Spatiotemporal Ensemble Modeling for Urban Taxi Travel Time Prediction

Krish Bharucha[1], Jeet Mehta[2], Devarshee Thopte[3], Dhruv Gohil[4], Nilesh Patil[5]

[1,2,3,4,5]SVKM'sDwarkadas J Sanghvi College of Engineering, Mumbai, India
Email: krishbharucha03@gmail.com[1], jeetmehta2026@gmail.com[2], devarshee.t@gmail.com[3], dhruvgohil03@gmail.com[4], nilesh.p@djsce.ac.in[5]

Received: 10.01.2024 Revised: 19.02.2024 Accepted: 27.02.2024

ABSTRACT
In this paper, we propose to address the challenge in predicting urban taxi travel time through spatiotemporal ensemble modelling techniques. Accurate travel time prediction is very important not just for planning urban mobility but also for running ridesharing platforms with excellent user experience. We compare a stacking model to a ridge meta-regressor optimised through gradient descent, KNN regressor tuned by hill-climbing, decision tree and random forest.The stacking model emerged as the best, underscoring again the efficacy of ensemble methods in coping with the spatiotemporal complexity of the data. KNN regressors, with their hyperparameters optimised through hill climbing yielded competitive results.Our experimental results show that the ensemble approach, especially the stacking model, generally has much higher performance in prediction than individual models for taxi travel time. This proves that the ensemble approach can effectively work as an alternative way to predict real-time urban transportation system states.

Keywords: KNN, Decision Tree, Random Forest, Hill Climb, Stacking Ensemble.

1. INTRODUCTION

With the rapid development of cities and the growing demand for transportation services, urban mobility is becoming increasingly complex. In this context, the ability to predict taxi travel time accurately in urban areas is crucial for city planners, transport network companies, and regular travellers alike. Accurate predictions can enhance the efficiency of ride-hailing services, improve traffic management, and contribute to the overall optimization of urban transportation systems.Yet, due to the spatiotemporal complexities of urban traffic patterns, travel time prediction remains a major challenge, despite the eminent strides in the development of machine learning.

Therefore, the problem this research tries to solve is to accurately estimate the travel time of taxis in urban cities with the use of advanced machine learning models that capture both spatial and temporal dependencies in data. Traditional practices in travel time prediction often use basic statistical methods or take independent models of machine learning into consideration which may not account for such interdependencies effectively-so, traffic congestion could be combined with road networking and temporal variations of traffic.

In the light of this, our contribution falls under a certain ensemble modeling technique that we use in an attempt to combine both time and space features in the obtaining of optimum prediction accuracies. We implemented an ensemble method, stacking, with a meta-ridge regressor optimised using gradient descent, and a KNN regressor optimised using hill climbing, apart from other established models like decision trees and random forests. Because of this, our ensemble strategy leveraged the strengths of each model by incorporating their different predictions and leading to a more accurate and robust outcome.

This research utilises a publicly available taxi trajectory dataset from Kaggle and through a comparative analysis between the proposed ensemble model and individual models, we aim to demonstrate the effectiveness of ensemble learning in capturing the complex spatiotemporal patterns inherent in urban taxi travel time data. The findings from this research have the potential to contribute to the development of more efficient and reliable urban transportation systems.

2. RELATED WORK

In a daily commuter's life, Travel Time Prediction is a critical service regarding precision and reliability. It can also be applied in logistics and urban mobility systems management. Machine learning techniques, on the other hand—more precisely, deep learning methods—have done wonders by boosting accuracy and

