AN EXPLAINABLE STACKED ENSEMBLE MODEL FOR STATIC ROUTE-FREE ESTIMATION OF TIME OF ARRIVAL

A PREPRINT

Sören Schleibaum
Department of Informatics
Clausthal University of Technology
Julius-Albert-Straße 4 in 38678 Clausthal-Zellerfeld, Germany
soeren.schleibaum@tu-clausthal.de

Jörg P. Müller
Department of Informatics
Clausthal University of Technology
Julius-Albert-Straße 4 in 38678 Clausthal-Zellerfeld, Germany
joerg.mueller@tu-clausthal.de

Monika Sester
Institute of Cartography and Geoinformatics
Leibniz University Hanover
Appelstraße 9a in 30167 Hanover, Germany
monika.sester@ikg.uni-hannover.de

ABSTRACT

To compare alternative taxi schedules and to compute them, as well as to provide insights into an upcoming taxi trip to drivers and passengers, the duration of a trip or its Estimated Time of Arrival (ETA) is predicted. To reach a high prediction precision, machine learning models for ETA are state of the art. One yet unexploited option to further increase prediction precision is to combine multiple ETA models into an ensemble. While an increase of prediction precision is likely, the main drawback is that the predictions made by such an ensemble become less transparent due to the sophisticated ensemble architecture. One option to remedy this drawback is to apply eXplainable Artificial Intelligence (XAI). The contribution of this paper is three-fold. First, we combine multiple machine learning models from our previous work for ETA into a two-level ensemble model—a stacked ensemble model—which on its own is novel; therefore, we can outperform previous state-of-the-art static route-free ETA approaches. Second, we apply existing XAI methods to explain the first- and second-level models of the ensemble. Third, we propose three joining methods for combining the first-level explanations with the second-level ones. Those joining methods enable us to explain stacked ensembles for regression tasks. An experimental evaluation shows that the ETA models correctly learned the importance of those input features driving the prediction.

Keywords Estimated Time of Arrival · Ensemble Learning · eXplainable Artificial Intelligence

1 Introduction

In intelligent transportation systems that coordinate a taxi fleet to, for instance, run a ridesharing service, the computation and comparison of taxi schedules is often supported by a component that estimates the duration of a trip or the time of arrival. To illustrate the problem of ETA, in Figure 1, we show two taxis Y and Z that aim to serve three passengers A, B, and C. Before serving the passengers, many alternative schedules have to be compared. Using estimated durations or an ETA component is common to avoid computing all routes in advance. As you can see, already in this small scenario, many alternative schedules can be taken into account.

Another use case for ETA is to provide insights into upcoming taxi trips, e.g. the time when a taxi will pick up a passenger or how long a trip will take for a driver/passenger. Approaches like [dE19; Kan+19; Li+18; SMS22] apply machine learning to achieve a high prediction precision. A promising option to further increase the prediction precision are ensemble models [Gan+21]; a special form of ensemble models is stacked ensembles in which the output of multiple

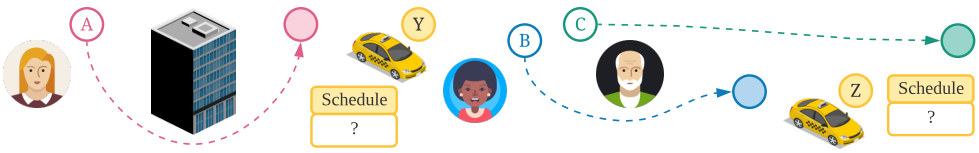


Figure 1: Sample scenario to motivate an ETA for the planning of taxi schedules.

first-level models is combined on the second level via another model, for example to the final estimation of the ETA for a taxi trip. The higher variety achieved via multiple models—potentially of different types—can better represent/learn the diversity of the data and potentially increase the prediction precision. One drawback is that the previously already hard-to-explain models now become even less transparent through their combination. This is caused by enhancing the architecture to a more sophisticated one with, among others, interactions between the various first-level and second level models. One option to remedy this drawback is to apply XAI-methods like Shapley Additive Explanations (SHAP), which aim to explain complex non-linear machine learning models like neural networks.

Related Work. To estimate the time of arrival of a trip, many previous approaches like [Li+18] used the metadata of a route or the route itself and are, therefore, *route-based* approaches. However, we focus on *route-free* ETA because we do not depend on a road network, historic data such as trajectories on certain roads, or the relatively complex calculation of routes. Al-Abbasi, Ghosh, and Aggarwal [AGA19] and Jindal et al. [Jin+18] develop route-free ETA models as part of a larger service like ridesharing and even though the authors do not prioritize ETA, achieve a remarkable prediction precision; others like [dE19; Kan+19; Li+18; SMS22] focus solely on ETA.

As regards related work that explains ensembles, Silva, Fernandes, and Cardoso [SFC19] propose three ensemble models for different datasets from medicine and finance to generate joint explanations for the generated classifications. They use three heterogeneous first-level models—a Scorecard, a Random Forest (RF), and a neural network—and generate local explanations. The explanations are evaluated using their previously defined framework called the *three C's of interpretability*. Another similar work is [KRM21]; Kallipolitis et al. ensemble therein several variants of the EfficientNet—see [TL19]—with a fully connected layer to identify forms of cancer in whole slide images—a type of images widely used in medicine. Additionally, the authors apply the XAI-methods Grad-CAM and Guided Grad-CAM to jointly explain their classification. While they achieve a remarkable classification accuracy, they are also able to combine the explanations of the first-level ensemble models to a joint local explanation in form of a heatmap. Rozemberczki and Sarkar [RS21] design an algorithm, which is based on the XAI-method SHAP, to build and improve ensembles for binary classification. There, an approximation of SHAP is used as an importance metric or valuation score for the models of an ensemble—models with low scores might be excluded from the final ensemble.

The approach by Utkin and Konstantinov [UK21] combines multiple explanations—generated via the XAI method SHAP—in a classification or regression task to reduce the computational expenses required by SHAP. Therefore, Utkin and Konstantinov [UK21] propose three alternative realizations of their method and compare those with multiple black-box models. Note that their work neither uses ETA nor explains their ensemble.

Research Gap and Aim. While many previous static route-free ETA approaches [AGA19; Jin+18; dE19; Kan+19; Li+18; SMS22] achieved a high prediction precision, none tried to use a stacked ensemble; moreover, none of the previous approaches considered the explainability of their models. As regards the explanation of ensembles, we observed that very few works combine explanations for different machine learning models that contribute to a prediction. We found even fewer—a total of only three—related works that generate explanations for ensembles of machine learning models. In contrast to the works that locally explain ensemble models—[SFC19; KRM21; RS21]—we focus on the explanation of ensemble models for regression. While for [SFC19] it is not clear whether the same or different inputs are used for the first-level models, [KRM21; RS21] use the same input for all first-level models; we will ensemble machine learning models with (partly) different feature sets.

Based on the related work, we conclude that combining multiple models in an ensemble to perform ETA and explaining both—the classical single-level models and the stacked ensemble—is an open research gap. Therefore, our research aim is threefold. First, we plan to apply XAI methods to explain ETA locally. Second, we propose an ensemble model for ETA and consequently increase the prediction precision further. Third, we also aim to explain this model to provide a basis for making predictions more transparent to users of ETA.

Outline. After providing some preliminaries in Section 2, we explain the dataset and methods used throughout this paper in Section 3. Subsequently, we take multiple machine learning models from our previous work—[SMS22]—for

ETA and combine them into a heterogeneous—different model types like tree- and neural network-based—and stacked ensemble model of two levels (see Section 4). In Section 5.1, we apply existing XAI methods to explain the first-level models and later combine the explanations on the second level to generate joint explanations of the ensemble in Section 5.2. In Section 6, we discuss our results with respect to the research aims and point out limitations as well as future work; we conclude the paper in Section 7. The source code used in this paper can be accessed online via [Sch22].

2 Preliminaries

Estimating the Time of Arrival. Estimating the duration or time of arrival for a planned trip is a classical regression problem $f(X) = \hat{y}$, in which we aim to find a function f that minimizes the difference between $\hat{y}$ and the real values y; while $y \in \mathbb{R}$, each X_i of the dataset contains multiple or m features so that $X_i \in \mathbb{R}^m$. Because we consider static route-free ETA, information about the route—like the number of turns on the route—or information not known before a trip starts—like a traffic accident on the route happening after the start of a trip—are excluded from X. We only include features like the degree-based coordinates of the pickup/dropoff location or the start time of the trip in X.

