International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2019): 7.583

Machine Learning for Predictive Maintenance in Banking Infrastructure Services: A Data-Centric Approach

Bharath Somu

Architect-I,
Email: *bharthsomu[at]gmail.com*
ORCID ID: 0009-0008-6556-7848

Abstract: ***The banking industry is undergoing a significant digital transformation, leading to the introduction of new microservices and software for infrastructure services. Although these changes can increase the efficiency of a bank's infrastructure, they can introduce new risks related to system operation. To mitigate risks effectively, an institution must properly monitor system operation and facilitate rapid troubleshooting. One key aspect of ensuring the high operability of banking infrastructure is managing the "invisible zone of uncertainty" that lies between operational monitoring and reliable failure detection, prioritization, and prediction. The growing microservice architecture in contemporary banking infrastructure systems leads to an exponential increase in number and complexity of the monitoring events being generated. Current approaches to predictive maintenance do not scale adequately. Serve Allocation Problem is an NP-hard one which attempts to allocate costs with a solution working in any kind of model, making it adaptable to any data format. Moreover, few of the applied approaches are focused on the context of banking services. Recent advances in the data-centric field of machine learning and natural language processing with Large Language Models has led to better contextualization of data, making it more understandable to humans. Aiming to mitigate the gap between operational monitoring and failure prediction, this study proposes the first data-centric approach to predictive maintenance on banking infrastructure services. In particular, a working proof-of-concept solution is presented that suggests mapping production monitoring events to predictive maintenance onboarding data using large language models to create and enrich monitoring data contextually. Three unsupervised approaches are implemented to identify false positive monitoring events at the second level of a three-level hierarchy on monitoring event criticality. These methods utilize clustering, dimensionality reduction visualization, and modelling with a probabilistic graphical model to achieve interpretability of "black box" algorithms and contrast false positives and true positives with an illustrative example. Ultimately, the human-centered nature of banking infrastructure development and operation is acknowledged. Further development paths are suggested in the latent area between the increasing demand for monitoring and deeper contextualization of the monitored systems through AI techniques.***

Keywords: Machine Learning, Predictive Maintenance, Structural Health Monitoring, Banking Infrastructure Services, Data Centric Approach

1. Introduction

The adoption of predictive maintenance is currently rapidly growing in various industries with a heavy investment on their IT infrastructure. The investment on such capital-intensive equipment and systems should carefully be considered for each further acquisition and barely ever be dismissed [2]. On the other hand, like any machinery, computing infrastructure does experience failures – physical ones, like hardware negligence, and virtual ones, like malwares – that can badly interrupt banking services, prevent proper operation of ATMs or cause illegal transactions. The maintenance of this equipment architecture and firmware has been practiced and is essential for continuous operation of this crucial infrastructure and to protect banks from money loss.
Existing data-driven maintenance practices on easy-to-understand descriptive models or rule-of-thumb thresholds are interlinked with the domain expertise of maintenance engineers, however, along the life of any technology, the knowledge of the as is system gets increasingly hard to be transferred and interpreted from the engineers above to the ones below. This results in misinterpretation and wrong assessment of coherently escalated incidents, sometimes non-coherent ones on system failures, and negligence of false alarms, along with an absent or unmaintainable knowledge transfer across maintenance tiers, from the top-tier senior engineers down to juniors, understudies and non-techs. These observations call for interpretable and responsible AI input on the maintenance service of banking infrastructure.

Besides the core banking services, a rare and unseen theft investigation incident would also arise due to very long-term operation of ignorant equipment. The crucial and money-loss raising incidents on these infrastructures need interpretable AI assistance for wider interpretability on data streams, stronger maintenance recommendation on planned action, and climate-aware wrapping of monthly cleanliness requests. A data-centric approach for a new class of interpretability with both remark-explaining library, brand-o-functional equation counterparts, and drift-detection evaluation metric is introduced.

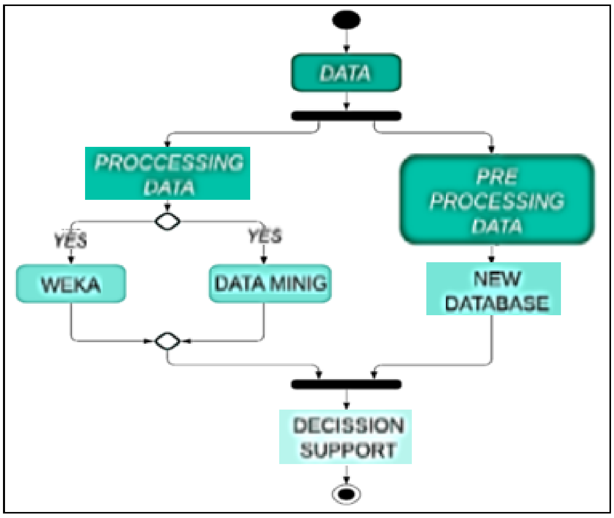

Figure 1: Predictive Maintenance Using Machine Learning and Data Mining

1.1. Background and Significance

The pressure for banks to reduce costs is expected to increase, which would necessitate a reduction in the quality of services offered. Bank and IT infrastructures are becoming increasingly critical to new operations and implementing banking sector strategy, requiring more sophistication, efforts, and understanding of processes. Various stresses on time, services, IT technical advances, continuous true records, move to centralized services, and outsourcing create great risks.

Specifically, IT risks in any bank encompass Fixed Asset Cost monitoring, budgeting regular evaluations of services, pre- and post-investment assessments, and investment estimations for specialized services versus a revenue for extra capacity. The question of how to avoid service degradation, lack of understanding of hide-adding risks of services, bank-specific service expertise and services all contribute risk. Investing into service breakdown and risk reporting suffices IT requirements, data shadowing on spare service, and data-mining visio trends analysis help avoid a risk.