reliability in TTP models, some of which are the ensemble methods. Tree-based ensemble methods, including but not limited to Random Forest, XGBoost, and LightGBM, make a huge amount of applications in estimating travel time inside urban areas. These models are best suitable in addressing feature co-linearity and high dimensionality, which are phenomena most likely to occur in traffic data analysis. They also take much less computational power when compared to complex deep learning models. Studies have shown that the effectiveness of these models can outperform traditional methods by including relevant features such as weather conditions, time of day, and historical travel times. [1]-[2] For example, isolated XGBoost regression models exhibit high effectiveness when applied for the prediction of static travel times from vast datasets, like New York City taxi trip data. These models are particularly robust in relation to handling outliers and extreme conditions that frequently occur in urban traffic scenarios. Application of the XGBoost technique gives much improvement in prediction accuracy compared to other models such as neural networks or support vector regression. That is the reason why hybrid-ensemble models promise to be very promising in TTP, which is an approach that combines the strengths of different machine learning techniques in an attempt to capture complex nonlinearities in traffic data. [3]-[4] For instance, advanced hybridized models, such as deep-feature-space-aggregating convolutional neural networks, multilayer perceptrons, long short-term memory, and gated recurrent unit models, with support vector regression, are proposed for high improvements of prediction accuracy in travel time. Hybrid models will be trained based on such varied data sources to effect better travel time predictions using dimensionality reduction and optimization of feature space. [5] On the other hand, the Stacking ensembles build upon a number of other models: they are a combination of machine learning algorithms XGBoost, Random Forests, Extra Trees as base learners and MLPs as meta-learners for speed prediction in traffic. [7-11] Research has shown that these models are superior in performance by significant reduction of prediction error and improvement in accuracy over the conventional approaches. Thus, use of ensemble method, particularly those hybridized with others so as to optimize the feature space, proves to be a very potent approach in travel time prediction. Furthermore, they not only increase predictive accuracy but also provide a more solid solution against the intrinsic complexities of urban traffic systems.

2. DATASET

Our study utilizes a comprehensive dataset [6] encompassing a full year of taxi trajectory information from New York City, spanning July 1, 2013, to June 30, 2014. This data, compiled in a CSV file titled "train.csv", covers the operations of 442 taxis equipped with mobile data terminals and connected to a central dispatch system.

We've classified each taxi journey into three distinct categories based on how the service was initiated: through the central dispatch, from a designated taxi stand, or via direct street hailing. For centrally dispatched rides, we've included an anonymized identifier when available from the call data.

The dataset comprises nine key attributes for each completed trip:

1. Trip Identifier: A unique string assigned to each journey.
2. Service Initiation Type: A character code indicating how the ride was requested - 'A' for central dispatch, 'B' for taxi stand, and 'C' for street hailing.
3. Customer Phone ID: An integer uniquely identifying the caller's phone number for dispatch-initiated trips (null for other types).
4. Taxi Stand Code: An integer identifying the starting location for stand-initiated trips (null for other types).
5. Driver Identifier: A unique integer for each taxi operator.
6. Trip Start Time: Recorded as a Unix timestamp in seconds.
7. Day Classification: A character code categorizing the day of travel - 'B' for holidays or special occasions, 'C' for days preceding type 'B' days, and 'A' for regular days.
8. GPS Data Integrity: A boolean flag indicating whether the GPS data stream is complete (false) or has gaps (true).
9. Route Coordinates: A string containing GPS coordinates (in WGS84 format) recorded at 15-second intervals throughout the journey, with the first and last entries representing the trip's start and end points, respectively.

In our preprocessing phase, we've converted the route coordinate data into total travel time for each trip. This dataset provides a rich foundation for analyzing urban taxi travel patterns and developing predictive models for journey times in New York City.

3. METHODOLOGY

This section describes the different regression models used in the research.

KNN Regressor with Hill Climbing Optimizer

A KNN regression model was applied to predict travel time in an urban taxi. The KNN algorithm is based on the principle of similarity: neighbours in feature space normally have similar target values. In this context, the KNN model predicts the travel time of a new taxi trip based on the average travel times of its K nearest neighbours in the training dataset.The approach was, then, implemented: hill climbing search to optimise the performance of the KNN model. This is done by initialising the K value that will be used by the algorithm in the beginning. The K value is a hyperparameter that defines how many neighbours are considered for predictions. Subsequently, neighbouring values of k are explored one at a time, choosing each in turn that best minimises the training error until convergence or until a maximum number of evaluations are carried out.The KNN model was fitted on historical taxi trip datasets, in which the features are the pick-up and drop-off coordinates, time, day of the week, and traffic conditions. The reoptimized KNN model was then subjected to generalisation performance testing on a hold-out sample, using metrics like RMSE and R-squared. The RMSE is calculated as:

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

whereN is the total number of predictions, y_i is the actual target value, and $\hat{y}_i$ is the predicted targetvalue.

Decision Tree

The Decision Tree regression model has been applied to predict travel time for urban taxis. Decision Trees operate on the principle of recursively splitting the feature space so that each region corresponds to a different target value range. In this context, the Decision Tree model predicts the travel time for a new taxi trip by learning from the historical relationships among input features such as pick-up and drop-off coordinates, time of day, day of the week, and traffic conditions, in relation to the target travel time.

