied on conventional statistical models. This deficiency presents a significant challenge in incorporating air pollution into traffic forecasting. Given that the volume of traffic is the primary contributor to pollution, it is logical to utilize pollution levels as a proxy for estimating the number of vehicles on the road.

This study aims to assess the feasibility and effectiveness of a methodology that incorporates both air pollutants and traffic density to generate reliable outcomes. Moreover, if the findings are favorable, the model could lead to a reduction in the necessity for traffic sensors, potentially lowering maintenance costs. This shift from targeted sensing and monitoring to comprehensive sensing could enhance infrastructure management in expansive urban areas. This implies that instead of relying solely on sensors for traffic prediction, the model could utilize air quality data alone for this purpose.

We have developed a novel approach for forecasting traffic volume using air pollution data. This involved constructing six ensemble regression models, each employing distinct regression techniques. Further details regarding the methodology of this innovative approach are elaborated in Section III. We evaluated the effectiveness of the regression models, bagging ensemble models, and stacking ensemble models using four evaluation criteria: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared. This was done to determine the most accurate statistical model for estimating traffic intensity.

In Section 2, a summary of relevant literature is provided. Section 3 elaborates on the architecture, methodology, and regression models utilized. Section 4 thoroughly examines the findings. Lastly, Section 5 presents the concluding remarks.

2 Literature Review

In this section, we briefly review the chosen methodologies introduced earlier in traffic prediction. Awan FM et al(2020)[6] have utilized datasets collected from open datasets in Madrid, Spain to demonstrate that the LSTM (Long-Term Memory Recurrent Neural Network) is more effective in predicting traffic flow when combined with time series traffic flow data, air pollution data (CO, NO, NO2, NOx, and O3) and atmospheric data, than when only using traffic flow data to forecast traffic. However, the air pollution data did not include PM2.5 data. Braz F J et al(2022)[7] looked at how to predict traffic on motorways based on weather data. They tested a bunch of different prediction methods, including CNN, which had the best results in terms of both error and execution time. They also looked at how long it took to learn and predict traffic at 10-minute intervals. The influence of air quality is not taken into account by them in their experiment. Artificial neural networks are used by Loumiotis I et al(2018)[8] to figure out how fast cars are going on the road as a sign of how busy the road is. They have not used any air quality data, resulting in a higher mean absolute error value. Singh S et al(2023)[9] came up with a new way to figure out how busy a road is based on distance, speed, and time interval using the K-means Algorithm (The effect of air pollution was not considered). In these crowd-sourced locations, with the aid of Bayesian Classifiers, they have estimated air quality using eight common air quality parameters. Neelakandan et al(2021)[10] introduced an efficient OWENN algorithm to forecast traffic efficiently. This model is compared with the existing ENN method, CNN method, NN method, and ANFIS method. The proposed one provides an average MAE value of 0.254 and an RMSE value of 0.345. We think that the error rate can be reduced by considering the impact of air pollution. Shepelev V et al(2023)[11] proposed a hybrid model that integrates convolutional neural networks with recurrent deep learning networks to enhance the precision of traffic intensity predictions for the purpose of calculating pollutant emissions associated with vehicles. The proposed model was found to accurately predict vehicle fleet size and significantly outperform competing models in predicting accuracy. Almeida et al(2022) [12] investigated how to understand and predict city traffic patterns using both statistical models and deep learning. They used algorithms like SARIMA and neural networks like Feed Forward Networks, LSTMs, and Hybrid LSTMs to do this. They found that statistical models were much better than neural networks at predicting traffic counter data over time, even when they noticed unusual traffic. They did not use air pollution data in their study. A hybrid model for the prediction of road movement was proposed by Tang et al(2019)[13] which includes noise mitigation methods as well as support vector machines (SVM). They have simply applied 3 characteristics to their experiment: Volume, speed, and occupancy. However, due to the lack of data on air quality, the error rate recurs significantly. To Analyze Traffic Flows in UK Zhuang W et al(2023)[14] used numerous machine learning algorithms, including SVR, LSTM, GRU, KNN-LSTM, and CNN-

LSTM models, along with real-time recordings of time, flow, and speed from all detection points on all motorways in the city. However, the proposed K-Nearest Neighbor-Bidirectional Long Short-Time Memory model has a better prediction. The experiment did not take into account air pollution data. Using a long-term memory network, Majumdar et al(2021)[15] predicted how long it would take for traffic to spread out on the roads. They used data from an Internet of Things (IoT) device that had a speed sensor in Buxton, UK. They looked at things like speed, how long it took, how fast it was moving, and headway. But they did not take into account air pollution factors, which caused the error value to be high.

