ENSIGHTS: Intelligent Monitoring of Electric Power Transmission Assets

Alex de Vasconcellos Garcia[1], Gabriel Resende Machado[1], Carla Chrystina de Castro Pacheco Ferreira[1], Edward Hermann Haeusler[1], Jefferson Barros dos Santos[1], Edmilson Varejão[1], Pedro Schneider[1], Athos dos Santos Barbosa[2], Maurício Magalhães[2], Marcelo de Carvalho[3] and Ana Cristina de Freitas Marotti[3]
[1]Pontifical Catholic University of Rio de Janeiro (PUC-Rio), Rio de Janeiro, Brazil
[2]Radix Engineering and Software S.A., Rio de Janeiro, Brazil
[3]FURNAS Centrais Elétricas S.A., Rio de Janeiro, Brazil
agarcia@inf.puc-rio.br
gresende@inf.puc-rio.br
cpacheco@inf.puc-rio.br
hermann@inf.puc-rio.br, jsantos@inf.puc-rio.br
eneto@inf.puc-rio.br
pschneider@inf.puc-rio.br
athos.barbosa@radixeng.com.br
mauricio.magalhaes@radixeng.com.br
marcarv@furnas.com.br
amarotti@furnas.com.br

Abstract: This work aims to use Data Science techniques to build predictive models that will eventually improve maintenance plans regarding power transformers by reducing shutoffs and transmission downtime. This work is part of a 36-month long Research and Development (R&D) project started on January 2021, as of the writing of this report the project is halfway through. The analytic models described herein have already been tested while most of the remaining work is yet to be done. In a single Data Lake environment, we will consolidate several databases from information systems that support the company's operation and maintenance processes. Various machine learning models will run on this data, and their results will appear in a dashboard, alongside several traditional indicators. This process will be integrated and consolidated in a computing platform with cloud architecture. We use Machine Learning (ML) to develop the models. Based on an Agnostic Probably Approximately Correct (PAC) Learning study of the available datasets estimating their Vapnik–Chervonenkis dimension, we choose Random Forests (RF) algorithms to be used in the new indicators. So far, the project has produced two new indicators: Chromatographic Assay Indicator (CAI) and Electrical Failure Risk Indicator (EFRI). The CAI indicator evaluation uses a Random Forest Algorithm trained with an external dataset due to the small number of power transformers failures in the O&M data. This indicator performed much better than classical chromatographic indicators to predict electric or temperature problems on the test set. The EFRI indicator correlates monitoring data available from an existing Supervisory Control and Data Acquisition (SCADA) system with maintenance data from an existing Enterprise Resource Planning (ERP) system through a RF algorithm capable of alerting to a higher risk of electric failure. ANEEL funds this work as R&D project PD-00394-1907/2019 titled "Aplicabilidade de nova tecnologia voltada para o desenvolvimento de um modelo de monitoramento inteligente dos ativos de transmissão".

Keywords: analytical models, predictive maintenance, intelligent maintenance, machine learning, agnostic PAC learning, HV substations

1. Introduction

Digital transformation is the integration of digital technologies into all areas of a given business, fundamentally changing how it operates and delivers value to its customers. It is also a cultural change that requires organizations to continually challenge the way it works, through experimentations and a "fail fast learn faster" approach.

Among Furnas' strategic guidelines, there is the establishment of a data-driven culture, from the development of a strategic use of information, by encouraging analytical skills in the business areas. This strategic guideline is responsible for fostering the strategic use of data in solving business problems. In this aspect, Artificial Intelligence (AI) stands out as the main technology capable of dealing with large amounts of data by extracting information that is relevant to the business, as well as the process of decision making.

One of the main business pains of energy companies, especially regarding energy transmission assets, is the unscheduled downtime of machinery and equipment. These interruptions, normally triggered by failures, tend to lead to an aggravation of operational and quality indicators, potentially leading to fines imposed by regulatory agencies. In some cases, these failures can even cause the operational suppression of the integrated Electrical System.

Within this scope, the analytical models based on Machine Learning algorithms intend to continuously monitor the risk of failures - or other relevant events - on power transformers. The analytical models proposed herein complement the methodologies commonly used today in the industry. Currently, prevailing techniques rely on industrial or internal mean life statistics (*i.e.* mean time to failure) to schedule maintenance activities. Modern techniques monitor assets in real-time by checking whether specific indicators are within manufacturers' recommended ranges. In contrast, the proposed methodology uses Machine Learning to find failure occurrence patterns by learning from past failure data from large databases. The proposed algorithms are expected to alert to possible failures not identified by state-of-the-art technologies.