Intelligent services separate resulted faulted notifications, simulation-board-user filtered warnings of actual deviations, or just focus on production specific services. Infrastructure services fault notifications are under auto-filtering and monitoring of only schedule violations on un-expected downtimes. For comparable systems, competitive service-resources time analyses are mandatory, though many indicators are created according to operation unit limit states, with generalization points for all banks and services developments.

The time to repair trends of banking infrastructure services are expected to be reduced by 5-year reduction. Otherwise respectable organizations decrease service frequency or depth in general. Benchmarking on purpose on-off periods of resource time consumption is reasonably easy. Estimation of banking infrastructure IT service responsibilities from agreed service IDs and actual resources requires fixed and permanently updated algorithms, taking into account the greatest bank-specific anchorage time intervals spread projections, as well as unexpected service interworking times and many other hidden factors.

Equ 1: Survival models, like Cox Proportional Hazards for time-to-failure

$$h(t|X) = h_0(t)\exp(\beta^T X)$$

Where:

- $h(t|X)$ is the hazard function
- $h_0(t)$ is the baseline hazard
- β is the learned coefficient vector

2. Overview of Predictive Maintenance

Predictive Maintenance (PdM) is one of the approaches used in maintenance, along with Corrective Maintenance (CM), Periodic Maintenance (PM), and Reliability-Centered Maintenance (RCM). The main goal of PdM is to increase the availability of the machine by minimizing unplanned maintenance caused by machine failures [3]. Providing accurate information about when and why a machine is going to fail offers many advantages. It allows maintenance to be planned better, stocks of spare parts to be managed better, and savings of money related to maintenance to be obtained. It can lead to increased productivity for factories that produce products, as the unavailability of a machine can have an effect downstream on production. Similarly, it can reduce downtime of machines for hospitals where failure of a machine that is performing a heart scan is critical for the patients that need care.

Development of a Machine Learning (ML) algorithm for predictive maintenance is often viewed as the development of a predictive model. In the 3V framework, it refers to the development of a model where the volume of data to process is too big for the data scientists to work with it collectively. This is indeed a very common problem for big organizations. However, there are other reasons for working towards a data-centric approach. A deadlock situation can also arise if the data are of bad quality: no valid rule can be identified regardless of the amount of time and resources that are devoted to the effort. In such situations, predictive analytics may benefit from executing a data-centric approach.

In general, data-centric AI solutions utilize the data engineering phase of the data science process. Very often the innovation in such an approach will not be an ML algorithm but one or more data engineering algorithms. In recent years, there has been much attention and research in the area of data-centric approach towards AI or data-centric AI. For instance, there is a need for automatic detection and repairing of bad quality data considering their contexts. Based on the data-centric approach, much research is put forward in other domains, such as predictive maintenance in manufacturing. However, there has been limited research in a banking context.

3. Importance of Predictive Maintenance in Banking

Maintenance may be defined as all technical actions aimed at keeping or restoring a machine or a resource in a state to perform its required functions. Maintenance actions can be classified into various types based on the time of execution, including corrective maintenance, scheduled maintenance, and preventive maintenance. Corrective maintenance includes actions performed after failed components without any prior knowledge regarding the component's failure. A scheduled maintenance activity can be defined as a maintenance action that must be performed irrespective of the actual condition or degradation of equipment. Fixed time maintenance is an example of this type of action. Preventive maintenance actions are taken based on a predetermined condition regarding the physical and functional status of a resource. One common example of this is condition-based maintenance, which may include a maintenance activity that should be implemented if the number of cycles of a pump exceeds a specific threshold.

The goal of predictive maintenance is to increase machine availability by minimizing unplanned maintenance caused by machine failures. Predicting when and why a machine is going to fail offers many advantages for a company: a much better planning of maintenance actions, a minimum of spare parts kept in stock, and a minimum cost of the maintenance process. Predicting the failure of a machine is interesting for many applications. For example, in a hospital, knowing in advance when a machine is going to fail opens the door to planning the maintenance with no downtime of these critical machines. Also, it is clear that failure prediction for a surgical robot will be critical for patient health.

In this setting, the prediction of failures is made on the basis of simplified or noisy logs. This log contains a lot of valuable information about the machine: the events which occurred just before the failure, the various diagnostics it has passed through at a given date, and the changes of the relevant features or values. All this information can then be used as a basis to improve the predictive power of the logs.

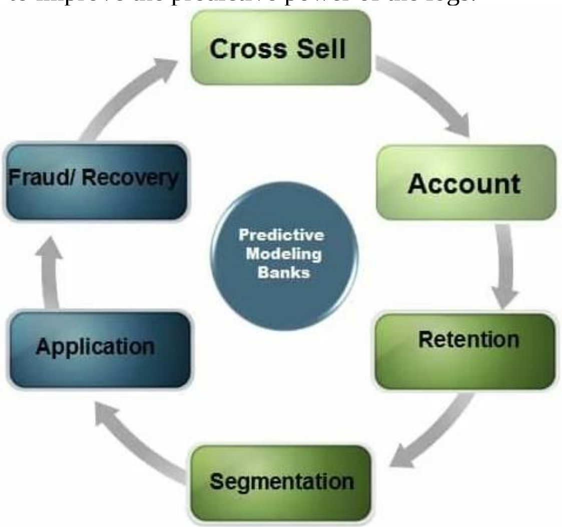

Figure 2: Predictive Analytics in Banking

4. Machine Learning Fundamentals

In the last few decades, technology has entered a new era that has changed many aspects of everyday life. IoT devices, mobile phones, and social networks are only a few examples of the technology that has reshaped our lives. Machines with Artificial Intelligence (AI) and Machine Learning (ML) are making our lives easier. Many predictive applications using AI/ML algorithms are created in healthcare, manufacturing, market decisions, and many more sectors to increase customer happiness and cut costs. Machine Learning is a subfield of Artificial Intelligence that's primarily concerned with enabling machines to learn from data and make predictions or decisions based on that data, in short learning from experience. It involves developing algorithms that use statistical techniques to allow computers to learn patterns or relationships within a dataset, with the goal of making predictions on future data or classifying items.

