2025 20th European Dependable Computing Conference Companion Proceedings (EDCC-C) | 979-8-3315-3741-8/25/$31.00 ©2025 IEEE | DOI: 10.1109/EDCC-C66476.2025.00043

Harnessing Explainable AI in Railway: A Decision Tree-based Approach

*Note: Sub-titles are not captured for https://ieeexplore.ieee.org and should not be used

Mario Barbareschi, Antonio Emmanuele, Nicola Mazzocca, Franca Rocco di Torrepadula
Department of Electrical Engineering and Information Technologies
University of Naples Federico II
Naples, Italy
{name.surname}@unina.it

***Abstract*—In recent years, Artificial Intelligence has gained significant popularity for solving various tasks, including service optimization, system monitoring, and industrial control. Despite its success, adoption in critical systems, such as the railway domain, remains limited. This is primarily due to the high stakes in these systems, where failures can lead to damage to critical infrastructure and risks to human lives. As a result, software in these domains must be deterministic, ensuring that all behaviors can be statically verified.**

Machine Learning models, due to their complexity, are often perceived as black-box systems and exhibit seemingly nondeterministic behavior, making their integration into such infrastructure challenging. To address this issue, one potential solution is the use of eXplainable Artificial Intelligence (XAI) techniques, which enable the construction of human-interpretable explanations for model predictions.

In this paper, we propose a time-series prediction framework for the railway domain by combining XGBoost, a highly accurate tree-based model, with SHAP, a widely used explainability technique.

***Index Terms*—eXplainable Artificial Intelligence, SHAP, XGBoost, Railway Domain, Time Series Forecasting**

I. Introduction

Artificial Intelligence (AI) is increasingly applied in diverse sectors of society, including healthcare, industrial control systems, and public transportation. Applications like passenger flow prediction [1], anomaly detection [2], and system monitoring [3] utilize Machine Learning (ML) models and techniques to deliver adaptable and precise solutions. Also within the railway domain, ML models are increasingly employed for crucial tasks such as accident prediction [4] and monitoring track health conditions [5].

However, despite the potential benefits, integrating AI into critical domains presents significant challenges due to stringent constraints. For instance, software systems employed for railways must exhibit a deterministic behavior, ensuring that every input condition results in predefined outputs and that timing constraints are fully and statically specified. This is particularly emphasized by the EN 50128 standard from the EN-CENELEC framework, which sets stringent guidelines for the development and deployment of software in safety-critical railway applications [6]. Traditional software can achieve determinism through specific development processes, adherence to coding standards, use of real-time operating systems, and exhaustive testing. Conversely, ML models often lack such determinist properties due to their inherent complexity. Indeed, in order to obtain more accurate models, their complexity is increased, making it challenging to specify their behavior for every possible input condition. This creates a significant barrier to their adoption in critical systems.

To facilitate the use of ML in railway systems – particularly in safety-critical tasks such as train control, speed regulation, or predictive maintenance – it is essential to understand the internal behavior of these models fully. This understanding not only aids in ensuring safety and compliance but also supports the comprehensive characterization of models required for certification in the railway domain. In this context, recently eXplainable Artificial Intelligence (XAI) has gained substantial attention as a means to provide human-interpretable explanations of ML models. Among several XAI approaches, feature attribution is a well-established technique that explains individual predictions by quantifying the contribution of each feature to the model inference [7]. Building on these advancements, this paper proposes an interpretable time-series predictive framework tailored for railway applications. The framework leverages XGBoost, a tree-ensemble model known for its accuracy and computational efficiency, combined with SHAP (SHapley Additive exPlanations) values to enhance interpretability. This combination enables the development of accurate models whose predictions can be readily explained, thus addressing critical challenges of determinism and compliance.

The proposed framework is evaluated on a synthetic dataset designed to predict train occupancy in a major Italian city. In this context, explainability enables railway operators to understand key factors influencing model predictions, thereby facilitating informed decisions regarding resource allocation and management. Moreover, transparent predictions enhance

This work has been partially supported by Spoke 9 *Digital Society & Smart Cities* of ICSC - Centro Nazionale di Ricerca in High-Performance-Computing, Big Data and Quantum Computing, funded by European Union - NextGenerationEU (PNRR-HPC, CUP: E63C22000980007).