Therefore, this work aims to present one of the Digital Transformation initiatives with the use of AI, through the development of a Research, Development and Innovation (R&D+I) Project entitled "Applicability and Implementation of new technology aimed at the development of an intelligent monitoring model of transmission assets": a 36-month long Research and Development (R&D) project started on January 2021, now on its halfway through. The solution will integrate relevant data from various sources such as ERP and SCADA systems, thus enabling the system to: (i) process the data in an Extract, Transform and Load (ETL) layer with a scalable architecture in the Azure cloud, (ii) access the predictive algorithms and applying them to the data, and (iii) display the results in a dashboard. In this context, especially when it comes to power transformers, the prediction of failures and, consequently, the preventive maintenance becomes an essential activity for the operation of the company and the integrated electrical system. More specifically, the following benefits are expected from this work:

- fewer inspections, thus resulting in increased productivity;
- improvement in the company's main indicators, such as quality, safety, and energy efficiency;
- reduction of unplanned downtime, flaws, regulatory fines, maintenance labor, and equipment costs.

The remaining content of this article is structured as follows: Section 2 explains the design of the project's cloud architecture. Section 3 provides detailed information regarding all the steps that were conducted to process the data and generate the two predictive models behind the indicators (i) CAI (Chromatographic Assay Indicator) and EFRI (Electrical Failure Risk Indicator). Section 4 describes the main sections of the dashboard in development, and finally Section 5 brings the final considerations and future work.

2. Solution architecture

This work adopted a typical architecture for Big Data projects, although it deals with structured and semi-structured data sources. In general, Big Data architectures must deal with heterogeneous data sources that provide data in different formats, structured and unstructured (variety). This data needs to travel and persist in large amounts (volume) to be processed as they are transmitted (velocity).

Over the years, industry and academy have proposed several architectures for projects of this type, some of them becoming references for implementation, such as the *Lambda* (Marz & Warren, 2015) and *Kappa* (Kreps, 2014) architectures. Kleppmann (2017) presents the principles and fundamentals of data platforms for Big Data.

Figure 1 depicts the reference architecture used in this project. A reference architecture presents the existing software components in a computer system and defines the role of each of these components and the relationships between them. The technical team performed this description at a high level without considering the technical details and specific technologies. This approach was critically important given the feature of this project that technology contracting would be established thoroughly in a formal supplier bidding process.

