ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

Simple Non Regressive Informed Machine Learning Model for Predictive Maintenance of Railway Critical Assets *

Simone Minisi[1], Andrea Garrone[2], Luca Oneto[2], Renzo Canepa[3], Carlo Dambra[1] Davide Anguita[2]

[1]ZenaByte s.r.l., Via Cesarea 2, 16121, Genova, Italy
[2]University of Genoa, Via Opera Pia 11a, 16145, Genova, Italy
[3]Rete Ferroviaria Italiana, Via Don Vincenzo Minetti 6/5, 16126, Genova, Italy

Abstract. Signals, track circuits, switches, and relay rooms are simultaneously the most critical and most maintained railway assets. A fault of one of these assets may strongly reduce the railway network capacity or even disrupt the circulation. Effectively predicting what assets may need maintenance allows to anticipate the intervention thus avoiding a failure. Currently, this problem is tackled by infrastructure managers mostly relying on operators' experience and with limited support of decision supporting tools. In this paper, we propose a Simple Informed Machine Learning (ML) based model able to automatically predict what asset need to be maintained fully leveraging on the operator experience. However, ML models in modern industrial MLOps pipelines demand continuous data collection, model re-training, testing, and monitoring, creating a large technical debt. In fact, one of the main requirements of these pipelines is to not be regressive, i.e., not simply improve average performances but also not incorrectly predicting an output that was correctly classified by the reference model (negative flips). In this work we face this problem by empowering the proposed ML with Non Regressive properties. Results on real data coming from a portion of an Italian Railway Network managed by Rete Ferroviaria Italiana, the Italian Infrastructure Manager, will support our proposal.

1 Introduction

The classical approach to maintenance of railway assets consists in a combination of both Corrective Maintenance (i.e., maintenance aimed at resolving a failure) and Preventive Maintenance (i.e., maintenance aimed at avoiding the failure), primarily relying on the operators' experience with limited support of artificial intelligence [1]. In particular, Preventive Maintenance activities are usually scheduled according to both manufacturer's or legislation's prescriptions and the experience of the operators and normally are performed without disturbing the traffic. Corrective Maintenance activities, instead, are scheduled every time a failure occurs and typically require track possession or speed reductions, thus heavily impacting railway traffic.

*This research received funding from the Shift2Rail Joint Undertaking (JU) under grant agreements No. 881574 (IN2SMART2) and No. 101008913 (DAYDREAMS). The JU receives support from the European Union's Horizon 2020 research and innovation programme and the Shift2Rail JU members other than the Union.

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

Faults on Signals, Track Circuits, Switches, Switch Maneuvers, and Relay Rooms [1] are responsible for a large part of the disruptions generated by signalling assets on the Italian Railway network often requiring Corrective Maintenance.

Therefore, the first objective is to focus on this set of assets and to predict if a failure may occur in a two-weeks window thus allowing Predictive Maintenance without interfering with railway traffic.

For this purpose, we will leverage on a Simple Informed Machine Learning (ML) based model [2–5]. As we will describe later, our problem is characterized by two main issues: data regarding faults are scarce and the resulting dataset is strongly unbalanced. For these reasons, we needed to keep the approach as simple as possible [5]. Hence, in our application we will rely on the simple yet effective Shallow ML model, the XGBoost [6] carefully tuned with rigorous statistical procedures [7], which also easily allow to being informed [3, 4] with the domain knowledge and the experience of the operators by careful handcrafting feature able to fully and synthetically represent the operators experience. Then, methods to mitigate problems related to the unbalance of the dataset will be exploited [8].

Nevertheless, in real, high-stake, and mission-critical applications like the one we are facing in this paper, building the initial ML model is only the beginning of the process. Maintenance of ML models in modern industrial development pipelines (known as MLOps [9]) requires implementing processes for continuous data collection, model updating, and monitoring, thereby creating a large technical debt [10]. New models usually improve the overall performance accuracy (i.e., the average number of error). Nevertheless, they can still introduce errors on specific predictions that the previous versions of the models did not make (the so called negative flips) showing a so called regressive behavior [11]. ML models regression can cause post-processing pipelines to break, requiring specialized maintenance interventions, since modern ML empowered architectures contain several components in addition to ML-based modules. For this reason, in this paper, we propose to empower our Simple Informed ML model for Prescriptive Maintenance of TCs with non regressive properties building a Simple Non Regressive Informed ML model.