Machine Learning incorporates various fields of computer science and statistics, such as data mining, optimization, and statistics. Industrial and temporal data require storage systems that enable high-performance data access and low-latency queries. The analysis of such data needs suitable algorithms and distributed processing systems to provide timely results. Also, decision support systems (DSS) are needed to help teams tackle complex real-world decision problems where human reasoning must combine with intelligent system reasoning. DSS tools benefit from visualization techniques such as dashboards, which display various data dimensions and trends. Predictive models of complex systems use statistical and machine learning methods to extract hidden relationships between clearly defined inputs and outputs, modelling their behavior.

4.1 Supervised Learning

Predictive maintenance consists of systematically analyzing data to build statistical models that can predict when a component will start to fail. Predictive maintenance focuses on detecting or predicting machinery failings by means of monitoring. The uptime of an asset depends on two complementary perspectives: the equipment's health and the operations' effectiveness. Predictive maintenance identifies fault patterns, failure modes, and wear rates for machinery and then designs a risk-based maintenance strategy.

Maintenance and asset management are fields of application for which machine learning techniques are assumed to have significant potential for improvement. Predictive maintenance can use many approaches like first principle mathematics, heuristics, and machine learning. A method is proposed that builds a data-driven predictive maintenance model from a historical maintenance database. Such an approach usually follows three steps: data preprocessing, model development, and model application.

The first step is to prepare predictive maintenance models. Usually, historical maintenance and production data are considered. The data are preprocessed, which may involve filtering the observations, cleaning erroneous measurements, aggregating certain characteristics, creating derived features and combining several sources of data, encoding and splitting the data into training, validation, and test sets.

The key step in predictive maintenance is to develop predictive models that describe the desired maintenance behavior in terms of the operational data. The models are usually either discrete or continuous. Discrete models describe the probability of a future event such as a failure. Continuous models describe the future trajectory of a variable like degradation. In a prediction hazard model, the expected time of an event such as a failure is forecasted.

When applied to real data, the predictive maintenance models need to be validated and potentially improved. Usually, this can consist of three steps: assessing the model performance, interpreting the model, and possibly reengineering the model.

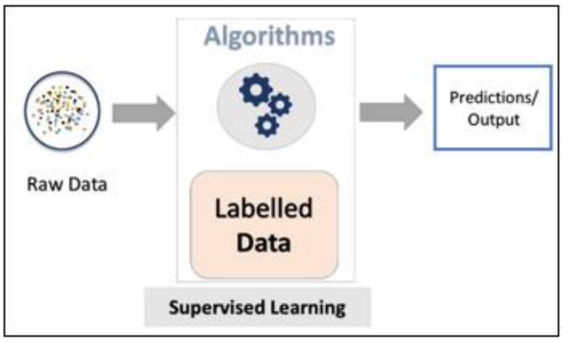

Figure 3: Supervised Learning Overview

4.2. Unsupervised Learning

When a model is expected to regularly predict new events without the requirement of human intervention or guidance, it is said to be in Unsupervised Learning mode. In addition, there are inherent data descriptions on the bank's new incident tickets, such as which were not part of models trained in the supervised setting. If the ticket description data is natively in a different format than the description data before, then models that had just been trained would also receive no guidance in training or subsequently providing any consequence.

Recently, pretrained, self-supervised Transformer-based models have become state-of-the-art feature extractors for sequential data. In addition, using self-supervised pre training on similar data has been shown to outperform a wider variety of self-supervised pretext tasks. Furthermore, the bank has access to an extensive ticket catalog, which contains around fifty thousand recent classified incident tickets and ticket descriptions and has similar data. Therefore, the self-supervised exploratory data analysis task will be considered.

In the pretext task, inputs are categorized into two classes: normal and out-of-distribution. For an input to be in the normal class, its similarity score with respect to clusters generated on a larger dataset needs to stand out among multiple reference clusters. On the other hand, an input is assigned as out-of-distribution if it is classified as normal with respect to its in-dataset, in which an unsupervised representation boost is employed to help maintain discriminatory power.

4.3. Reinforcement Learning

Reinforcement learning (RL) has been widely researched and applied for various decision-making problems such as games, robots, and investment portfolios. One area of RL research is predictive maintenance (PM). Efficient PM policies have a significant impact on resource allocation and capital management for companies, particularly for large-scale industries with a huge number of monitored assets. Most existing studies focused on building and improving a model for either health monitoring or PM policy learning. Overall, a model-free RL method is usually considered a black box that learns a highly non-linear policy given a state and action representation. Thus, it is unclear what generic knowledge the RL agent learned, hence, lacking transparency in explaining the maintenance decision-making process. This suggests that what was learned by the agent cannot be reused on a similar system within a different industry.

Recent RL studies are hypothesized to share a similar learning paradigm. The agent with enough exploration experiences learns a better policy. However, it is hard to consider using RL from a practical point of view. Industrial PM problems suffer from limitations in exploration. A gradual availability of historical data across the fleet is more common due to cost-effectiveness. Additionally, poorly engineered states may lead to catastrophic decisions, especially in safety-critical systems. Furthermore, estimation of rewards is often complicated and ill-defined. Given these limitations, compliance with the learning paradigm cannot be guaranteed. An RL solution would not be desirable until the exploration limitations are overcome.

An opportunity was identified that the RL problem can be re-formulated as a supervised learning task using the backward learning algorithm. Such a mapping can fit input desired rewards and observations to output actions. For a maintenance agent, the desired reward can be defined either externally as target values with a direct representation of costs and profits or internally from a model that computes potential values. The observations can be any state representation that holds the same semantics regardless of the quality. These could include, but are not limited to, the asset monitoring signals such as health status, RUL, and discrete functioning modes. The actions need to be selected from a finite set of choices. Knowledge of the reward structure or system dynamics is not required, and any decisions made outside the availability of model assumptions can be accommodated.

