Computers in Industry 144 (2023) 103781

Contents lists available at ScienceDirect

Computers in Industry

journal homepage: www.sciencedirect.com/journal/computers-in-industry

Edge intelligence-based proposal for onboard catenary stagger amplitude diagnosis

Itxaro Errandonea [a,b,*], Pablo Ciáurriz [a,b], Unai Alvarado [a,b], Sergio Beltrán [a,b,c], Saioa Arrizabalaga [a,b,c]

[a] CEIT-Basque Research and Technology Alliance (BRTA), Manuel Lardizabal 15, 20018 Donostia / San Sebastián, Spain
[b] Universidad de Navarra, Tecnun, Manuel Lardizabal 13, 20018 Donostia / San Sebastián, Spain
[c] Institute of Data Science and Artificial Intelligence, DATAI, Universidad de Navarra, Pamplona, Spain

ARTICLE INFO

Keywords:
Digital Twins
Railway
Catenary
Edge computing
Maintenance
Diagnostic analytics

ABSTRACT

In recent years, the integration of Digital Twins (DT) for the adoption of smarter maintenance strategies has grown exponentially in different industrial sectors. New IoT and edge computing systems are being developed for this purpose, however, there are still some open issues and challenges to be solved. Firstly, this paper presents new approaches to the initial dependencies of the studied solution and make a new proposal to improve the interoperability of the presented system. Secondly, this paper provides a methodology applicable to similar developments of edge-based AI (Artificial Intelligence) solution, which comprises of four phases: the presentation of the multi-objective problem and the pre-selection of AI-based models, the description of the evaluation architecture, the profiling of the different models for the selection of the most suitable one and explainable AI strategies for getting insights of the selected model. Finally, it presents a use case of an edge-solution for the railway catenary geometry diagnostic (stagger amplitude of the overhead wire), saving the interoperability of the message exchange with other systems is provided.

1. Introduction

A Digital Twin (DT) can be defined as a virtual model that accurately represents a physical object, process, or service and is used to help optimizing its management during its whole life cycle. It relies on real-time and historical data to represent the past and present and eventually forecast the future, allowing the simulation and prediction of failures (Tao et al., 2019). DTs usually comprise five dimensions including physical entities, virtual models, data, services, and connections, and have a great potential to radically improve the design, manufacturing, maintenance and renewal processes across multiple industries (Tao and Zhang, 2017; Errandonea et al., 2020).

DT is a fast-growing technology and every year it is applied to a wider range of business and processes, which include construction, transportation, smart cities, healthcare, or energy efficiency. Due to the growing interest that this technology is having in the industry, the expected global market of DT-based solutions is expected to grow from US $ 10.3 billion in 2021 to reach US$ 54.6 billion by 2027 (Digital Twin Market, 2022–2027).

One of the market niches where DT provide huge benefits is product and infrastructure maintenance. The models and data included in the virtual representation of the DT provide a great tool to enable a transition from traditional corrective maintenance strategies (repairing after failure) to the more interesting and effective condition-based (real-time monitoring of assets health) or predictive maintenance (anticipate to the future condition of assets). These two latter maintenance strategies require a continuous monitoring and diagnosis to assess the health condition of assets. Going a step further, the ultimate goal is to achieve prescriptive maintenance to optimize maintenance operations: being able to anticipate the future condition of the different components by means of a combination of physical modelling and advanced data analysis of the collected data is key to enable complex decision support scenarios.

Railway industry is one of the fastest-growing industries in the application of DT technology. It has initially been mostly used for the digitalization of assets for construction operations (e.g., using BIM), but it is widening its application to operations management and optimization and to improve rolling stock and infrastructure maintenance

* Corresponding author at: CEIT-Basque Research and Technology Alliance (BRTA), Manuel Lardizabal 15, 20018 Donostia / San Sebastián, Spain.
E-mail address: ierrandonea@ceit.es (I. Errandonea).

https://doi.org/10.1016/j.compind.2022.103781
Received 10 June 2022; Received in revised form 1 September 2022; Accepted 6 September 2022
Available online 18 September 2022

operations and costs. At European level, railway infrastructure and vehicle maintenance and renewal costs are estimated to be above €25 billion per year (Using analytics to get European rail maintenance on track, 2022), Additionally, global growing population and an estimated shift to cleaner transportation modes like railway forecast an increase of railway operations and infrastructure, with the associated increase of maintenance costs (Publication: The Future of Rail, 2022b).

In the railway sector, there are several approaches for anomaly detection using technologies such as edge-computing and IoT (Internet of Things). Most of the proposed solutions focus on train-specific failure modes. For example, publications such as (Hodzic et al., 2020) show an example of a solution for detecting anomalies in the traction motor. In contrast, (Liu et al., 2018) shows the work to be done for the detection of anomalies within critical subsystems of rolling stock. The work reflected in (Teng et al., 2021) focuses more on wheel-rail interaction. One can also find approaches for signal anomaly detection such as the one presented in (Rabatel and Bringay, 2009). More specifically, (Kang et al., 2018) is focused on the anomaly detection of speed signals.

The use of Machine Learning (ML) models to monitor and diagnose stagger amplitude has been mainly limited to artificial neural networks (ANNs) for image processing (Karakose et al., 2017; Yang et al., 2020; Zhang et al., 2020). The main disadvantage of these approaches is the computational cost of training the convolutional neural networks, together with many hand-labelled images. This is also a problem for Support Vector Machine (SVM) type models when several features are extracted from the images (Cho and Ko, 2015).

No previous work is known in the railway sector that describes the process of developing an edge solution based on theoretical results for the diagnostic of stagger amplitude. This paper contributes to the process of transferring theoretical results to real-life applied solutions. A new methodology is presented to create solutions based on edge intelligence for stagger amplitude diagnosis. An AI model selection criterion is defined based on a multi-objective problem for the proposed solution. Considering the possible difficulties in the interpretability of a ML model, some guidelines are also included to improve the comprehension of selected models.

This article is organized in 5 sections. Section 2 describes the preliminary solution to address the stagger amplitude diagnostic challenge, and presents the latest enhancements made for interoperability. Section 3 presents the proposed methodology and Section 4 shows the evaluation criteria and the results of the methodology applied to a stagger amplitude diagnosis use case. Finally, some conclusions are derived from the proposed work in Section 5.

2. Background on catenary diagnosis: limitations and new enhancements

SIA is a research project funded by the European Union's Horizon H2020 research and innovation programme (Home - SIA project, 2022) and supported by EUSPA, the European Union Agency for the Space Program (About EUSPA, 2022). The main objective of the project is to develop several services to provide real-time prognostic information on the health condition of the railway's most demanding assets in terms of maintenance costs. The higher costs occur on the components where the moving vehicle interacts with the surrounding infrastructure, i.e., wheel–rail and pantograph-catenary interactions.

To monitor both interactions and the components involved, several modules have been developed: iWheelMon and iRailMon provide real-time information about wheel and rail condition respectively, whereas iPantMon and iCatMon supervise the real-time condition of the pantograph and catenary. These systems provide a continuous monitorization of the assets using non-invasive and low-cost components.

Within iCatMon, and in particular focused on the health monitoring of the overhead line, an accurate characterization of the involved assets was made, as well as a virtual framework to simulate, train and test the developed ML algorithms that provide a diagnostic information about the current condition of the components.

Next, the previously developed stagger amplitude diagnosis module, its limitations and new contributions regarding speed-variable analysis are described. Then, the IoT-based monitoring system architecture of the SIA project and further contributions for gaining interoperability are introduced.

2.1. Stagger amplitude diagnosis module

For asset health diagnosis, there are different techniques used in different industrial sectors. One of them makes use of physics-based models, where the accuracy of the results is very precise since they are very faithful to the dynamic behaviour of the asset as such. Another technique commonly used is the deployment of data-driven models, also called ML models (Wang et al., 2020; de et al., 2019). Both techniques show difficulties for its applicability in the described context. In the case of physics-based models, they are computationally demanding and time-consuming for an edge solution. On the other hand, in case of data models, a large volume of data would be needed to train the models.

To overcome these issues and to take advantage of their complementary benefits, a solution using hybrid models is proposed. This is done by training a model based on synthetic data, generated by physics-based model of the asset.

Fig. 1 shows the followed steps to extract the hybrid model for catenary stagger diagnosis. A mathematical model of the physical asset has been developed, a series of scenarios of interest have been defined and the simulations of these scenarios launched. Then, a processing and analysis of the simulated signals has been performed and the necessary features have been defined and extracted to finally train the data-based model or ML model.

In this case, the mathematical model simulates the behaviour of the different accelerometers installed in a pantograph. The simulated signals are processed to extract the necessary characteristics for the training of the ML models. To train a hybrid model, simulations of the different scenarios that may occur in real life must be performed. All details about the mathematical model, the signal processing performed, and the simulated scenarios are published in (Blanco et al., 2022). From all the simulations carried out, 36 features have been extracted (described in the Table 1).

