ForecastBoost: An Ensemble Learning Model for Road Traffic Forecasting

Syed Muhammad Abrar Akber[1], Sadia Nishat Kazmi[2], Ali Muqtadir[3]
and Syed Muhammad Zubair Akber[4]

[1]*Department of Computer Graphics, Vision and Digital Systems, Faculty of Automatic Control, Electronics and Computer Science, Silesian University of Technology, 44-100 Gliwice, Poland*

[2]*Department of Distributed Computing and Informatic Devices, Faculty of Automatic Control, Electronics and Computer Science, Silesian University of Technology, 44-100 Gliwice, Poland*

[3]*School of Electronic and Electric Engineering, North China Electric Power University, Beijing 102206, China*

[4]*Department of Electrical Engineering, University of Sialkot, 513010 Sialkot, Pakistan*

{*abrar.akber, sadia.nishat.kazmi*}*@polsl.pl, alimuqtadir@ncepu.edu.cn, smzubair1974@gmail.com*

Keywords: Deep Learning, Ensemble Learning, Road Traffic, Time Series, Traffic Prediction.

Abstract: Accelerated urbanization is causing ever-increasing road traffic around the world. This rapid increase in road traffic is posing several challenges, such as road congestion, suboptimal emergency services due to inadequate road infrastructure and lack of economic sustainability. To overcome such challenges, intelligent transportation systems have recently become increasingly popular. Traffic prediction is an important part of such intelligent traffic management systems. Accurate traffic prediction leads to improved traffic flow, avoids congestion and optimizes the timing of traffic signals, resulting in higher vehicle fuel efficiency. Lower fuel consumption due to better fuel efficiency also limits the carbon footprints that help in combating global warming. To accurately predict road traffic, this paper proposes the *ForecastBoost* model, which leverages an ensemble learning approach to predict road traffic. ForecastBoost integrates two regression learning algorithms, namely *Extreme Gradient Boosting* and *Categorical Boosting*, to predict road traffic. The first component handles missing values and sparse data and the second handles categorical features without overfitting. We train the proposed ForecastBoost with a publicly available real-world traffic dataset. The obtained results are evaluated using similar state-of-the-art algorithms such as *Neural Hierarchical Interpolation for Time Series Forecasting (N-HiTS)*, *Series-cOre Fused Time Series (SOFTS)* and TimesNET. We use a well-known performance metrics containing several performance parameters, including mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean square error (RMSE), to evaluate the performance of the proposed ForecastBoost. The evaluation results show that the proposed ForecastBoost outperforms the other models.

1 INTRODUCTION

A report published by the United Nations states that more than half of the world's population currently resides in urban areas, and it is estimated that this proportion will rise to 68% by 2050 [1]. This rapid urbanization poses significant challenges for countries as they strive to meet the needs of growing urban populations while ensuring sustainable development. Transportation is one of the biggest challenges. Inrix reports that drivers in Bucharest and Bogota lose an average of 134 and 133 hours per year respectively due to traffic congestion (Zheng et al., 2023).

[1]https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html

Rapidly growing urbanization is encouraging vehicle manufacturers to increase the number of vehicles they produce. Vehicle manufacturers are producing more advanced vehicles by taking advantages of advances in Information and Communication Technologies (Akber et al., 2018), (Mohsin et al., 2021). As a result, road traffic has increased significantly in today's modern world. We are witnessing an increasing volume of traffic on the roads. This increased volume of traffic needs to be managed effectively for several reasons, such as improving safety, optimizing urban infrastructure (roads, bridges, etc.) and ensuring efficient emergency preparedness. Furthermore, effective traffic management can lead to avoiding traffic congestion, reducing fuel consumption and limit-

Akber, S. M. A., Kazmi, S. N., Muqtadir, A. and Akber, S. M. Z.
ForecastBoost: An Ensemble Learning Model for Road Traffic Forecasting.
DOI: 10.5220/0013155100003890

In *Proceedings of the 17th International Conference on Agents and Artificial Intelligence (ICAART 2025) - Volume 3*, pages 488-495
ISBN: 978-989-758-737-5; ISSN: 2184-433X

ing the carbon footprint, thus making the environment greener.