Fernandes et al. (2020) [16] performed a comprehensive empirical evaluation of multiple suburban roundabouts, analysing their effectiveness in managing traffic flow as well as their impact on air and noise pollution levels. However, their model is site-specific, limiting its applicability to roundabouts with similar traffic conditions. Luo X et al. (2019) [17] proposed a traffic flow prediction method that combines KNN and LSTM. The KNN model selects spatial stations, and the data from these stations are then inputted into LSTM for prediction. Song Xiang et al. (2018) [18] devised a model for daily traffic flow prediction using group data processing and time series prediction. They segmented data into three groups based on factors such as days of the week, weekends, holidays, and seasons, and used this group data as input for the time forecasting model, yielding promising results. Ma et al. (2021) [19] processed data into 288 sampling intervals per day to predict daily traffic flows, aggregating them into a matrix periodically. The data for each period was fed into a CNN model, which extracted spatial features. Finally, the data was fed into a linear time-stamping machine (LSTM) model for fusion across the entire connection layer, resulting in an accuracy of more than 90%. Qu et al. (2018) [20] introduced a traffic flow prediction method that explores spatial relationships within context using a supervised learning algorithm, followed by training a deep neural network (DNN) with the obtained data. This approach surpasses traditional traffic prediction methods in accuracy. Ma et al. (2021) [21] utilized a genetic algorithm to categorize input context factors for traffic flow forecasting, converting their significance into weights. Historical data sets were selected as inputs for prediction algorithms based on the similarity of these weights, consistently yielding reliable prediction outcomes.

Several of the aforementioned models are designed for predicting time series data, while others are deep mixture models that incorporate both spatial and temporal aspects of traffic flow, recognizing their significance in road traffic forecasting. However, the majority of these models have overlooked the influence of air quality. Consequently, an increasing number of researchers have focused on investigating the effects of air pollution on road traffic prediction to achieve optimal forecasting conditions.

3 Areas of Study and Methods

The experiments utilized open data sourced from the Data Streams of the Year 2014 provided by the City of Denmark, Aarhus. The city administration has deployed 449 sensor pairs across its main roads, which record the vehicle count every five minutes. The air pollution dataset comprises measurements of pollutants such as carbon monoxide, nitrogen dioxide, sulfur dioxide, particulate matter, and ozone emitted into the air by vehicles.

Initially, the air pollution datasets and traffic datasets were combined using timestamps. For the analysis, only vehicle intensity is retained from the traffic data, and all the features from the pollution data are used. The experiments were split into two sub-tests: one to compare different regression techniques, and two to analyze various proposed bagging and stacking ensemble techniques. Various optimization techniques were employed for the model, which included feature engineering, feature transformation, standardization, and hyperparameter tuning. The dataset is standardized using StandardSclaer and GridSearchCrossValidation is used to find the best hyperparameter combinations to improve the model performance. Figure 1 illustrates the location of the data source. Figure 2 is the suggested framework Regression Ensemble Model.

Fig 1: Location of the Data source

3. 1 Regression Methods

3.1.1 Decision Tree Regression: Using training data, decision trees, also known as classification and regression trees, or CARTs, determine the best points to divide the data to reduce the cost metric. The mean squared error is the default cost metric for regression decision trees.

3.1.2 KNN: The K-Nearest Neighbors (KNN) algorithm identifies the k most similar data instances from the training dataset for a new data point. Then, it derives a prediction by taking the mean or median output variable from these k neighbors. It is important to consider the distance metric employed, with the default being the Minkowski distance. This metric is a broader form that encompasses both the Euclidean distance (suitable when all inputs share the same scale) and the Manhattan distance (applicable when input variable scales differ).

3.1.3 Catboost Regression: CatBoost Regression is a specialized machine learning method crafted to manage

categorical features within regression scenarios. It employs gradient boosting on decision trees to forecast continuous target variables. A standout characteristic of CatBoost is its capability to directly handle categorical variables, eliminating the need for preliminary encoding. Additionally, it applies a range of regularization techniques to mitigate overfitting and enhance overall predictive accuracy.

3.1.4 Linear Regression: The foundation of linear regression is the idea that the distribution of the input variables is Gaussian. Additionally, it is assumed that while there is some correlation between the input and output variables, the correlation is not very strong.

3.1.5 Lasso Regression: Lasso regression, referred to as L1 regularization, is a linear regression method that integrates a penalty term proportional to the absolute value of coefficient magnitudes. This penalty fosters sparsity within the model, causing certain coefficients to be reduced to zero, thereby facilitating feature selection. Lasso regression proves beneficial in managing datasets with numerous dimensions, aiding in the identification of pivotal features for prediction while diminishing the influence of less significant coefficients.