979-8-3315-3741-8/25/$31.00 ©2025 IEEE
DOI 10.1109/EDCC-C66476.2025.00043

trust and provide a basis for continuous monitoring. The results of the experimental protocol demonstrate that our approach achieves both high accuracy and interpretability, making it suitable for deployment in railway systems where transparency and reliability are paramount.

II. Related Work

In this section, we present a brief introduction to decision-tree-based models, which form the foundation of the model adopted in our framework, as well as an overview of ML techniques. Lastly, we discuss the major applications of ML in the railway domain.

A. Decision Tree Models

Decision Trees (DTs) are ML models structurally represented as full-binary tree structures. The nodes of these trees can be categorized as internal or leaf nodes. Internal nodes encapsulate logical conditions, referred to as *decision rules*, that are evaluated on input features. These rules typically involve comparing a feature against a fixed threshold using a mathematical operation, such as the $\leq$ operator. On the other hand, leaf nodes represent inference results. For regression tasks, these values are continuous, whereas, for classification tasks, they are categorical numbers representing distinct classes [8]. Inference in such models involves a tree-visiting procedure, where decision rules are iteratively evaluated to traverse the tree until a leaf node, corresponding to a classification or regression value, is reached [9].

Despite their lightweight inference capabilities, DTs are prone to overfitting. To mitigate this issue and enhance accuracy, they are often used in ensembles. One of the most widely adopted techniques is boosting, where tree ensembles are constructed through an iterative training process. During each iteration, a new tree is trained to improve the accuracy of the ensemble. Among tree-ensemble methods, XGBoost is particularly notable for its efficiency in training and inference, resulting in models with high accuracy [10]. Building on these recognized benefits, this work utilizes XGBoost regressors to predict train occupancy in a major Italian city.

B. Interpretability

Explaining the predictions of a model implies being able to fully understand its inner reasoning and to represent such a process in a human-interpretable format. To this end, global-interpretable ML models allow for an immediate representation of all their possible outcomes through a set of decision rules [8]. Among such models, DTs are the most known example, as they allow explaining each outcome—i.e., a leaf of the tree—through a set of decision rules—i.e., the conditions of the internal nodes that evaluate to true during the tree traversal.

In contrast, models such as XGBoost do not allow interpreting all their possible outcomes directly. Nevertheless, different techniques enable the explanation of individual predictions, making these models local-interpretable, as only single predictions can be locally interpreted [11]. Among these techniques, feature attribution is the most widely used [12]. These techniques mathematically represent single predictions as a combination of contribution values—also referred to as attribution values—assigned to each feature. The challenge of interpreting predictions through feature attribution lies in assigning these attribution values to the features [13].

The authors of [12] demonstrated that SHAP values are the only attribution values satisfying the following properties: (i) Local accuracy: the linear combination of feature attributions approximates the prediction; (ii) Missingness: if a feature contributes nothing to the prediction, its attribution value is 0; (iii) Consistency: if the model changes such that a feature's contribution to the prediction increases, its attribution value does not decrease. In light of these findings, in this work we use SHAP values to provide explanations for model predictions.

C. Artificial Intelligence in the Railway domain

AI has been extensively utilized in the railway sector for various tasks, including optimizing complex railway systems, detecting component defects and faults, enhancing safety, and improving service quality, as highlighted in several recent surveys (e.g., [14]). Among several AI techniques, Neural Networks (NNs) and particularly Deep Neural Networks (DNNs) are among the most widely adopted solutions due to their ability to capture intricate data relationships and effectively solve complex tasks [14].

For instance, Acikgoz and Korkmaz [15] developed a classification system for railway track defect detection using residual Convolutional Neural Networks (CNNs), where residual blocks were introduced to enhance network effectiveness. Similarly, Islam et al. [16] proposed a framework that integrates neural networks with a clustering algorithm to improve anomaly detection in railway systems. Furthermore, Kang et al. [17] employed a Long Short Term Memory Network (LSTM) network to predict Passenger Flow (PF) at the Xizhimen subway station of the Beijing Metro, leveraging recurrent connections to capture temporal patterns.