The implementation began with the construction of the Decision Tree by recursively partitioning the dataset based on the feature values that most effectively minimise the prediction error. At every node of the tree, that algorithm picks the feature and a threshold for it such that one can get maximum impurity reduction. Common metrics for this impurity are MSE and variance in regression tasks. This goes on until all the leaves represent homogenous subsets of the data or when any stopping criterion (a predefined maximum tree depth or minimum number of samples per leaf) is reached.

We have trained a Decision Tree model on historical taxi trip data; it learned the mapping from input features to travel time. Next, we tested the performance of this trained model on the holdout test set by using generalisation performance. Metrics used to quantify the accuracy of the model and the amount of variance in travel time that the features can explain include the RMSE and R-squared values, respectively. Together with the interpretability of this model, the possibility to capture nonlinear relationships presents it as an effective tool for the analysis of the comparison of predictive models for urban taxi travel time. Entropy is an important concept in decision trees, used to measure the impurity or uncertainty in a dataset. It is computed as:

$$Entropy = \sum_{i=1}^{c} -p_i \times \log_2 p_i$$

wherep_i is the proportion of instances in class *i*.

Random Forest

Random Forest regression model was implemented for the prediction of urban taxi trip travel time. Random Forest is an ensemble learning technique that works in constructing many Decision Trees to improve predictive accuracy and increase generalisability. The intuition behind the method is that since the predictions from a host of trees are averaged out, this leads to low risk of overfitting by the model and increases generalization performance. This random forest model will make an average prediction for the travel time of this new taxi trip using different predictions obtained by a collection of trees trained on different subsets of data.

In practice many Decision Trees are fit on a randomly sampled subset of the training data. A random subset of features is chosen at each split in a tree. Ensuring the trees of the forest will be uncorrelated. Therefore, each tree predicts the travel time, and these predictions are averaged in order to give the final output. It is also suitable for the nature of the problem at hand, which is spatiotemporal taxi travel time prediction, since this model is capable of capturing complex interactions among features while resisting overfitting.

We then fit the Random Forest model on the historic taxi trip data and evaluated the generalization performance using a hold-out test set. Standard regression metrics included the RMSE, besides R-squared

to assess the accuracy and the proportion of variance in travel time explained by the features. This ensemble method yielded a robust model to handle the variability and complexity of the data that underlies the prediction of urban taxi travel time. The formula for MSE is as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(f_i - y_i)^2$$

Stacking Ensemble

The research presented in this paper employs a sophisticated ensemble technique known as stacking to forecast urban taxi journey durations. Stacking is an advanced approach to combining multiple predictive models, which aims to enhance overall predictive accuracy by leveraging the unique strengths of various base models.In this study, we implement a two-tier stacking architecture. The foundation consists of several base regression models, each contributing its predictions to the ensemble. These predictions are then synthesized by a second-tier model, referred to as the meta-regressor, which is responsible for producing the final travel time estimate.

To optimize the performance of our stacking ensemble, we've made two key design choices:

1. We've selected Ridge regression as our meta-regressor. Ridge regression is a linear model that incorporates L2 regularization, which helps prevent overfitting and can handle multicollinearity among input features effectively.
2. To train the meta-regressor, we employ gradient descent optimization. This iterative algorithm adjusts the model parameters to minimize the prediction error, allowing the meta-regressor to learn the optimal way to combine the base models' outputs.

By integrating these elements - multiple base regressors, a Ridge regression meta-model, and gradient descent optimization - our stacking ensemble is engineered to capture complex patterns in urban taxi travel times, potentially offering superior predictive performance compared to individual models or simpler ensemble methods.

Base Regressors

The following are the base regressors that make up this stacking model: Decision Tree Regressor, Lasso Regressor, and KNN Regressor. Each model individually has its strength that it brings to the ensemble:

- **Decision Tree Regressor:** Useful in modeling complex datasets, it picks up nonlinear relationships and interactions between features.
- **Lasso Regressor:** A linear model that does feature selection through the penalization of the absolute size of the regression coefficients to reduce overfitting and for model interpretability.
- **K-Nearest NeighborsRegressor:** A nonparametric model that predicts its target by averaging out the outcomes of the k-nearest neighbors, thus capturing local patterns within the data.
  These base regressors are initialized and trained using the training dataset. They learn from the spatiotemporal features of the taxi trajectory data.