Ensemble Learning. Several ensemble techniques like boosting or stacking are possible. In this paper, we use a two-level stacking-based approach that combines the outputs of the first-level models on the second level with another model. As shown in Figure 2, each of the first-level models receives an input like X_{L1-1} derived from the original input data X. The inputs $X_{L1-1}, X_{L1-2}, \ldots, X_{L1-n}$ of the n first-level models can be different feature sets. The outputs of the first-level models ($\hat{y}_{L1-1}, \hat{y}_{L1-2}, \ldots, \hat{y}_{L1-n}$) are fed into the second-level model which outputs $\hat{y}_{L2}$ or the ETA for a given trip. To improve the prediction precision further, ensembles can be constructed of heterogeneous models with different types like a tree- and neural network-based one, which can increase the diversity of the ensemble.

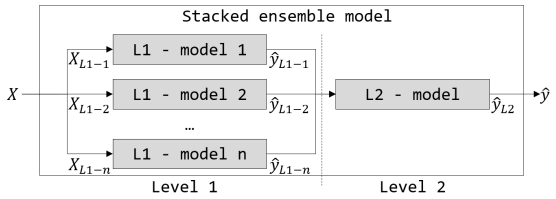


Figure 2: The architecture of a stacked ensemble with two levels.

Explanations. Throughout this paper, we consider an explanation as a vector e of length $|X|$ assigning a value to each input feature $x \in X$ for a given prediction $\hat{y}$: $e = e_1, e_2, \cdots, e_x$; so, $e \in \mathbb{R}^{|X|}$. As we consider the prediction as given, we explain the model post hoc. To differentiate the explanations in an ensemble, we add a model superscript to an explanation: e^M.

3 Methodology

3.1 Dataset

Selection and Features. We select two datasets: the New York City Yellow taxi trip data from 2015 and 2016—see [New19]—and one recorded in Washington DC in 2017—see [Kag19]. We select the former because it was used several times for demonstrating an ETA approach and it is the dataset mainly used in this paper. We additionally include the Washington DC dataset to increase the generalizability of our experiment as regards the usage of ensembles to increase the prediction precision. For both datasets, we rely on the feature engineering described by Schleibaum, Müller, and Sester [SMS22], which makes use of or enhances the dataset by the following features: the location-based ones with (1) the pickup/dropoff as degree-based coordinates and (2) the indices of a 50 meter square grid as an alternative representation. To represent the start time of a trip, (3) the *month*, (4) *week*, (5) *weekday*, and (6) the indices of a *5-minute time-bin*, which represents the hour and minute, are used. Moreover, we use (7) the *temperature* at the hour a trip starts and calculate (8) the *Haversine distance* between pickup and dropoff location.

Outlier Removal. For removing outliers, we also use the criteria from Schleibaum, Müller, and Sester [SMS22] and the description of the following method partly reproduces their wording. Overall, around three percent of the trips from the New York City dataset and around 19 percent from the Washington DC dataset are filtered out. A trip can be an outlier because one of its locations is not in the area studied, which is shown in Figure 4, or not in a district like

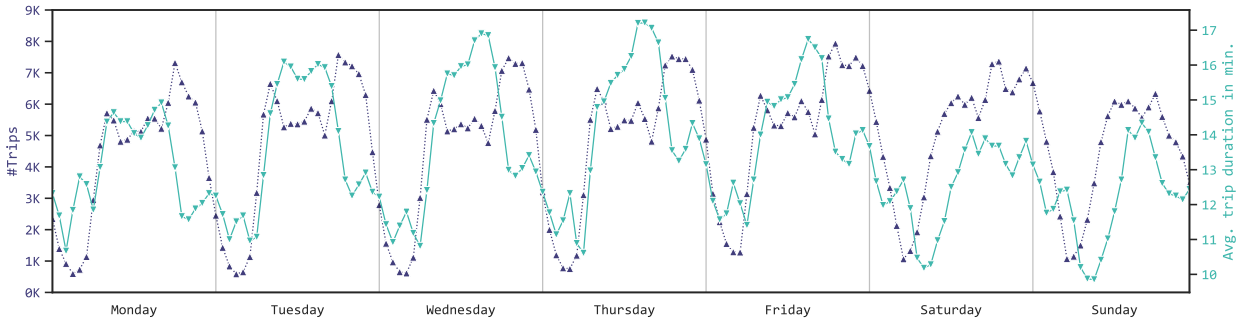
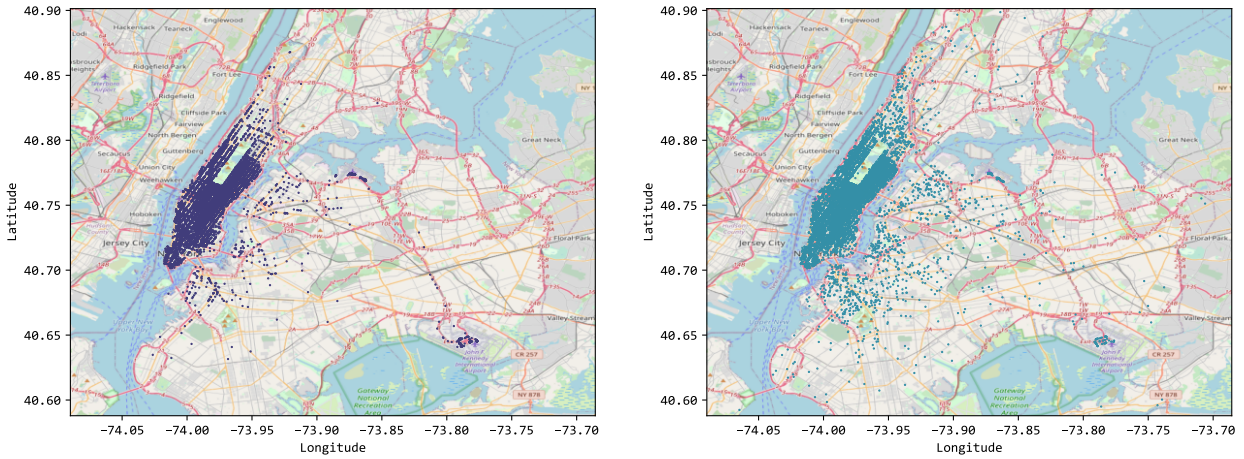


Figure 3: Distribution of the number of trips and their average duration over the week in New York City.



(a) (b)

Figure 4: Distribution of the pickup (a) and dropoff (b) locations for randomly selected trips from the training data of the New York City dataset.

erroneously being recorded in the Hudson River. Moreover, a trip's reported duration could be unreasonable low or high, or could not correlate logically with the distance between pickup and dropoff locations; we also remove trips with a distance of zero. Compared to other papers, the criteria are relatively moderate and, therefore, the comparison to approaches not reproduced is fairer.

Characteristics. To better understand the data, in Figure 3, we visualize the number of trips and the average trip duration per weekday for the New York City dataset. The number of trips is relatively low during the early morning; during weekdays, it has one peak at around 8 am and another one at around 5 pm. As expected, during the weekend, the decrease in the number of trips occurs later and the morning peak does not exist. In general, the average duration of a trip highly correlates with the number of trips.

To show the area considered and to get a better understanding of the distributions of pickup and dropoff locations, we visualize both in Figure 4. As expected for the Yellow taxis in New York City, the vast majority of trips start in Manhattan; most of the trips not starting in Manhattan begin at John F. Kennedy Airport. As regards the dropoff locations, the general behavior is similar, but more trips end outside of Manhattan.

3.2 Estimated Time of Arrival Models

We take the three ETA models proposed by Schleibaum, Müller, and Sester [SMS22] and their hyperparameters as our first-level models; we choose these because, with a bagging-, a boosting- and a neural network-based model, they provide a good basis for a heterogeneous ensemble. As alternatives for the second-level model, we consider the same machine learning methods and add a relatively simple Multiple Linear Regression (MLR). As regards the main dataset or the one from New York City, we use 1M trips for training and validation from 2015 and another 250K from 2016

for testing. For the dataset from Washington DC we use less or 600K trips for training and validation, and another 50K for testing. As training data for the second-level models, we use the predictions of the first-level models on the validation data and use the same test data as before. We do not use the same training data twice or for the first- and second-level models to reduce overfitting. We do not tune the hyperparameters of the second-level models and, therefore, consider not using a validation dataset as fine. Except for the MLR, which does not have any hyperparameters, for the second-level models, we use the same hyperparameters as for the first-level models. The only difference is that we decrease the number of trees for the RF and XGBoost from 300 to 100 and the number of hidden layers for the neural network-based model from four to two; we choose smaller models compared to the first-level models because the number of input features or the variety of the model input is reduced substantially.