In order to test the quality of our proposal we will exploit three years of historical maintenance data about a subsection of the Italian Railway Network[2].

2 Problem Formalization and Available Data

The problem that we face in this paper to predict whether an asset needs to be maintained in the next two weeks to avoid a failure occurrence. For this purpose we will exploiting data coming from different heterogeneous sources: (i) assets failures data, (ii) assets characteristics, (iii) train traffic, (iv) past maintenance activities, and (v) weather data. Data exploited in this research are related to three years (2018 to 2020 full years) and are coming from a subsection

[1] For more details on railway assets please refer to `https://www.orr.gov.uk/glossary`

[2] All data and some steps have been anonymized through the paper (i.e., name of the subsection of the Italian Railway, the number of assets, the number of faults, etc.) because of confidentiality issues.

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

of the Italian Railway Network managed by Rete Ferroviaria Italiana (the Italian Infrastructure Manager) and from the Agenzia Regionale per la Protezione dell'Ambiente Ligure (the Regional Agency for the Environment Protection). Assets failures data give information about the occurrence and the duration of each failure. Assets characteristic data provide information about installation date, manufacturer, and other information specific for each asset type. Train traffic data consist in the list of train passages through the locations where the assets operate. The past maintenance activities contain past maintenance interventions along with their duration and cost. Weather data cover precipitation, temperature, wind, and humidity measurements collected and predicted on a network of 188 weather station spread over the considered geographical region. Failures are not frequent, and perceptually a limited number of failures are available making the problem strongly imbalanced, i.e., most of the time there will be no failure.

3 Simple Non Regressive Informed Data Driven Model

The proposed Simple Non Regressive Informed Data Driven Model will be constructed in five steps: (i) Engineering the Features to Inform the ML models, (ii) Choosing a Simple ML model[3], (iii) Empowering the ML model with the ability of effectively handling unbalanced datasets, (iv) Empowering the ML model of Non Regressive properties during models updates[4], and (v) Tuning (the hyperparameters) and assessing the performance of the ML model.

Regarding Step (i), starting from the raw data provided by Rete Ferroviaria Italiana, we transformed the data in order to obtain a structure that was better suitable for our analysis. In particular we engineered a series of features guided by the operators hints on the problem. These features contains both classical signal processing techniques able to fully yet compactly represent the data described in Section 2 and both complex feature able to inform the mode with the knowledge of the operators. The final dataset is composed by samples including a series of informed features (i.e., the ones just described computed on the past information regarding an asses) and as target -1 if in the next two weeks a failure will not occur or $+1$ if it will occur.

Regarding Step (ii), one can observe that the datasets created at Step (i) are typical datasets for ML regression. After testing several Shallow and Deep ML models we decided to opt for the XGBoost [6] because of its simplicity and effectiveness. XGBoost is characterized by several hyperparameters, the most important ones are the L2 regularization hyperparameters λ_2 (default value 1) the learning rate of the gradient η (default value 0.3), the max dept of arch tree d (default value 6), the minimum loss reduction γ (default value 0), fraction of training to randomly sample from the whole training set for each tree creation f_s (default value 1), and the fraction of feature to randomly sample from the whole featured during each node of each tree creation f_f (default value 1).

[3]We choose XGBoost after test many different other algorithms that we do not report here because of space constraints.

[4]In fact, every week the model is updated on the system by retraining it with the additional data collected during that week.

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

Regarding Step (iii), what we noticed is that XGBoost tended to make more mistakes on the minority class of the datasets (Full and Half Day) created at Step (i). In fact, number of faults per asses is very limited resulting in a strongly unbalanced dataset. To address this issue we rely on the combination of a general techniques for handling unbalanced datasets, namely undersampling [8] (since SMOTE in this case tends to generate not plausible samples), plugged inside the XGBoost sampling procedure performed during the creating of each tree composing the ensemble [6]. Moreover, we tuned the threshold in the probability output t (default value .5) before making a decision.