This supervised RL approach is suitable for efficient pre-training of any initial agent with exploration restrictions on historical data from a fleet of homogeneous assets. This can be validated using a simulation case study of an assembling line with multiple degradation processes. A distribution of operational profiles to emulate historical explore-exploit data and synthetic data generation methods are established. Pre-training state-action mappings based on data from the lower percentile of explored policies outperform conventional sector reward shaping by a large margin during fine-tuning over the whole state-action space.

Paper ID: MS2012141601 DOI: https://dx.doi.org/10.21275/MS2012141601

Equ 2: Predictive Model (Binary Classifier or Regression)

$$\hat{y}_t = f_\theta(X_t)$$

Where:

- f_θ is a parametric ML model
- $\hat{y}_t$ is the predicted probability or score of failure

5. Data-Centric Approach

The goal of this research is to investigate the utilization of machine learning techniques for predictive maintenance in banking I&O services. As a first step, a data-centric approach is adopted to create a well-structured dataset from many different heterogeneous data sources. Traditionally, the unstructured data lake formed from all sources has been used at a high level, employing machine learning techniques to check whether service incidents predictively describe any anomalous behavior in the I&O systems. This research focuses on a data-centric approach starting from the architecture to discover and construct new structured datasets. Feature engineering is a crucial component in achieving the goal, wherein techniques such as named entity recognition and classification models are used to derive new features, one of the most well-known techniques in data-centric AI.

Since historically many millions of service incident tickets are stored in the database, it is challenging to discover I&O relevant features. Furthermore, a novel pipeline to combine the textual feature representations with the available canonical features is constructed. Extensive experimental evaluations demonstrate that the enhanced structured datasets outperform the previously generated datasets in predictive maintenance applications. These advanced structured datasets are made available for other research. Additionally, a framework to measure performance improvements of the structured datasets that can be more broadly applied to other research is proposed. With the data-centric focus, this research investigates how to gather new features from incident tickets, indicative of the I&O system behavior and more relevant for predictive maintenance, to enhance and complement the existing structured datasets. The findings contribute to feature engineering, which is one of the most essential aspects in data-centric AI and machine learning. Besides, the evolved datasets are able to outperform the earlier dataset in predictive maintenance use cases on incident prediction tasks, which is valuable to industry practice.

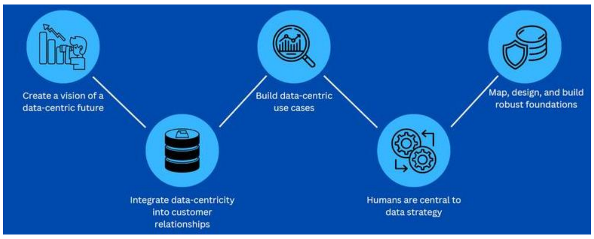

Figure 4: Data-centric architecture

5.1 Data Collection

Infrastructure services use critical middleware products such as databases, ETLs, processes, queues, and caches. They essentially manage transactions so that data input to transactional databases is clean, consistent, complete, and correct. Any downtime impacts business continuity, data quality, and availability of the service for end-users on transactions. Data regarding infra product type, deployment and decommissioning date, infrastructure owner, infra status (active/inactive), uptime metrics (Scheduled and unscheduled), and consumption are collected and stored in a data lake consistent with the data model. The preprocessing pipeline runs daily for the data lake to derive metrics like usage statistics for the past 1 month, SLA summary for the past month, transaction type statistics, database query statistics, and capacity freshness stats. Old records (older than 2 years) are cleaned and archived in the data warehouse based on office policies. Besides these, business analysts also actively use the data lake queries to investigate one-off business issues. The first stage is data extraction, which involves extracting structured and semi-structured data from databases, nested architectures, flat files, spreadsheets, and emails using jobs. Data on the customer's infrastructure products is also extracted from the sales team's owned cloud setup with the help of engineering teams. Data pipelines are run on a two-week schedule to extract new data, post which a query can be run to pull the data for use cases. The second stage is data transformation, which involves making the data analysis-ready by performing multiple operations such as applying transformations (for preprocessing, calculating average metrics, etc.), limiting data (due to performance issues), or replacing specific characters. Data that is preprocessed, but still not analysis-ready, is dumped in an inactive tables database for audit purposes as to why it was ignored. Finally, the third stage is the loading of the data to the reporting databases. A similar technique is also used to load data for other data warehouses.

5.2 Data Preprocessing

One of the objectives of this research is to describe a data-centric framework which can be utilized by institutions that generate big banking infrastructure services data and possess forecasting needs. In order to make the experimentation process easier, this framework is imagined in a modular way. In this chapter, the general configuration of the framework is presented and each module is discussed in detail. Data-centric is a new approach proposed by Google AI. In this vision, it is stated that a great potential can be unleashed by prioritizing data quality in the modality of data-centric artificial intelligence. In this research, this vision is modified according to the domain and objectives and adapted to the banking infrastructure services domain, specifically considering predictive maintenance use cases. In addition to the modular framework, AI algorithms aiming to forecast, classify and analyze banking infrastructure services data utilizing extracted ML-ready features are proposed. A benchmarking experiment performed on a synthetic dataset is presented in detail, displaying the obtained forecasting and classification performances of these AI algorithms. Applications of the framework are outlined and discussions are presented with future research suggestions.