3.1.6 Elasticnet Regression: Elasticnet regression merges the regularization principles of Lasso(L1) and Ridge(L2) regression, achieving a compromise between feature selection and coefficient shrinkage by utilizing both penalty types. This method adeptly tackles the shortcomings of Lasso and Ridge regression, providing enhanced model adaptability and predictive accuracy, especially in datasets with highly correlated features.

3.1.7 Kernel Ridge Regression: Kernel Ridge Regression, a derivation of Ridge Regression, employs the kernel trick to manage nonlinear associations between features and the target variable. By transforming the input space into a higher-dimensional feature space, Kernel Ridge Regression can effectively grasp intricate data patterns. This approach proves especially potent in handling nonlinear relationships, often yielding superior predictive accuracy over conventional linear models.

3.1.8 Gradient Boost Regression: Gradient Boosting Regression is a machine learning method that iteratively combines weak regression models, such as decision trees, to create a strong predictive model. It sequentially trains new models on the residual errors of the previous ones, to reduce overall prediction error. This iterative approach steadily enhances model accuracy and effectiveness, rendering gradient-boosting regression a potent asset for predictive modeling endeavors.

3.1.9 XGB: XGB regression, an adaptation of gradient boosting, employs the XGBoost algorithm for predictive modeling purposes. By sequentially integrating weak regression models, like decision trees, it constructs a sturdy predictive framework. Employing iterative training on residual errors, XGB regression reduces prediction errors and improves model accuracy, rendering it an asset for regression analysis.

3.1.10 LGBM: LGBM regression, a form of gradient boosting, utilizes the LightGBM algorithm for regression applications. It adopts a distinctive approach to tree-based learning, growing tree leaf-wise to enhance computational efficiency and model effectiveness. Renowned for its capacity to manage extensive datasets and intricate relationships, LGBM regression proves instrumental in achieving precise regression analyses across diverse fields.

3.2 Data Collection

In our study, we utilized extensive real-time Internet of Things (IoT) data sourced from the publicly accessible City Pulse Aarhus dataset. This dataset consists of two distinct sets: one dedicated to pollution levels and the other to traffic flow. Aarhus City maintains numerous sensors that monitor vehicle counts every five minutes, while the air pollution dataset provides details on various emissions from vehicles, including carbon monoxide, nitrogen dioxide, sulfur dioxide, particulate matter, and ozone.

3.3 Data Processing

In our research, we acquired two separate datasets—one dedicated to pollution levels and the other to traffic flow—at a particular location and time. As both datasets originated from the same location and time, we amalgamated them to form an integrated dataset for forecasting traffic based on pollution levels in that vicinity. However, the obtained dataset necessitated normalization and standardization procedures, which we conducted using the MinMax Scaler and Normalization techniques. Furthermore, we handled missing data by utilizing the mean method: identifying columns with absent values, calculating the mean for those columns, and replacing the missing values with the computed mean.

3.4 Developing Ensemble Regression Models

3.4.1 Bagging

We employed the "BaggingRegressor" from the scikit-learn library to train the base regressors. Two bagging ensemble models were introduced in our approach. Our ensemble model combines the predictions of the base regressors by averaging them to generate a final prediction. We utilized 10 base estimators in the ensemble. Additionally, we developed a method to evaluate the performance of each base estimator on the test data and obtain their corresponding prediction values. The predictions generated by each base estimator are stored in individual columns, and the final prediction for Mean Squared Error, Mean Absolute Error, Root Mean Square Error (RMSE) and RSquare calculations are computed by averaging all the respective predictions. Below are the base regressors utilized in our proposed bagging ensemble models:

1) Bagging Ensemble Model 1: Linear Regression, Lasso, Elastic Net and Kernel Ridge
2) Bagging Ensemble Model 2: Gradient Boosting, Catboost, XGB, and KNN

Among these two models, the Bagging ensemble model2 demonstrates the most favorable outcome. Consequently, Figure 2 illustrates Gradient Boosting, Catboost, XGB, and KNN as base regressors.