3.3 Selection of XAI Methods

To demonstrate our approach, we select two commonly used XAI methods—Local Interpretable Model-agnostic Explanations (LIME) and SHAP—which are described in the following. We chose these XAI methods because both are model-agnostic and can, therefore, be applied to all models of the heterogeneous ensemble. Moreover, both create local post-hoc explanations that can be used to explain to ETA users like taxi drivers and passengers. While all first-level models are explained via the XAI methods, only the best-performing second-level model will be explained.

Local Interpretable Model-agnostic Explanations. Ribeiro, Singh, and Guestrin [RSG16] present LIME, which explains predictions based on a linear surrogate model by minimizing two aspects: the goodness of the local approximation of the interpreted model in the observations neighborhood and the complexity of the surrogate model. This post-hoc XAI method outputs a vector- or graphics-based explanation that is visualized differently by software libraries. The main formula presented by Ribeiro, Singh, and Guestrin [RSG16] is:

$$\xi(x) = \underset{g \in G}{\arg\min}\, \mathcal{L}(f, g, \pi_x) + \Omega(g) \tag{1}$$

The importance of feature x from a sample is extracted from the surrogate model g from all possible surrogate models G that describe the black box model f and the neighborhood of x—denoted as π_x—best, while also minimizing the complexity of the surrogate model $\Omega(g)$.

Shapley Additive Explanations. Another model-agnostic XAI-method—SHAP—was proposed by Lundberg and Lee [LL17]. It is able to generate local explanations for a given sample or the feature importance in this sample. Therefore, SHAP utilizes the famous Shapley values from cooperative game theory. More concrete, Lundberg and Lee [LL17] presented the following formula:

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} [f_{S \cup \{i\}}(x_{S \cup \{i\}}) - f_S(x_S)] \tag{2}$$

Here, the contribution of a feature ϕ_i is estimated by iterating over all feature subsets S of the feature set F without the feature i. The fraction in the sum weights the difference between the output of the model to be explained—represented by the function f—with and without i or the contribution of i.

3.4 Evaluation

Prediction Precision of ETA models. Similar to Schleibaum, Müller, and Sester [SMS22], we apply three evaluation metrics common for regression tasks: (1) The Mean Absolute Error ($MAE = {}^1/_N \sum_i |y_i - \hat{y}_i|$), which in our case returns the average error per trip in seconds, (2) the Mean Relative Error ($MRE = \sum_i |y_i - \hat{y}_i| / \sum_i y_i$), and (3) the Mean Absolute Percentage Error ($MAPE = {}^1/_N \sum_i |(y_i - \hat{y}_i)/y_i|$), which is robust to outliers. Because the latter two produce percentage values, they are also relatively easy to understand and put the error in perspective to a trip's duration.

Scenarios for Explanations. To demonstrate and evaluate our explanation approach, we randomly select ten trips from the New York City test data for four scenarios. Each scenario has two opposing characteristics that are described in the following along with the scenarios:

- SC1 - *Off city-center vs. city-center*: We compare trips that start outside of the city-center—a rectangle with the bottom left at coordinate (40.7975, -73.9619) and top right at (40.8186, -73.9356)—with those that do start in the city-center—a rectangle with the bottom left at (40.7361, -73.9980) and top right at (40.7644, -73.9770).
- SC2 - *Night-time vs. rush-hour*: Here, we choose some trips that start early in the morning—3 am to 5 am—and some that start during the NYC rush-hour—4 pm to 6 pm.

SC3 - *Low vs. high temperature*: In this scenario, we compare trips with a relatively low temperature—trips that are in the 0.25 quartile and not in the 0.1 decile—with trips that took place at a high temperature—trips that are in the 0.75 quartile and not in the 0.9 decile.

SC4 - *Low vs. high distance*: We select trips with a relatively high/low distance between pickup and dropoff locations—we use the same boundaries as for SC3 for the feature *Haversine distance* to select the trips.

4 Models for Estimating the Time of Arrival

We take the three ETA models proposed by Schleibaum, Müller, and Sester [SMS22] as our first-level models as well as their hyperparameters, which have been chosen via Bayesian optimization. The first model is an RF-based one (*L1-RF*) with 300 trees and a maximum tree depth of 89; the number of maximal features per node is chosen automatically and the minimum number of samples per leaf and split are set to four. The second model is based on XGBoost (*L1-XGBoost*) and also consists of 300 trees, but has a maximum tree depth of eleven; the minimum weights of instances needed in a child is set to seven, the subsample ratio of the training data per tree to one, the minimum loss reduction required for making a further partition on a child to zero, and the subsample ratio of features for a tree to one. The third model is based on a Fully-Connected Feedforward Neural Network (FCNN) (*L1-FCNN*) with four hidden layers and 300, 150, 50, and 25 corresponding neurons. Similar to Schleibaum, Müller, and Sester [SMS22], we set the batch size to 128, the learning rate to 0.001, train the network for 25 epochs, and select the best model along the training to minimize overfitting.

Besides the first-level models, we propose four second-level models or ensembles. Because we use all three first-level models for each ensemble, all four ensembles are heterogeneous. The first second-level model is a relatively simple one based on an MLR referred to as *L2-MLR*. The second one is based on RF (*L2-RF*) with 100 trees in the forest; for the third, XGBoost-based model or *L2-XGBoost*, we chose the same number of trees. For both—*L2-RF* and *L2-XGBoost*—we do not train the hyperparameters as these methods usually achieve a high prediction precision without any hyperparameter tuning. The fourth ensemble (*L2-FCNN*) combines the output of the first-level models via a FCNN with two fully-connected hidden layers—50 and 25 corresponding neurons—and otherwise similar hyperparameters to the *L1-FCNN*.

Table 1 shows that for the New York City dataset the MAE or average prediction error in seconds per trip is around 178 seconds for the L1-FCNN and a couple of seconds higher for the L1-RF and L1-XGBoost. The results for the other evaluation metrics—the MRE and MAPE—are similar and put the prediction error in perspective to the trip duration. As regards the New York City dataset and the second-level models, all models are able to outperform the first-level models as regards the MAE and MRE. Only the L2-FCNN is able to outperform all first-level models in all evaluation metrics with an MAE of 169 seconds or an MRE of around 20 percent. Interestingly, the L2-MLR achieves a remarkable prediction precision which is better than the one of the L2-RF and similar to L2-XGBoost. For the models trained and tested on the Washington DC dataset, we observe that on the first level the L1-FCNN with an Mean Absolute Error (MAE) of around 169 seconds is able to outperform the L1-RF by 10 seconds and the L1-XGBoost by 20 seconds. As regards the second-level models, we observe that except for the L2-RF all models achieve a remarkable prediction precision. In contrast to the models trained and tested on the New York City dataset, none of the second-level models is able to outperform the Mean Absolute Percentage Error (MAPE) achieved by the L1-FCNN.

Table 1: Comparison of our ETA models of the first and second level based on different evaluation metrics

	Dataset	New York City			Washington DC		
	Evaluation metric	MAE [seconds]	MRE	MAPE	MAE [seconds]	MRE	MAPE
Level 1	L1-RF	180.694	0.2158	27.8689	179.5912	0.2373	30.1512
	L1-XGBoost	183.4192	0.219	27.1137	190.2613	0.2514	30.4033
	L1-FCNN	178.2321	0.2129	23.7561	169.8152	0.2244	**24.372**
Level 2	L2-MLR	172.2439	0.2057	25.2758	171.178	0.2262	27.1985
	L2-RF	183.2319	0.2188	26.9828	183.7377	0.2428	29.5762
	L2-XGBoost	173.6526	0.2074	25.3077	172.7287	0.2283	27.5419
	L2-FCNN*	**169.4285**	**0.2023**	**22.9121**	**167.9959**	**0.222**	24.6133

*This prediction precision is better than the one presented by Schleibaum, Müller, and Sester [SMS22]