Traffic forecasting is an important aspect of modern intelligent traffic management systems. Traffic forecasting provides valuable insights for decision makers in urban planning and in the design of effective and timely emergency services. In addition, accurate traffic forecasting can optimize the timing of traffic signals, which can reduce waiting times at intersections and significantly improve fuel efficiency. Furthermore, predicting traffic patterns enables proactive measures to be taken to ensure safety in high-risk areas, which can reduce the number of accidents. With the existence of smart cities and connected vehicles, traffic prediction has become an integral part of the development of advanced transportation systems.

Anticipating the significance of traffic forecasting, a lot of research efforts are underway in this domain. The fundamental objective of traffic forecasting is to predict future traffic volumes on road networks using historical traffic data. The wide popularity of deep learning (DL) models in various domains (Akber et al., 2023) has encouraged their use for traffic forecasting (Trachanatzi et al., 2020), (Liu et al., 2023). These DL-based models are capable of examining spatio-temporal relationships to make more accurate traffic forecasts (Chen et al., 2020). Accordingly, such models are successfully being used to analyze road traffic and predict future road traffic.

Several transformer-based neural network models, which have an enhanced capability of processing input data in parallel, are also widely being applied for forecasting (Guo et al., 2019) (Mohsin et al., 2018). Such transformer-based models use self-attention to capture temporal dependencies and spatial correlations in the traffic data and make accurate predictions (Jiang et al., 2023). In addition to DL- and transformer-based models, time series forecasting (TSF) models are also very popular for predicting traffic (Zhang and Guo, 2020) (Shao et al., 2022). Such models leverage historical traffic data to create temporal patterns for predictions.

In recent decades, TSF solutions have evolved rapidly. Initially, traditional statistical methods such as ARIMA were very common. Later, machine learning techniques such as GBRT became increasingly popular. Nowadays, DL models are widely used. For predictive purposes, there has been a surge of transformer-based solutions for time series analysis in the recent past. However, there are several researchers (Zhou et al., 2022) (Liu et al., 2021) who propose models that focus on the challenges of the long-term time series forecasting (LTSF) problem, an area that is less explored. The objective of such models is to identify and utilise temporal patterns from historical traffic data and other relevant information to improve predictions.

Transformer-based models, when making predictions, focus multi-head self-attention mechanism for identifying semantic connections between elements in the data. However, this self-attention mechanism ignores the order of elements and is *permutation invariant* and *anti-order*. Although various encoding techniques can be applied to preserve the order information, still self-attention may potentially lose the temporal details. This might be of lesser significance for natural language processing (NLP) tasks, however, for TSF, preserving order becomes vital. Considering this fact, Zheng *et al.* (Zeng et al., 2023) coined an intriguing argument *Are transformers really effective for long-term time series forecasting*, which has become very popular and attracted the attention of researchers working on forecasting and prediction domain.

Furthermore, Zheng and his colleagues (Zeng et al., 2023) also presented a hypothesis that *"long-term forecasting is only feasible for those time series with a relatively clear trend and periodicity. As linear models can already extract such information"*. In support of this hypothesis, in this work, we propose *ForecastBoost* an ensemble learning model for TSF that integrates two popular regression learning algorithms, namely *Extreme Gradient Boosting (XGBoost)* (Li et al., 2024) and *Categorical Boosting (CatBoost)* (Zhang and Jánošík, 2024) (Fan et al., 2024) for predicting road traffic. We train our proposed approach by using a real-world traffic dataset. The obtained results are evaluated with similar state-of-the-art algorithms such as *Neural Hierarchical Interpolation for Time Series Forecasting (N-HiTS)* (Challu et al., 2023), *Series-cOre Fused Time Series (SOFTS)* (Han et al., 2024) and TimesNET (Wu et al., 2022). We use a well-known performance metrics that contains several performance parameters, including mean absolute error (MAE), mean absolute percentage error (MAPE) and root mean square error (RMSE). The specific contributions of this work are as follows;

- Propose *ForecastBoost* an ensemble learning TSF model to predict the road traffic
- Train our proposed model by using a real-world traffic dataset
- Evaluate the performance of the proposed model by comparing it with state-of-the-art models by considering several performance parameters such as MAE, MAPE, and RMSE.

The rest of the paper is organized as follows: Section 2 provides an insight into the existing literature review of the domain. Section 3 briefly describes the proposed approach, its working and the pseudocode. Section 4 presents the details of the experiment, dataset, methodology and the obtained results along with their evaluation. Lastly, section 5 concludes the paper.