H2O AutoML (Hutter et al., 2019) has been used for training and cross-validation of models for the diagnosis of stagger at 100, 200 and 300 km/h. A total of 150 different ML models have been trained for this purpose. To reduce the risk of overfitting, all training has been performed with 10 folds in the cross-validation. The metric used for the evaluation of the trained models is the Root-Mean-Square-Error (RMSE). This metric measures the error between the observed values (v_i) and those predicted by the model ($\widehat{v}_i$):

$$RMSE = \sqrt{\frac{1}{n}\sum\nolimits_{i=1}^{n}(v_i - \widehat{v}_i)^2} \quad (1)$$

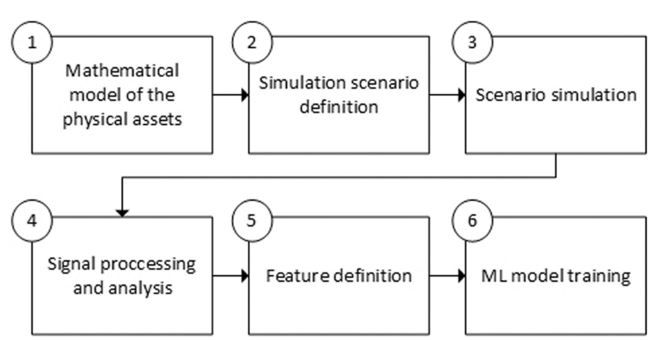

Fig. 1. Virtual scenarios for synthetic data generation for ML model training.

Table 1
Stagger features.

Index	Feature	Description
1–2	mean ($SAWP_i$)	Mean value of SAWP for each *i*-th sensor
3–4	std ($SAWP_i$)	Standard deviation of SAWP for each *i*-th sensor
5	mean (Rt_{stag})	Mean value of Rt_{stag}
6	std (Rt_{stag})	Standard deviation of Rt_{stag}
7	mean [max (Rt_{stag})]	Mean of relative maximums of Rt_{stag}
8	mean [min (Rt_{stag})]	Mean of the relative minimums of Rt_{stag}
9	max (Rt_{stag})	Absolute maximum of Rt_{stag}
10	min (Rt_{stag})	Absolute minimum of Rt_{stag}
11–14	$WL[PSD(SAWP_i)]_{1,2}^{st,nd}$	Wavelengths of the 1st and 2nd dominant peaks of the $SAWP_i$ PSD for the *i*-th sensor
15–18	$\lvert PSD(SAWP_i)\rvert_{1,2}^{st,nd}$	Magnitude of the 1st and 2nd dominant peaks of the $SAWP_i$ PSD for the *i*-th sensor

* $SAWP_i$ = Scale Average Wavelet Power, $Rt_{stag}=\log_{10}\left(\frac{SAWP_{right}}{SAWP_{left}}\right)$, PSD= Power Spectral Density

The trained models show good performance for the estimation of stagger (see Table 2).

Overall, the results indicate that the proposed virtual scenario simulations for data generation, together with the selected features are suitable to train ML models capable of infer the health condition of the stagger amplitude, as its RMSE is acceptable and independent of the speed (similar results for the three models). A more extensive explanation of these results is published in (Blanco et al., 2022).

However, there are still some limitations for this approach that have to be solved. For example, three different ML models were created (for discrete speeds), whereas in production, the ML model should be ready to be used for a range of speed values. Next section analyses the relevance of speed factor in the model and justifies a new alternative independent of the speed, before going further with the proposed methodology for ML model selection for edge-based systems.

2.1.1. Model re-factoring: speed variable analysis

To perform the speed factor analysis, the three input datasets have been unified obtaining one with 1800 observations. An extra variable has been included referring to the constant speed at which it has been simulated. For training, the same method used previously was used: 150 models were trained using the RMSE metric as a basis using AutoML. This training has been performed in two different ways in parallel: the first one considering the speed variable and the other one without considering the speed variable.

Table 3 shows the comparison of the best five models among the 150 trained: the results of the training considering the variable speed, and without taking such variable into account.

The results show that the RMSE values are maintained by unifying the three datasets. Furthermore, it can be seen that there is hardly any difference between training with and without the speed variable. To verify the importance of the speed variable, Fig. 2 shows the results of a sensitivity analysis of the 36 features (Y-axis) that has been performed on the 150 trained models (X-axis). It shows the most relevant models, where the sensitivity value is on a colour scale from blue (0 sensitivity) to red (1 sensitivity).

The results show that the speed variable (black box on Y-axis) has a sensitivity value close to zero. The dark blue result means that the sensitivity of the model to this variable is very low, so it has been decided not to include it as an input to the ML model.

Table 2
RMSE results for stagger amplitude ML models.

Model	RMSE [mm]	CI_{95} [mm]
Model 100 km/h	4.5	[3.6, 5.7]
Model 200 km/h	4.9	[4.3, 5.3]
Model 300 km/h	4.6	[3.9, 5.2]

Once having decided that the input features will be those 36 already identified and speed will not be taken into account for the stagger amplitude prediction, the final selected ML model (see Section 4) will be deployed in the edge computing system based on IoT that is explained next.

2.2. IoT based monitoring system

The project has identified several defects that stand out for the interest they arouse in the end users (e.g., infrastructure managers and maintenance service providers), such as, for example, the appearance of cracks in the rail, the stagger or the contact force between pantograph and catenary. For early detection of identified defects, a monitoring system based on IoT technologies has been designed (Errandonea et al., 2021).

The information transmitted from the monitoring system is complemented with positioning information. This way, the events detected by the sensors can be located along the route taken by the train. As another part of the monitoring system implementation, a series of sensors have been installed both on the train wheels and on the pantograph. For example, in the case of the pantograph, some accelerometers are placed at the pantograph head and lower arm (see Fig. 3).

The implemented architecture centralizes all the information generated by the positioning subsystem, pantograph monitoring subsystem and the wheelset monitoring subsystem in a module called DataHub. This module is responsible for collecting and transmitting two types of information to the central server: all the raw data captured from the different systems, and the detection of anomaly events.

Fig. 4 shows the architecture deployed in the development of the DataHub module. The transfer of raw information is done in batches of data in different files and transmitted to the central server via FTP (file transfer protocol). This transfer is done through the Wi-Fi installed in the stations, in order to ensure that the bandwidth is sufficient to send the files and not lose the signal at the time of transmission.

For the notification of anomaly events, a publisher/subscriber system using LTE technologies has been developed. For events messages management, an MQTT broker has been deployed with three topics for the different types of messages: *SIA/pos* for positioning system messages, *SIA/pant* for pantograph system messages and *SIA/aba* for wheelset system messages. Several hierarchical topics have been created to organize the messages transmitted.

Two specific topics have been defined for sending event messages: *SIA/aba/event* and *SIA/pant/event*. The following two sections describe how the event message structure has been defined and how the module that detects the events has been developed. In this work, the focus has been set on developing a module to detect catenary stagger anomalies.

However, there are still some limitations for this approach that need to be solved, such as the interoperability with other existing systems in the rail sector. In the following section, a proposed architecture for CDM-based message exchange is presented.

2.2.1. CDM-based data model for event messages

Within the European Linx4Rail project (LINX4RAIL, 2022), the objective is to address the issue of a missing unified approach towards a Railway System Architecture. To this end, work has been done on the definition of a Common Data Model (CDM), which defines a unified conceptual structure and data model representing the components of the railway system, and the kind of information they exchange during their operation identifying the relations between them and providing a common language and data dictionary to identify them and describe them.

This stagger amplitude diagnosis system makes use of the structure defined for observations made in train operations defined in the CDM. Since the monitoring system has been installed in a passenger train, the measurements were taken during regular services of the train. Fig. 5

Table 3
Speed variable importance comparation.