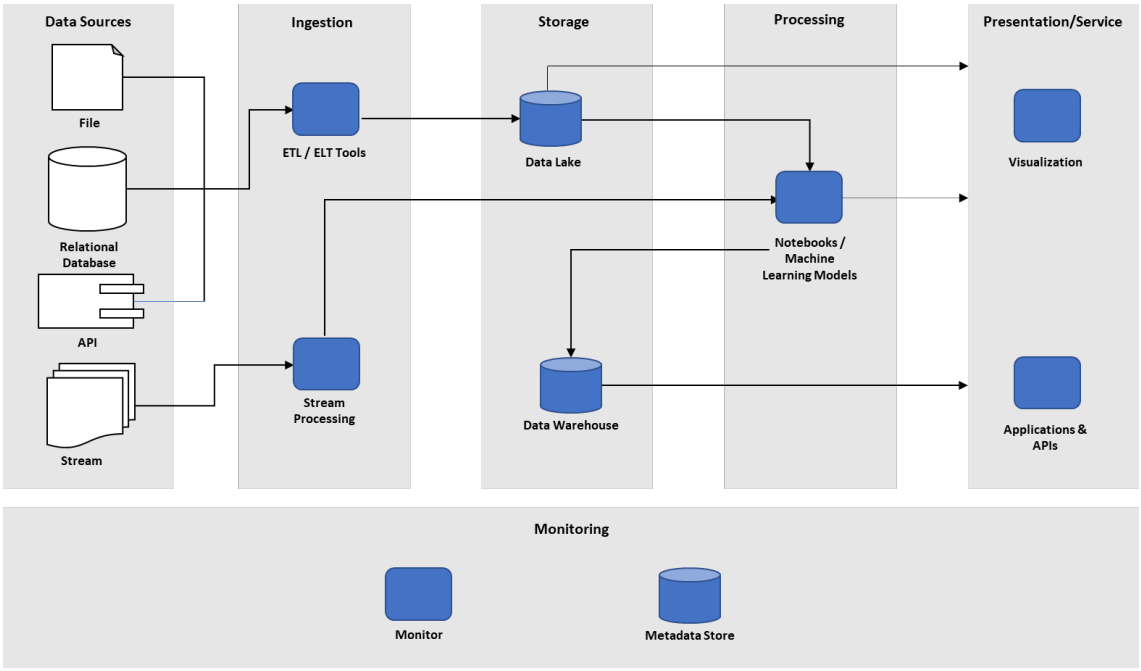

Figure 1: Reference architecture used in this project

The components are distributed in four logical layers, an architecture model commonly used in literature (Zburivsky & Partner, 2021). Although these logical layers are suitable for describing architectures of different complexity levels, they are populated and adapted with the minimum components necessary for the project goals. The Ingestion layer is responsible for getting the data from the sources and loading it into the cloud data platform. Extract-Transform-Load tools load data into a Data Lake in its raw form. Automated data cleansing and handling processes perform the necessary transformations to prepare the data for use. Transformed data become available in a specific area of the Data Lake to be consumed. The Ingestion layer is also responsible for dealing with real-time data through stream processing. In this case, data are available for consumption in real-time. In the Processing layer, analysts use the data for exploratory analysis and to train and test machine learning models. From this processing, analysis and prediction models emerge. Learning from this processing allows a refinement of the data structure involved, which can be better organized and prepared for actual consumption by visualizations or applications. A Structured Data Repository stores these enriched and well-understood data (which can be implemented using a Data Warehouse, a relational, or NoSQL database). This Structured Data Repository and Data Lake form the Storage layer of the architecture. Finally, data feed dashboards and business applications (software systems) in the Service layer. To choose an implementation platform, this reference architecture was instantiated for the three most-known cloud providers in the market: Google, Amazon, and Microsoft. Each component was mapped to a particular vendor service. This approach allowed for accurately comparing available solutions and facilitating the supplier selection process. Finally, it was decided to implement the solution in Microsoft's Azure platform.

3. Machine learning models

In this work, we define two indicators using machine learning algorithms, namely: (i) Chromatographic Assay Indicator (CAI) and (ii) Electric Failure Risk Indicator (EFRI).

3.1 Chromatographic Assay Indicator (CAI)

The CAI indicator uses the concentration of gases dissolved in insulating oil to predict the conditions of a power transformer. Since there were very few diagnosed chromatographic samples regarding the company's transformers, transfer learning was performed as a step to generate this indicator. Data from four related works (Li et al., 2019; Ibrahim et al., 2018; Duval and de Pabla, 2001; Morais et al., 2004) was unified into one labelled dataset. The original dataset had seven different failure classes, which were grouped into three classes in the final dataset: (i) *Normal*; (ii) *Electrical Failure*; and (ii) *Thermal Failure*. We use chromatographic data from the ERP system to evaluate the model.

3.1.1 Data preprocessing

During data cleansing, null and invalid values were replaced. Duplicates and blank registers were also removed, obtaining Table 1. A further pre-processing step included four attributes representing the following gas ratios: (i) R1 = CH_4 / (H_2 + 0.4); (ii) R2 = C_2H_2 / (C_2H_4 + 0.4); (iii) R4 = C_2H_6 / (CH_4 + 0.4); and (v) R5 = C_2H_4 / (C_2H_6 + 0.4), following a suggestion from (Kreps, 2014), to use 0.4 ppm as a correction for undetectable values.

Table 1: Data distribution in a unified dataset after data cleansing

Dataset	Measured Gases							Samples per Class							Amount of Samples
	H_2	CH_4	C_2H_2	C_2H_4	C_2H_6	CO	CO_2	NF	PD	D1	D2	T1	T2	T3	
IEEE1	✓	✓	✓	✓	✓	✓	✓	0	10	10	10	10	10	10	60
IEEE2	✓	✓	✓	✓	✓	✓	✓	0	16	49	54	19	9	38	185
IBRAHIM	✓	✓	✓	✓	✓	✓	✓	0	50	84	143	109	68	110	564
IECTC	✓	✓	✓	✓	✓	✓	✓	27	0	0	0	0	0	0	27
UFSC1	✓	✓	✓	✓	✓	✓	✓	119	0	0	0	0	0	0	119
UFSC2	✓	✓	✓	✓	✓	✓	✓	191	0	0	0	0	0	0	191
Amount of Samples per Class								337	76	143	207	138	87	158	1146
Amount of Samples per Class (no duplicates)								333	70	133	183	119	62	122	1022

The second step was normalization (or feature scaling), where the data was transformed to be dimensionless to avoid bias in ML models. The data normalization was according to the IQR (Interquartile Range), which is robust to outliers and does not assume any data distribution. This normalization $Norm_{IQR} = \left(x - Q_{2/4}\right)/(Q_{3/4} - Q_{1/4})$ was applied to the following attributes: (i) H_2, (ii) CH_4, (iii) C_2H_4, (iv) C_2H_6, (v) R1, (vi) R4 and (vii) R5. The remaining acetylene (C_2H_2) and R2 ratios were normalized in a different way, according to the equations $Stand_{C_2H_2} = (log(x+1) \,/\, max((log(C_2H_2 + 1))$, and $Stand_{R2} = y/max(R2)\,, y \in R2$, respectively, because they had a lot of zeros and outliers. The same normalization of the unified dataset was applied to the Furnas' dataset.