2 LITERATURE REVIEW

In recent years, several urban areas have suffered from limited road networks due to ever-expanding road traffic. This imbalance between road infrastructure and road traffic leads to frequent congestion on roads, which may also lead to accidents. In such a situation, the traffic management system becomes vital in predicting road traffic and optimizing the decision-making process. Time series forecasting is a fundamental forecasting approach that is widely used for traffic forecasting. One possible reason for the wide adoption of time series forecasting is its simplicity. As a result, several authors utilize its capabilities in designing advanced traffic forecasting models.

Zheng *et al.* (Zheng and Huang, 2020) use a deep learning (DL) approach to perform traffic flow prediction based on time series analysis, aiming to address traffic congestion in urban areas. The authors employ DL techniques to make predictions for traffic flow through time series analysis Specifically, Zheng *et al.* (Zheng and Huang, 2020) propose a traffic flow forecasting model based on the long short-term memory (LSTM) network. The authors utilize the capabilities of DL for handling large data volumes and observe the patterns and consistency of traffic flow data to find out the periodicity in a city in China. Subsequently, Zheng *et al.* (Zheng and Huang, 2020) develop a traffic flow prediction model using LSTM. By utilizing massive raw data, this model outperforms classical methods such as ARIMA and backpropagation neural networks (BPNN) in terms of prediction accuracy. The experimental results show the superiority of the LSTM network and provide valuable insights into the dynamic development of traffic flow.

Luo *et al.* (Luo et al., 2019) propose a time series prediction of the traffic flow by presenting a methodology to integrate K-nearest neighbors (KNN) and LSTM networks. The proposed approach utilizes the spatiotemporal correlation in the traffic data for making traffic predictions with higher accuracy. The KNN component of the proposed method identifies and extracts the neighbouring nodes in the traffic data and the LSTM is used to make the predictions for the traffic flow at the identified nodes. The individual predictions of all these nodes are aggregated by weighting the predicted values to produce the final prediction. This approach also identifies the busiest traffic node at a particular location.

The hybrid approach proposed by *et al.* (Luo et al., 2019) effectively captures both temporal and spatial dependencies in the traffic data. The KNN and LSTM modules each individually identify the spatial and temporal correlations in the traffic data. This enables the approach to make more accurate traffic predictions.

In another work, Shao *et al.* (Shao et al., 2022) propose a time series model for traffic prediction called D^2STGNN. The authors focus on the challenges faced by Graph Neural Networks (GNNs) in identifying spatio-temporal correlations in traffic data. D^2STGNN improves the identification of spatio-temporal correlations in traffic data by decoupling them into two distinct components: (i) the diffusion model and (ii) the inherent model.. This decoupling enables the D^{2}STGNN to effectively handle the dynamic nature of traffic data and effectively learn the spatio-temporal correlations. The diffusion model and the inherent component of D^{2}STGNN separate the diffusion signal (which captures the propagation of traffic conditions) from the inherent signal (which represents direct spatiotemporal relationships), resulting in improved prediction accuracy.

Zheng *et al.* (Zhang and Guo, 2020) propose a time series traffic forecasting model called GALSTM, which uses a graph attention mechanism for traffic prediction. This graph-attention approach in GA-LSTM is applied within an LSTM network. The proposed GA-LSTM combines the capabilities of LSTM and spatio-temporal graph convolutional networks to capture correlations in traffic data. GA-LSTM models the road network as a graph, with nodes representing road segments and edges indicating connections between them. The graph-attention mechanism of GA-LSTM assigns weights to the nodes in the graph, enabling the learning and prediction of traffic patterns.

Shah *et al* (Shah et al., 2022) forecast traffic flow using a functional time series (FTS) approach. The proposed approach provides the traffic information for the entire day and facilitates traffic prediction at any desired time. The authors use a functional autoregressive (FAR) model to predict traffic flow for the next day. The authors demonstrate the effectiveness of their proposed approach by using the Dublin airport link road traffic data and making the prediction.

Chui *et al* (Cui et al., 2022) propose a two-stage hybrid learning model to address the challenges such as handling the nonlinear and stochastic nature of traffic data associated with short-term traffic prediction. The authors make a hybrid model by combining Particle Swarm Optimization (PSO) and Gravitational Search Algorithm (GSA) with the Extreme Learning Machine (ELM). The PSO component of the model identifies the population distribution for GSA that helps achieve global optimum search. Afterwards, ELM optimizes the results obtained by the PSO. The authors test their approach using a real-world dataset from Amsterdam and evaluate the model's performance using RMSE and MAPE values as the performance metrics.