	Without Speed variable		With Speed variable	
	Model	RMSE [mm]	Model	RMSE [mm]
1	StackedEnsemble_BestOfFamily_7	4.467479	StackedEnsemble_BestOfFamily_8	4.439279
2	GBM_grid_1_model_17	4.581003	StackedEnsemble_BestOfFamily_4	4.545824
3	StackedEnsemble_BestOfFamily_4	4.583752	StackedEnsemble_BestOfFamily_5	4.547984
4	GBM_grid_1_model_15	4.599797	GBM_grid_1_ model_5	4.550151
5	GBM_grid_1_model_5	4.617085	GBM_grid_1_ model_7	4.617055

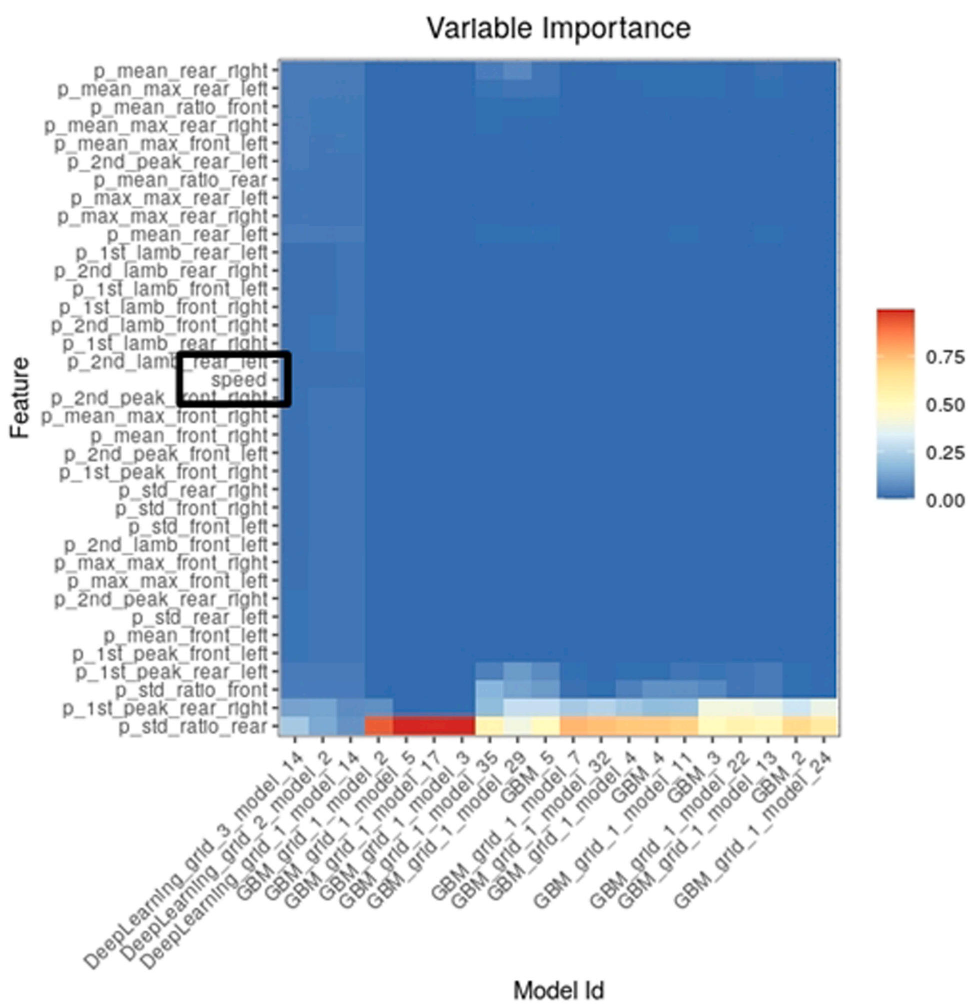

Fig. 2. Speed variable importance in models.

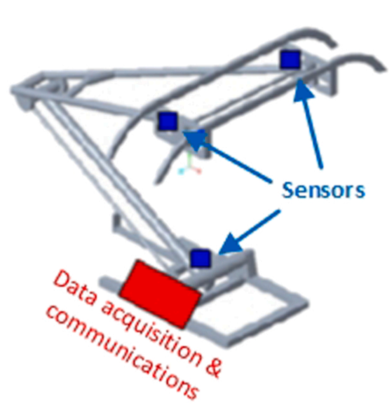

Fig. 3. SIA Pantograph monitoring system.

shows the structure on which the information is exchanged by the monitoring system.

The information exchanged by the developed monitoring system is that of the observations captured by the stagger amplitude diagnosis model. This observation is not collected directly from the physical sensors placed on the pantograph, but through the processing performed by the ML model in the edge (in the vehicle). Therefore, it could be said that the sensor referred to in this observation is virtual and not physical. The observation is completed with the information of the observation ID, the feature of interest, the estimated value, the unit, and the timestamp.

The information provided also relates to an element of the railway infrastructure, in this case the catenary. Information on the position of the catenary must also be submitted, together with some extra information about train operation, train ID, travel direction, etc. With the use of these mechanisms, the interoperability of the designed system with other railway systems is ensured.

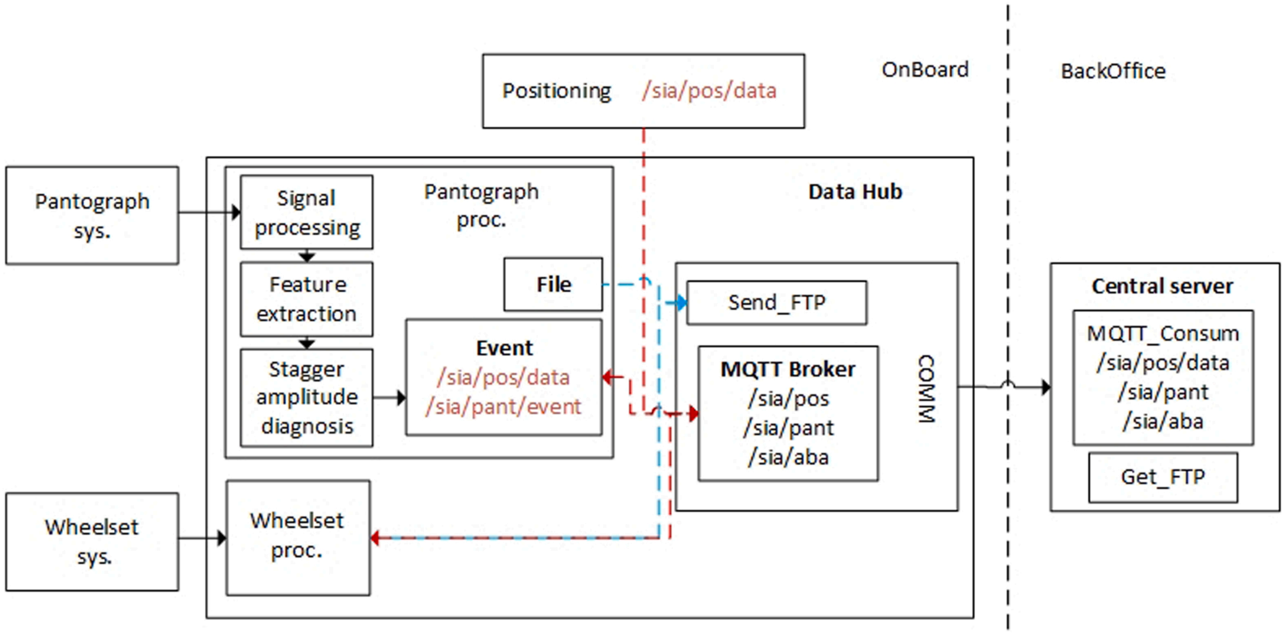

Fig. 4. SIA Data Hub system.

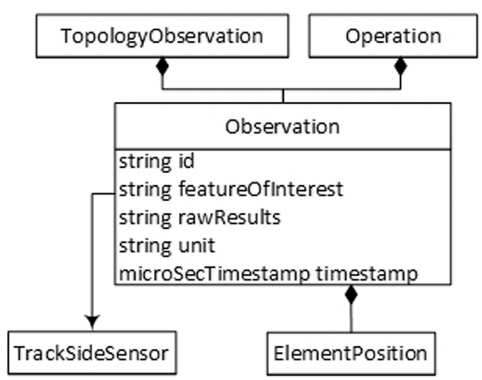

Fig. 5. CDM Data model diagram.

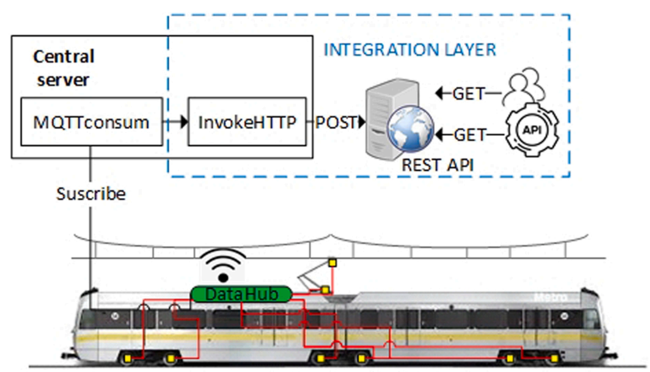

Fig. 6. CDM-based interoperability system.

With the information extracted from the diagnostic system, a message with the structure defined by CDM, described in Fig. 5, is created in JSON format. As defined in the "Integration Layer" of CDM, there are several possibilities for the transmission of such messages. For this work, it was chosen to deploy a REST API for message exchange between systems. Since a publisher/subscriber system for message exchange has already been deployed for our monitoring system, it has been decided to implement a complementary system that would comply with the CDM guidelines.