3.4.2 Stacking

We utilized the Stacking Regressor module from the scikit-learn library for our regression task. We experimented with various combinations of different models in stacking to achieve the best possible outcomes. Among the ten regression models tested, "KNN" emerged as the top performer. Consequently, we employed it as the meta learner, while utilizing other algorithms as base estimators in Stacking Ensemble Model 1 and Stacking Ensemble Model 2. Conversely, "Lasso" performed the least effectively. Therefore, we designated Lasso as the meta learner and employed other algorithms as base estimators in Stacking Ensemble Model 3 and Stacking Ensemble Model 4. In summary, we proposed four Stacking Ensemble Models which are:

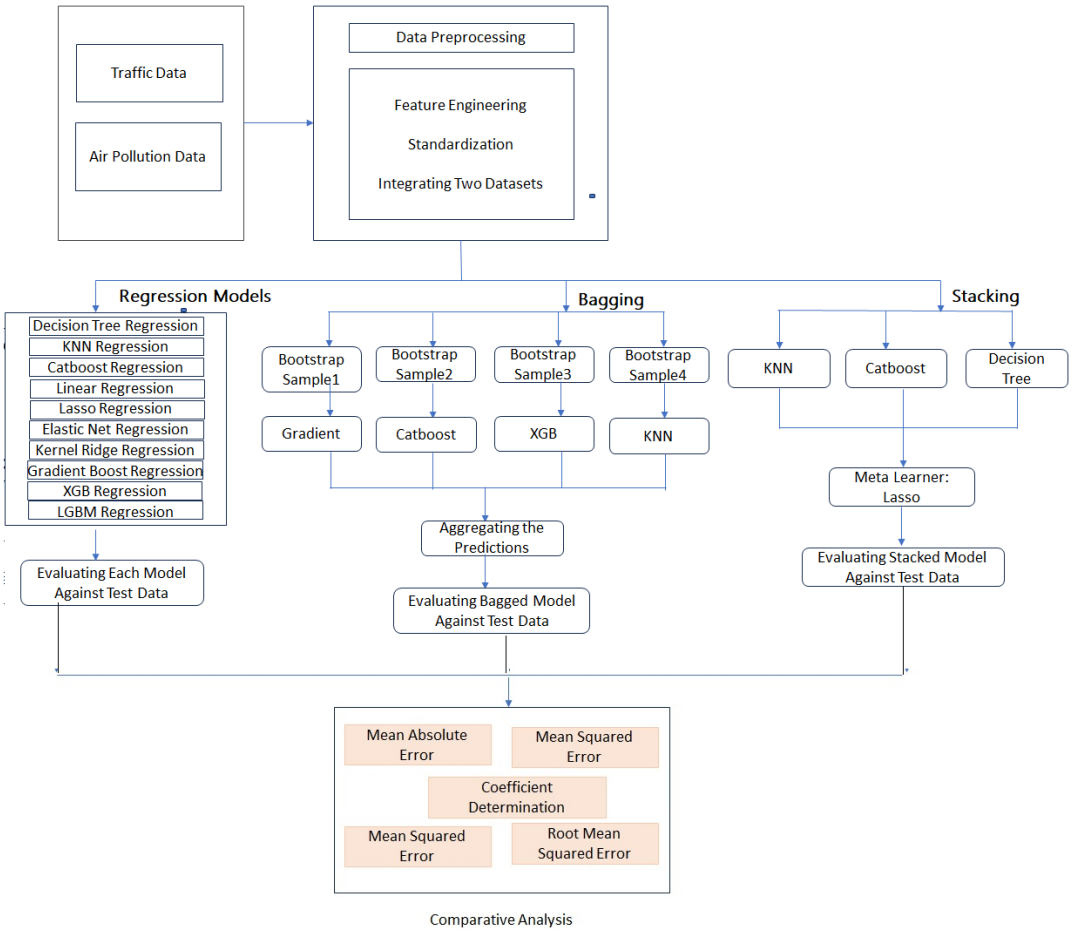


Fig 2: Suggested framework – Regression Ensemble Model

1) Stacking Ensemble Model1: KNN meta learner, Catboost Regressor, Gradient Boosting Regressor, and Decision Tree Regressor as base estimators.
2) Stacking Ensemble Model2: KNN meta learner, XGB, Catboost and Decision Tree as base estimators.
3) Stacking Ensemble Model3: Lasso meta learner, KNN, Catboost and Decision Tree as base estimators.
4) Stacking Ensemble Model4: Lasso meta learner, KNN Catboost, Gradient Boosting Regressor as base estimators.

Among these four models, the Stacking Ensemble Model3 with CatBoost, KNN, and Decision Tree as base learners and Lasso as meta learner yields the most favorable outcome than KNN and Bagging models. Consequently, in Figure 2, for stacking CatBoost, KNN, Decision Tree, and Lasso are depicted as the machine learning algorithms utilized.

Out of the four models considered, Stacking Ensemble Model3, incorporating CatBoost, KNN, and Decision Tree as base learners with Lasso as the meta learner, demonstrates superior performance compared to both the KNN and Bagging models. Therefore in Figure 2, the machine learning algorithms utilized for stacking include CatBoost, KNN, Decision Tree, and Lasso.