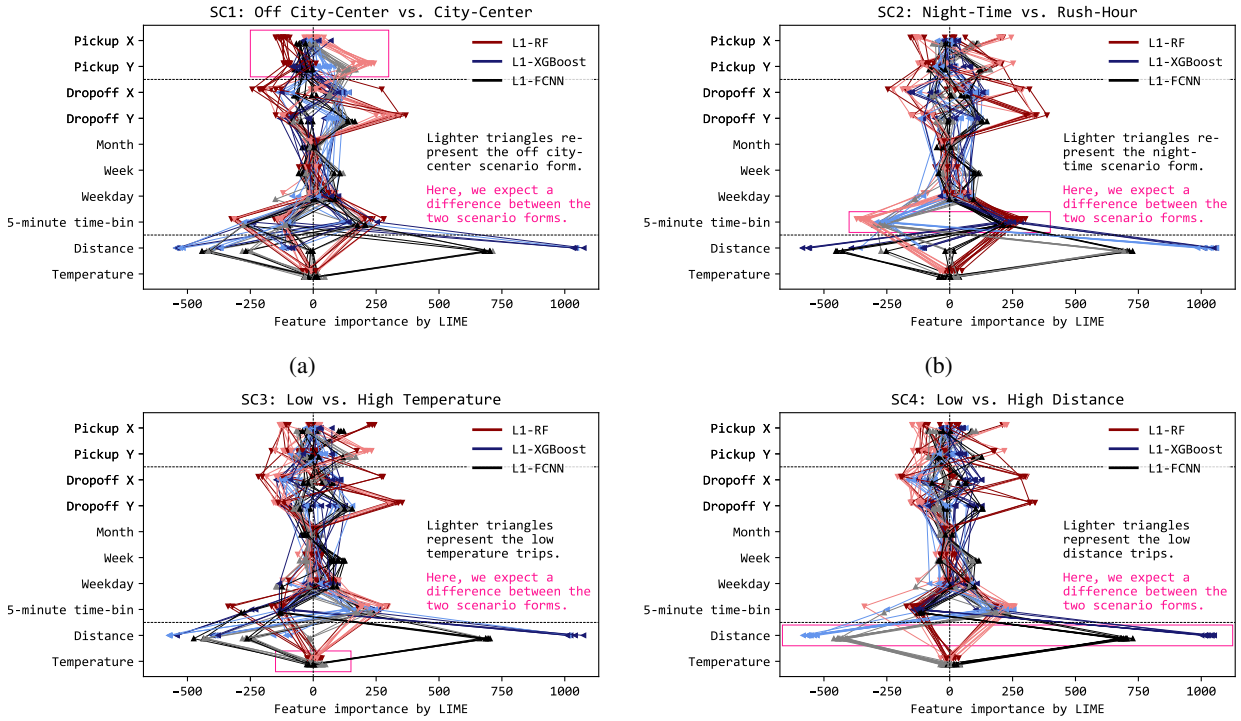


Figure 5: Local feature importance via LIME per feature of the samples in the scenarios for the first-level models; each plot refers to one scenario, (a) for SC1, (b) for SC2, (c) for SC3, and (d) for SC4; the ten trips with an expected lower influence are marked with lighter triangles—the ones with an expected higher influence with less light triangles; each line connects the feature importances for one trip along the various features used by the corresponding model.

5 Explaining the Estimated Time of Arrival Models

5.1 First-Level Model Explanation

In Figure 5, we visualize the explanations generated by LIME per scenario, its characteristics, feature, and ETA model. Each triangle represents the importance of a feature for one trip from a scenario—the lighter triangles are from the first or 'lower' characteristic of the scenario. The lines connect the feature importances for one sample or trip. The concrete values can be interpreted as follows: for instance, the left-most triangle in SC1—(a) of Figure 5—has a value of around -575. This refers to a relatively strong negative influence of the concrete distance value on the corresponding estimated duration of the trip. This most likely refers to a trip with a small distance between the pickup and dropoff locations.

In SC2—top right—and SC4—bottom right—the two characteristics of each scenario are visually separated for the features that constitute the scenario—the 5-minute time-bin for SC2 and the distance for SC4. As expected, the other features are not separable because they are more or less randomly distributed over the space of each feature—not completely random as some features might slightly correlate with the feature of interest. Moreover, the separation is in the correct order, meaning that the 'lower' characteristic also has lower feature importance than the 'higher' characteristic of the scenario. Therefore, the ETA models appear to have properly learned the expected behavior in these scenarios. Even though for the majority of trips in SC1 the difference between the pickup off city-center and in city-center is learned, some trips of the 'lower' and 'higher' characteristics interfere. For instance, for *pickup X*, the *L1-FCNN* feature importances of both scenario characteristics overlap. As regards SC3, we observe that the reported feature importances for the temperature are low or close to zero. While this could indicate that the ETA models have not learned the underlying pattern, similarly to Schleibaum, Müller, and Sester [SMS22], we argue that the overall feature contribution of the weather or temperature is low.

While the concrete values or feature contributions generated via SHAP differ from the feature importances of LIME, we observe similar results in Figure 6: For SC2 and SC4 the two characteristics of the scenarios are visually separated only by the feature of interest; the separation for SC1 and SC3 is not clear.

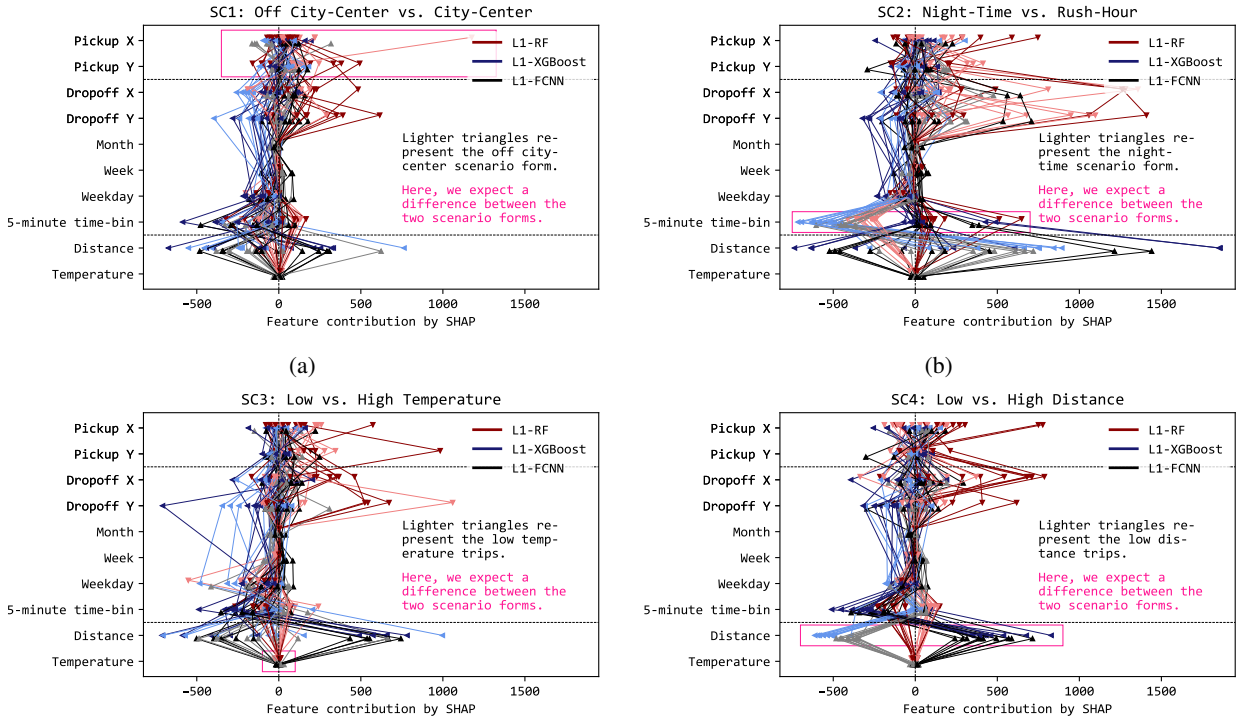


Figure 6: Explanations via SHAP per feature, sample, and scenario for the first-level models; each subfigure refers to one scenario, (a) for SC1, (b) for SC2, (c) for SC3, and (d) for SC4; the ten trips with an expected lower influence are marked with lighter triangles—the ones with an expected higher influence with fewer light triangles.

5.2 Explaining the Ensemble

In the following, we describe three relatively simple but novel methods to join the explanations from the first and second level. To be able to compare the outputs for all three joining methods and samples in a better way, we first normalize the second-level explanation per sample so that they sum up to one. The joining methods (JMs) are:

(JM1 - *Adding a Dimension*) Here, we simply output both, the first and second-level explanations—which is meant by 'additional dimension'—simultaneously. Therefore, we weigh the feature contribution or importance of a feature on the first level with the contribution or importance of the first-level prediction in the second level. Consequently, we join the first and second-level explanations without losing any information.