Regarding Step (iv), every week the model is updated by retraining it with the new data collected during that week[5]. During model update we need to be sure that the model does not actually regress, namely introduces new errors that previously were not there. We drew inspiration from the work of [11] and we added a constraint to the learning phase of XGBoost. In particular, let us define f_j the old model built at week j and f_{j+1} the new model built at week $j+1$. Let us also define as $\mathcal{D}_j = \{(X_1, Y_1), \cdots, (X_{n_j}, Y_{n_j})\}$ as the dataset built at week j where X is the feature vector (for Full or Half day) described at Step (i) and $Y \in \{\pm 1\}$ indicates the fact that the maintenance does $(+1)$ or does not (-1) need to be prescribed. Note that $\mathcal{D}_j \subseteq \mathcal{D}_{j+1}$. What we want is that for f_{j+1}

$$\textstyle\sum_{(X,Y) \in \mathcal{D}_{j+1} : f_j(X)Y \leq 0} [f_{j+1}(X)Y > 0] = 0, \tag{1}$$

namely to not introduce with f_{j+1} errors that f_j did not make. Unfortunately, Constraint (1) is non convex and for this reason we will relax it as follows [12]

$$\textstyle\sum_{(X,Y) \in \mathcal{D}_{j+1} : f_j(X)Y \leq 0} \max[0, 1 - Y f_{j+1}(X)] = 0, \tag{2}$$

which is a convex approximation of Constraint (1). Since plugging this constraint in the XGBoost training phase is not trivial we will rely on the Tikhonov principle [13] adding to the classical XGBoost objective the term

$$\lambda_{nr} \textstyle\sum_{(X,Y) \in \mathcal{D}_{j+1} : f_j(X)Y \leq 0} \max[0, 1 - Y f_{j+1}(X)], \tag{3}$$

where λ_{nr} is a large enough constant (in our case 1000).

Finally, Step (v) is devoted to tune and assess the performance of the final model we proposed. For this purpose we rely on a resampling procedure adapted to the fact that data are "sorted" in time and that every week the model is updated [7]. In particular, at week j we train the model f_j with the data $\mathcal{D}_j$ and we tested it (assess its performance) with $\mathcal{D}_{j+1} \setminus \mathcal{D}_j$. We measured the quality of the model according to four metrics: percentage of errors on positively labeled samples (False Positive %), percentage of errors on negatively labeled samples (False Negative %), percentage of negative flips (i.e., errors made by f_j not made by f_{j-1}) on positively labeled samples (Negative Flips on Positive %), and percentage of negative flips on negatively labeled samples (Negative Flips on Negative %). In order to tune the performance (find the optima hyperparameters) of XGBoost for f_j we perform classical leave-one-out [7] using $\mathcal{D}_j$. During this phase we performed a grid search for $\lambda_2 \in \{0.001, 0.01, 0.1, 1, 10, 100\}$,

[5]Obviously, in our case, we simulated thi process with our three years of data. We started from the end of month year one, in order to have enough samples, and then we retrained every week from the second year on.

	XGBoost	+Tuning	+Unbal	+NoRegr
False Positive %	69.7 ± 25.8	50.6 ± 20.9	20.1 ± 4.3	18.3 ± 2.9
False Negative %	18.9 ± 13.7	2.2 ± 1.7	15.5 ± 3.8	10.2 ± 3.4
Negative Flips on Positive %	43.3 ± 20.3	38.3 ± 18.1	10.9 ± 5.1	1.1 ± 0.5
Negative Flips on Negative %	7.4 ± 5.2	1.9 ± 1.3	9.7 ± 4.9	0.7 ± 0.5

Table 1: Results on real data coming from a portion of an Italian Railway managed by Rete Ferroviaria italiana for the different models developed in the paper (XGBoost, +Tuning, +Unbal, and +NoRegr) measured with different performance metrics (False Positive %, False Negative %, Negative Flips on Positive %, and Negative Flips on Negative %) to predict whether a failure on the asset will occur in the next two weeks.

$\eta \in \{0.01, 0.02, 0.03, 0.04, 0.05\}$, $d \in \{3, 5, 10\}$, $\gamma \in \{0, 0.01, 0.02, 0.04, 0.1\}$, $f_s \in \{0.6, 0.8, 1\}$, $f_f \in \{0.1, 0.2, 0.4, 1\}$, and $t \in \{.2, .4, .5, .6, .8\}$.

4 Experimental Results

In this section we will show the results of applying the methodology presented in Section 3 to solve the problem formalized in Section 2 with real data coming from a portion of an Italian Railway Network managed by Rete Ferroviaria Italiana.