Nevertheless, despite their high accuracy, NN-based models often suffer from increased computational complexity and a lack of interpretability—a significant challenge in railway applications where adherence to safety standards is paramount. Hence, addressing this trade-off between accuracy and interpretability remains a crucial open research problem in the development of AI-driven railway solutions [18].

III. Time Series Forecasting in the Railway Domain

In this section, we present our framework for explainable time-series forecasting. Our approach facilitates the construction of accurate predictive models by utilizing XGBoost. Additionally, it enables the interpretation of model predictions using SHAP values. This capability opens numerous opportunities in the railway domain, as explaining predictions fosters trust in

TABLE I
HYPERPARAMETER SETTINGS FOR XGBOOST, RANDOM FOREST, AND REGRESSION TREES.

Hyperparameter	XGBoost (XGB)	Random Forest (RF)	Regression Trees (RT)
`learning_rate`	0.1	-	-
`gamma`	0.2	-	-
`max_depth`	7	20	20
`subsample`	0.75	-	-
`alpha`	0	-	-
`lambda`	3	-	-
`min_samples_split`	-	2	50
`min_samples_leaf`	-	5	20
`max_features`	-	1.0	1.0

the internal reasoning of models, which is crucial for critical-domains. In the following, we describe the four sequential phases of our framework.

A. Data Preprocessing

In this step, the data is pre-processed to optimize the accuracy of the final model while minimizing training time. Dataset entries with missing values are removed, and all data is normalized according to specific procedures, such as min-max scaling or z-score normalization. Additionally, features that do not contribute to the model, such as constant columns and redundant ones, are eliminated. This is achieved using established techniques like Principal Component Analysis (PCA) or through feature importance analysis. As noted by the authors of [1], this approach can also involve directly training DTs on the selected time-series.

B. Hyperparameterization

During this phase, the preprocessed time-series is utilized to determine the optimal set of hyperparameters for XGBoost models. Methods such as grid search or randomized search are employed to identify the best configuration. It is important to note that, due to the temporal dependencies inherent in time-series data, traditional validation procedures like k-fold cross-validation may cause unintentional data leakages. To address this issue, suitable techniques such as rolling-window cross-validation are adopted. This approach involves comparing models by iteratively using a portion of the series as the training set and a set of future samples as the validation set. The process is repeated k times by progressively shifting the starting point of the validation set forward, thereby incrementally expanding the size of the training set with each iteration.

C. Model Training

After determining the optimal set of hyperparameters, the final model is trained using the preprocessed time series. Once trained, the model is exported in a format compatible with the deployment platform. For non-time-critical applications, the model can be serialized and re-imported directly into the target platform, assuming the training framework, such as the Python library [1], is natively supported on the platform.

For time-critical applications, the model must be converted into an appropriate format. For instance, if Field Programmable Gate Arrays (FPGAs) accelerators are used, the model can be synthesized directly into a digital circuit by leveraging the open-source framework proposed by the authors in [19]. Alternatively, if the target platform relies on microprocessors, the model can be translated into C code using open-source tools like m2cgen [2].

D. Model Inferences and Interpretability

After deploying the model on the target platform, SHAP values can be used to explain individual predictions. SHAP satisfies the previously discussed attribution requirements, making it an effective tool for providing solid explanations of model outputs. Once SHAP values are computed, various plots can visually represent the explanations. For instance, a force plot graphically illustrates the positive or negative contribution of each feature to the final prediction value. In contrast, a dependence plot shows the SHAP values of a feature across its range of values, along with its most correlated feature.

IV. EXPERIMENTS AND RESULT

In this section, the proposed approach is applied to a synthetic dataset that simulates train-scheduled arrivals at various stations along with their corresponding occupancy levels. The dataset was generated using the renowned Eclipse Simulator of Urban MObility (SUMO)[3], starting from publicly available, real-world Origin-Destination matrices from the city of Genoa, Italy [4]. Each record captures the arrival of a metro train at a station, storing information such as the timestamp, the vehicle identifier, the route number, the station identifier, and the number of passengers on the train. For both training and inference, we utilize the standard XGBoost library, while SHAP values are computed using the open-source library available at https://shap.readthedocs.io/en/latest/.