(JM2 - *Basic Join of the Contributions*) To determine the contribution or importance of a feature, we compute the dot product between the vector that contains the contribution or importance of that feature for each first-level model and the vector that contains the contribution or importance of each first-level model on the second level; the product is then the joined contribution or importance of that feature for a given sample.

(JM3 - *Diversifying the Contributions*) Here, we use the result from JM2 as a basis and define a threshold α—which is the mean value of the distribution or influence of each first-level model on the second level or e.g. $1/3$ with our three first-level models. Next, every value below that threshold is shrunk by a value β to be increased by the collected value in the next step. In case values cannot be shrunk by β, because they would otherwise become negative, only the difference to zero is used and redistributed to the values above α. Thus, the second-level influence is diversified. In the following, we set β to 0.5 or relatively high as the number of first-level models is only three.

All three joining methods are compared to a *Baseline (BL)* method. This method generates explanations by explaining a function that wraps the whole ensemble. Within this function, features that are an alternative representation of other features, like the X-index of a 50-meter square grid, are also generated within that function from the corresponding base feature, i.e. the latitude of the pickup location.

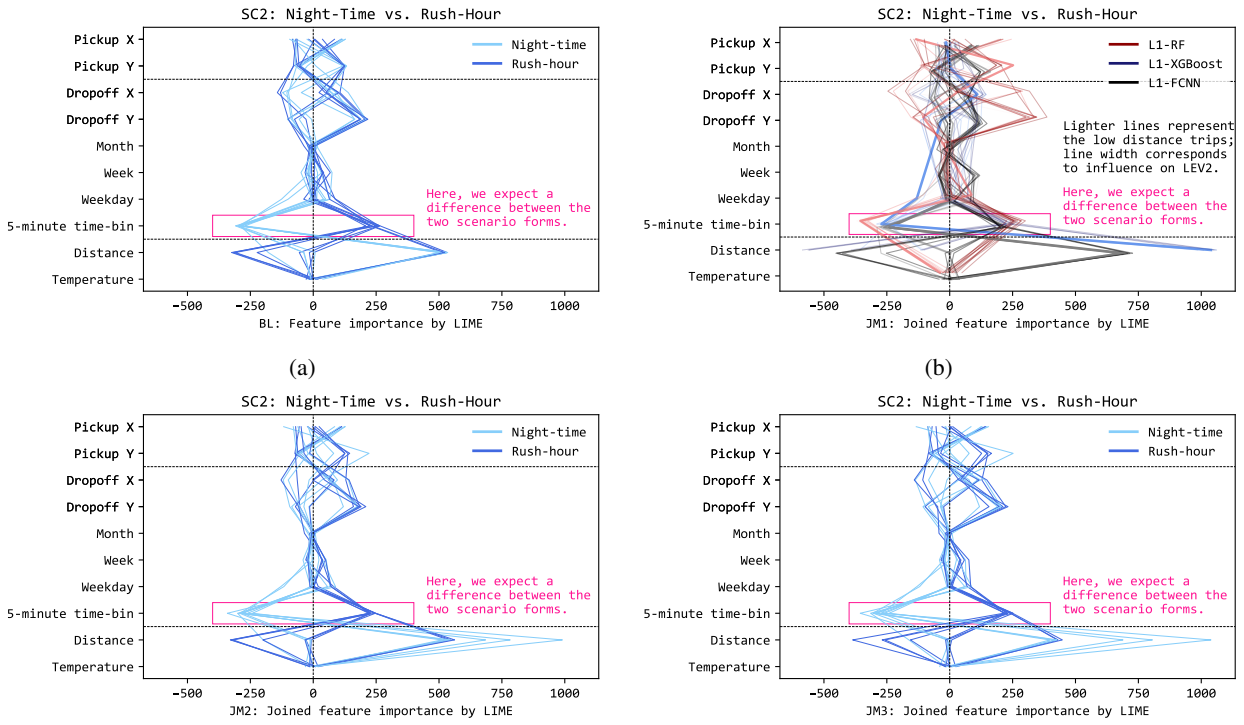


Figure 7: Joined local feature importance via LIME for each feature of the samples in the SC2 for JM1 (b), JM2 (c), and JM3 (d) compared to the BL (a); each line connects the feature importances for one trip.

In Figure 7, we show the feature importance for LIME joined via all three proposed joining methods as well as for the BL (top left). Similar to previous findings, in each graphic of Figure 7, for each trip and method a line—or three for JM1—is shown, this time without triangles. As we only want to demonstrate the joining methods, we omit all scenarios except for SC2 here but show the corresponding graphics for SC1, SC3, and SC4 in Figure 10 in the Supplementary Material.

In Figure 7, the difference between JM1 and BL/JM2/JM3 is obvious: JM1 shows much more information, including the proof that all three first-level models are used by the *L2-FCNN*. Even though the two scenario characteristics have the expected difference at the 5-minute time-bin, verifying the difference among the first-level models is hard for JM1. As regards JM2, we observe a relatively high difference to the BL joining method as for instance visible in the feature importances of the 5-minute time-bin or the distance of the night-time characteristic of the scenario. As expected, the JM3 makes the smaller feature importance values smaller and the larger ones larger, thereby diversifying the feature importances along all features slightly.

When applying the joining methods to the SHAP values for the same scenario, as shown in Figure 8, we observe similar results. While the difference between the night-time and rush-hour characteristic of SC2 is visible for all joining methods, this time JM2 and JM3 in general reduce the feature importances. This is in contrast to the explanations generated via LIME.

In Figure 9, we visualize the Shapley values for the features used to build the scenarios via the joining methods JM2 and JM3 per scenario and their two opposing characteristics—'lower' and 'higher'—to further investigate the differences to the BL; JM1 is omitted in the figure as it is hard to compare in the visualized regard. As expected, the Shapley values generated via JM2 and JM3 do not vary much compared to the BL; like for the 5-minute time-bin and SC2H, JM2 and JM3 slightly change the Shapley values in the positive direction. For the distance and SC4L, the Shapley values are moved in the opposite direction. In general, the difference expected in the scenarios gets slightly smaller but is still clearly shown. A similar figure for LIME can be found in Figure 12 the Supplementary Material.

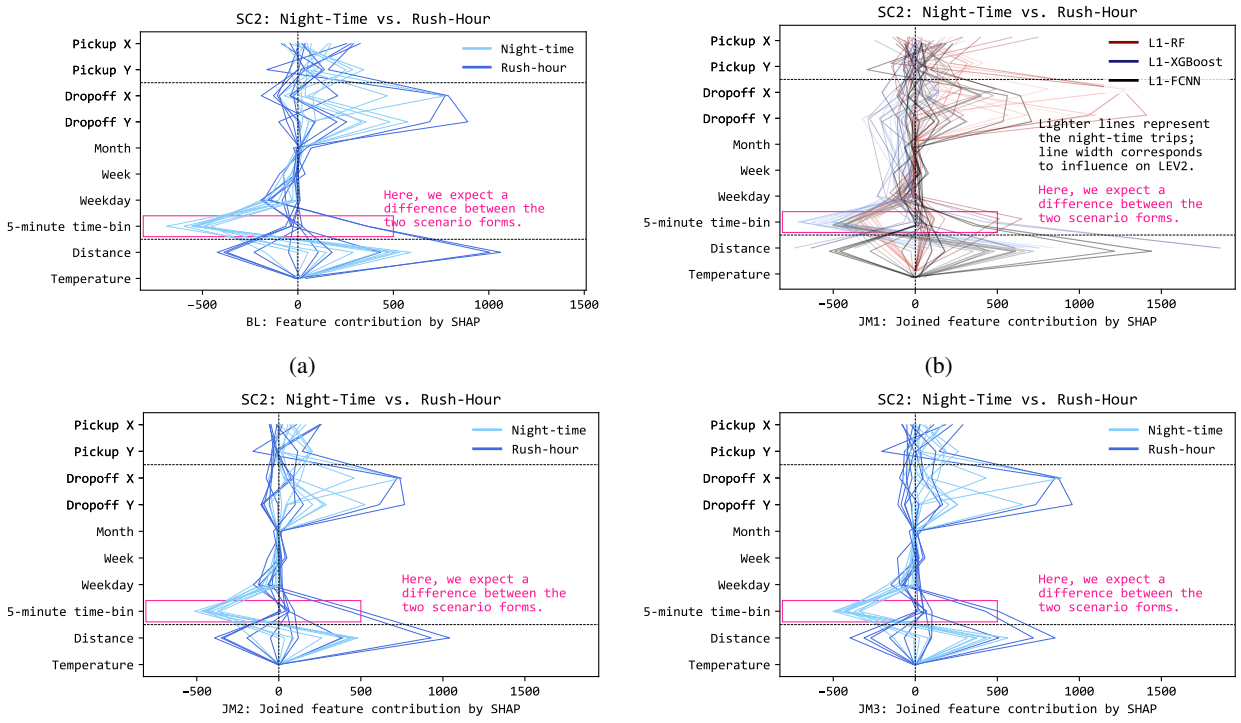


Figure 8: Joined local feature contribution via SHAP for each feature of the samples in SC2 for JM1 (b), JM2 (b), and JM3 (b) compared to the BL (a); each line connects the feature importances for one trip.