Regressors and Meta Regressor

In a stacking framework, the base regressors like Decision Tree, Lasso, and KNN are separately trained. It can be said that all these regressors capture different aspects of data: non-linear interactions, feature sparsity, or local patterns. These predictions from the base regressors are then passed to the input features of the next layer of a meta-regressor, which, in this case, is again a Ridge Regressor.

The Ridge meta-regressor learns the optimal way of combining the predictions of the base models to leverage the individual strengths of the base models and produce a more accurate final prediction. By using the Ridge Regressor, the model makes it possible to minimize the prediction errors efficiently through gradient descent so that this stacking model has high accuracy in predicting taxi travel times.The output from the base regressors is fed as an input to the metaregressor. Accordingly, the metaregressor used is Ridge Regression, which was trained by a gradient descent algorithm. Gradient Descent works in such a way that in iterative steps, the coefficients are adjusted so that the minimal error can be achieved. This step helps in making sure the metaregressor is combining the outputs of base regressors correctly.

Such an approach effectively integrates various diverse strategies and modeling attempts into one in the light of predictive modeling effort for a generalized robust, and accurate prediction model, suitable for complexities arising from the urban taxi travel-time data set.

4. RESULTS

To evaluate the performance of the proposed ensemble models, extensive experiments were conducted using a dataset comprising 15,000 taxi routes. The results are summarized below:

KNN Regressor with Hill Climbing Optimizer

The KNN regressor, optimized using hill climbing, performed very well. It achieved a root mean squared error (RMSE) of 30.0458 and an R-squared score of 0.9979 with an optimal k-value of 5. These metrics suggest that the model is promising in predicting taxi travel times. For example, a randomly selected taxi trip with ID 3937 was predicted to last approximately 1,335 seconds, which was quite close to the actual duration.

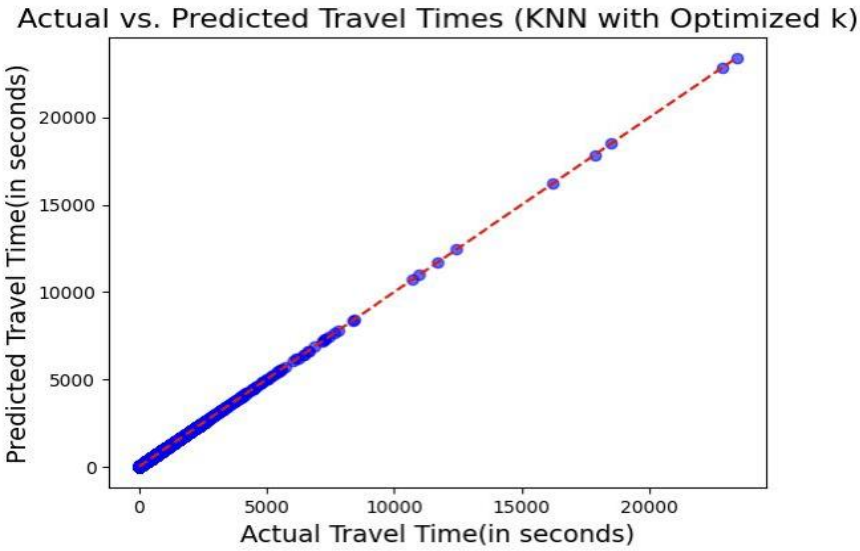

Fig. 1: Prediction using KNN Regressor with Hill Climbing Optimizer

Decision Tree Regressor

The decision tree regressor showed decent performance, only slightly lagging behind the optimized KNN. It achieved an RMSE of 31.9797 and an R-squared of 0.9976. The model predicted that a taxi trip with ID 6420 would take around 1,020 seconds, while the actual time was 910 seconds.