A. Evaluating Model Accuracy

We began our experimental campaign by executing the first three steps of the proposed framework. Since the dataset is synthetically generated, it does not contain samples with missing columns. The data was initially scaled using a min-max scaler, followed by the application of decision tree-based feature importance filtering, as discussed in [1]. These models

[1] https://xgboost.readthedocs.io/en/stable/python/python_intro.html

[2] https://github.com/BayesWitnesses/m2cgen

[3] https://eclipse.dev/sumo/

[4] https://dati.comune.genova.it/dataset/matrici-dei-viaggi-origine-destinazione

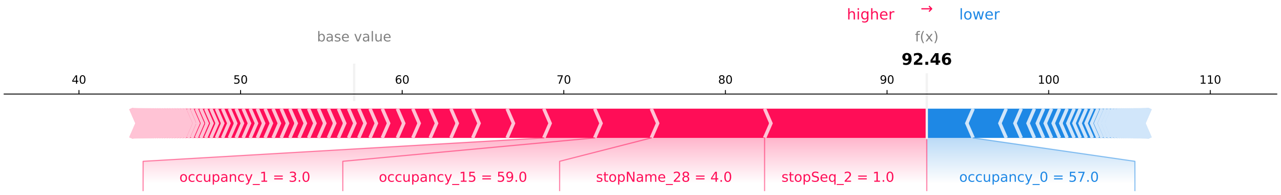

Fig. 1. Force plot of a single forecasted occupancy value.

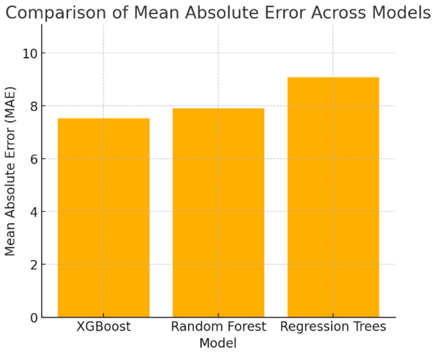

Fig. 2. Mean Absolute Error of tree based models on the considered dataset.

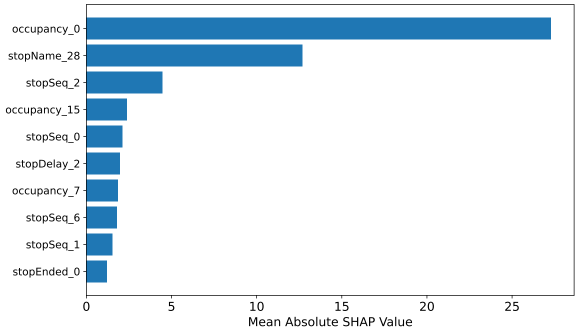

Fig. 3. Mean absolute SHAP value of the considered features.

are trained using a historical window of the 30 most recent samples from the training dataset, which are used to predict the subsequent occupancy value. From the entire dataset, only the top 30 features were selected.

The filtered time series is then utilized for tuning the hyperparameters of the predictive model, through a grid search. To prevent data leakage during the search, we apply a 5-fold rolling window validation. The validation of each parameter set begins with an initial training set that corresponds to 70% of the time series. As the approach targets XGBoost while also being applicable to tree-ensemble models in general, experiments are conducted on Regression Trees, Random Forest, and XGBoost. We report in Table I the best set of parameters.

Finally, Figure 2 presents the Mean Absolute Error (MAE) of the three considered models on the dataset. The reported values are obtained using the same cross-validation procedure employed during hyperparameter tuning. The results clearly indicate that ensemble models outperform single Decision Tree Regressors (DTRs). Among the ensembles, XGBoost exhibits the lowest error, proving to be best performing model.