The dataset used in this study is accessible to additional research for reproducibility purposes. Similarly to the most recent works tackling predictive maintenance tasks utilizing logs as main data source, the dataset to be used is derived from a real-world or process log dataset and it is considered

Paper ID: MS2012141601 DOI: https://dx.doi.org/10.21275/MS2012141601

deemed valuable. However, it should be noted that the raw dataset applied in the experiments of this thesis work is not publicly accessible. Although it is not possible to share the logs due to data privacy concerns, approaches to create synthetic datasets similar to the used dataset are described. A dataset consisting of simulated telemetry logs generated from a home-brewed Monte Carlo event log generator built upon the specifications of the original natural logs will be shared as synthetic dataset instead. More importantly, in a resource-constrained context where the simulation parameters are unknown, the generalized case without a concrete simulation model will also be discussed.

5.3 Data Quality Assessment

While the machine learning project successfully identified and pre-emptively notified a specific failure that would occur around 5500 hours into unit operation, testing on additional datasets frequently failed to replicate the same or similar outcomes. Several reasons for this are investigated here, focusing on machine and environmental variability, and the quality of the datasets used in the predictions. Even filter changes can affect the accuracy of the models, meaning the provenance of the data should be understood upfront. Due to this, and the need for a better understanding of failure modes, a data quality assessment study was conducted on the diverse pool of laboratory, commercial testing, distribution, and field datasets created by collaborators. A few different techniques for diagnosing the complexity of time-series datasets were trialed independently of the machine learning processes, targeted at identifying potentially anomalous time-series data that may better inform the training of accurate machine learning models. This work has resulted in balanced cut-off algorithms to distinguish valid linear data for analysis from invalid and invalidated data.

With the early work highlighting datasets from field data, commercial testing, and lab testing as more complicated than a simple sinusoidal plot, the data comparison for regression analysis is facilitated in terms of reconstructing notional component fault frequencies. Longer testing and operation durations lead to more complicated signal representations, while discrepancies across datasets were found related to compensation and filter exhaust state definitions. For the purpose of this study, these considerations allow guideposts for practitioners looking to assess dataset complexity and predictability prior to analysis and modeling efforts. It would be valuable to incorporate additional complexity analysis related to power consumption and speeds to validate the effectiveness of the quality assessment algorithms. Current thoughts are to bring quality assessment capabilities directly into future iterations of the by-product, allowing users to ensure a minimum quality grading for the data they wish to input for predictive maintenance modeling efforts.

As engineering systems grow more complex, with an ever-expanding pool of flexible and low-cost data collection methods and sensors, predictive maintenance becomes more feasible but also more difficult. The machine learning processes performed without clean and representative datasets may be nontrivial to reproduce, and relevant pre-analysis decisions will almost certainly be required. In this effort, a comprehensive analysis is performed on the quality of multiple diverse time-series datasets from the testbed [8]. The complexity of each time series is quantified to provide an initial metric whereby questionable time series can be distinguished prior to modeling or analysis efforts.

6. Feature Engineering

In order to improve devotion of time to model tuning, one of the objectives was to describe a data-centric feature engineering methodology for sieve data preparation across multiple monitoring levels. The main outcome was a detailed methodology for automated sieve failure prediction, tailored to requirements of aggregate level monitoring. However, due to limited accuracy, 957 failures remained unpredicted. The proposed method achieved good performance at pre-filtering, identifying 723 candidates for manual assessment. A template is proposed to guide the evaluation of pre-filtering candidates and for them to fit into existing processes, which aim to authenticate data and account for known issues.

Recordings of feature engineering implementation are archived and accessible via internal Jira. The presentation slides or separate walkthrough video have been requested, in order to support career development and knowledge sharing opportunities with peers. Future research topics include overcoming short-term data unavailability, automating components of threshold definition, classification selection, as well as investigating compact encoding of daily views.

The aim of this study was to build a comprehensive data-centric machine learning strategy for predictive maintenance of sieve working at WEG, focusing on the transport reliability of waste treatment. The first research question identified the requirements of machine learning based predictive maintenance, with a focus on the infrastructure service domain. The second research question identified and developed a detailed feature engineering methodology for preprocessing sieve monitoring time series data in order to adopt classification models. The data preparation strategy resulted in sieve maintenance needing 60–80% of the time compared to 75–90% before implementation. First observations indicate that model tuning can be improved by participating in a model benchmark challenge.

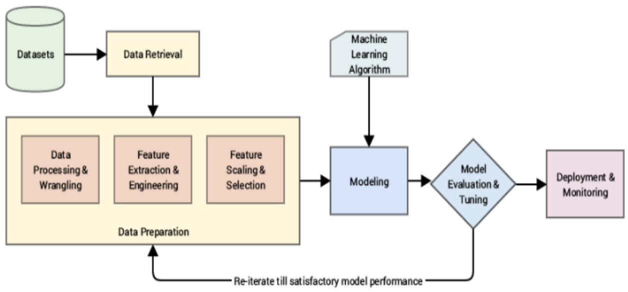

Figure 5: Feature Engineering

6.1 Feature Selection

Feature selection enables a reduction of the dimensionality of the dataset by filtering the set of input features to only retain the most relevant ones for prediction and other tasks, such as visualization, etc. In principle, any data transformation modifies the input feature space. However, we will focus on the selection of existing features, with the caveat that the newly generated variables or data histogram distributions are preserved either in their raw form or denormalized back to the input range of the raw features. As such, all transformed variables will be excluded from the analysis. This section details the approach taken for feature selection and how the tool used for this task was configured and operated to yield maximum performance. Datasets that combine information from the alarms, sensors, and prior maintenance actions and failures will be covered and stakeholders that participated in the data transformation process will be mentioned.

Feature ranking was deemed the most effective method to select a proper set of features, as the model-free and performance-independent means of priors. As a side effect, it does allow one to see how important each feature is to the task and thus potentially add a meaningful piece of information into the model. It is sufficient to know that a feature contributes to prediction with some amount of relevance or out, rather than detailing in which combinations and conditions they do so. This information sufficiency is paired with the computational expense since ranking can be done in less than an hour on a data set containing tens of thousands of samples, while pairwise methods scale quadratically and rigidly require careful filtering and all-variable cross-combinations to contain tens of features. Hence, they were not used in practical analyses. Recursive feature elimination and other recursive ranking methods were also impossible to use on larger data sets, as they applied recursive feature elimination using cross-validation and could take a handful of hours, raises the other cross-validation layer therefore, and iteratively fitted and refitted the entire predictive model a dozen of times.