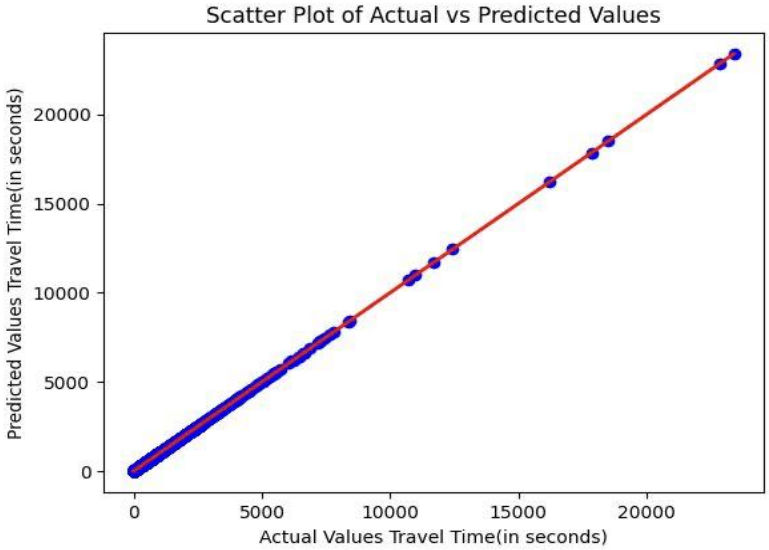

Fig. 2: Prediction using Decision Tree Regressor

Random Forest Regressor

The Random Forest Regressor (RFR), an ensemble method itself, outperformed both the KNN and decision tree models. It had the best results with an RMSE of 26.9539 and an R-squared of 0.9983. For instance, a random taxi trip with ID 9087 was accurately predicted to have a travel time of 645 seconds, matching the actual time.

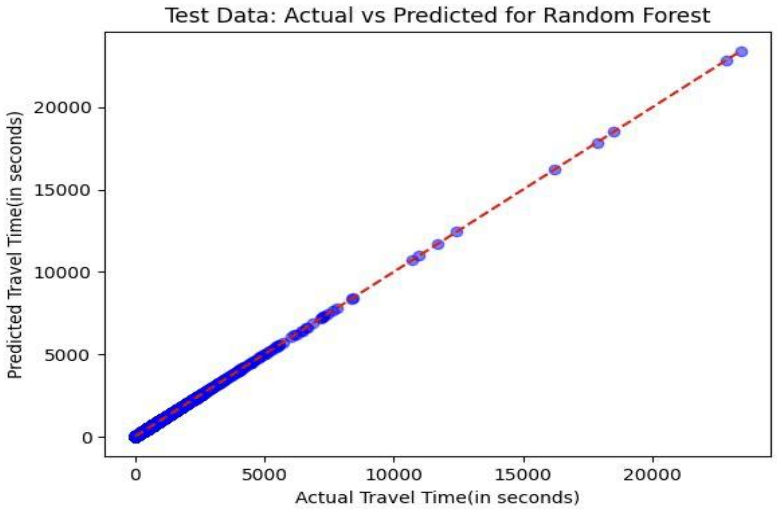

Fig. 3: Prediction using Random Forest Regressor

Stacking Ensemble Model

The stacking ensemble model, combining KNN, decision tree, and lasso regressors with a ridge meta-regressor, was the top-performing method. It achieved an extremely low RMSE of 0.004752 and an R-squared of 0.9999999999479441, indicating almost perfect prediction accuracy. For example, the model predicted a travel time of 855.0005744858202 seconds for a taxi trip with ID 6662, which was virtually identical to the actual time of 855 seconds.

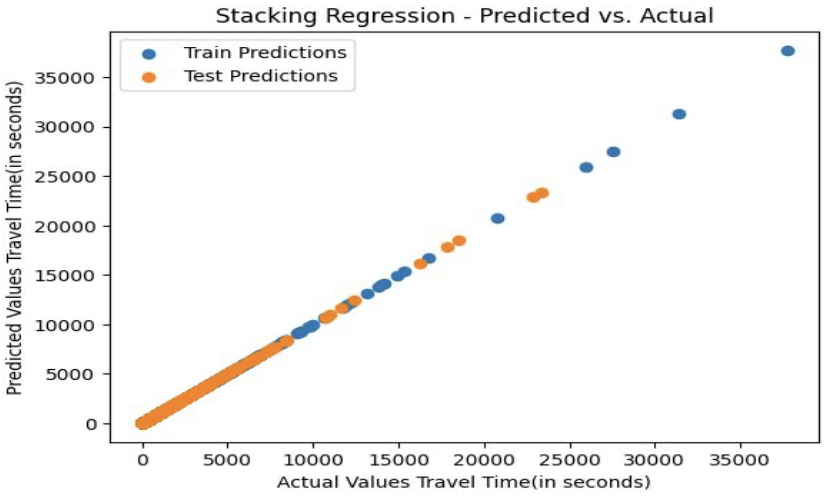

Fig. 4: Prediction using Stacking Ensemble Model

Table 1 presents a quantitative evaluation of model accuracy, showing the accuracy percentages for each model. This table allows for a direct comparison of how well each model predicts taxi travel times.