4 Assessment of Experiments

In our experiment, we exclusively incorporated the vehicle count data and synchronized it with the pollution dataset according to the timestamp. We opted for this approach because the pollution and traffic data sensors were located in the same area. Furthermore, there exists a direct correlation between the number of vehicles and air pollution levels; as traffic density increases, so do concentrations of carbon monoxide, nitrogen dioxide, sulfur dioxide, particulate matter, and ozone.

The motive behind utilizing air pollution data for traffic forecasting is to decrease the reliance on traffic sensors, thereby cutting down maintenance costs and enabling the development of a more expansive environmental monitoring infrastructure. This shift involves moving away from specific sensing and monitoring towards broader coverage across large urban areas. As a result, instead of depending on traffic sensors, the model can forecast based solely on air pollution data.

Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R-Square (R2) are utilized to assess the effectiveness of the proposed model.

4.1 Regression Models Comparison

This comparative analysis aims to identify the optimal prediction model for traffic forecasting. Table 1 presents an overview of all evaluation metrics for each regression model, while Figure 3(a) specifically displays the RMSE values of these models. RMSE serves as a conventional method for quantifying a model's forecasting error when predicting numerical outcomes. It is widely recognized as an excellent general-purpose metric for predictive forecasts. A smaller RMSE value indicates a closer fit to the data. In this context, KNN exhibits the lowest RMSE value of all models, with a value of 2.80 (as shown in Table 1), indicating its superior performance.

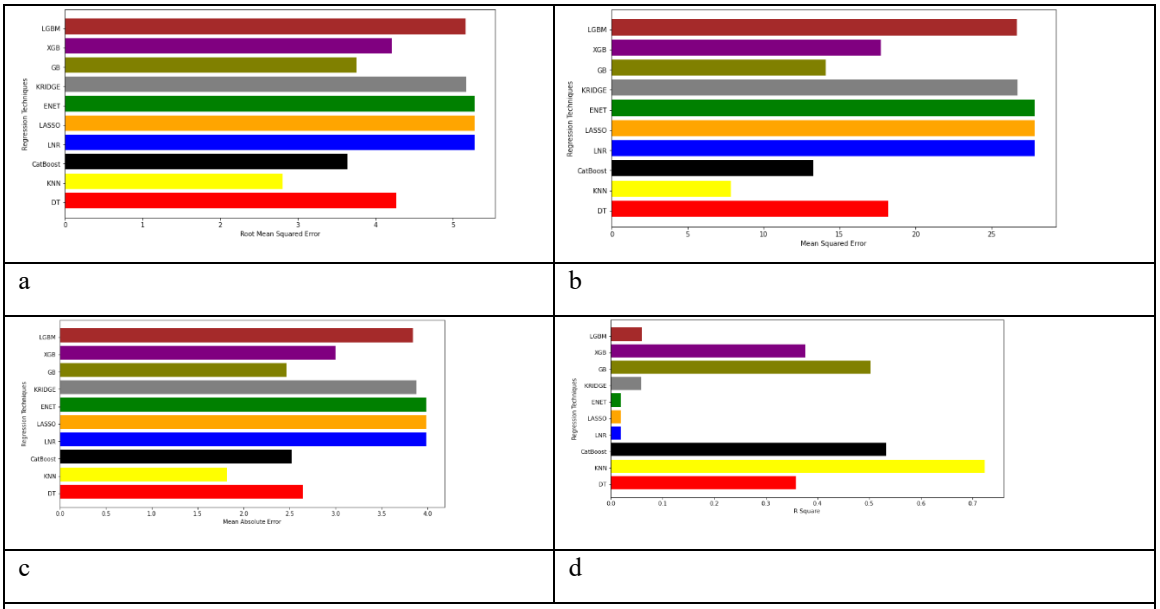


Fig. 3. (a): Root Mean Square Error across various regression methods. (b): MSE across various regression methods. (c): MAE across various regression methods. (d): RSquared across various regression methods.

Figure 3(b) illustrates the MSE values of the 10 regression models. Among these models, KNN exhibits the lowest MSE value, which is regarded as the most favorable value at 7.85 (refer to Table 1).

Figure 3(c) illustrates the Mean Absolute Error (MAE) values of the 10 regression models. Among these models, KNN exhibits the lowest value, 1.82, representing the most favorable outcome.

Figure 3(d) illustrates the R-squared values of the 10 regression models. R-squared serves as an assessment metric indicating the goodness of fit of the regression model. An ideal R-squared value is 1, signifying a perfect fit. The closer the R-squared value is to 1, the better the model fits the data. In this instance, KNN exhibits the highest value among all models, with a value of 0.72, which is considered the most favourable.