3 PROPOSED ForecastBoost MODEL FOR ROAD TRAFFIC FORECASTING

This section describes the details of the proposed model and its working.

3.1 Overview of the Proposed ForecastBoost

The proposed ensemble learning model, ForecastBoost, aims to predict road traffic by integrating XGBoost and CatBoost, which both are regression algorithms. This makes ForecastBoost an ensemble learning model. ForecastBoost leverages the capabilities of both XGBoost and CatBoost to make traffic predictions with higher accuracy. XGBoost can efficiently handle missing values and sparse data and CatBoost is known to efficiently process categorical features without overfitting. Consequently, ForecastBoost is designed to achieve higher prediction performance in traffic prediction tasks. After the model design, it is trained on a comprehensive real-world traffic dataset. The effectiveness of ForecastBoost is evaluated by a comparative analysis with other state-of-the-art algorithms. We present an overview of ForecastBoost in Figure 1.

3.2 Mathematical Modelling of ForecastBoost

This subsection presents the mathematical modelling for the proposed ForecastBoost. Since ForecastBoost integrates XGBoost and CatBoost, we present the modelling of both components separately, followed by their integration into ForecastBoost.

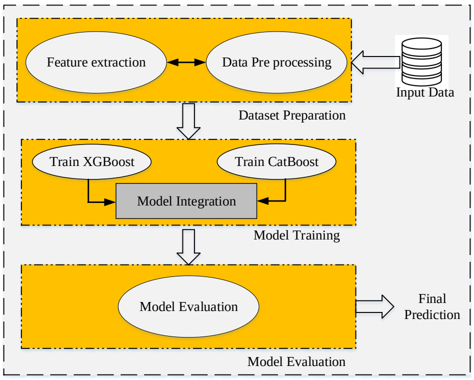


Figure 1: An overview of ForecastBoost.

3.2.1 XGBoost Component

It is a powerful ensemble learning algorithm widely being used for TSF. The mathematical modelling of XGBoost includes an objective function that compensates for the training loss and regularization to prevent overfitting. The objective function of XGBoost is shown in Eq.(1). It optimizes the following objective function to minimize the prediction errors:

$$O(\chi) = \sum_{i=1}^{n} l(x_i, \hat{x}_i) \tag{1}$$

where $O(\chi)$ represents the overall loss function, and $l(x_i, \hat{x}_i)$ is the point-wise loss function. l is the mean squared error as $l(x_i, \hat{x}_i) = (x_i - \hat{x}_i)^2$ and x_i represents the actual values, and $\hat{x}_i$ represents the predicted values. The modelling of regularization to prevent overfitting is presented in Eq. (2).

$$\varphi(\chi) = \sum_{k=1}^{K} \omega(f_k) \tag{2}$$

where $\varphi(\chi)$ represents the regularization function, $\omega(f_k)$ is the regularization term that penalizes model complexity to prevent the overfitting of the model. The regularization term is defined as in Eq (3).

$$\omega(f) = \gamma\tau + 0.5\left(\lambda \sum_{j=1}^{T} \mu_j^2\right) \tag{3}$$

where τ is the number of leave nodes, μ_j are the leaf weights, and γ is the regularization parameter.

The XGBoost component in the proposed ForecastBoost is modelled in Eq. (4).

$$\rho(\chi) = O(\chi) + \varphi(\chi) \tag{4}$$

where $\rho(\chi)$ represents the loss function.

3.2.2 CatBoost Component

CatBoost minimizes the following objective function while efficiently processing categorical features. The

Eq. (5) and Eq. (6) represent the equations for the loss function and the regularization parameter for the CatBoost component, respectively.

$$\delta(\chi) = \sum_{i=1}^{n} \delta(x_i, \hat{x}_i) \tag{5}$$

where δ is the loss function component.

$$\mathcal{L}(\Theta) = \kappa \sum_{j=1}^{m} \left(\frac{\partial \hat{y}_i}{\partial x_{ij}} \right)^2 \tag{6}$$

where Θ is the regularization parameter, and $\frac{\partial \hat{y}_i}{\partial x_{ij}}$ shows the sensitivity of model to feature j. We present the CatBoost component in the proposed ForecastBoost modelled in Eq. (7).