Table 1: Model Accuracy

MODELS	**ACCURACY(in percentage)**
KNN Regressor(with Hill Climb)	99.79
Decision Tree(with Hill Climb)	99.76
Random Forest	99.83
Stacking	99.999

Table 2 provides a qualitative comparison by displaying both actual and predicted travel times for selected taxi trips. This table offers a concrete demonstration of the models' effectiveness in real-world scenarios, illustrating how closely the predictions align with actual travel times.

Table 2: Predicted Vs Actual Travel Time of Taxi

Taxi Trip Id (Randomly Selected From Test Splits)	Predicted Travel Time (In Seconds)	Actual Travel Time (In Seconds)
3937	1330	1245
6420	1020	910
9087	645	645
6662	855	855

In conclusion, the stacking ensemble model was identified as the most effective approach for predicting taxi travel times in this study, outperforming the individual models significantly. This demonstrates the benefits of combining multiple models to utilize their strengths and minimize their respective weaknesses.

5. CONCLUSION

In this paper, we addressed the challenging problem of urban taxi travel time prediction using advanced spatiotemporal ensemble modeling. Building on a rich one-year dataset of taxi trajectories in New York city, we conducted an extensive benchmark study through several machine learning models in order to distill the most effective approach for this challenge. We explored KNN regressors optimized by hill climbing, Decision Trees, Random Forests, and a stacking ensemble models respectively.

Our test results show that while individual models, such as KNN Regressor, Decision Trees, and Random Forests, are good, the stacking ensemble has significantly better accuracy. The stacking ensemble fuses together many base models: Decision Trees, Lasso, and KNN with a Ridge meta-regressor fit through gradient descent. This has the feature of fusing together the diversity strength brought out by each base model, hence giving a much more accurate and robust prediction of travel time.

The superior performance of the stacking ensemble underlines the efficacy of ensemble methods in dealing with the spatiotemporal complexities by default in urban taxi data. Each of the base models has its own strengths: Decision Trees are very good at catching non-linear relationships, Lasso is very good with feature selection to counter overfitting, and KNN catches the local pattern of the data. When combined, the stacking ensemble model removes the pitfalls of a single model and successfully delivers improved predictive power. Further improvement comes in the form of the Ridge meta-regressor that tunes the final predictions by gradient descent so that they are very accurate.The results of this study suggest that ensemble learning models could greatly improve the predictive accuracy of urban taxi travel time, thereby making important contributions to a series of services in the field of urban mobility management and ride-hailing, ensuring that routes are maximized, service efficiency is improved, and user satisfaction is enhanced. By this nature, the stacking ensemble could further furnish greater strength to the predictive information for the improved decision-making process in an urban transportation system, which would, in turn, contribute to more effective and responsive transport solutions.

In conclusion, the stacking ensemble model has proven to be a powerful and efficient solution for predicting urban taxi travel time. Its ability to incorporate a variety of modeling strategies and yield accurate predictions highlights it as a fruitful way to overcome urban transport forecasting challenges.

6.FUTURE SCOPE

The most promising progress for the future is incorporating real-time traffic data into these prediction models. While we had used historical taxi trajectory data for travel time forecasting in this paper, the model can be significantly improved using real-time information about the traffic flow. For instance, given a massive congestion of traffic, road blocks, or even accidents, the current travel time may be affected enormously. It means that such dynamic factors in the models ensure that the prediction is sensitive to the instant changes of traffic conditions. Development of such real-time traffic feeds and part of them being associated with strong data pipelines would therefore ensure that models process and make use of information effectively towards improving both accuracy and timeliness of travel time forecasts.