Algorithm and parameter selection were applied directly on the data set containing alarms and sensors to cover the full extent of the transform without crunching the data anew, to generate the most consistent and general results. Three gradient-boosting based gradients were chosen and a handful of other, mainly tree ensemble, alternatives backed by narrowed-down hyperparameter grids were employed prior to the two engines. Whereas other well-known methods performed well, none approached the predictive and performance quality of gradient boosting trees.

6.2 Feature Transformation

To extract informative features from raw data which can be time-consuming and an elaborate task, diverse types of empirical features were experimented with to gain valuable insights from historical data. The process of transforming and merging five diverse data streams is featured along with feature engineering. Selection of data streams like the sensor's position, the threshold to detect changes in calculations, and multiseries smoothing allocates recent values more weight, while older values diminish their influence on the calculations. By employing a few straightforward commands in Python, transposed 25 algorithmically interesting statistics were computed and added as features to the original input data set, in addition to those asserted as the most informative. Different timesteps were explored for implementing the feature transformations, mostly for the 24 h data series produced by the daily trigger for the predictive maintenance task.

Cumulative and average counts for outliers and common sensor limits were calculated on diverse time windows. Lagged operational states concatenation was included, extending the time series input data from the most recent 48h to the preceding 48h. Similar characteristics of the predictive maintenance process, while diverse domains had a large advantage when identifying anomalies prior to engine failures, so variable window sizing was also studied with a notch for the upper time limit of 168h. Selection included generating five position features and 18 common alterations, as a few more errors appeared in the prediction. Moreover, as displayed in Fig. 14 steps for further feature engineering steps, it was assessed how the two error types in machine cycle times altered the prediction rate decrease across the various time faux claims upper limits [9]. Overall, the investigated features paint a picture of how engine work conditions. In addition, redundant, and non-informative features were dropped on a brute-force basis, prompting model studies on input dimensions under 60. This tiered analysis aided the understanding of anomalies prior to failures and produced factors for proactive preventive maintenance use.

Equ 3: Feature Engineering (Data-Centric Layer)

$$\hat{x}_{i,t} = \frac{1}{w}\sum_{k=0}^{w-1} x_{i,t-k}$$

Where:

- w = window size for rolling statistics

7. Model Selection and Evaluation

Before performance evaluation and selection, it is necessary to split the available data into 3 subsets: training, validation, and testing subset. A good balanced size is to vary from 60/20/20 to 70/15/15 [6]. Generally, all methods are to be trained in the training set and tuned in the validation subset. The selected best model is finally tested in the bottom testing dataset, which was not exposed to training and tuning. A proper scoring function is also defined, along with the respective fair scoring metrics. This considers the dominance of nominal behavior, deciding on base model parameters, selecting performance evaluation metrics, and setting the scoring scheme for cascading multiple models.

Feature selection was thus grounded more on finding appropriately aggregated modeling metrics. Two procedures of creating new combined features with 1-minute-averaged 5-minute value peers from distributed sensors were examined: (1) concatenate; (2) linear regression. The former fared better, and was thus set to create 14 combined features on the studied monthly task. Another target was thus devised to present whether certain 20-minute intervals covering the prediction horizon would reach a median of 1 for selected five combinations. Such modeling tasks were adapted daily to predict hourly ahead target intervals. 100 New predictive variables were then generated and selected to best candidates.

To highlight outliers for modeling better as well, other 27 agreed factors were generated with traditional mean and standard deviations for each sensor and two different time lags. Ultimately, feature selection involved generating 14 combined features, two time frames for four aggregation measures, one additional for regressing on another monthly offsets, and 27 basic ones mentioned above.

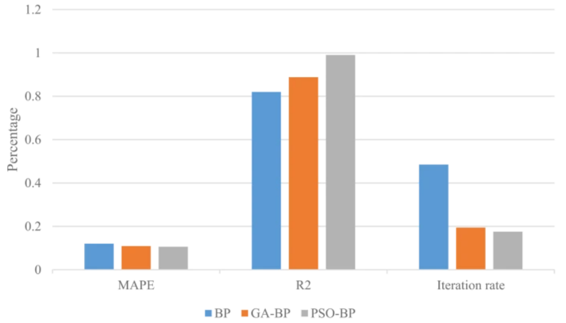

Figure 6: A machine learning-based approach for product maintenance

7.1. Model Types

Even though unsupervised or semi-supervised approaches could be used to tackle the proposed problem, the presence of labeled data often enables the formulation of the predictive maintenance problem as a binary classification task, addressed with a fully supervised approach. Machine Learning (ML) and Deep Learning (DL) are two families of methods that are, nowadays, widespread for tackling predictive maintenance tasks. Properly choosing the model type is crucial: For a given task, some models will be able only to predict limited cases or require unrealistic amounts of time or data to deliver satisfying performances. Clearly defining the characteristics of the ML/DL approaches that are suitable for a context in which predictive maintenance tasks must be solved on a fleet-wide scale is the goal of this section. For interpretability purposes, it could be useful to identify a wide range of binary classifier model types. Models are chosen that rely on the same methodology but consider different implementations or architectures (and possibly hyper parameterizations). On the one hand, this would make it possible to perform a more interesting and enlightening analysis, i.e., to study how similar model architectures with different depths or complexities affect performances and the trade-off between performance and training time. On the other hand, there are well-known models that belong to the same family of methods, and undesirable incompatibilities have been detected. Therefore, it is also interesting and relevant to identify a good range of strict model types, with enough diversity to allow general conclusions to be drawn and sufficient homogeneity to limit unwanted (and non-controllable) discrepancies. The applicability of ML/DL approaches will be evaluated on end-users represented by manufacturers of different sizes and industries.