Fig. 6 shows the architecture deployed to maintain message exchange compatibility with the CDM.

The deployed monitoring system receives all detected events through a subscription to the topic of an MQTT broker. After receiving the message, the message structure is edited to conform to the CDM format. The InvokeHTTP process is the one that establishes connection with the Web server to POST the message to the defined topic. In this way, other applications or users could access this information.

For the integration of ML model into an Edge solution, there are several things to consider. Usually, the most important factor to consider is the response time of the model. Normally a monitoring system has a very high sampling rate, and the equipment does not have a large computational capacity.

3. ML model selection methodology proposal for edge computing

Normally, when ML models are developed to provide a solution to a given problem, the theoretical focus is usually on the model's success rate or accuracy. However, there are times when other problems are encountered when deploying the solution in a practical case. For example, the computational complexity of the model selected for deployment must be considered (Baskakov and Arseniev, 2021; Lee and Chen, 2020). Several works in the literature show the need to take into account the resource consumption of the chosen model as well as the processing time, latency and, of course, accuracy (Tsiropoulou et al., 2017; Yang et al., 2021).

The proposed methodology, presented in Fig. 7, consists of four phases. First, a pre-selection is made among the models considered for the solution by means of the Pareto front definition method. Secondly, a processing sequence is deployed by integrating the pre-selected models into an architecture of smaller Hardware resources. Thirdly, the results obtained are analysed to select the final model. Finally, an analysis of the interpretability of the selected model is performed.

The first phase consists of a preselection among all the trained models. The objective is to get a subset of ML candidate models whose performance analysis will be carried out in more detail. For this

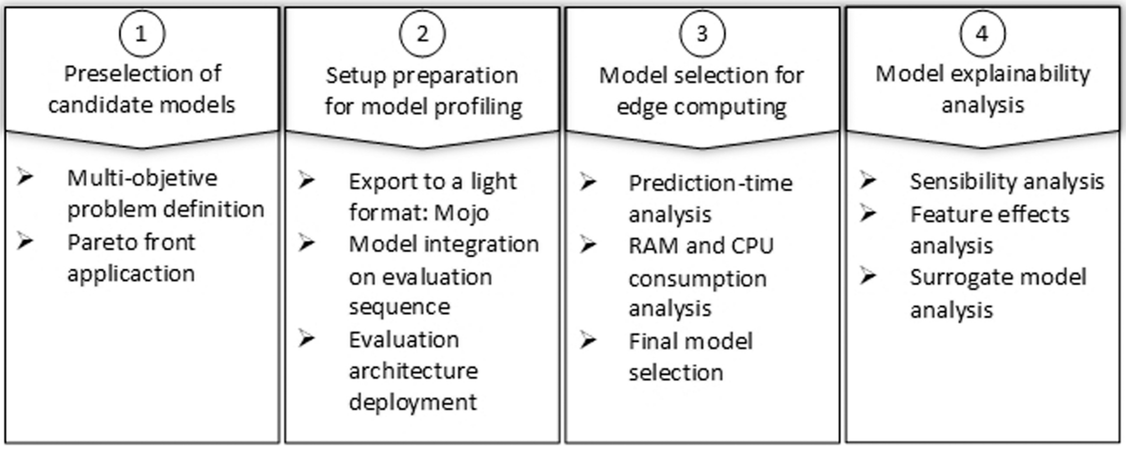

Fig. 7. Methodology diagram.

preselection, the multi-objective problem needs to be defined, which will depend on the type of ML problem that is being addressed and the use case where it is applied. At least one of the metrics will be related to the performance evaluation metric of the ML model: for classifiers, accuracy, confusion matrix, precision, recall, F-Score or AUC (Area Under the Curve)-ROC (Receiver Operating Characteristic curve) might be used. For regression models, others like MAE (Mean Absolute Error), RMSE (Root Mean Squared Error), and RMSLE (Root Mean Squared Logarithmic Error) can be used. It is also interesting to include some metric related to the computational performance (e.g., prediction time). The result of this phase is a list of ML models that will be further analysed.

The second phase consists of the preparation of a setup for the preselected models profiling. The objective is to prepare an evaluation environment (with a range of related hardware (HW) resources) where comprehensive model profiling can be carried out. On the one hand, candidate models need to be exported as they would be for an embedded solution. The software (SW) application for executing the model for different inputs needs to be prepared to be deployed in the HW-scalable setup. On the other hand, the scalable computation infrastructure with monitoring capabilities needs to be prepared. This implies that, in addition to gathering information about model quality metrics, information about HW resources consumption needs also to be gathered. The result of this phase is a test scenario capable of monitoring performance in different hardware capabilities.

The third phase consists of the final selection of the model. Considering the results obtained from the profiling performed in the test scenario defined in the previous phase, a model is selected, which will be deployed in the edge computing solution. This selection is made based on the specific needs of the problem to be addressed. Depending on the context of the solution and the stakeholders involved, the model that better suits the desired solution is selected. As a result of this phase, the selected model to be deployed in the edge computing system is obtained.

The fourth phase consists of an analysis of the model's interpretability. The objective of this phase is to understand in more detail how the selected ML model behaves. One of the drawbacks of the application of ML is the lack of comprehensiveness of the models. In applications aimed at more critical aspects, it is understandable that the user wants to understand how the applied model behaves, instead of treating it as a black box (Hall et al., 2019).

To address this drawback, there are certain methods to describe the behaviour of the selected model. These methods make a ML model more explainable. During this work, three aspects of the selected model have been analysed: the sensitivity of the features, the effect of the features on the target variable and, finally, an intrinsically interpretable ML model.

The sensitivity analysis results in a ranking of the features that have a bigger influence on target variable. Two methods are the most common ones for carrying out this analysis (Interpretable Machine Learning, 2022): the permutation method, and the Shapley values method. The permutation method consists of measuring the importance of a feature by calculating the increase in the model's prediction error after permuting the feature. A feature is "important" if shuffling its values increases the model error, because in this case the model relied on the feature for the prediction. This way, the importance of each of the variables in the prediction of the target variable is observed. The Shapley values method consists in assess every combination of features to determine each impact in the prediction (Lundberg and Lee, 2016).

An analysis of the effects of the different features on the value of the target variable has also been carried out. Specifically, two studies have been performed: Individual Conditional Expectation (ICE) y Partial Dependence (PDP). The PDP shows a curve of the average of how much the prediction varies in response to changes in one of the features, all other features being held constant (Friedman, 2001). The ICE shows how the prediction varies for each of the observations, only modifying the characteristic under study and keeping the rest constant (Goldstein et al., 2014).

As a last analysis, a surrogate model has been developed. It aims at drawing summary conclusions about the original model. This is done by providing an intrinsically interpretable model that approximates the predictions. The selected intrinsically interpretable model is a decision tree. Decision trees are directed graphs in which each interior node corresponds to an input feature. The terminal nodes (or leaf nodes) represent a value of the target variable given the values of the input variables represented by the path from the root to the leaf. To predict the outcome in each leaf node, the average outcome of the training data in this node is used. The paths can be visualized with simple *if-then* rules. In short, decision trees are data-derived flowcharts that follow a Boolean-like logic. As such, they are displayed graphically in a natural way that is easy to interpret.

Variable importance and interactions displayed in the surrogate model are assumed to be indicative of the internal mechanisms of the complex model. Variables that are higher or used more frequently are more relevant. Variables that are above and below one another can have interactions. The decision tree surrogate model has a global focus of interpretation. Nonetheless, local behaviour can also be visualized by highlighting the paths of specific instances through the internal nodes. By using the three analyses defined for this phase, more intuitive results are obtained for gaining more understanding about the behaviour of the model selected.

The proposed methodology not only addresses the problem from a multi-objective point of view, but also presents an evaluation and selection architecture based on the interaction of the model in environments closer to reality. It provides a systematic methodology for the selection of models, knowing its behaviour in the deployment machine, from a logical and physical point of view. The following section describes a practical use case, oriented to the diagnosis of catenary stagger amplitude, giving more detailed information on each of the steps established by the proposed methodology.

4. Use case: stagger amplitude diagnosis edge solution

In this section, a practical step-by-step use case of the methodology is presented regarding the stagger amplitude diagnosis ML model selection for edge computing. A detailed description of each of the phases of the methodology is given, showing the results obtained.

4.1. Preselection of ML models

Using H2O.ia's AutoML tool, 150 trained models of different typologies have been obtained: deep learning models, gradient boosting machines, generalized linear models and distributed random forests. In addition, it makes use of the stacked ensemble technique to improve overall model accuracy (Wolpert, 1992).

Among the 150 trained models, H2O provides several variables commonly used to determine the accuracy of the model. Since it is a regressor model, of all the variables provided, the RMSE (see (1)) is taken as a reference to determine the accuracy of the model. To help the user, H2O provides two extra variables for each trained model to evaluate the computational complexity of the model: the training time in milliseconds and the prediction time per row in milliseconds. Considering that the objective is to analyse the behaviour of the model in its final deployment, the prediction time per row is taken as the second variable of interest.