B. Explaining predictions through SHAP

In this section, a practical example of prediction interpretability for individual XGBoost predictions is presented using SHAP values. The analysis begins by investigating the features that have the most significant global impact on the model's predictions. To this end, Figure 3 presents a bar plot of the mean absolute SHAP value for each feature, highlighting their importance in influencing the forecasts of the model. Each feature is reported with respect to the timestamp within the input window. For instance, `occupancy_0` represents the occupancy value at the previous stop. The bar plot clearly shows that the most impactful feature is the train's occupancy at the preceding stop, which aligns with expectations, as the occupancy at the next stop is strongly correlated with that of the previous stop. Additionally, the stop name—i.e., its unique identifier—at lag 28 emerges as the second most influential feature. This suggests a periodic pattern in the dataset, where the initial stops significantly affect the predicted occupancy value.

Given the high attribution of the last occupancy value to the final prediction, Figure 4 presents the dependence plot for this feature. The x-axis represents the current occupancy value, while the left y-axis shows its corresponding SHAP value. The observed linear trend suggests that higher occupancy values tend to positively influence the predicted occupancy. Additionally, the right y-axis displays the most correlated features according to SHAP values, with colors indicating the strength of the correlation. The presence of `StopName` at lag 4 as a correlated feature suggests that, in addition to occupancy history, spatial information—i.e., the sequence of stop values—also plays a significant role in forecasts.

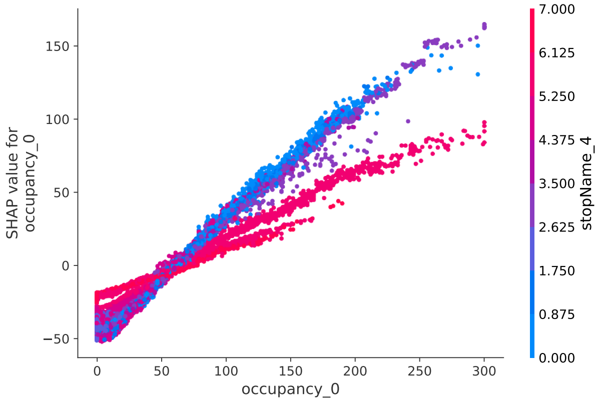

Fig. 4. Dependence plot of the last occupancy value.

Finally, after analyzing the global effects of features, the focus shifts to explaining individual predictions. Force plots are employed as graphical tools for interpreting forecasts, representing each prediction as a linear combination of feature attribution values. Specifically, these attribution values are summed to a base value to obtain the final predicted output. Depending on their influence, these attributions can be either positive or negative, reflecting their respective contributions to the prediction. For instance, Figure 1 presents the force plot for a specific prediction. This visualization highlights how recent features, such as `StopName` and `occupancy_1`, contribute positively to the predicted value, indicating an increasing trend in occupancy. However, the most recent occupancy value provides a negative contribution, suggesting that the increasing trend was interrupted at the last stop.

V. Conclusions

Artificial Intelligence is becoming increasingly pervasive across various sectors of society, raising several ethical and safety concerns regarding its extensive use. Among these sectors, the railway domain is particularly critical, as software-driven decision-making must be supported by a certifiable and reliable process. This paper proposes the use of explainability techniques to enhance the transparency and adoptability of predictive models in critical domains. Specifically, a time-series predictive framework is introduced, leveraging Decision Tree-based models and SHAP feature attribution values to interpret predictions. The approach is evaluated on an industrial synthetic dataset, demonstrating that it enables accurate models while facilitating the interpretability of predictions.

References

[1] M. Barbareschi, A. Emmanuele, N. Mazzocca, and F. Rocco di Torrepadula, "Designing on-board explainable passenger flow prediction," *Engineering Applications of Artificial Intelligence*, vol. 139, p. 109648, Jan. 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197624018062

[2] F. Vitale, F. De Vita, N. Mazzocca, and D. Bruneo, "A Process Mining-based unsupervised Anomaly Detection technique for the Industrial Internet of Things," *Internet of Things*, vol. 24, p. 100993, Dec. 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2542660523003165

[3] J. V. Abellan-Nebot and F. Romero Subirón, "A review of machining monitoring systems based on artificial intelligence process models," *The International Journal of Advanced Manufacturing Technology*, vol. 47, no. 1, pp. 237–257, Mar. 2010. [Online]. Available: https://doi.org/10.1007/s00170-009-2191-8