Another exciting direction of further research is the application of genetic algorithms to optimize model parameters. For example, in combination with ways of optimizing the KNN model, one can apply hill climbing in Ridge regression and gradient descent. But now, there comes the fine-tuning of the genetic algorithm to obtain a more sophisticated approach to hyperparameter tuning. They represent the laws of natural selection in evolving a population of solutions over generations and can be generalized to wide explorations of hyperparameter space. Additionally, the genetic algorithms could also optimize the parameters of the base model in the Stacking ensemble. Therefore, it enables us to fine-tune

regularization parameters for Ridge Regression with genetic algorithms and thereby hopefully leads to better generalization capabilities of the model. This will ensure good performance and accuracy overall.
One of the important lines of works is providing future improvements in model generalization. Further integration with state-of-the-art optimization techniques such as a genetic algorithm could be able to enhance maximal adaptability to the next versions of the models under different conditions. The resulting ensemble models should not only be accurate for historical data but also preserve the accuracy for different scenarios and datasets. This definitely guarantees better generalization of a model with respect to robustness and applicability of a predictive system for a complex urban transport system.
In short, there are mainly two opportunities to enhance the estimation of travel time in urban taxis: integrating real-time traffic data and parameter tuning with a genetic algorithm. These will be even better to improve the efficiency and effectiveness of predictive models through enhancing accuracy, responsiveness, and adaptability of the model.

REFERENCES

[1] K. D. Kankanamge, Y. R. Witharanage, C. S. Withanage, M. Hansini, D. Lakmal and U. Thayasivam, "Taxi Trip Travel Time Prediction with Isolated XGBoost Regression," 2019 Moratuwa Engineering Research Conference (MERCon),Moratuwa, Sri Lanka, 2019, pp. 54-59. DOI: 10.1109/MERCon.2019.8818915.

[2] M Poongodi, MohitMalviya, Chahat Kumar, MounirHamdi, V Vijayakumar, Jamel Nebhen&HasanAlyamani, "New York City taxi trip duration prediction using MLP and XGBoost", International Journal of System Assurance Engineering and Management, vol.13, 2022, pp. 16-27. DOI: 10.1007/s13198-021-01130-x

[3] B. Qu, W. Yang, G. Cui and X. Wang, "Profitable Taxi Travel Route Recommendation Based on Big Taxi Trajectory Data," in IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 2, pp. 653-668, Feb. 2020. DOI: 10.1109/TITS.2019.2897776.

[4] J. -U. -R. Chughtai, I. U. Haq, O. Shafiq and M. Muneeb, "Travel Time Prediction Using Hybridized Deep Feature Space and Machine Learning Based Heterogeneous Ensemble," in IEEE Access, vol. 10, pp. 98127-98139, 2022.DOI: 10.1109/ACCESS.2022.3206384.

[5] A. A. Awan, A. Majid, R. Riaz, S. S. Rizvi and S. J. Kwon, "A Novel Deep Stacking-Based Ensemble Approach for Short-Term Traffic Speed Prediction," in IEEE Access, vol. 12, pp. 15222-15235, 2024. DOI: 10.1109/ACCESS.2024.3357749.

[6] Dataset: Meg Risdal. (2017). New York City Taxi Trip Duration. https://kaggle.com/competitions/nyc-taxi-trip-duration

[7] Chughtai, Jawad-ur-Rehman, IrfanulHaq, Saiful Islam, and Abdullah Gani, "A Heterogeneous Ensemble Approach for Travel Time Prediction Using Hybridized Feature Spaces and Support Vector Regression", Sensors, vol. 22, no. 24, 9735, 2022.

[8] B. Santhosh Narayanan and L. Shyamala, "Adopting Ensemble Learning and Machine Learning Techniques for Predictive Modeling in Traffic Data Analysis," 2024 3rd International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 2024, pp. 528-533, DOI: 10.1109/ICAAIC60222.2024.10575594

[9] J. Raiyn, "Real-time Intelligent Speed Adaptation in Heterogeneous Road Networks," 2021 International Symposium on Networks, Computers and Communications (ISNCC), Dubai, United Arab Emirates, 2021, pp. 1-7, DOI: 10.1109/ISNCC52172.2021.9615840.

[10] Putatunda, S., Laha, A.K., "Travel Time Prediction in Real time for GPS Taxi Data Streams and its Applications to Travel Safety", Human-Centric Intelligent Systems, vol. 3, 2023, pp. 381–401.

[11] Huang, H., Pouls, M., Meyer, A., Pauly, M. (2020), "Travel Time Prediction Using Tree-Based Ensembles". In: Lalla-Ruiz, E., Mes, M., Voß, S. (eds) Computational Logistics. ICCL 2020. Lecture Notes in Computer Science(), vol 12433. Springer, Cham. DOI: 10.1007/978-3-030-59747-4-27