6 Discussion

Ensemble for ETA. We tried multiple alternatives to combine an RF, XGBoost, and FCNN model output via another model. Even though multiple models, like an MLR-based one, achieved a high prediction precision on the New York City dataset, only an FCNN-based one was able to outperform our previous models from Schleibaum, Müller, and Sester [SMS22] in all evaluation metrics. Interestingly, for the Washington DC dataset the results were not that clear: while the FCNN-based ensemble performed better than the first-level FCNN-based model as regards the MAE and Mean Relative Error (MRE), for the MAPE the observed pattern is the opposite. We believe that this is caused by three reasons. First, the feature selection and hyperparameter tuning were done for the New York City dataset and consequently not optimal for the Washington DC dataset. Second, the Washington DC dataset with around 650K trips used is much smaller than the New York City one with 1.25M trips causing the second-level models to have much fewer data to be trained on. Third, the gap between the performance of the three first-level models is much closer for the models trained on the New York City dataset compared to those first-level models trained on the Washington DC dataset. Therefore, we assume that a better performing XGBoost model or excluding it from the ensemble could improve the prediction precision further. As we already outperformed the approaches of [AGA19; dE19; Jin+18] in our previous work—[SMS22]— we consider the usage of a stacked heterogeneous ensemble as an effective method to increase the prediction precision for static route-free ETA. With the dataset considered, we reduced the MAE by nine seconds to around 169 seconds per trip on average; both MRE and MAPE were reduced by around one percentage point.

Explaining First-Level ETA Models. We applied the two model-agnostic XAI methods LIME and SHAP to evaluate and explain our first-level ETA models post-hoc and locally. In SC2 and SC4, we could show that all three models learned the expected behavior. For SC3, all ETA models that include the temperature consistently learned a low influence of the temperature on the ETA. As described previously, this is most likely caused by the low influence of weather-related data in general rather than a pattern not properly learned by our ETA models. Even though Schleibaum, Müller, and Sester [SMS22] showed the positive influence of including the feature in the models on the prediction precision, as their general influence is low, the explanation or feature importance value assigned is not very meaningful. In case someone focuses on explainable ETA models, removing the temperature or the month feature might be worth considering. As regards the SC1, we could show that information like the pickup location, which is encoded into

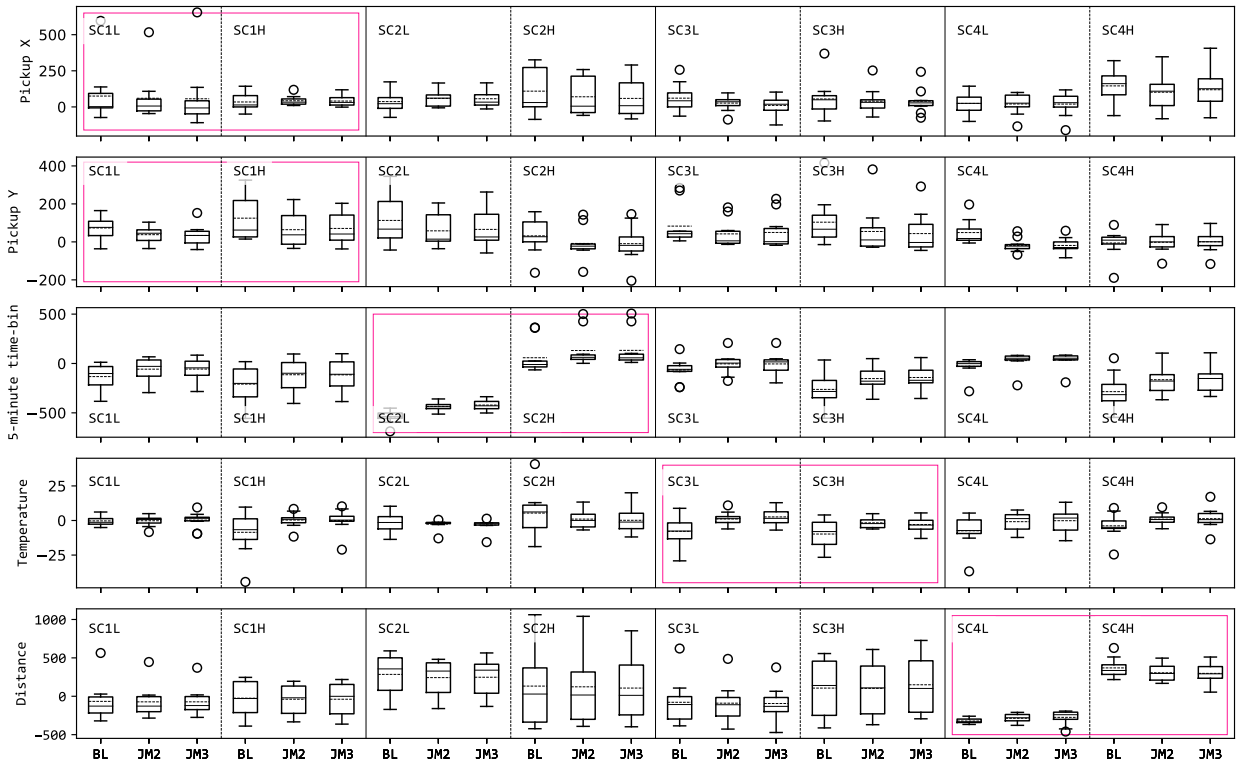


Figure 9: Box plot per feature—those affected by the scenarios like the 5-minute time-bin for SC2—and joining method compared to the BL for each scenario in its lower (e.g. *SC1L*) and its higher (e.g. *SC1H*) characteristic; the dashed lines in the boxes of each box plot are the mean values and the pink rectangles mark the features of the scenarios.

multiple and, therefore, correlating features, is hard to explain via LIME and SHAP. We observed that the explanations produced by LIME are more separated in our scenarios compared to those of SHAP. As the focus of our work is not to compare LIME and SHAP, we refer the interested reader to the work from Belle and Papantonis [BP21]; but in general, LIME has a relatively low runtime and SHAP the advantage of producing additive explanations.

Joint Explanation of Ensembles for ETA. We presented three relatively simple methods for joining the first and second-level explanations of an ensemble to generate a joint explanation. The main advantage and at the same time drawback of JM1 (*Adding a Dimension*) is that more information or all explanations are shown. When—as we did—multiple trips are shown in one graphic, we assume that it is harder to understand, but on the other hand, especially when only one trip is visualized, this provides additional insights not provided by JM2 and JM3. For instance, it might be interesting to see if different first-level models disagree on a feature's importance for a specific sample, how strong the influence of each of the models is, and if some relation was not learned correctly. For such a case, a smarter choice of colors or an alternative to the used line plots could improve the understandability. However, with respect to larger ensembles or those that have more or many first-level models, the less dense explanation created by JM1 might be confusing or not understandable anymore.

As regards JM2 (*Basic Join of the Contributions*), we observed an unexpectedly high difference to the BL method. We believe that this difference is at least partly caused by correlated features such as the pickup latitude and longitude. Nevertheless, the general direction of the feature importances is similar. Even though the XAI methods might not be built for correlated features, especially in stacked ensembles and practice, correlated features exist. Interestingly, when considering LIME, the larger values are made larger; for SHAP, the effect is the opposite. For JM3 (*Diversifying the Contributions*), we observed the same but slightly stronger effect. In contrast to JM1, JM2, and the BL, JM3 has a hyperparameter that has to be chosen by the user, which makes this method more complicated to apply.