Considering the above, it can be said that we are facing a multi-objective problem, since we want to minimize the model error (RMSE) and also minimize the model execution time. For this reason, an analysis has been carried out to identify, among the 150 trained models, those that mark the frontier to achieve the two objectives.

Fig. 8 shows the relationship between the two variables involved in the multi-objective problem. The x-axis shows RMSE, the error of each of the trained models. The y-axis shows the prediction time per row. For this figure, models with a RMSE greater than 6 mm and prediction execution time higher than 0.1 ms have been filtered out.

The red line that is defined is the one corresponding to the Pareto front (Roocks, 2016). It is the boundary that marks the solution to the multi-objective problem of minimizing the error and execution time of the model. Considering the interest of the railway sector, the preselected ML models for further analysis are those from the pareto front with the lowest RMSE values. Table 4 shows the specific values of RMSE and prediction time in milliseconds for the selected candidate models.

It has also been considered that the prediction times given by the H2O cluster depend on the resources of the machine. For the training of the different models, a Docker container with R server has been deployed. The computer where the container is deployed has the following characteristics is an Intel® Core™ i7–8700 CPU @ 3.2 GHz 3.19 GHz with a RAM of 32 GB. The resources allocated to Docker are 5 CPUs, 26.50 GB of memory and 1.5 GB of swap.

The model cannot be selected without considering how it would behave on machines with tighter resources than the cluster where it has been trained. Therefore, an analysis and comparison between the behaviour of the selected models on different machines with different resources must be performed. Next step describes how this analysis can be done in order to carry out a model profiling.

4.2. Setup preparation for model profiling

As explained in the previous section, in order to carry out the candidate models' profiling, the specific test setup needs to be prepared both in terms of SW application and the computation infrastructure, together with the corresponding monitoring capabilities for model profiling.

For the model profiling, the RMSE as the quality metric and prediction time per row, the RAM and CPU consumption in percentages of use are considered. A summary of the evaluation criteria and their units can be found in Table 5.

First of all, the trained candidate models have been exported to a format called MOJO (Model Object Optimized), which is specific to H2O, due to the following advantages (Kraljevic, 2016):

- The resulting artifact does not need to be compiled.

Table 4
Pareto front defining models features.

	Model	RMSE [mm]	Prediction time [ms]
1	StackedEnsemble_BestOfFamily_7	4.467479	0.014873
2	GBM_grid_1_model_17	4.581003	0.012007
3	GBM_grid_1_model_15	4.599797	0.010886
4	GBM_grid_1_model_5	4.617085	0.009256
5	GBM_grid_1_model_27	4.650621	0.008230

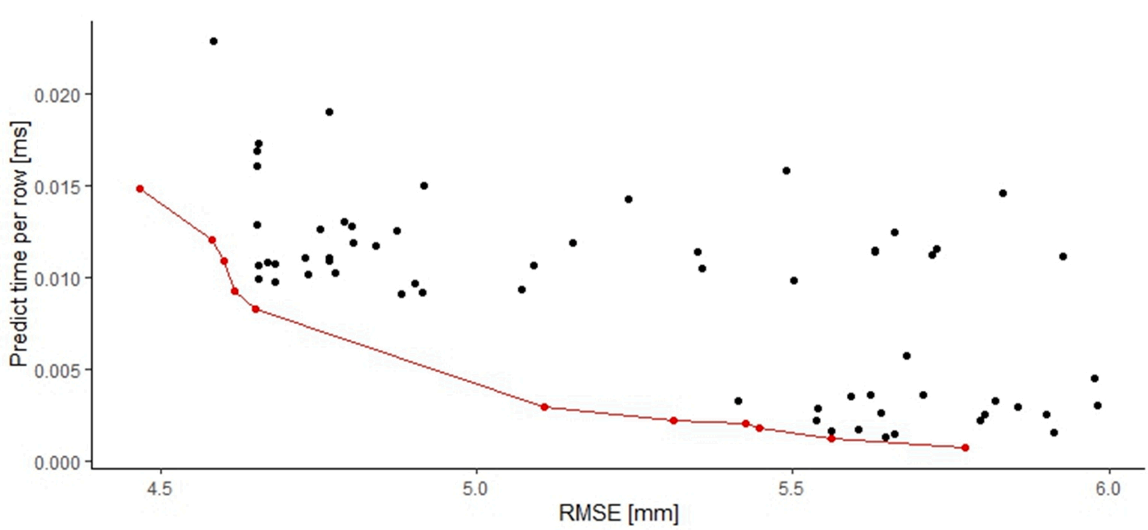

Fig. 8. Pareto front.

Table 5
Evaluation criteria summary.

Variable	Units
RMSE	mm
Predict time per row	ms
RAM consumption	%
CPU consumption	%

- It is supported by the Java runtime.
- Low latency, perfect for real-time applications.
- Efficient working in a row per time, perfect for streaming applications.

To perform the analysis, the exported models have been inserted into a Java application (see Fig. 9). This Java application reads the five selected models in MOJO format and performs a batch of 1000 predictions for each one of them, to take measurements of prediction execution times per observation and per batch of 1000 observations.

A Docker image has been created with the compiled Java application. This Docker image deploys a container with Alphine operating system with the java JDK (version 1.8). It executes the JAR for the program to perform the 5000 predictions (5 models with 1000 predictions each), while making measurements of execution times (per prediction and per 1000 predictions), RAM consumption and CPU consumption.

To perform the comparisons with different compute resources, an architecture has been deployed on AWS (Amazon Web Services), making use of ECR (Elastic Container Registry), ECS (Elastic container Services) and S3 (Simple storage service) services (see Fig. 10).

ECR is a repository to store and manage Docker container images where the configured Docker image has been stored. This service is complementary to the ECS service, since it makes use of the images registered in this repository. This service deploys Docker containers in different Amazon EC2 (Elastic compute cloud) instances of the characteristics determined by the user. For this work, five EC2 instances have been deployed and they are described in Table 6:

Finally, all measurements performed on the different instances are collected in S3, a service provided by AWS for data storage.

4.3. Final model selection for edge computing

The RAM memory and CPU consumption readings are shown in Fig. 11: on the y-axis shows the consumption percentage and on the x-axis the time. It shows that the higher the RAM and CPU capacity, the less both resources are saturated. It is also observed that when the RAM and CPU resources are doubled, the RAM memory becomes half as saturated. This shows that the bottleneck is the CPU and not the RAM memory. On the other hand, the CPU still maintains a significant saturation level, but the execution time improves considerably.

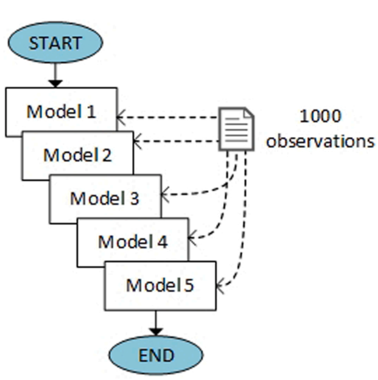

Fig. 9. Execution sequence.

On the other hand, the Fig. 12 shows the results of the models in prediction execution times: by batches of 1000 and the average per observation and model.

The results show that machines with more resources such as t2.large and t2.medium help to sustain a better prediction execution time. Probably the reason is that these instances have two CPUs and more RAM. In the comparison of the execution times per row with those given by the H2O cluster, the reason for the time difference is the machine resources.

If the difference in prediction execution times is analysed, the ensembled model (model 1) has a higher mean than the GBMs. Looking at the batch results of 1000 predictions, the ensembled model (model 1) shows times around 1.5 s or below 1 s, if the machine has more than 2 CPUs and 4 GB of RAM. In contrast, rest of models show times below of 400 ms or below 200 ms with better machines.

In a monitoring system installed in a passenger train running at a maximum speed of 80 km/h, two approaches are possible: batch or row-by-row predicting. In the first, considering the sampling frequency of the sensors is very high (e.g., 200 Hz) and continuous during the entire route, it could be estimated to perform 1000 samples in 5 s. In that case the model would respond in 30% of the capture time. In the second case, the times of any model are below 0.3 ms. Both approximations estimate a good prediction execution time for either model, therefore, the best model would be model 1 (StackedEnsemble_BestOfFamily_7, with 4.46 mm of RMSE), thus prioritising minimising the RMSE.

Once the model to be implemented in the monitoring system has been selected, a more exhaustive analysis of the model's behaviour has been performed. The following section shows different characteristics of the selected model.

4.4. Machine Learning model explainability analysis

In the fourth phase of the proposed methodology, three types of analysis are defined, in order to have a better comprehension of the selected model, and the one that will finally be deployed in the solution. The first of the analyses to be carried out is the sensitivity analysis. To do this, IML (Interpretable Machine Learning) library is used, which provides several tools for the analysis (Molnar et al., 2018).