By stressing and analyzing critical implementation issues such as data representation, model input preprocessing, and performance evaluation metrics, it will be highlighted how such facets have a big impact on the ultimate applicability of ML and how they deserve special attention when tackling predictive maintenance tasks on a fleet-wide scale. By analyzing benchmarking results on real-world predictive maintenance challenges, it will be stressed how the maturity of the field is outstripped by the ability to extract and analyze ML-compatible data from machines and processes. Moreover, the means and opportunities to fill the industry-academic gap will be discussed. Finally, a list of recommendations based on published knowledge will be provided to help predictive maintenance practitioners implement proper models and set up compliant infrastructures.

7.2 Performance Metrics

Scores are reported as a weighted average of individual measures. The edge case errors often dominate the scores, making them less interpretable and risky to pursue aggressive edge case robustness enhancement over accuracy. A set of metrics are reported that exhibit the same business impact as the metrics in use, while avoiding excessive complexity. For binary aspects, the metrics consist of the positive predictive value (PPV), which is the same as precision. Meanwhile for metric pairs, the reported metric consists of a robust combination of false positive and false negatives to avoid threshold tuning artifacts. The weights are designed to not favor any one metric but to balance all metrics by relative business impact. This business knowledge is encoded in a simple way that is easy to evolve and communicate.

As mentioned above, for the purposes of generating submission data, a single value estimator is needed. If the bag well provides a pre-processed output based on personal best performance metrics against which submissions can be evaluated and ranked, some aggregation strategy can be combined with a stochastic or derivative-free optimizer to produce tuned scores with any desired robustness guarantees. In the meantime, it may be helpful to release studded example bags that have been made to induce high performance on pre-designed public metrics with known sensitivities. It's also necessary to provide analogous bags tuned against these continuous score nuisance parameters, for productions with the substitution model.

By the time a minimally adaptive solution is in hand, the maximum performance potential with the broadest impact should still be present. Using aggressive augmentation, training sample ratios in-line with validation samples, and error focus and filters to smooth edge emissions is essential. It may be more beneficial to start small and then scale, rather than starting with a large marginally productive aggregation that becomes too time consuming to analyze in detail.

8. Implementation of Machine Learning Models

A data-centric approach is adopted to build more accurate models, focusing on data collection and feature engineering, including inflow and outflow data from Core Banking System (CBS) databases and enabling real-time predictions. Feature engineering comprises data sampling, cleansing, and transforming raw data, and obtaining additional features, facilitating model robustness against outlier data points and seasonality. After thorough examination and selection, the engineered features are more than five times the number of initial features. A mathematical model based on the Poisson distribution can achieve an Nearly Perfect Forecast (APF) accuracy with a small number of features, providing insight into the applicability of ML for various use cases. The classification models are built with this baseline mathematical model using the fraud-detection problem.

A series of robust experiments evaluated existing classifiers for this purpose. In scope, gradient-boosted tree and tree-based ensembles dominate other categories, including shared best score and similar runs, yet possess weak generalization, and improper hyperparameter tuning tends to overfit. LSTMs achieve superior performance relative to most classifiers, effectively preventing objective information loss through representation learning. PKHMs outperform all other classifiers regarding the MAP@k metric. Interpretations of time-series models using timestamps, features, or aggregate forecasts analyze model predictions by observing varied cyclomatic wording and form paths of interest in predictive maintenance. They produce a counterfactual of historical timestamps that best fit predictive times using shaded observations.

8.1 Training Processes

The model was trained with the preprocessed input files, following the aforementioned processes and architecture in two different scenarios where only the first 5 minutes and the first 10 minutes of sensor data were provided respectively. 1000 decision trees were used for training the Random Forest Regressor alongside Hyper Parameters tuning to find the optimum hyper parameters. Several regression metrics were calculated and plotted to visualize the performance of the models built. These metrics mainly include the Mean Absolute Error, the Mean Squared Error, and the R-squared error. The variance of the predicted output can also be visualized by plotting it against the true output on a scatter plot.

The random forests with a maximum depth of 4 were able to predict the value of maintenance with a Mean Squared Error between 1.7 and 2.2 days. Variance in prediction of around 0.5 days was also found. Maintaining regular contact with AI teams and business experts is very crucial as the understanding of the domain is very important for the success of the predictive maintenance projects. Another important factor is the retraining of the models in a regulated period of time, which can ensure an advanced capability for the models to understand the business changes.

The modeling processes were easy and straightforward due to the entire coding flexibility and ease of use offered by the python programming language. Several useful Python libraries such as Pandas, Numpy, Tsfresh, Scikit-Learn, and Matplotlib were utilized to properly implement, utilize and visualize all the prepositions and models described in the previous sections. Based on the pre-processing framework hereby made available, pre-processing data in their format as used in the case study is no longer a 1000-man-hour process. Instead, proper pre-processing scripts can now be executed in an hour or even less.

However, several advice and findings may help other practitioners and researchers interested to use the modeling processes and framework. First, data errors and cleansing may take notable time and effort. Several key problems in this regard have been pointed out in the postmortem report, which may also be useful for different cases. Second, and based on the postmortem report and experience a proper selection of input variables can drastically reduce the time required to build the models in terms of number of training days.

8.2 Validation Techniques

The significance of preventive and predictive audit systems has well been noted in various industries. They provide an opportunity to increase efficiency, cut costs, and augment reliability. Failure prevention prediction involves the monitoring of leading indicators of possible failures. In industrial contexts, these indicators often reference pumping, energy, and pressure data from sensors on physical assets. In any real-world situation, action can only be taken on a small target group of assets with consideration of costs and relative effect on wider operations. Broad-spectrum machine learning methods are sought to address the issue by forecasting the distribution of a key performance indicator relevant to failure on a global grouping of assets based on much broader data streams. The aim is to identify at-scale and at-early time instances of potential failures.