Interestingly, none of the related work—[SFC19; KRM21; RS21]—has used the BL method to generate an explanation or compare their explanation to it. As we did not find other literature as regards explaining ensembles, we assume that our work presents three novel joining methods. While the general concept we applied for creating a joint explanation of a stacked ensemble with two levels that performs a regression is relatively simple, the proposed concept is specific neither

to the underlying XAI method nor to the underlying regression models. It could even be applied to the probabilities generated by classification models. Additionally, the concept does not depend on the number of first-level models and can, as we did, be applied to first-level models that only share a part of their input features. Also, the joining methods are model-agnostic and a combination of different XAI methods is possible. When, for instance, considering one or multiple complex models on the first level, explaining them with an XAI method that has a faster inference time, and combining that on the second level with an XAI method like SHAP, is possible.

Limitations and Future Work. As argued before, we applied relatively moderate criteria for identifying outliers before training the various ETA models. We did this to make the comparison to non-reproduced papers fairer. However, we expect that we could increase the prediction precision of the composed ensemble model further. Another option to potentially achieve a higher prediction precision is to include other ETA models into the ensemble as additional first-level models, for instance [AGA19; dE19; Kan+19; Li+18].

While the evaluation of the performance of ETA models is relatively straightforward, the evaluation of explanations is not; especially, determining the influence of the slight differences between our joining methods is affected by this problem. Moreover, we considered only four self-chosen scenarios to demonstrate and evaluate the generated explanations; many more scenarios like the ones that combine features—for instance, pickup at city-center during rush hour—might be interesting and valuable for evaluation. While we focused on generating explanations in a general way so that others can adapt and build upon our work, correlated features or information that span over multiple features like the pickup location could be explained better when the features are explained jointly or the x and y values of the pickup location and their influence on the ETA are visualized on a map.

As regards the explanation, in future work, it might be worth looking into ways to explain information that spans over multiple features. Another obvious option to extend our work is to use other XAI methods or use different ones for different models to generate more accurate explanations per model type. The latter could also be used to generate explanations relatively fast, for instance by using LIME for first-level models with a greater feature space and SHAP on the second-level models with smaller feature space. Moreover, the explanations generated here are vectors of values and, therefore, still hard to understand by affected users like taxi drivers or passengers. The explanations could be translated into more human-friendly ones, for instance by linking an explanation of the locations‘ influence on the ETA to points of interest like the main train station that is close to the dropoff location and thereby possibly increasing the ETA for an upcoming trip. Moreover, our explanation could be enhanced from route-free to route-based ETA as such approaches are more likely to be used by taxi drivers and passengers thanks to their increased prediction precision. Additionally, using the generated explanations might be beneficial not only for users of an ETA model, but also for the designers of such models. Based on the explanations, some first-level models or features used in a model might be excluded—leading to a smaller and more precise ETA model.

7 Conclusions

Estimating the time of arrival of taxis is relevant for the comparison and computation of their schedules. In this paper, we took multiple machine learning models from our previous work for ETA and combined them into a stacked heterogeneous ensemble—which, on its own, is novel. We showed that our ensemble model outperforms previous state-of-the-art static route-free ETA approaches. Additionally, we applied two existing XAI methods to explain the first-level models of the ensemble. Finally, we proposed three novel methods for joining the first and second-level explanations of the sophisticated ensemble and consequently explained it. We compared the explanations to a baseline that wraps the whole ensemble in one function and demonstrated our approach on the Yellow taxi trip dataset from New York City.

Acknowledgments

This work was supported by the Deutsche Forschungsgemeinschaft under grant 227198829/GRK1931. The SocialCars Research Training Group focuses on future mobility concepts through cooperative approaches. We thank Steven Minich and Julian Teusch for providing helpful feedback—especially as regards the explanation part of the paper.

References

[AGA19] Abubakr O. Al-Abbasi, Arnob Ghosh, and Vaneet Aggarwal. "DeepPool: Distributed Model-Free Algorithm for Ride-Sharing Using Deep Reinforcement Learning". In: *IEEE Transactions on Intelligent Transportation Systems* 20.12 (Dec. 2019), pp. 4714–4727. ISSN: 1524-9050, 1558-0016. DOI: 10.1109/TITS.2019.2931830.

[BP21] Vaishak Belle and Ioannis Papantonis. "Principles and Practice of Explainable Machine Learning". In: *Frontiers in Big Data* 4 (July 2021). DOI: 10.3389/fdata.2021.688969.

[dE19] Arthur Cruz de Araujo and Ali Etemad. "Deep Neural Networks for Predicting Vehicle Travel Times". In: *2019 IEEE SENSORS*. Montreal, QC, Canada: IEEE, Oct. 2019, pp. 1–4. ISBN: 978-1-72811-634-1. DOI: 10.1109/SENSORS43011.2019.8956878.

[Gan+21] Mudasir. A. Ganaie, Minghui Hu, A. K. Malik, Mohammad Tanveer, and Ponnuthurai N. Suganthan. "Ensemble Deep Learning: A Review". In: *arXiv:2104.02395 [cs]* (Apr. 2021). arXiv: 2104.02395 [cs].

[Jin+18] Ishan Jindal, Zhiwei Tony Qin, Xuewen Chen, Matthew Nokleby, and Jieping Ye. "Optimizing Taxi Carpool Policies via Reinforcement Learning and Spatio-Temporal Mining". In: *2018 IEEE International Conference on Big Data (Big Data)*. Seattle, WA, USA: IEEE, Dec. 2018, pp. 1417–1426. ISBN: 978-1-5386-5035-6. DOI: 10.1109/BigData.2018.8622481.

[Kag19] Kaggle. *DC Taxi Trips*. [online; Visited on 17. March 2022]. 2019. URL: https://www.kaggle.com/bvc5283/dc-taxi-trips.

[KRM21] Athanasios Kallipolitis, Kyriakos Revelos, and Ilias Maglogiannis. "Ensembling EfficientNets for the Classification and Interpretation of Histopathology Images". In: *Algorithms* 14.10 (Sept. 2021), p. 278. ISSN: 1999-4893. DOI: 10.3390/a14100278.

[Kan+19] Kusal D. Kankanamge, Yasiru R. Witharanage, Chanaka S. Withanage, Malsha Hansini, Damindu Lakmal, and Uthayasanker Thayasivam. "Taxi Trip Travel Time Prediction with Isolated XGBoost Regression". In: *2019 Moratuwa Engineering Research Conference (MERCon)*. Moratuwa, Sri Lanka: IEEE, July 2019, pp. 54–59. ISBN: 978-1-7281-3632-5. DOI: 10.1109/MERCon.2019.8818915.

[Li+18] Yaguang Li, Kun Fu, Zheng Wang, Cyrus Shahabi, Jieping Ye, and Yan Liu. "Multi-Task Representation Learning for Travel Time Estimation". In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. London United Kingdom: ACM, July 2018, pp. 1695–1704. ISBN: 978-1-4503-5552-0. DOI: 10.1145/3219819.3220033.

[LL17] Scott M. Lundberg and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS'17. Long Beach, California, USA: Curran Associates Inc., 2017, pp. 4768–4777. ISBN: 9781510860964.

[New19] City of New York. *TLC Trip Record Data*. [online; Visited on 14. March 2022]. 2019. URL: https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page.

[RSG16] Marco Ribeiro, Sameer Singh, and Carlos Guestrin. ""Why Should I Trust You?": Explaining the Predictions of Any Classifier". In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*. San Diego, California: Association for Computational Linguistics, 2016, pp. 97–101. DOI: 10.18653/v1/N16-3020.

[RS21] Benedek Rozemberczki and Rik Sarkar. "The Shapley Value of Classifiers in Ensemble Games". In: *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. New York, NY, USA: Association for Computing Machinery, 2021, pp. 1558–1567. ISBN: 9781450384469. URL: https://doi.org/10.1145/3459637.3482302.

[Sch22] Sören Schleibaum. *Estimated Time of Arrival*. https://gitlab.tu-clausthal.de/ss16/stacked-eta-and-explanation. Mar. 2022.

[SMS22] Sören Schleibaum, Jörg P. Müller, and Monika Sester. "Enhancing Expressiveness of Models for Static Route-Free Estimation of Time of Arrival in Urban Environments". In: *Transportation Research Procedia* 62 (2022), pp. 432–441. DOI: 10.1016/j.trpro.2022.02.054.