Figure 4 provides a summary of the results discussed earlier. It indicates that KNN consistently outperforms other regression techniques across various aspects.

4.2 Comparison of Ensemble Models

This comparative analysis aims to identify the most effective prediction ensemble model among the six proposed models for traffic forecasting. We have developed six distinct models: two based on bagging and four based on stacking. Figure 5(a)–(d) display the evaluation metrics, including RMSE, MSE, MAE, and R-squared, for each model.

Based on the findings depicted in Figures 5(a)–(d), it can be concluded that Stacking Ensemble Model3 consistently outperforms other ensemble regression models across all evaluation parameters. Stacking Ensemble Model3 demonstrates the most accurate traffic predictions compared to the alternatives. Specifically, its RMSE value of 2.09 is the lowest among all models, while its MSE value of 7.80 also ranks as the lowest. Additionally, Stacking Ensemble Model3, with Lasso as meta learner and KNN, Catboost and Decision Tree as base learners, exhibits the lowest MAE value of 1.82 and the highest R-squared value of 0.73 among all models.

Table 2 displays the parameter values for each ensemble regression model, while Figure 6 provides a visual representation of the above-mentioned results.

Table 1: Regression Models Comparison

Regression Models	RMSE	MSE	MAE	R-squared
Decision Tree	4.268587	18.220831	2.642944	0.358218
KNN	2.802102	7.851778	1.815638	0.723441
Catboost	3.642688	13.269178	2.526262	0.532627
Linear Regression	5.276512	27.841577	3.985246	0.019352
Lasso Regression	5.276527	27.841737	3.985317	0.019346
Elastic net Regression	5.276526	27.841727	3.985315	0.019347
Kernel Ridge Regression	5.169532	26.724058	3.879650	0.058714
Gradient Boosting	3.756299	14.109780	2.465573	0.503019
XGB	4.208979	17.715506	3.001670	0.376017
LGBM	5.164498	26.672042	3.842367	0.060546

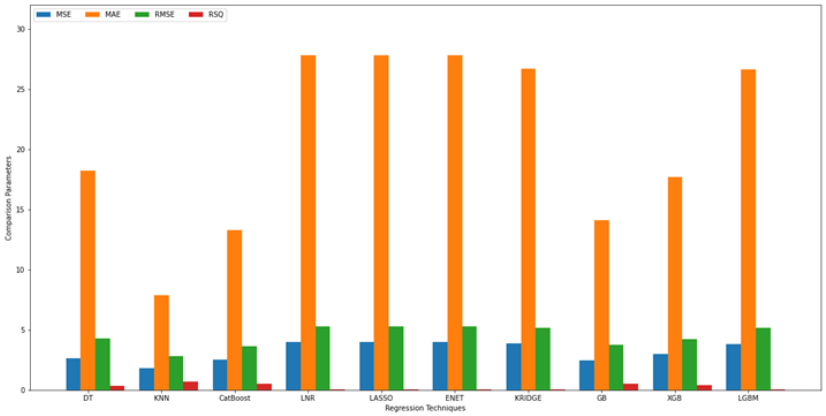
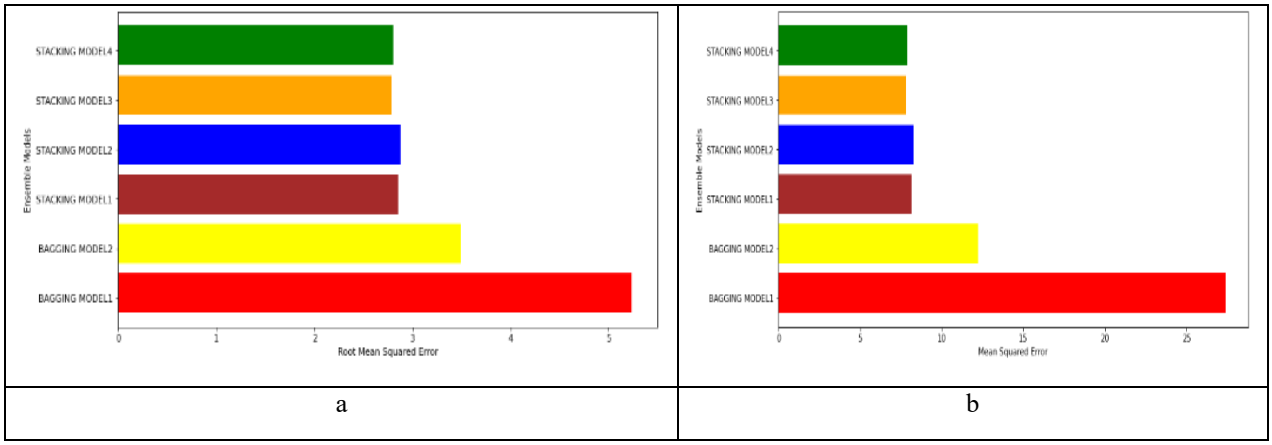


Fig. 4. Comparison of different Regression techniques.



a b

c d

Fig. 5. (a): RMSE for Ensemble Regression Models. (b): MSE for Ensemble Regression Models. (c): MAE for Ensemble Regression Models. (d): RSquared for Ensemble Regression Models.