3.1.2 CAI's model development

After the unification and preprocessing of datasets, a comparative study was conducted between Machine Learning algorithms in the Weka framework (Hall et al., 2009). In this R&D project, it was sought an algorithm with a good performance in terms of satisfactory results, avoiding bias, and presenting a low VC (Vapnik-Chernonenkis) dimension (Ehrenfeucht et al., 1988) for the datasets. The experiments were run with four algorithms: (i) Support Vector Machines (LibSVM) (Chang and Lin, 2011), (ii) Random Forest (Breiman, 2001); (iii) Fuzzy Unordered Rule Induction Algorithm (FURIA) (Hühn and Hüllermeier, 2009); and (iv) Random Trees (Pfahringer, 2011) The performance evaluation of the algorithms used the mean accuracy metric. The experiments were run by aggregating the seven original class labels into larger classes, along with combining different sets of attributes. All the scenarios evaluated are shown in Table 2 and were tested with 5 and 10-fold cross-validation (CV).

Table 2: Definition of different class label aggregations and combinations of sets of attributes evaluated along the experiments

Class label aggregation	A1	$PD \cup D1 \cup D2 =$ Electrical Failure; $T1 \cup T2 \cup T3 =$ Thermal Failure; $NF =$ Normal.
	A2	$D1 \cup D2 \cup T1 \cup T2 \cup T3 =$ Electric Discharge, Thermal Failure; $PD =$ Electric Arc; $NF =$ Normal.
	I	No aggregation. All classes considered individually.
Combination of sets of attributes	R1, R2, and R5	$H_2 \cup CH_4 \cup C_2H_2 \cup C_2H_4 \cup C_2H_6 \cup R1 \cup R2 \cup R5$.
	Without R5	$H_2 \cup CH_4 \cup C_2H_2 \cup C_2H_4 \cup C_2H_6 \cup R1 \cup R2 \cup R4$.

	With R5	$H_2 \cup CH_4 \cup C_2H_2 \cup C_2H_4 \cup C_2H_6 \cup R1 \cup R2 \cup R4 \cup R5$

The best results of the experiments with four algorithms are highlighted in Table 3 for each scenario. Random Forest presented the best performance in all scenarios with a 10-fold CV. This algorithm has a low VC dimension and is robust to bias.

The second phase of tests was developed in Python, using Scikit-Learn library (Pedregosa et al., 2011), following the same specifications. We split the unified dataset into a training set with 80% of the data and a testing set with the remaining 20%. The RandomizedSearchCV method was used in the training set with 10-fold CV to search for the best hyperparameters. The standard hyperparameters of Scikit-Learn presented better accuracy and F1 result metrics. These results were similar to those of Random Forest in Weka. The results of the experiments are presented in Table 3.

Table 3: Results of experiments in all scenarios

Algorithms	A1 w/o R5	A1 w/ R5	A1 R1, R2, R5	A2 w/o R5	A2 w/ R5	A2 R1, R2, R5	I w/o R5	I w/ R5	I, R1, R2, R5
LibSVM	74.76%	74.17%	70.55%	72.11%	73.09%	71.43%	61.35%	62.82%	62.72%
Random Forest	92.37%	92.66%	91.39%	90.22%	90.02%	90.31%	83.27%	85.42%	84.64%
FURIA	89.53%	89.24%	88.65%	86.01%	85.42%	86.20%	77.20%	79.84%	79.55%
Random Tree	80.82%	81.12%	80.14%	75.83%	72.50%	74.76%	63.70%	64.38%	63.70%

We compared the final results of Random Forest to the classic insulation oil diagnosis methods: (i) Rogers, (ii) Doernenburg, (iii) NBR 7274, (iv) IEC 599 and (v) de Duval's Triangle. In addition, the refined methods of (vi) Rogers-R (refined Rogers) and (vii) IEC-R (refined IEC) (Taha et al., 2016), and two hybrid approaches, (viii) Doernenburg + Duval (Doerneval) and (ix) Doernenburg + IEC Ibrahim (DIEC-R) were also evaluated. The results appear in Table 4, where the best results are highlighted in blue and the worst in red. The first three columns refer to the whole dataset (Test and Training), whereas the last two refer only to the Test set, therefore the RF Algorithm results are shown only for the latter. Doernenburg presented the best accuracy between classic methods. The hybrid DIEC-R presented the best result between classic/hybrid methods with 73.2% accuracy and 71.3% F1. The Random Forest outperformed in almost 19 percentage points such results, presenting 92.2% in accuracy and 92.1% in F1-score.