$$\mathcal{L}(\chi) = \delta(\chi) + \mathcal{L}(\Theta) \tag{7}$$

3.2.3 Integration of XGBoost and CatBoost Components

The final prediction $\hat{y}$ of ForecastBoost is derived from the weighted combination of the outputs from XGBoost and CatBoost as modelled in Eq. (8).

$$\hat{y} = \mu_1 \cdot \hat{y}_{\text{XGB}} + \mu_2 \cdot \hat{y}_{\text{Cat}} \tag{8}$$

where $\hat{y}_{\text{XGB}}$ and $\hat{y}_{\text{Cat}}$ are the predictions from XGBoost and CatBoost, respectively, and μ_1 and μ_2 are the weights assigned to XGBoost and CatBoost predictions respectively.

3.3 Working of the ForecastBoost

ForecastBoost is an ensemble architecture that aims to increase prediction performance for road traffic prediction by leveraging the capabilities of the XGBoost and CatBoost algorithms. The detailed working of ForecastBoost can be divided into several phases. In the first phase, the data is pre-processed to remove missing values and noise. In addition, feature extraction is performed in this phase to extract temporal and event-related features. The final step in this phase is the normalization of the data so that the data may be scaled uniformly to achieve effective model training and convergence. Once data prepossessing is complete, ForecastBoost concurrently implements the XGBoost and CatBoost algorithms. Both algorithms are gradient-boosted decision trees (Zhang and Jánošík, 2024). XGBoost is a popular algorithm for efficiently handling sparse data and missing values (Hakkal and Lahcen, 2024), while CatBoost is known for effectively managing categorical features and avoiding overfitting (Fan et al., 2024).

The third phase of ForecastBoost is the model training phase, in which the model is trained with the training dataset, which is a fraction of the input dataset. Before training begins, hyperparameters are set to achieve optimal model performance. The individual components of ForecastBoost are trained individually, then their outputs are integrated and synchronized to obtain the final forecast values. The individual outputs of XGBoost and CatBoost are synchronized using a weighted average approach.

3.4 Pseudocode for ForecastBoost

We present the pseudo-code for the proposed ForecastBoost as Algorithm 1.

Algorithm 1: ForecastBoost: An Ensemble Learning Model for Road Traffic Prediction.

Input: Traffic dataset $\mathcal{D}$
Output: Predicted traffic values
Divide $\mathcal{D}$ into $\mathcal{D}_{\text{train}}$, $\mathcal{D}_{\text{val}}$, and $\mathcal{D}_{\text{test}}$
Initialize XGBoost with hyperparameters
Use $\mathcal{D}_{\text{train}}$ to train XGBoost and get M_{XGB}
Use $\mathcal{D}_{\text{val}}$ to optimize hyperparameters for XGBoost
Initialize CatBoost with hyperparameters
Use $\mathcal{D}_{\text{train}}$ to train CatBoost and get M_{Cat}
Use $\mathcal{D}_{\text{val}}$ to optimize hyperparameters CatBoost
Obtain predictions $\hat{\mathbf{y}}_{\text{XGB}}$ from M_{XGB}
Obtain predictions $\hat{\mathbf{y}}_{\text{Cat}}$ from M_{Cat}
Learn optimal weights μ_1 and μ_2 on $\mathcal{D}_{\text{val}}$
Compute final predictions $\hat{\mathbf{y}}$ as:

$$\hat{\mathbf{y}} = \mu_1 \cdot \hat{\mathbf{y}}_{\text{XGB}} + \mu_2 \cdot \hat{\mathbf{y}}_{\text{Cat}}$$

Return Predicted traffic values $\hat{\mathbf{y}}$

The algorithm takes the preprocessed traffic dataset $\mathcal{D}$ as the input. The input data contains feature vectors $\mathbf{X}$ and corresponding target values $\mathbf{y}$. The output of the algorithm is the predicted traffic values $\hat{\mathbf{y}}$. The dataset $\mathcal{D}$ is divided into three subsets: (i) a training set $\mathcal{D}_{\text{train}}$, (ii) a test set $\mathcal{D}_{\text{test}}$, and (iii) a validation dataset $\mathcal{D}_{\text{test}}$. The training data is used to train the model, validation data is employed for hyperparameter tuning and validation of the model and the test data is used for final performance evaluation.