Table 2: Ensemble Regression Models Comparison

Regression Models	**RMSE**	**MSE**	**MAE**	**R-Squared**
Bagging Ensemble Model 1: Linear Regression, Lasso, Elastic Net, and Kernel Ridge	5.234646	27.401520	3.945002	0.034852
Bagging Ensemble Model 2: Gradient Boosting, Catboost, XGB, and KNN	3.496845	12.227951	2.295272	0.521837
Stacking Ensemble Model1: Meta Learner: KNN Base Estimators: Catboost Regressor, Gradient Boosting Regressor, and Decision Tree Regressor	2.853295	8.141295	1.821564	0.713244
Stacking Ensemble Model2: Meta Learner: KNN Base Estimators: XGB, Catboost, and Decision Tree	2.876635	8.275027	1.827550	0.708533
Stacking Ensemble Model3: Meta Learner: Lasso Base Estimators: KNN, Catboost and Decision Tree	2.092050	7.795545	1.821481	0.825422
Stacking Ensemble Model4: Meta Learner: Lasso Base Estimator: KNN Catboost, Gradient Boosting Regressor	2.805884	7.872984	1.824692	0.722694

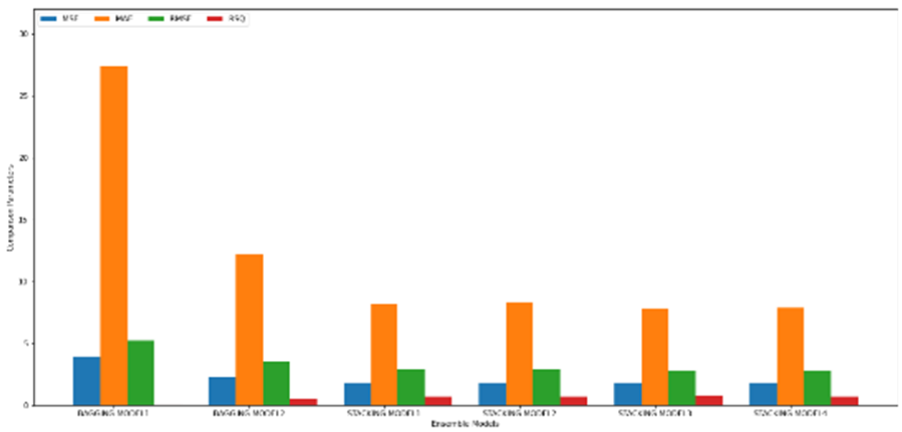


Fig. 6. Comparison of different Ensemble Regression Models.

5 Conclusion

In this research, we evaluated the efficacy of various Regression Models in accurately forecasting traffic flow. We introduced a method incorporating both bagging ensemble regression model techniques and stacking ensemble regression model techniques. This study was conducted in two distinct phases. In the first phase , we conducted a comparative analysis of 10 different regression models to identify the most precise model, with KNN emerging as the top performer. Secondly, we proposed a framework for bagging and stacking utilizing regression models. Our suggested stacking ensemble framework yielded superior outcomes in comparison to the 10 regression models and bagging models. The experimental results confirm the overall efficacy of the integrated approach we introduced. Especially noteworthy, the Stacking Ensemble Model, which utilized CatBoost, KNN, and Decision Tree as base learners, and Lasso as the meta learner, demonstrated robust performance with an RMSE of 2.09.

References

[1] Ouallane, A. A., Bahnasse, A., Bakali, A., & Talea, M. (2022). Overview of road traffic management solutions based on IoT and AI. Procedia Computer Science, 198, 518-523.

[2] Gouveia, N., Kephart, J. L., Dronova, I., McClure, L., Granados, J. T., Betancourt, R. M., ... & Diez-Roux, A. V. (2021). Ambient fine particulate matter in Latin American cities: Levels, population exposure, and associated urban factors. Science of the Total Environment, 772, 145035.

[3] Thangavel, P., Park, D., & Lee, Y. C. (2022). Recent insights into particulate matter (PM2. 5)-mediated toxicity in humans: an overview. International journal of environmental research and public health, 19(12), 7511.