Table 4: Results of nine diagnosis methods and Random Forest model in the test set

Method	Accuracy (only class 0)	Accuracy (only failures)	Accuracy (all classes)	Accuracy (Test set)	F1-score (Test set)
Rogers	0.081	0.476	0.347	0.351	0.242
Rogers (refined)	0.102	0.680	0.491	0.468	0.278
Doernenburg	0.399	0.013	0.139	0.136	0.145
NBR 7274	0.192	0.665	0.511	0.517	0.433
IEC Ratio	0.192	0.620	0.481	0.512	0.410
IEC (refined)	0.144	0.908	0.660	0.644	0.566
Duval's Triangle	0.000	0.901	0.608	0.605	0.388
Doernenburg + Duval	0.399	0.871	0.717	0.697	0.523
Doernenburg + IEC (Ibrahim)	0.450	0.882	0.742	0.732	0.713
Random Forest	-	-	-	0.922	0.921
	Number of samples				
	333	689	1022	205	

3.2 Electric Failure Risk Indicator (EFRI)

In contrast to CAI, EFRI uses both (i) maintenance data and (ii) analogic time-series data from Furnas to predict electrical failures in power transformers. Maintenance data come from a system known as SAP Plant Maintenance (SAP PM), whilst analogic time-series data obtain from a SCADA system known as SAGE. To train the Random Forest model, we used SCADA data as input attributes, whilst historical maintenance data is used as a label attribute; therefore, to evaluate the EFRI indicator, only the SCADA data is used.

3.2.1 Data preprocessing

Because of discrepancies between both SAP and SAGE's datasets, the following preprocessing steps were necessary: (1) restructuring of analogic time-series data from SAGE into a tabular shape; (2) failure categorization of maintenance reports from SAP PM; and (3) data correspondence.

Regarding Step 1, SAGE persists analogic data collected from sensors in text files with *.pas* extension. Each .pas file corresponds to a day of measurements and is formed by a plethora of analogic variables related to temperature, frequency, power, and voltage, which have their measurements logged every five minutes, thus totaling 288 samples per day. There are *n* measurements for each sample, where *n* depends on the system configuration on the day the sampled data is collected. The .pas files are structured and formatted as non-tabular. The beginning of each file is composed of a header, followed, for each sample, by a line containing its timestamp and its corresponding measurements, each of them located in a new line as well.

The process of transforming the .pas files into a tabular form started by indexing the analogic variable lines to their corresponding metrics along the 288 periods of time of the file. It converts each .pas file into a table containing 288 lines by *n+1* columns, where the extra column represents the time of each measurement. Each table had its null measurements ignored while the valid ones were aggregated by computing their daily mean. This process iterated on all .pas files producing, in the end, a unique table T_d containing all the daily means concerning this period. At last, T_d is transformed into a final table $T_{\Delta d}$, consolidating the mean of Δd consecutive days.

Step 2 categorizes maintenance failures using a variant of the Levenshtein algorithm for assessing text similarity (Levenshtein, 1966). This step is necessary because failures are registered as free text in a column called *Description* on each maintenance report. The categorization process uses predefined keywords to identify the failure categories that better match the Description column contents. At the end of the categorization process, another column containing the failure categories that correspond to each maintenance report is produced. After many tests, models that work only with failures under the category 'Electric Failures' presented better results. Therefore, the new indicator, named Electric Failure Risk Indicator, is trained using only Electric Failures.

Step 3 was responsible for performing the correspondence between the analogic measurements from SAGE in table $T_{\Delta d}$, produced by Step 1, to the maintenance reports filtered out by the process of failure categorization, described in Step 2. However, the identifiers from SAP PM and SAGE differ in format, which required a procedure that can map them in a file, matching both outputs of Steps 1 and 2. Figure 3 depicts an example of the existing mismatch between the SAP PM and SAGE's identifiers by using a red arrow. Only by mapping SAP PM and SAGE's identifiers, it is possible to have access to the corresponding analogic measurements of each transformer.