Fig. 13 relates the results of the permutation method (x-axis) to the results of the Shapley values method (y-axis). On both axes, the normalised results are shown, where 0 represents the lowest sensitivity and 1 the highest. They show of the sensitivity of the 36 features, only naming the two most important ones.

The results show that for both methods the two most sensitive variables are the following: p_std_ratio_rear, p_std_ratio_front. These variables are the standard deviation of the Rt_{stag}, which is the ratio between SAWP (Scale-Averaged Wavelet Power) from right and left accelerometers. The expressions rear and front refer to the front or rear strips of the pantograph.

The second analysis to be carried out is the analysis of the features effects. To perform this exercise, IML (Interpretable Machine Learning) library is used for this analysis too. In Fig. 14, the effect of the most important feature detected above is studied: p_std_ratio_rear. For this purpose, the ICE and PDP curves are plotted, with the study feature on the x-axis and the predicted value on the y-axis.

The results show that when the value of the feature (p_std_ratio_rear) increases, the prediction value increases as well. The ICE curves show in more detail, with what probability the result can vary. However, in all observations, the trend is the same. It can also be observed that stagger amplitude is more sensitive to the lower values of feature.

The third analysis consists of developing a subrogated model, in this case a decision tree model, to analyse the result. In this case, the tool used to carry out the analysis is Rpart (Therneau and Atkinson, 2022). The decision tree surrogate model is trained on the original inputs and predictions of the final stack ensemble model already selected in the previous section. A depth of four nodes is chosen as a trade-off between

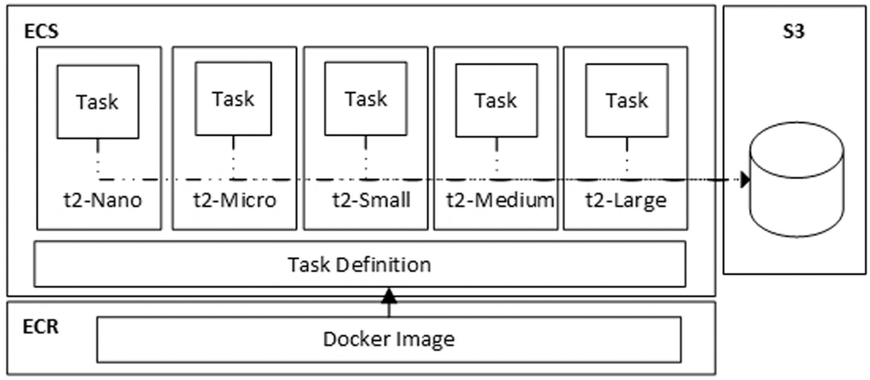

Fig. 10. Performance evaluation architecture.

Table 6
Deployed EC2 instance description.

EC2 Instance	CPU [3.3 GHz]	RAM [GB]
t2.nano	1	0.5
t2.micro	1	1
t2.small	1	1.5
t2.medium	2	4
t2.large	2	8

accuracy and interpretability. The decision tree is tuned by cross-validation. The decision tree is shown in Fig. 15.

The figure shows how the value of the stagger amplitude is classified, considering the different depth nodes of the decision tree. For example, if no variable was taken into account in the first depth node, the model would only assign the value of 0.13 as stagger amplitude prediction to 100% of the observations, in this case 1800.

Considering the depth of four nodes that have been assigned to it, all decisions of the surrogate model are dependent on the same feature: p_std_ratio_rear. This is natural, considering the importance it has demonstrated in previous analyses. The fact that the other features do not appear does not mean that they have no effect on the prediction of the original model. It simply means that the surrogate model did not consider it relevant enough to incorporate it.

5. Conclusions

This paper presents a methodology identifying the process to be followed to implement a railway catenary stagger amplitude diagnosis model in an onboard edge computing solution in-service trains. The

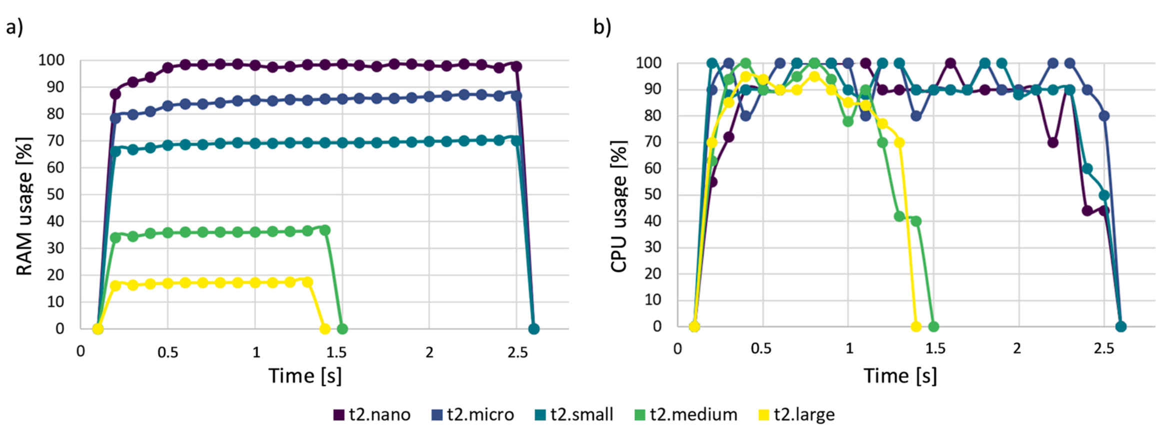

Fig. 11. a) RAM consumption and b) CPU consumption.

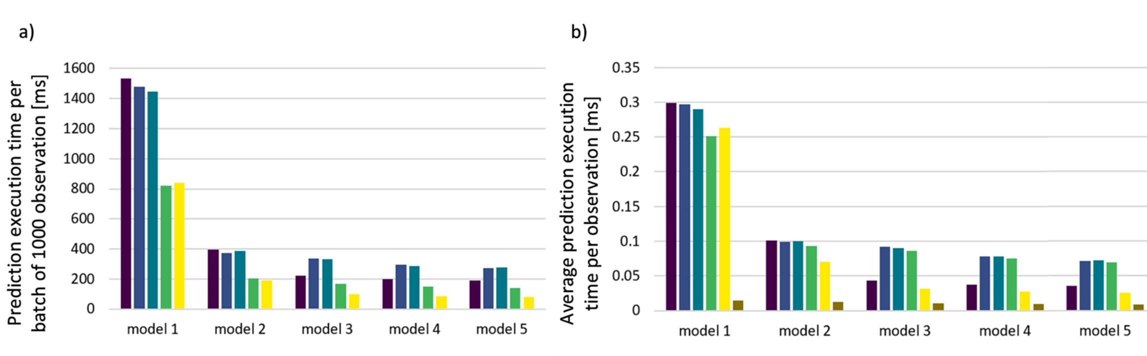

Fig. 12. a) Prediction Execution time for 1000 observation b) average prediction execution time per observation.

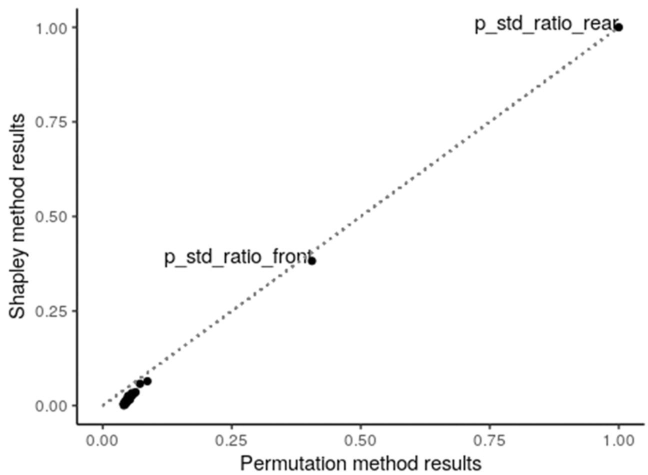

Fig. 13. Sensibility analysis.

presented pipeline is divided into four parts: a) the presentation of the multi-objective problem and the pre-selection of the model, b) presentation of the evaluation architecture, c) the analysis of the profiling results of the different models for the selection of the most suitable one, and d) the interpretation of the selected model.

For this purpose, the previous work on which the proposal is based has been presented. The design of the monitoring system developed has been described. Then, the mechanisms used for the interoperability of the system with other railway systems have been described (LINX4RAIL, 2022) and, finally, the methodology followed for the development of the stagger amplitude diagnosis model has also been reported.