After dividing the dataset, the XGBoost component of ForecastBoost is initiated and trained on the $\mathcal{D}_{\text{train}}$ and create XGBoost model M_{XGB}. At the same time, CatBoost is also inilize and trained in analogous to XGBoost and produces CatBoost model M_{Cat}. The ForecastBoost optimizes the Hyperparameters for XGBoost and CatBoost model through cross-validation on $\mathcal{D}_{\text{val}}$. Once the individual models

Table 1: Experimental Setup.

Description	Details
Hardware	- CPU : Intel(R) core(TM) i7 920 @2.67 GHz - RAM: 16 GB - OS: Windows 10 (64 bits) - Graphics Card: NVIDIA GeForce GT 440
Software	- Python 3.7.9 (64 bits) - PyTorch - VS Code 1.91.1
Baseline	N-HiTS, SOFTS, TimeNet

have generated their outputs, predictions are generated $\hat{\mathbf{y}}_{\mathrm{XGB}}$ and $\hat{\mathbf{y}}_{\mathrm{Cat}}$ from the respective models. These predictions are then integrated and synchronised to identify the optimal weights μ_1 and μ_2. The final prediction $\hat{\mathbf{y}}$ is performed by $\hat{\mathbf{y}} = \mu_1 \cdot \hat{\mathbf{y}}_{\mathrm{XGB}} + \mu_2 \cdot \hat{\mathbf{y}}_{\mathrm{Cat}}$

4 EXPERIMENT, RESULTS AND EVALUATION

This section describes our experiments, obtained results and their evaluation for the proposed ForecastBoost. We first present the experiment environment.

4.1 Experiment Environment

We conduct practical experiments to verify the working of the ForecastBoost. All experiments are conducted on a PC with Intel core i7 processor with 16 GB RAM. We use Python 3.7.9 (64-bit version) and VS Code version 1.91.1 as the IDE. The detailed environment setup is presented in Table 1.

4.2 Dataset and Methodology

For the experiment, we use a real-world traffic dataset, PEMS-08, obtained from a publicly available repository from Kaggle [2]. The dataset contains information about traffic flow, time step, location, speed of vehicles, etc. To implement ForecastBoost, we first pre-process the input traffic dataset and extract the features, both temporal features and external features (traffic flow and speed), for training. This feature extraction phase is the key to capturing the temporal dependencies. The target variable is separated from the features, enabling the creation of a prediction model. We then define the prediction horizon and the size of the inputs such that the size of the inputs is twice the size of the horizon to ensure a comprehensive set of lagged inputs. The function is called to generate the lagged dataset, which is then split into training and testing sets, with 80% of the data used for training and the remaining 20% for testing.

Table 2: Hyperparameters.

Model	Hyperparameter	Value
XGBoost	Learning Rate	0.1
	Max Depth	6
	Min Child Weight	1
	Number of Estimators	100
CatBoost	Learning Rate	0.1
	Max Depth	6
	L2 Leaf Regularization	3
	L2 Leaf Regularization	100

[2] https://www.kaggle.com/datasets/elmahy/pems-dataset

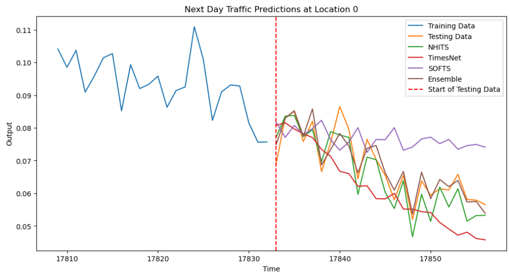


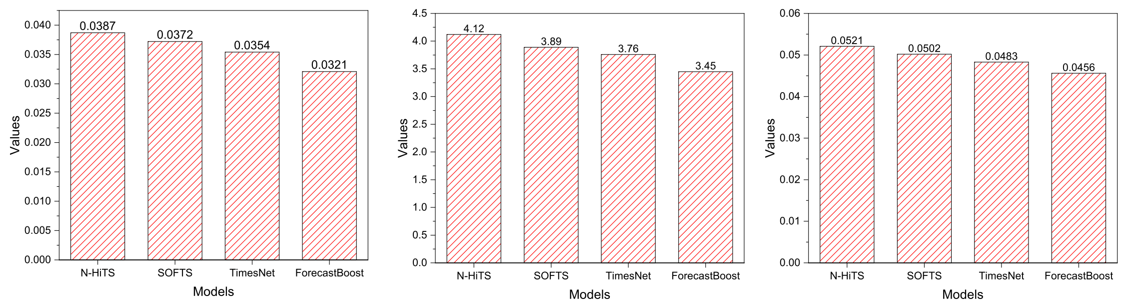
Figure 2: Training and testing of different models.