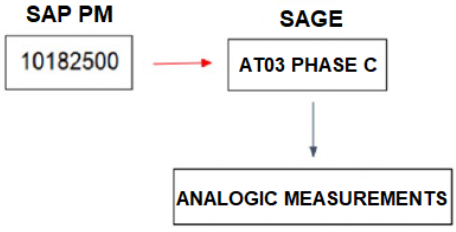

Figure 3: The existing mismatch between SAP PM and SAGE's identifiers

3.2.2 EFRI's model development

After generating the table, we started the development of the EFRI's model. The proposed methodology is formed by two main steps: (1) Preprocessing and (2) Experimental Evaluation, as depicted in Figure 4. It is worth

mentioning that the whole flow can be resumed again from the step of *Outliers treatment* in cases where the generated model presents non-satisfactory results.

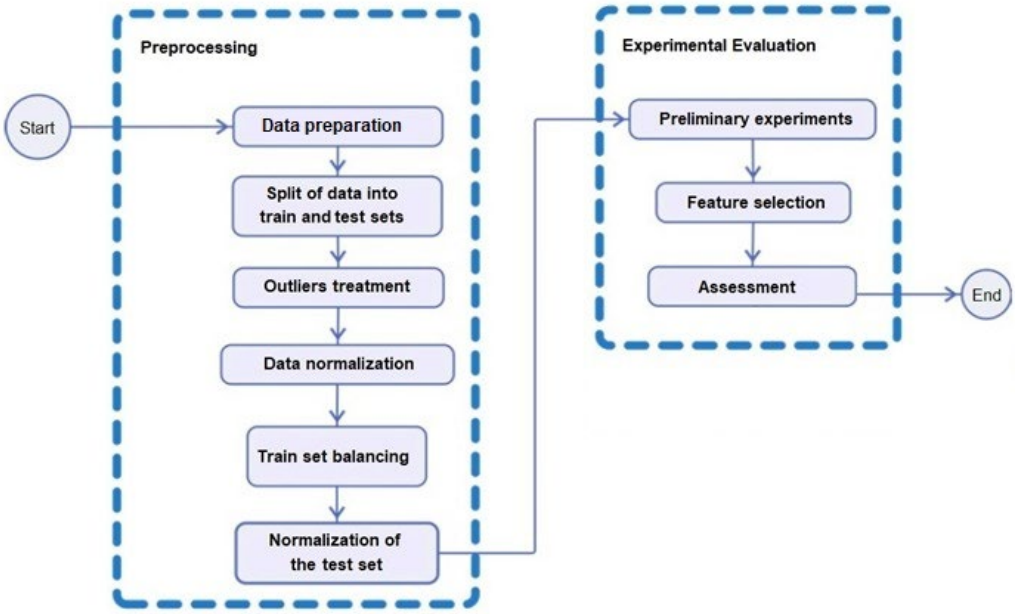

Figure 4: Applied methodology for developing the EFRI's model

The *Preprocessing* step began with *data cleansing*, where the zeroed values were transformed into null values and were ignored throughout the subsequent processes. After that, we have all data split into two sets: (i) 80% was considered as training data (7,322 records), whereas the remaining (ii) 20% (1,810 records) was set apart for evaluating purposes. With the Training set's definition, an outlier´s treatment was conducted by computing the Interquartile range (IQR). However, with a more conservative approach, according to Han et al (2012): for a value *x,* it is an outlier if (i) $x < Q_1 - 3 * IQR$, or (ii) $x > Q_3 + 3 * IQR$. All samples containing at least one value outside the IQR interval are not considered and discarded in those cases. In the end, 34 samples from class "Electrical Failure" and 1,192 samples from class "Non-electrical Failure" were discarded. After removing the outliers, the Training set was processed based on a class-independent IQR normalization, according to equation $Norm_x = \frac{x - median}{IQR}$. Later, it was necessary to balance the Training set since there were about 40 times fewer samples from class "Electrical failure" compared to its counterpart class. Better results show up if appealing to oversampling, where an algorithm known as SMOTE (Chawla et al. 2002) was used in Weka (Witten and Frank, 2002). This algorithm synthetically increases by 3,835% the amount of "Electrical Failure" samples, totalling 5,941 samples, against 5,945 samples of "Non-electrical Failures". At last, the Test set was also normalized based on the parameters computed from the Train set, according to Equation $Norm_{x_{TEST}} = \frac{x_{TEST} - median_{TRAIN}}{IQR_{TRAIN}}$. It is worth highlighting that the outliers present in the test set were also removed per attribute, based on the Training set's parameters.