Once the previous steps have been analysed, the next step has been to refactor the model to fit the real context. Thanks to this refactoring, it has been possible to obtain a model that is speed-independent, and thus more applicable to different scenarios. Considering the objective of integrating the selected ML model into an Edge solution, it has been necessary to analyse the problem from a multi-objective perspective. In conclusion, five models that meet the objectives of minimizing model error and minimizing execution time have been pre-selected. Following the analysis, 5 models has been pre-selected: StackedEnsemble_BestOfFamily_7, GBM_grid_1_model_17, GBM_grid_1_model_15, GBM_grid_1_model_5 and GBM_grid_1_model_27. All have obtained RMSE results of less than 4.7 mm and prediction execution times of less than 0.015 ms per observation.

The next step has been to perform a profiling of the pre-selected models. A test environment has been deployed to measure RAM consumption, CPU consumption and execution times. With the results obtained, the model with the best characteristics to implement in an Edge solution has been selected. It has also been observed that machines with higher CPU and RAM availability may be a better option to improve execution time. The model selected was StackedEnsemble_BestOfFamily_7 (model 1), with an RMSE of 4.46 mm. The prediction execution time in batches of 1000 observations has been less than 1.5 s in all instances. It is suggested to use

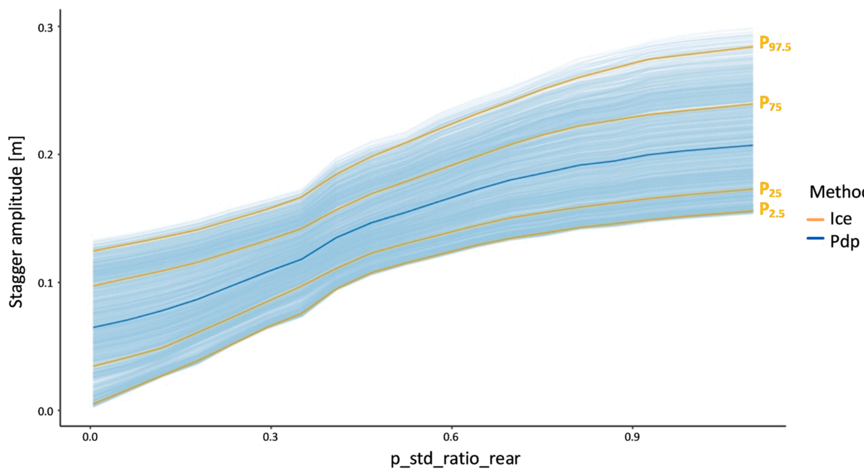

Fig. 14. PDP and ICE.

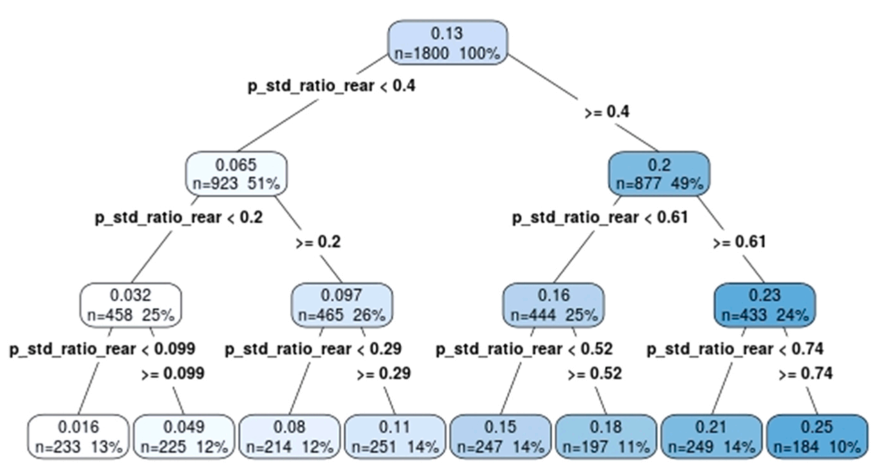

Fig. 15. Surrogate model.

machines similar to the characteristics of instances such as t2.medium or t2.large, as it has been shown that prediction execution times can fall around 800 ms.

Finally, the selected model has been analysed by various methods to provide interpretability to its behaviour. In this way, an interpretable model is obtained instead of a black box, giving greater reliability in its application as a solution to the early infrastructure diagnosis in railway infrastructure. It has been observed that the sensitivity of the model for the characteristic p_std_ratio_rear marks its behaviour.

In conclusion, a procedure has been defined for the selection of a ML model to be integrated into an edge solution. Considering the characteristics of the procedure, it is applicable to any context where the objective is similar to the one analysed here as a use case.

Several possible future works have been identified to continue with this research. The first one would be to apply the methodology here proposed to other failure modes of interest. For example, also related to the catenary, this methodology could be used to create an edge-computing system for steady arm diagnosis. It would also be interesting to apply it for the diagnosis of other components failure modes as part of an inspection system. For example, track inspection system.

Another line of future work is to complement certain aspects of the methodology proposed in this paper. New explainable AI methods could be included in the fourth phase of the methodology. Another option is to develop a framework to automate the execution of the methodology. As a final step, this framework would automatically deploy the selected model in the on-board system, following operations defined by MLOps strategies.

Finally, other future work identified would be to develop a system to complete the event information provided by the IoT system (diagnosis and position), with complementary information of the asset itself.

Funding

This project has received partial funding from the European Union's Horizon 2020 research and innovation programme and from the European Union Agency for the Space Program (EUSPA) under grant agreement No. 776402. Also, received partial funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 881826.

CRediT authorship contribution statement

Itxaro Errandonea: Conceptualization, Methodology, Investigation, Resources, Visualization, Software, Validation, Data curation, Writing – original draft, Writing – review & editing. **Pablo Ciáurriz**: Resources, Investigation, Visualization, Writing – original draft, Writing – review & editing. **Unai Alvarado**: Resources, Investigation, Writing – review & editing, Project administration and Funding acquisition. **Sergio Beltrán:** Conceptualization, Methodology, Formal analysis, Writing – review & editing. **Saioa Arrizabalaga**: Conceptualization, Methodology, Data curation, Supervision, Writing – review & editing, Project administration, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

About EUSPA | EU Agency for the Space Programme. ⟨https://www.euspa.europa.eu/about/about-euspa⟩ (Accessed Jun. 07, 2022).

Baskakov, D., Arseniev, D., 2021. On the computational complexity of deep learning algorithms. Smart Innov. Syst. Technol. vol. 220, 343–356. https://doi.org/10.1007/978-981-33-6632-9_30.

Blanco, B., Errandonea, I., Beltrán, S., Arrizabalaga, S., Alvarado, U., 2022. Panhead accelerations-based methodology for monitoring the stagger in overhead contact line systems. Mech. Mach. Theory vol. 171, 104742. https://doi.org/10.1016/J.MECHMACHTHEORY.2022.104742.

Cho, C.J., Ko, H., 2015. Video-based dynamic stagger measurement of railway overhead power lines using rotation-invariant feature matching. IEEE Trans. Intell. Transp. Syst. vol. 16 (3), 1294–1304. https://doi.org/10.1109/TITS.2014.2361647.

Digital Twin MarketGlobal Industry Trends, Share, Size, Growth, Opportunity and Forecast 2022–2027. ⟨https://www.researchandmarkets.com/reports/5562576/digital-twin-market-global-industry-trends?utm_source=CI&utm_medium=PressRelease&utm_code=73f6ft&utm_campaign=1683648+-+Global+Digital+Twin+Market+(2022+to+2027)+-+Industry+Trends%2C+Share%2C+Size%2C+Growt%2C+Opportunity+and+Forecasts&utm_exec=jamu273prd⟩ (Accessed May 30, 2022).

Errandonea, I., Beltrán, S., Arrizabalaga, S., 2020. Digital Twin for maintenance: a literature review. Comput. Ind. vol. 123 https://doi.org/10.1016/j.compind.2020.103316.

Errandonea, J. Goya, U. Alvarado, S. Beltrán, and S. Arrizabalaga, IoT Approach for Intelligent Data Acquisition for Enabling Digital Twins in the Railway Sector, in 2021 International Symposium on Computer Science and Intelligent Controls (ISCSIC), Nov. 2021, pp. 164–168, doi: 10.1109/ISCSIC54682.2021.00039.

Friedman, J.H., 2001. Greedy function approximation: a gradient boosting machine. Ann. Stat. vol. 29 (5), 1189–1232. https://doi.org/10.1214/aos/1013203451.

A. Goldstein, A. Kapelner, J. Bleich, E. Pitkin, Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation, 2014.

T. Kraljevic, "Ways to productionize H2O", 2016 https://github.com/h2oai/h2o-meetups/blob/master/2016_07_19_H2O_Open_Tour_NYC_Prod/Prod16x9.pdf (accessed May 26, 2022).

P. Hall, H. Ai Washington, N. Schmidt, and L. Philadelphia, Proposed Guidelines for the Responsible Use of Explainable Machine Learning, Jun. 2019, doi: 10.48550/arxiv.1906.03533.