[4] Kumar, K., & Pande, B. P. (2023). Air pollution prediction with machine learning: a case study of Indian cities. International Journal of Environmental Science and Technology, 20(5), 5333-5348.

[5] Bekkar, A., Hssina, B., Douzi, S., & Douzi, K. (2021). Air-pollution prediction in smart city, deep learning approach. Journal of big Data, 8(1), 1-21.

[6] Awan, F. M., Minerva, R., & Crespi, N. (2020). Improving road traffic forecasting using air pollution and atmospheric data: Experiments based on LSTM recurrent neural networks. Sensors, 20(13), 3749.

[7] Braz, F. J., Ferreira, J., Gonçalves, F., Weege, K., Almeida, J., Baldo, F., & Gonçalves, P. (2022). Road traffic forecast based on meteorological information through deep learning methods. Sensors, 22(12), 4485.

[8] Loumiotis, I., Demestichas, K., Adamopoulou, E., Kosmides, P., Asthenopoulos, V., & Sykas, E. (2018, September). Road traffic prediction using artificial neural networks. In 2018 South-Eastern European Design Automation, Computer Engineering, Computer Networks and Society Media Conference (SEEDA_CECNSM) (pp. 1-5). IEEE.

[9] Singh, S., Singh, J., Goyal, S. B., Sehra, S. S., Ali, F., Alkhafaji, M. A., & Singh, R. (2023). A novel framework to avoid traffic congestion and air pollution for sustainable development of smart cities. Sustainable Energy Technologies and Assessments, 56, 103125.

[10] Neelakandan, S. B. M. A. T. S. D. V. B. B. I., Berlin, M. A., Tripathi, S., Devi, V. B., Bhardwaj, I., & Arulkumar, N. (2021). IoT-based traffic prediction and traffic signal control system for smart city. Soft Computing, 25(18), 12241-12248.

[11] Shepelev, V., Slobodin, I., Almetova, Z., Nevolin, D., & Shvecov, A. (2023). A Hybrid Traffic Forecasting Model for Urban Environments Based on Convolutional and Recurrent Neural Networks. Transportation Research Procedia, 68, 441-446.

[12] Almeida, A., Brás, S., Oliveira, I., & Sargento, S. (2022). Vehicular traffic flow prediction using deployed traffic counters in a city. Future Generation Computer Systems, 128, 429-442.

[13] Tang, J., Chen, X., Hu, Z., Zong, F., Han, C., & Li, L. (2019). Traffic flow prediction based on combination of support vector machine and data denoising schemes. Physica A: Statistical Mechanics and its Applications, 534, 120642.

[14] Zhuang, W., & Cao, Y. (2023). Short-Term Traffic Flow Prediction Based on a K-Nearest Neighbor and Bidirectional Long Short-Term Memory Model. Applied Sciences, 13(4), 2681.

[15] 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.

[16] Fernandes, P., Tomás, R., Acuto, F., Pascale, A., Bahmankhah, B., Guarnaccia, C., ... & Coelho, M. C. (2020). Impacts of roundabouts in suburban areas on congestion-specific vehicle speed profiles, pollutant and noise emissions: An empirical analysis. Sustainable Cities and Society, 62, 102386.

[17] Luo, X.; Li, D.; Yang, Y.; Zhang, S. Spatio-temporal traffic flow prediction with KNN and LSTM. J. Adv. Transp. 2019, 2019. [Google Scholar] [CrossRef][Green Version]

[18] Song, X.; Li, W.; Ma, D.; Wang, D.; Qu, L.; Wang, Y.

A Match-Then-Predict Method for Daily Traffic Flow Forecasting Based on Group Method of Data Handling. Comput. Civ. Infrastruct. Eng. 2018, 33, 982–998. [Google Scholar] [CrossRef]

[19] [19] Ma, D.; Song, X.; Li, P. Daily Traffic Flow Forecasting Through a Contextual Convolutional Recurrent Neural Network Modeling Inter- and Intra-Day Traffic Patterns. IEEE Trans. Intell. Transp. Syst. 2021, 22, 2627–2636. [Google Scholar] [CrossRef]

[20] Qu, L.; Li, W.; Li, W.; Ma, D.; Wang, Y. Daily long-term traffic flow forecasting based on a deep neural network. Expert Syst. Appl. 2018, 121, 304–312. [Google Scholar] [CrossRef]

[21] Ma, D.; Ben Song, X.; Zhu, J.; Ma, W. Input data selection for daily traffic flow forecasting through contextual mining and intra-day pattern recognition. Expert Syst. Appl. 2021, 176, 114902. [Google Scholar] [CrossRef