[SFC19] Wilson Silva, Kelwin Fernandes, and Jaime S. Cardoso. "How to Produce Complementary Explanations Using an Ensemble Model". In: *2019 International Joint Conference on Neural Networks (IJCNN)*. Budapest, Hungary: IEEE, July 2019, pp. 1–8. ISBN: 978-1-72811-985-4. DOI: 10.1109/IJCNN.2019.8852409.

[TL19] Mingxing Tan and Quoc Le. "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks". In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, June 2019, pp. 6105–6114. URL: https://proceedings.mlr.press/v97/tan19a.html.

[UK21] Lev V. Utkin and Andrei V. Konstantinov. “Ensembles of Random SHAPs”. In: (Mar. 2021). arXiv: 2103.03302 [cs.LG].

A Appendix

In Figure 10, we visualize the LIME explanations generated via the proposed joining methods for SC1, SC3, and SC4; in Figure 11, we do the same for the SHAP explanations. Figure 12 shows the content of Figure 9 for LIME.

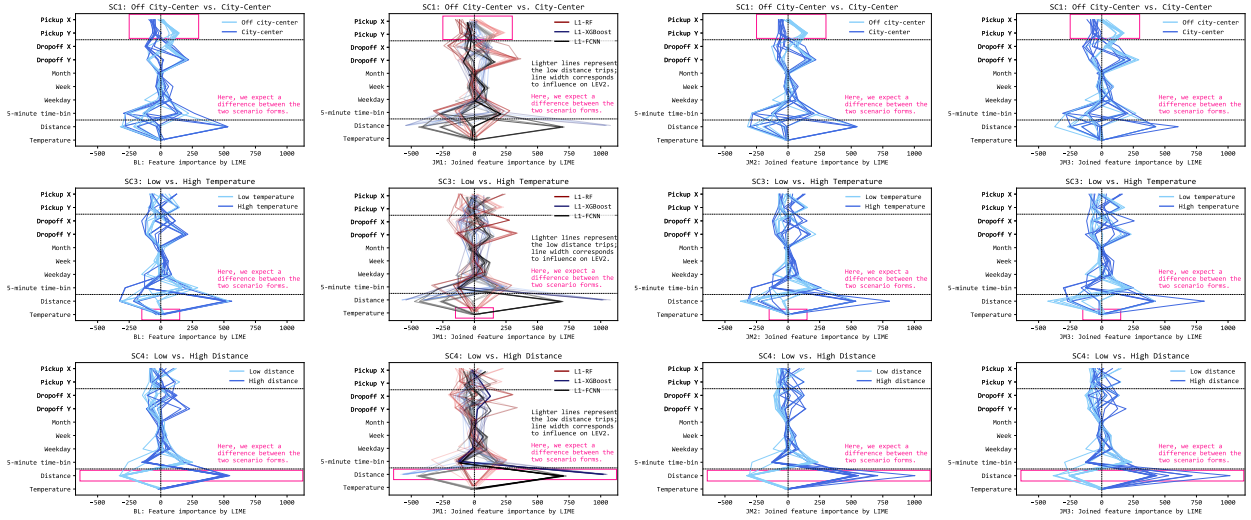


Figure 10: Content of Figure 7 for SC1, SC3, and SC4.

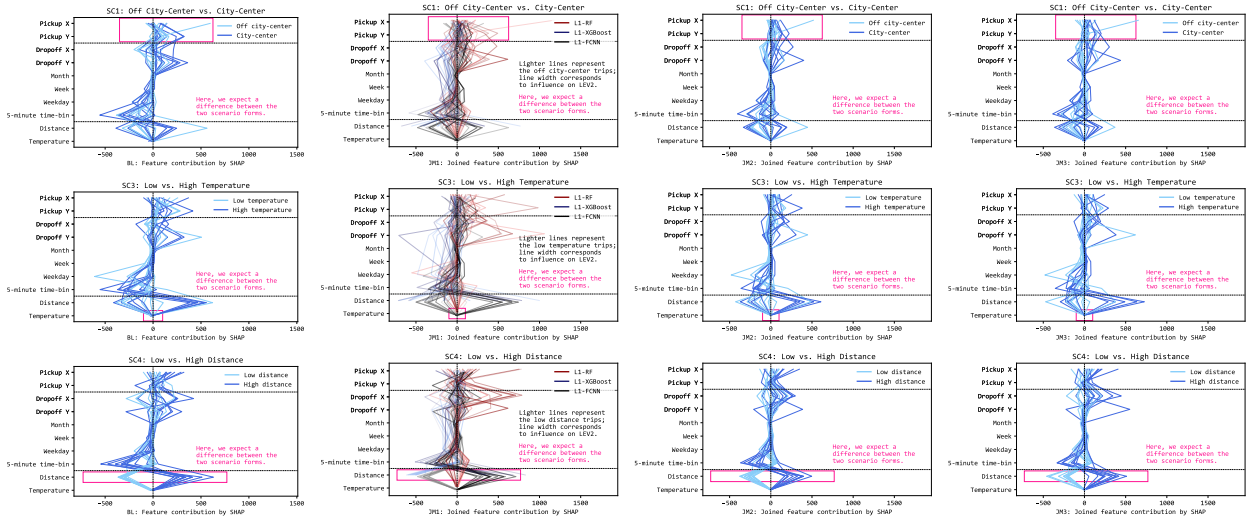


Figure 11: Content of Figure 8 for SC1, SC3, and SC4.

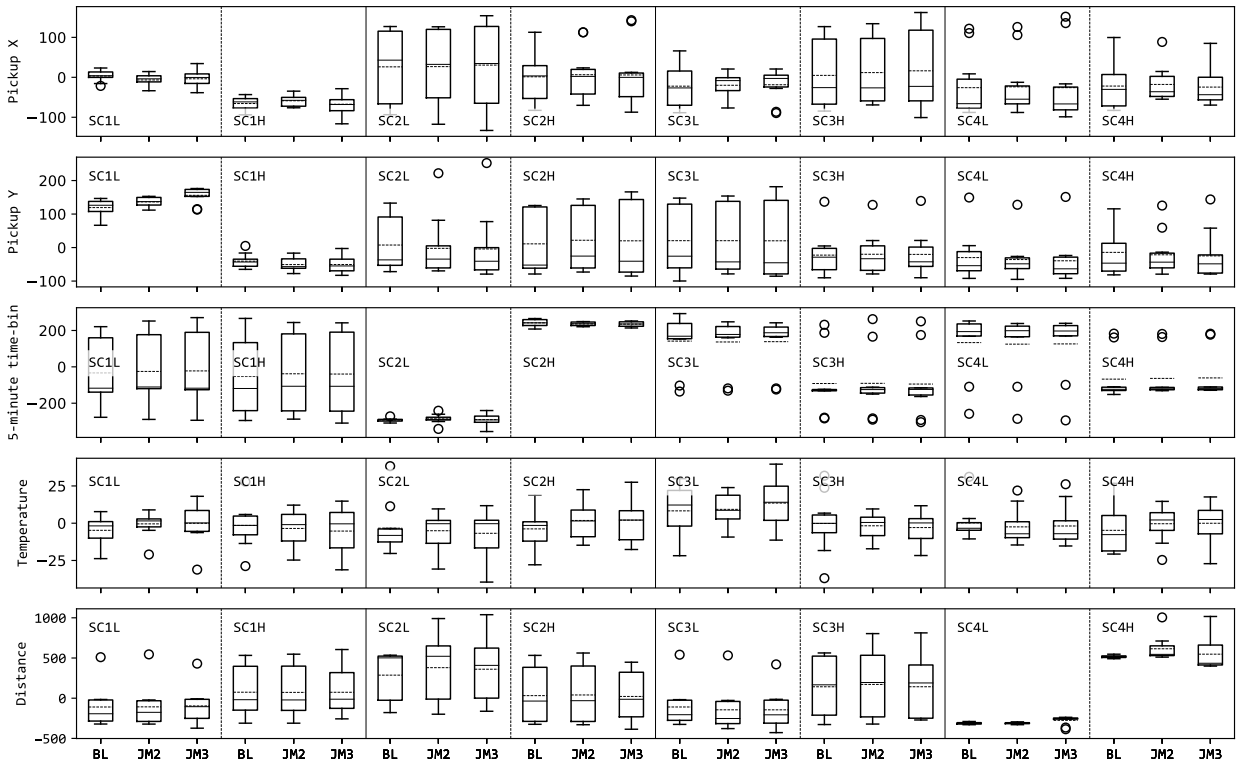


Figure 12: Content of Figure 9 for LIME