After feature extraction, we perform the normalization process. Both components of ForecastBoost, XGBoost and CatBoost, are trained independently and their hyperparameters are optimized by cross-validation on the validation data. We present the hyperparameters in Table 2. The individual predictions of the two components are merged using a weighted average to produce the final prediction. We evaluate the performance of our proposed ForecastBoost by comparing it with other similar models such as NHITS, SOFTS and TimesNET. For the evaluation, we use a well-known performance metrics consisting of MAE, MAPE and RMSE.

4.3 Results and Evaluation

After training of the ForecastBoost is completed, we record the results and perform the evaluation. We present the training and testing of different models in Figure 2. For the evaluation, we perform a comprehensive comparative analysis and compare our results with the existing similar models. We compare our results with NHITS, TimesNet and SOFTS.

We present the details of our comparative analysis and the results we obtained for the different models in terms of all performance parameters are shown in Figure 3. The values in Figure 3 show that the proposed ForecastBoost outperforms the other models in all performance parameters. Specifically, ForecastBoost achieves the lowest value for MAE of 0.0321, outperforming NHITS (0.0387), SOFTS (0.0372) and



(a) MAE for different models (b) MAPE for different models (c) RMSE for different models

Figure 3: Comparative Analysis for different models.

TimesNet (0.0354) as shown in Figure 3a.

Figure 3b shows the results obtained for MAPE for different models. For MAPE, ForecastBoost shows a value of 3.45%, while the values for MAPE for NHITS, SOFTS and TimesNet are 4.12%, 3.89% and 3.76% respectively. Figure 3c shows the values obtained for the RMSE It can be seen that the RMSE value for ForecastBoost is the lowest at 0.0456, while the RMSE values for NHITS SOFTS, and TimesNet are 0.0521, 0.0502, and 0.0483 respectively. The comparative analysis shows that ForecastBoost is better able to limit the deviations between the predicted and actual traffic values, thus confirming the robustness and effectiveness of the proposed ForecastBoost.

5 CONCLUSION

In this paper, we address the challenge of effective road traffic prediction to facilitate the decision-making process for the provision of road infrastructure to cope with future traffic needs. Considering the significance of traffic prediction, we propose ForecastBoost, an ensemble learning model for road traffic forecasting. ForecastBoost integrates the capabilities of two popular regression-based models, XGBoost and CatBoost, to produce time-series forecasts for road traffic. The XGBoost component can efficiently handle missing values and sparse data, while CatBoost handles categorical features without overfitting. The individual predictions of the two models are integrated before ForecastBoost makes a final prediction. We implement ForecastBoost and train it with real traffic data.

We evaluate the performance of our proposed ForecastBoost by comparing it with other similar models such as NHITS, SOFTS and TimesNET. For the evaluation, we use a well-known performance metrics consisting of mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean square error (RMSE). The evaluation results show that the proposed ForecastBoost approach outperforms the existing model. The comparative analysis shows the higher performance of ForecastBoost in limiting the deviations between the predicted and actual traffic values, thus confirming the robustness and effectiveness of the proposed ForecastBoost.

ACKNOWLEDGMENT

This publication was supported by the pro-quality research grant under the framework of the Excellence Initiative - Research University programme, of the Silesian University of Technology, Gliwice, Poland under project number 32/014/SDU/10-22-66.

REFERENCES

Akber, S. M. A., Kazmi, S. N., Mohsin, S. M., and Szczesna, A. (2023). Deep learning-based motion style transfer tools, techniques and future challenges. *Sensors*, 23(5).

Akber, S. M. A., Khan, I. A., Muhammad, S. S., Mohsin, S. M., Khan, I. A., Shamshirband, S., and Chronopoulos, A. T. (2018). Data volume based data gathering in wsns using mobile data collector. In *Proceedings of the 22nd International Database Engineering & Applications Symposium*, pages 199–207.

Challu, C., Olivares, K. G., Oreshkin, B. N., Ramirez, F. G., Canseco, M. M., and Dubrawski, A. (2023). Nhits: Neural hierarchical interpolation for time series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 6989–6997.

Chen, C., Li, K., Teo, S. G., Zou, X., Li, K., and Zeng, Z. (2020). Citywide traffic flow prediction based on multiple gated spatio-temporal convolutional neural networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 14(4):1–23.