[4] H. Alawad, S. Kaewunruen, and M. An, "Learning From Accidents: Machine Learning for Safety at Railway Stations," *IEEE Access*, vol. 8, pp. 633–648, 2020, conference Name: IEEE Access. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8941092

[5] Y.-W. Wang, Y.-Q. Ni, and S.-M. Wang, "Structural health monitoring of railway bridges using innovative sensing technologies and machine learning algorithms: a concise review," *Intelligent Transportation Infrastructure*, vol. 1, p. liac009, Sep. 2022. [Online]. Available: https://doi.org/10.1093/iti/liac009

[6] European Committee for Electrotechnical Standardization (CENELEC), "EN 50128: Railway Applications - Communication, Signalling and Processing Systems - Software for Railway Control and Protection Systems," https://standards.cen.eu, 2011, standard published by CENELEC.

[7] S. M. Lundberg, G. G. Erion, and S.-I. Lee, "Consistent Individualized Feature Attribution for Tree Ensembles," Mar. 2019, arXiv:1802.03888 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1802.03888

[8] O. Sagi and L. Rokach, "Explainable decision forest: Transforming a decision forest into an interpretable tree," *Information Fusion*, vol. 61, pp. 124–138, Sep. 2020. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1566253519307869

[9] M. Barbareschi, "Implementing Hardware Decision Tree Prediction: A Scalable Approach," in *2016 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA)*. Crans-Montana, Switzerland: IEEE, Mar. 2016, pp. 87–92. [Online]. Available: http://ieeexplore.ieee.org/document/7471178/

[10] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. San Francisco California USA: ACM, Aug. 2016, pp. 785–794. [Online]. Available: https://dl.acm.org/doi/10.1145/2939672.2939785

[11] O. Sagi and L. Rokach, "Approximating XGBoost with an interpretable decision tree," *Information Sciences*, vol. 572, pp. 522–542, Sep. 2021. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0020025521005272

[12] S. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," Nov. 2017, arXiv:1705.07874 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1705.07874

[13] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why Should I Trust You?": Explaining the Predictions of Any Classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: Association for Computing Machinery, 2016, pp. 1135–1144. [Online]. Available: https://dl.acm.org/doi/10.1145/2939672.2939778

[14] R. Tang, L. De Donato, N. Besinović, F. Flammini, R. M. P. Goverde, Z. Lin, R. Liu, T. Tang, V. Vittorini, and Z. Wang, "A literature review of Artificial Intelligence applications in railway systems," *Transportation Research Part C: Emerging Technologies*, vol. 140, p. 103679, Jul. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0968090X22001206

[15] H. Acikgoz and D. Korkmaz, "Msrconvnet: classification of railway track defects using multi-scale residual convolutional neural network," *Engineering Applications of Artificial Intelligence*, vol. 121, p. 105965, 2023.

[16] U. Islam, R. Q. Malik, A. S. Al-Johani, M. R. Khan, Y. I. Daradkeh, I. Ahmad, K. A. Alissa, Z. Abdul-Samad, and E. M. Tag-Eldin, "A novel anomaly detection system on the internet of railways using extended neural networks," *Electronics*, vol. 11, no. 18, p. 2813, 2022.

[17] L. Kang, H. Liu, M. Chai, and J. Lv, "A lstm-based passenger volume forecasting method for urban railway systems," in *Robotics and Rehabilitation Intelligence: First International Conference, ICRRI 2020, Fushun, China, September 9–11, 2020, Proceedings, Part I 1*. Springer, 2020, pp. 368–380.

[18] N. Besinović, L. De Donato, F. Flammini, R. M. P. Goverde, Z. Lin, R. Liu, S. Marrone, R. Nardone, T. Tang, and V. Vittorini,

"Artificial Intelligence in Railway Transport: Taxonomy, Regulations, and Applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14 011–14 024, Sep. 2022, conference Name: IEEE Transactions on Intelligent Transportation Systems. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9652066

[19] M. Barbareschi, S. Barone, and N. Mazzocca, "Advancing synthesis of decision tree-based multiple classifier systems: an approximate computing case study," *Knowledge and Information Systems*, vol. 63, no. 6, pp. 1577–1596, Jun. 2021. [Online]. Available: https://doi.org/10.1007/s10115-021-01565-5