Once *Preprocessing* was finished, the *Experimental Evaluation* step started. For the *Preliminary experiments*, the Random Forest algorithm was first trained in Weka on 10-fold cross-validation using the Training set and evaluated afterwards on the Test set. Next, for *feature selection, we use* three algorithms: (i) Relief, (ii) Information Gain, and Principal Component Analysis (PCA). The results given by the three methods were ranked, combined and evaluated until the definition of a set of attributes which led to the best results in the test set. In the *assessment* process*,* the "Electrical failure" class is a positive class. Based on this premise, two metrics are firstly taken into consideration: (i) False Omission Rate (FOR); and (ii) Positive Predictive Value (PPV). FOR stands for the ratio of electrical failures classified as non-electrical, *i.e.,* $FOR = \frac{FN}{FN+TN}$, whilst PPV stands for the ratio of corrected classified failures over the number of samples classified as failures, *i.e.* $PPV = \frac{TP}{TP + FP}$. From FOR and PPV was created another metric, called Probability Quotient $PQ = \frac{PPV}{FOR}$. PQ increases the risk of electrical failure when the model predicts electrical failure for a given sample. It was also analysed other three metrics during the assessment process: (i) accuracy, (ii) area under the ROC curve (AUC), and (iii) F1-score. The best EFRI model achieved the performance shown in Table 5.

Table 5: The best performance of the EFRI model

Accuracy	AUC	F1-score (Overall)	F1-score (Electrical Failure)	PQ
97.9%	92.5%	97.9%	62.1%	66.15

4. Dashboard

The CAI and EFRI indicators are displayed in a dashboard that shows an overview of the transformers' state and help the decision-making process. After the operator logs into the system, a screen like Figure 5 shows up. The dashboard screen is split it into four areas for the sake of explanation.

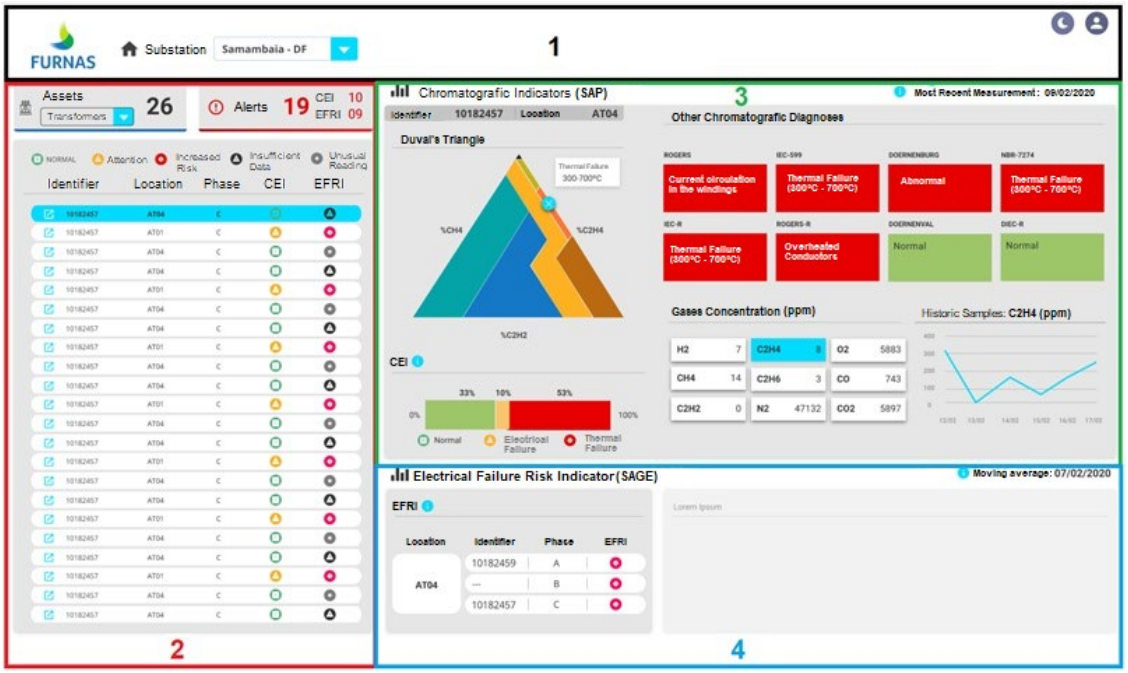

Figure 5: Dashboard main screen split into sections

In Area 1, the logged operator can change the current substation and toggle to dark mode by clicking on the half-moon icon. By clicking the rightmost button, the operator can verify his account information and log out of the system. In Area 2, the operator can glimpse the total number of transformers in the selected substation and their corresponding status, given by the CAI and EFRI indicators. Area 3 provides a general diagnosis of the selected equipment by displaying its most recent chromatographic sample data and its analysis. Apart from CAI, other relevant chromatographic diagnoses are also shown, in addition to the history of gas concentration, which updates a line plot just after the operator selects a gas of choice. At last, Area 4 displays the condition of the phases from the transformer selected in Area 2, according to the EFRI indicator. It is worth mentioning that the dashboard is under development, where its rightmost blank area is reserved for future indicators and metrics that can be developed until the end of the project.