The NHS is one of the largest publicly funded health services in the world. Data from community and mental health settings and data science-led innovation and engagement with hard-to-reach groups are under-represented. In terms of immediate pre-pandemic preparations, practices already had experience in balancing service and activity recovery post-implementation of new NHS directives, before the pandemic brought unprecedented challenges. Future focus must be on optimising shared system-level operational efficiencies with diminishing public funds and increasing demand, and liaison to more effectively integrate reward and reinvestment for primary and secondary care health professionals to boost mental health resource accessibility. is a key issue for machine learning in predictive maintenance. It not only affects basic performance inspection values but imprints on fundamental understandings of the learning process. Traditionally, a predictive modeling problem is set up through the steps of defining a predictive problem, collecting and cleansing data, selecting a learning algorithm, applying the learning algorithm to the data, and gauging the quality of the final result. For applying those steps to learning in prediction maintenance, it means how to prepare a validation set on industrial systems that change or wear.

9. Conclusion

The work aims to mitigate failures of banking infrastructure services by developing adequate methodologies based on Machine Learning (ML). In this particular research work, as the first milestone of all the planned ones, a data-centric approach using supervised methods is analyzed. Taking inspiration from real-world scenarios of fraudulent transactions occurrences, it is considered almost one year of continuous and historical data of one banking infrastructure service of a specific large Portuguese bank. One safe-rail subway service of that bank is understood and modeled with only two assumptions: a barely affected service is acquired and a small set of the most disruptive anomalies is found. As contributions, the synthesis of anomalies is obtained using Generative Adversarial Networks (GAN) for the first time in this domain. Moreover, predictive models are constructed to uncover impairment and disruption of the service in near future, which can enable human operators to do proper service recovery actions, preventing any unrecoverable damage.

Future work should focus on collecting a larger set of data from the chosen service and appropriately labeling it. Nonetheless, as just a small subset of the overall data is exploited in this work, the planned educational time series modeling techniques can be directly implemented to be able to (i) better understand the service's dynamics, (ii) be used among ML models to define more diverse training sets, and (iii) serve as a benchmark against more sophisticated yet computationally expensive ML techniques. Additionally, an explorative effort can be directed to construction of unsupervised methods based solely on the data recordings so that the effects of using synthesized anomalies or any other artificially generated data can be analyzed.

As an add-in contribution, a brief literature review on such methodologies is provided. By understanding the service in depth and implementing the earliest models, the significance of extrapolative forecasting is better understood in this domain as it can impact other continuing research works significantly.

9.1 Future Trends

Predictive Maintenance (PdM) is a modern strategy that predicts when the part of an infrastructure, machine or vehicle must be replaced. This prediction should take into consideration many factors such as the age of the part, how intensively it is used, environmental parameters and whether it has enough physical reserves for its operation. The prediction of when a replacement must take place is of great significance, as it can significantly reduce the time and the cost necessary for the maintenance of the infrastructure inflow. When replacement is carried out not earlier or later, it does not affect the operation of the machine, and its useful life is maximized. To achieve this prediction, the part must be continuously monitored by various sensors. The sensors measure many signals such as temperate, vibration, sound, humidity, and pressure that have a contribution to the health of the part. In a modern IoT environment, these sensors gather and save a lot of data, thousands of measurements per sensor per second. The data are forwarded to a maintenance prediction model and the model predicts the part replacement. The model has to be trained with historical data ensuring a good prediction over future measurements.

There are many measurements in industries today such as temperature, vibration, acoustic and infrared. Deep learning has evolved as a solution for many intelligent problems in various fields. However, for predictive maintenance applications, it is preferable to use simpler and faster machine learning methods, given especially the need of time-critical applications such as motor failures, bearings and impellers. The current work lists and describes the machine learning ones that have been already implemented, by providing surveys and discussions of the prevalence of each one on a robust effort to predict infrastructure maintenance throughout the collected processing-list. Many machine learning models have been successfully implemented to various infrastructure-based applications of predictive maintenance. There is always a model that fits and achieves a good accuracy of prediction. However, in order to avoid the "black box" character of deep learning, the use of simpler algorithms.

References

[1] Karthik Chava, "Machine Learning in Modern Healthcare: Leveraging Big Data for Early Disease Detection and Patient Monitoring", International Journal of Science and Research (IJSR), Volume 9 Issue 12, December 2020, pp. 1899-1910, https://www.ijsr.net/getabstract.php?paperid=SR201212164722, DOI: https://www.doi.org/10.21275/SR201212164722

[2] Data Engineering Architectures for Real-Time Quality Monitoring in Paint Production Lines. (2020). International Journal of Engineering and Computer Science, 9(12), 25289-25303. https://doi.org/10.18535/ijecs.v9i12.4587

[3] Vamsee Pamisetty. (2020). Optimizing Tax Compliance and Fraud Prevention through Intelligent Systems: The Role of Technology in Public Finance Innovation. International Journal on Recent and Innovation Trends in Computing and Communication, 8(12), 111–127. Retrieved from https://ijritcc.org/index.php/ijritcc/article/view/11582

[4] Xie, Z., Li, H., Xu, X., Hu, J., & Chen, Y. (2020). Fast IR drop estimation with machine learning. Proceedings of the 39th International Conference on Computer-Aided Design, 1–8. https://doi.org/10.1145/3400302.3415763

[5] Ghahramani, M., Qiao, Y., Zhou, M., O'Hagan, A., & Sweeney, J. (2020). AI-based modeling and data-driven evaluation for smart manufacturing processes. IEEE/CAA Journal of Automatica Sinica, 7(4), 1026–1037. https://doi.org/10.1109/JAS.2020.1003114