A. Hodzic, D. Skulj, and A. Causevic, Data-driven Anomaly Detection for Railway Propulsion Control Systems, IECON Proc. (Industrial Electron. Conf., vol. 2020-October, pp. 4351–4356, Oct. 2020, doi: 10.1109/IECON43393.2020.9255026.

Home - SIA project. ⟨https://siaproject.eu/⟩ (Accessed May 30, 2022).

F. Hutter, L. Kotthoff, and J. Vanschoren, Automated Machine Learning, p. 219, 2019, doi: 10.1007/978-3-030-05318-5.

Interpretable Machine Learning. ⟨https://christophm.github.io/interpretable-ML-book/⟩ (Accessed Jun. 08, 2022).

S. Kang, S. Sristi, J. Karachiwala, and Y.C. Hu, Detection of anomaly in train speed for intelligent railway systems, 2018 Int. Conf. Control. Autom. Diagnosis, ICCAD 2018, 2018, doi: 10.1109/CADIAG.2018.8751374.

Karakose, E., Gencoglu, M.T., Karakose, M., Member, S., 2017. A new experimental approach using image processing-based tracking for an efficient fault diagnosis in pantograph–catenary systems. IEEE Trans. Ind. Informatics vol. 13 (2), 635–643.

R. Lee and I.Y. Chen, The Time Complexity Analysis of Neural Network Model Configurations, Proc. - 2nd Int. Conf. Math. Comput. Sci. Eng. MACISE 2020, pp. 178–183, 2020, doi: 10.1109/MACISE49704.2020.00039.

LINX4RAIL. ⟨https://projects.shift2rail.org/s2r_ipx_n.aspx?p=LINX4RAIL⟩ (Accessed May 30, 2022).

Z. Liu, C. Jin, W. Jin, J. Lee, … Z. Z.-… on prognostics and, and undefined 2018, Industrial AI enabled prognostics for high-speed railway systems, ieeexplore.ieee.org, Accessed: Jul. 20, 2022. [Online]. Available: ⟨https://ieeexplore.ieee.org/abstract/document/8448431/⟩.

S.M. Lundberg and S.-I. Lee, An unexpected unity among methods for interpreting model predictions, 2016, doi: 10.48550/arxiv.1611.07478.

Molnar, C., Casalicchio, G., Bischl, B., 2018. iml: An R package for Interpretable Machine Learning. J. Open Source Softw. vol. 3 (26), 786. https://doi.org/10.21105/JOSS.00786.

Publication: The Future of Rail – Opportunities for energy and the environment - Event - IEA. ⟨https://www.iea.org/events/the-future-of-rail-opportunities-for-energy-and-the-environment⟩ (Accessed May 30, 2022b).

Rabatel, J., Bringay, S., 2009. P. P.-I. C. on D. Mining, and undefined 2009, SO_MAD: Sensor Mining for Anomaly Detection in Railway Data. Springer, pp. 191–205. https://doi.org/10.1007/978-3-642-03067-3_16ï.

Roocks, P., 2016. Computing pareto frontiers and database preferences with the rPref package. R J. 8 (2), 394–405. https://doi.org/10.32614/rj-2016-054.

de S. Soares, E.F., Carlos, C.A., de Lucena, S.C., 2019. Online travel mode detection method using automated machine learning and feature engineering. Futur. Gener. Comput. Syst. vol. 101, 1201–1212. https://doi.org/10.1016/J.FUTURE.2019.07.056.

Tao, F., Zhang, M., 2017. Digital Twin shop-floor: a new shop-floor paradigm towards smart manufacturing. IEEE Access vol. 5, 20418–20427. https://doi.org/10.1109/ACCESS.2017.2756069.

Tao, F., Zhang, H., Liu, A., Nee, A.Y.C., 2019. Digital Twin in Industry: State-of-the-Art. IEEE Trans. Ind. Informatics vol. 15 (4), 2405–2415. https://doi.org/10.1109/TII.2018.2873186.

Teng, F., Zhu, R., Zhou, Y., Chi, M., Zhang, H., 2021. A lightweight model of wheel-rail force inversion for railway vehicles. Concurr. Comput. Pract. Exp., e6443 https://doi.org/10.1002/CPE.6443.

T. Therneau and B. Atkinson, rpart: Recursive Partitioning and Regression Trees. 2022, [Online]. Available: ⟨https://cran.r-project.org/package=rpart⟩.

E.E. Tsiropoulou, S.T. Paruchuri, and J.S. Baras, Interest, energy and physical-aware coalition formation and resource allocation in smart IoT applications, 2017 51st Annu. Conf. Inf. Sci. Syst. CISS 2017, 2017, doi: 10.1109/CISS.2017.7926111.

Using analytics to get European rail maintenance on track | McKinsey. ⟨https://www.mckinsey.com/industries/public-and-social-sector/our-insights/using-analytics-to-get-european-rail-maintenance-on-track⟩ (Accessed May 30, 2022).
Wang, H., Nunez, A., Liu, Z., Zhang, D., Dollevoet, R., 2020. A Bayesian network approach for condition monitoring of high-speed railway catenaries. IEEE Trans. Intell. Transp. Syst. vol. 21 (10), 4037–4051. https://doi.org/10.1109/TITS.2019.2934346.
Wolpert, D.H., 1992. Stacked generalization. Neural Networks vol. 5 (2), 241–259. https://doi.org/10.1016/S0893-6080(05)80023-1.
Yang, X., Zhou, N., Liu, Y., Quan, W., Lu, X., Zhang, W., 2020. Online pantograph-catenary contact point detection in complicated background based on multiple strategies. IEEE Access vol. 8. https://doi.org/10.1109/ACCESS.2020.3042535.
Yang, Z., et al., 2021. Efficient resource-aware convolutional neural architecture search for edge computing with pareto-bayesian optimization. Sensors Vol. 21, 444. https://doi.org/10.3390/S21020444.
Zhang, D., Gao, S., Yu, L., Kang, G., Zhan, D., Wei, X., 2020. A robust pantograph-catenary interaction condition monitoring method based on deep convolutional network. IEEE Trans. Instrum. Meas. vol. 69 (5), 1920–1929. https://doi.org/10.1109/TIM.2019.2920721.

Itxaro Errandonea her M.Sc. degree in Computer Science at the Faculty of Informatics in San Sebastian (University of the Basque Country) in 2017. Since then, she has been working at Ceit-BRTA in San Sebastian as a research assistant in Information and Communication Technologies (ICTs). His research activity focuses on data analysis and information management. She has participated in research projects studying new methodologies for the digitization, monitoring, and diagnosis of asset health status for the railway sector. She is author of three articles in JCR journals and two international conference papers. She is currently developing her doctoral thesis at the University of Navarra on new methodologies for maintenance using digital twins.

Pablo Ciáurriz received the M.Sc. and Ph.D. degrees in mechanical engineering from the University of Navarra, San Sebastián, Spain, in 2010 and 2014 respectively. Since 2015, he is a researcher within the Transport and Energy Division at Ceit, San Sebastián, Spain. His main research interests include railway systems modelling, energy consumption optimization and energy efficient driving strategies.

Unai Alvarado is a senior researcher at Ceit, where he is the director of the Railway Research Group (Transport and Energy Division). He is also lecturer at Tecnun, the School of Engineering of the University of Navarra. He received his M. Sc. degree in Electrical Engineering in 2003 and his Ph.D. in Electronics and Communications in 2007 by the University of Navarra. His research interests have spanned from the design and development of ASICs for communications applications, to systems engineering. Now his research is oriented towards the digitalization of the railway system; in particular, to new methodologies and approaches towards the optimization of the maintenance of railway systems (e.g. railway infrastructure monitoring&inspection systems, anomaly detection and early prediction of failures, etc.). He has co-authored more than 50 papers in scientific journals and international conferences.

Sergio Beltrán received the B.Sc and Ph.D. degree in industrial engineering from the University of Navarra, Spain, in 2006 and 2013, respectively. He is currently an Associate Professor of Tecnun (University of Navarra). He has published papers in international journals, conferences, and book chapters. Since 2006, he has been working at Ceit research center carrying out applied R&D projects and technical studies in data analysis and information management. His main areas of research are applied data intelligence, mathematical modeling, simulation, optimization, system identification and control and automation techniques.

Saioa Arrizabalaga received her M.Sc. degree in Telecommunication Engineering from the Faculty of Engineering in Bilbao (University of the Basque Country) in 2003, and her Ph.D. degree in Engineering from TECNUN in 2009. She is a lecturer at TECNUN (University of Navarra), and a researcher at Ceit-IK4. She is the Head of Data Analytics and Information management Research group and is involved in the participation of European research projects, direction of doctoral theses, scientific and technical publications in national and international journals and conferences.