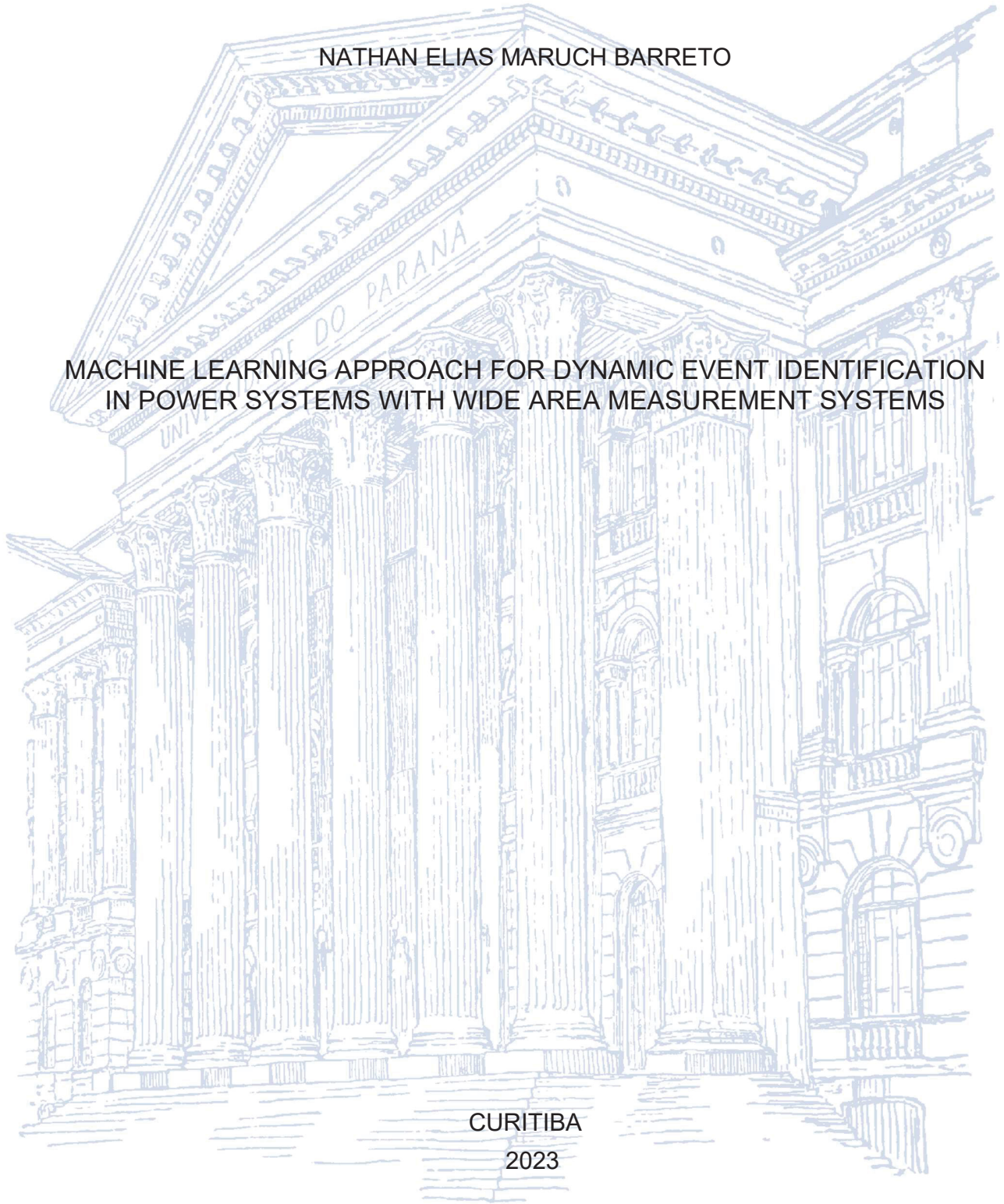


UNIVERSIDADE FEDERAL DO PARANÁ

NATHAN ELIAS MARUCH BARRETO

MACHINE LEARNING APPROACH FOR DYNAMIC EVENT IDENTIFICATION
IN POWER SYSTEMS WITH WIDE AREA MEASUREMENT SYSTEMS

CURITIBA
2023



NATHAN ELIAS MARUCH BARRETO

MACHINE LEARNING APPROACH FOR DYNAMIC EVENT IDENTIFICATION
IN POWER SYSTEMS WITH WIDE AREA MEASUREMENT SYSTEMS

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica – Sistemas de Potência, no Departamento de Engenharia Elétrica, pertencente ao Setor de Tecnologia da Universidade Federal do Paraná, como parte das atividades requeridas para a obtenção do título de Mestre em Engenharia Elétrica.