Cui, Z., Huang, B., Dou, H., Cheng, Y., Guan, J., and Zhou, T. (2022). A two-stage hybrid extreme learning model for short-term traffic flow forecasting. *Mathematics*, 10(12):2087.

Fan, Z., Gou, J., and Weng, S. (2024). A feature importance-based multi-layer catboost for student performance prediction. *IEEE Transactions on Knowledge and Data Engineering*.

Guo, S., Lin, Y., Feng, N., Song, C., and Wan, H. (2019). Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 922–929.

Hakkal, S. and Lahcen, A. A. (2024). Xgboost to enhance learner performance prediction. *Computers and Education: Artificial Intelligence*, page 100254.

Han, L., Chen, X.-Y., Ye, H.-J., and Zhan, D.-C. (2024). Softs: Efficient multivariate time series forecasting with series-core fusion. *arXiv preprint arXiv:2404.14197*.

Jiang, J., Han, C., Zhao, W. X., and Wang, J. (2023). Pdformer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 4365–4373.

Li, X., Shi, L., Shi, Y., Tang, J., Zhao, P., Wang, Y., and Chen, J. (2024). Exploring interactive and nonlinear effects of key factors on intercity travel mode choice using xgboost. *Applied Geography*, 166:103264.

Liu, H., Dong, Z., Jiang, R., Deng, J., Deng, J., Chen, Q., and Song, X. (2023). Spatio-temporal adaptive embedding makes vanilla transformer sota for traffic forecasting. In *Proceedings of the 32nd ACM international conference on information and knowledge management*, pages 4125–4129.

Liu, S., Yu, H., Liao, C., Li, J., Lin, W., Liu, A. X., and Dustdar, S. (2021). Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting. In *International conference on learning representations*.

Luo, X., Li, D., Yang, Y., and Zhang, S. (2019). Spatiotemporal traffic flow prediction with knn and lstm. *Journal of Advanced Transportation*, 2019(1):4145353.

Mohsin, S. M., Javaid, N., Madani, S. A., Akber, S. M. A., Manzoor, S., and Ahmad, J. (2018). Implementing elephant herding optimization algorithm with different operation time intervals for appliance scheduling in smart grid. In *2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, pages 240–249.

Mohsin, S. M., Khan, I. A., Abrar Akber, S. M., Shamshirband, S., and Chronopoulos, A. T. (2021). Exploring the rfid mutual authentication domain. *International Journal of Computers and Applications*, 43(2):127–141.

Shah, I., Muhammad, I., Ali, S., Ahmed, S., Almazah, M. M., and Al-Rezami, A. (2022). Forecasting day-ahead traffic flow using functional time series approach. *Mathematics*, 10(22):4279.

Shao, Z., Zhang, Z., Wei, W., Wang, F., Xu, Y., Cao, X., and Jensen, C. S. (2022). Decoupled dynamic spatial-temporal graph neural network for traffic forecasting. *arXiv preprint arXiv:2206.09112*.

Trachanatzi, D., Rigakis, M., Marinaki, M., and Marinakis, Y. (2020). An interactive preference-guided firefly algorithm for personalized tourist itineraries. *Expert Systems with Applications*, 159:113563.

Wu, H., Hu, T., Liu, Y., Zhou, H., Wang, J., and Long, M. (2022). Timesnet: Temporal 2d-variation modeling for general time series analysis. *arXiv preprint arXiv:2210.02186*.

Zeng, A., Chen, M., Zhang, L., and Xu, Q. (2023). Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 11121–11128.

Zhang, L. and Jánošík, D. (2024). Enhanced short-term load forecasting with hybrid machine learning models: Catboost and xgboost approaches. *Expert Systems with Applications*, 241:122686.

Zhang, T. and Guo, G. (2020). Graph attention lstm: A spatiotemporal approach for traffic flow forecasting. *IEEE Intelligent Transportation Systems Magazine*, 14(2):190–196.

Zheng, G., Chai, W. K., Duanmu, J.-L., and Katos, V. (2023). Hybrid deep learning models for traffic prediction in large-scale road networks. *Information Fusion*, 92:93–114.

Zheng, J. and Huang, M. (2020). Traffic flow forecast through time series analysis based on deep learning. *IEEE Access*, 8:82562–82570.

Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., and Jin, R. (2022). Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International conference on machine learning*, pages 27268–27286. PMLR.