In particular, in Table 1 we reported the mean and the standard deviation of the metrics defined in Section 3 (False Positive %, False Negative %, Negative Flips on Positive %, and Negative Flips on Negative %) for different models. First we reported the vanilla XGBoost with default values (XGBoost), then we show what happens by tuning the hyperparameters XGBoost maximizing the balanced accuracy (+Tuning), then we show what happens by handling the fact that classes are unbalanced (+Unbal), and finally we we show what happens by forcing the model to be non regressive (+NoRegr). For details about XGBoost, +Tuning, +Unbal, and +NoRegr please refer to Section 3.

From Table 1 we can make some observations. Vanilla XGBoost is not effective enough for a real application because of the high number of False Negatives (i.e., necessary maintenance not prescribed). The problem is a bit mitigated by the +Tuning model but yet not practical. The +Unbal model actually addressed the issue strongly reducing the False Negatives with a slight increase in False Positives (which are less of a problem, in practice it is better to make one more unuseful maintenance that missing a required maintenance). Nevertheless, XGBoost, +Tuning, and +Unbal exhibit high levels of regressivity which is bad for modern ML empowered software pipelines. The +NoRegr version of the model strongly reduces this phenomena and, moreover and not surprisingly, tends to increase the accuracy of the final model. These results are in agreement with what is discussed in Section 3.

Based on the reported results we can state that the proposed Simple Non Regressive Informed Machine Learning Model is actually effectively able to predict failures and then prescribed maintenance on the most critical Italian Railway

ESANN 2022 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 5-7 October 2022, i6doc.com publ., ISBN 978287587084-1. Available from http://www.i6doc.com/en/.

Network assets.

5 Conclusions

In this work we focused on developing an automatic intelligent tool able to predict failures of the most maintained and critical Railway asset. For this purpose, we propose a Simple Informed Machine Learning based model, namely a data-driven model able to both exploit the physical knowledge about the phenomena provided by the operators (via advanced feature engineering from the raw logs and data) and historical data. However, building our Simple Informed Machine Learning models is just the first step of our contribution. In fact our model is exploited in a modern industrial MLOps pipeline and demands continuous data collection, model re-training, testing, and monitoring that results in a large technical debt. An update in the model cannot be regressive, i.e., we do not have to simply improve average performance but also not introduce mistakes previously not present (negative flips). For this purpose, we empowered the proposed Simple Informed Machine Learning with Non Regressive properties with a simple yet effective approach. Results on real data coming from a portion of an Italian Railway managed by Rete Ferroviaria Italiana showed the effectiveness of our proposal in predict failures in a non regressive way.

References

[1] A. Zoeteman. Asset maintenance management: state of the art in the european railways. *International journal of critical infrastructures*, 2(2-3):171–186, 2006.

[2] V. D'Amato, L. Oneto, A. Camurri, and D. Anguita. Keep it simple: Handcrafting feature and tuning random forests and xgboost to face the affective movement recognition challenge 2021. In *International Conference on Affective Computing & Intelligent Interaction*, 2021.

[3] L. Von Rueden, S. Mayer, K. Beckh, B. Georgiev, et al. Informed machine learning-a taxonomy and survey of integrating prior knowledge into learning systems. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[4] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

[5] S. Shalev-Shwartz and S. Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[6] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[7] L. Oneto. *Model Selection and Error Estimation in a Nutshell*. Springer, 2019.

[8] H. Kaur, H. S. Pannu, and A. K. Malhi. A systematic review on imbalanced data challenges in machine learning: Applications and solutions. *ACM Computing Surveys*, 52(4):1–36, 2019.

[9] S. Mäkinen, H. Skogström, E. Laaksonen, and T. Mikkonen. Who needs mlops: What data scientists seek to accomplish and how can mlops help? In *IEEE/ACM 1st Workshop on AI Engineering-Software Engineering for AI*, 2021.

[10] D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, D. Ebner, V. Chaudhary, M. Young, J. F. Crespo, and D. Dennison. Hidden technical debt in machine learning systems. *Neural information processing systems*, 2015.

[11] S. Yan, Y. Xiong, K. Kundu, S. Yang, S. Deng, M. Wang, W. Xia, and S. Soatto. Positive-congruent training: Towards regression-free model updates. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[12] V. N. Vapnik. *Statistical learning theory*. Wiley, 1998.

[13] A. N. Tikhonov and V. Y. Arsenin. *Methods for solving ill-posed problems*. Nauka, Moscow, 1979.