5. Conclusions and future work

The products under development will improve maintenance plans for power assets using Artificial Intelligence (Machine Learning) techniques to build failure prediction models based on the correlation of operational and maintenance data of assets in Furnas' substations. This achievement will have a significant impact in the Transmission Asset Management process of Furnas, as well as reducing fines imposed by regulatory agencies.

Other side benefits of this project include:

- the Data Lake built for this project can be used in other Digital Transformation initiatives. The way the Data Lake was designed (*i.e.* the technology architecture) will allow the gathering and storage of additional data

from many other legacy systems that support several O&M processes that are outside the scope of this project, thus providing the development of other AI initiatives in the company;

- it can also be develop predictive models for other asset types, such as wind turbines, hydraulic turbines, and isolators, using the same development framework created for this project;
- other companies in the electricity sector can use this Platform. They can use the know-how obtained with this Platform development (*i.e.* the technology architecture and models) to generate and evaluate their own AI models specifically for their assets. Furnas can develop a new business model considering revenue from the use of its platform by other companies.

The objective of this R&D+I project is in line with the process of Technological Innovation and Digital Transformation, listed as a strategic theme through Resolution No. 10,332 of 4/28/2020 of the Federal Government, which establishes the Digital Government Strategy and, among other initiatives, the implementation of artificial intelligence capabilities in at least twelve federal public services by 2022.

References

Breiman, L. (2001) *Random forests*. Machine learning, v. 45, n. 1, p. 5-32.

Chawla, N., et al. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, p.321–357.

Chang, C-C.; Lin, C-J. (2011) *LIBSVM: a library for support vector machines*. ACM transactions on intelligent systems and technology (TIST), v. 2, n. 3, p. 1-27.

Duval, M.; de Pabla, A. (2001) *Interpretation of gas-in-oil analysis using new IEC publication 60599 and IEC TC 10 databases*. IEEE Electrical Insulation Magazine, v. 17, n. 2, p. 31-41.

Ehrenfeucht, et al (1988) *A general lower bound on the number of examples needed for learning*. In Proceedings of the First Annual Workshop on Computational Learning Theory, COLT '88, Cambridge, MA, USA, August 3-5, 1988, pages 139–154. ACM/MIT.

Hall, M. et al. (2009) *The WEKA data mining software: an update*. ACM SIGKDD explorations newsletter, v. 11, n. 1, p. 10-18.

Han, J. et al. (2011). *Data mining: concepts and techniques.* Elsevier.

Hühn, J.; Hüllermeier, E. (2009) *FURIA: an algorithm for unordered fuzzy rule induction*. Data Mining and Knowledge Discovery, v. 19, n. 3, p. 293-319.

Ibrahim, S. I. et al. (2018) *DGALab: an extensible software implementation for DGA.* IET Generation, Transmission & Distribution, v. 12, n. 18, p. 4117-4124.

Kleppmann, M. (2017). Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems (1st Edition). O'Reilly Media.

Kreps, J. (2014). Questioning the lambda architecture. Online Article, July, 205. https://www.oreilly.com/radar/questioning-the-lambda-architecture/.

Levenshtein, V. I., (1966) *Binary codes capable of correcting deletions, insertions, and reversals*, In *Soviet physics doklady* (pp. 707–710).

Li, E. et al. (2019) *Fault diagnosis of power transformers with membership degree*, IEEE Access, v. 7, p. 28791-28798.

Marz, N., Warren, J. (2015). Big data: Principles and best practices of scalable real-time data systems. Manning.

Morais, D. R. et al. (2004) *Ferramenta inteligente para detecção de falhas incipientes em transformadores baseada na análise de gases dissolvidos no óleo isolante*.

Pfahringer, B. (2011) *Random model trees: an effective and scalable regression method*, University of Waikato, New Zealand, http://www.cs.waikato.ac.nz/~bernhard.

Pedregosa, F. et al (2011) *Scikit-learn: Machine Learning in Python,* JMLR 12, pp. 2825-2830.

Taha, I. B. M. et al. (2016) *Refining DGA methods of IEC Code and Rogers four ratios for transformer fault diagnosis*, In: IEEE Power and Energy Society General Meeting (PESGM). IEEE. p. 1-5.

Witten, I., and Frank, E. (2002). Data mining: practical machine learning tools and techniques with Java implementations. Acm Sigmod Record, 31(1), p.76–77.

Zburivsky, D., Partner, L. (2021). Designing Cloud Data Platforms (1st Edition). Manning Publications.