Orientador:

Prof. Dr. Alexandre Rasi Aoki

Co-orientador:

Prof. Dr. Ricardo Schumacher

CURITIBA
2023

DADOS INTERNACIONAIS DE CATALOGAÇÃO NA PUBLICAÇÃO (CIP)
UNIVERSIDADE FEDERAL DO PARANÁ
SISTEMA DE BIBLIOTECAS – BIBLIOTECA DE CIÊNCIA E TECNOLOGIA

Barreto, Nathan Elias Maruch

Machine learning approach for dynamic event identification in power systems with wide area measurement systems / Nathan Elias Maruch Barreto. – Curitiba, 2023.

1 recurso on-line : PDF.

Dissertação (Mestrado) - Universidade Federal do Paraná, Setor de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica.

Orientador: Alexandre Rasi Aoki

Coorientador: Ricardo Schumacher

1. Aprendizado de máquinas. 2. Inteligência artificial. 3. Sistemas de energia elétrica. 4. Máquinas de medição por coordenadas. I. Universidade Federal do Paraná. II. Programa de Pós-Graduação em Engenharia Elétrica. III. Aoki, Alexandre Rasi. IV. Schumacher, Ricardo. V. Título.

Bibliotecário: Elias Barbosa da Silva CRB-9/1894



MINISTÉRIO DA EDUCAÇÃO
SETOR DE TECNOLOGIA
UNIVERSIDADE FEDERAL DO PARANÁ
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO ENGENHARIA
ELÉTRICA - 40001016043P4

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação ENGENHARIA ELÉTRICA da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **NATHAN ELIAS MARUCH BARRETO** intitulada: **MACHINE LEARNING APPROACH FOR DYNAMIC EVENT IDENTIFICATION IN POWER SYSTEMS WITH WIDE AREA MEASUREMENT SYSTEMS**, sob orientação do Prof. Dr. ALEXANDRE RASI AOKI, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

Curitiba, 13 de Fevereiro de 2023.

Assinatura Eletrônica
16/02/2023 14:45:18.0
ALEXANDRE RASI AOKI
Presidente da Banca Examinadora

Assinatura Eletrônica
14/02/2023 14:06:00.0
GUSTAVO HENRIQUE DA COSTA OLIVEIRA
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
10/03/2023 15:19:47.0
VITOR HUGO FERREIRA
Avaliador Externo (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
14/02/2023 13:47:56.0
ROMAN KUIAVA
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

I dedicate this dissertation to my mother, Raquel, grandmother Eraide and fiancée Maria who have been my pillars of support in different areas and stages of my life.

ACKNOWLEDGEMENT

First and foremost, I would like to thank my mother, Raquel Maruch, who has raised me into someone who can hold his head up high and who seeks fairness, truth and to achieve higher standards every day – be they professional, academic, or ethical. Without her and her herculean efforts as a divorced woman in our sexist society, I would have gotten nowhere and certainly wouldn't have had as many opportunities as I've had.

I would also like to thank my grandmother, Eraide Maruch, who has instilled a thirst for knowledge in me since my childhood and aided me and my mom during times of financial struggle by paying for my high school tuition fees.

I also thank Prof. Dr. Alexandre Rasi Aoki, whose continuous guidance has shaped me into the researcher and professional I am today. His mentorship inspires me and makes me hopeful for Brazilian higher education – and his friendship has helped me during times when problems weren't defined by mathematical expressions.

I would like to thank COPEL Geração e Transmissão for the financial support, research resources, and opportunities provided through R&D project PD-06491-0531/2019, regulated by ANEEL, coordinated by Prof. Dr. Alexandre Rasi Aoki and executed alongside GNARUS Institute. I'm also grateful to its employees, who have aided me with valuable information throughout the last two years.

I'd like to extend my gratitude to my peers at ExxonMobil. There, I've become a better problem solver and gained the confidence I lacked in my abilities to deliver something meaningful and valuable – which has extended over to my academic life.

I'd also like to thank my mentors at ExxonMobil, Danielle Nascimento, Jean Nascimento, and Kaushal Patel. My ascension to Information Technology Architect is owed to all of them; and exercising the holistic views, modular frameworks, and stakeholder negotiations required at my job has helped me become a better researcher. Furthermore, their support and friendship have kept me balanced and with the right mindset more times than I can count.

Lastly, I'd like to thank my fiancée, Maria Gabriele, for all the love, companionship, comprehension, and support given prior to, during, and after the COVID pandemic. To have someone like her by my side has been absolutely invaluable.

“The struggle itself toward the heights is enough to fill a man’s heart.”

Albert Camus

RESUMO

Ao longo dos últimos dez anos, a disponibilidade de WAMS (*Wide Area Measurement Systems*) tem constantemente aumentado e, com isso, a necessidade de se otimizar seu uso em relação a uma ampla gama de capacidades requeridas nos centros de operação. Concorrentemente, o sistema brasileiro tem observado diversos eventos em múltiplos níveis de criticalidade e, portanto, formas de rapidamente identificar irregularidades na rede elétrica têm sido requisitadas pelos operadores. Todavia, mesmo com tal diversidade de eventos registrados por PMUs (*Phasor Measurement Unit*), há dificuldades em se consolidar um banco de dados de eventos e, ademais, sistemas diferem uns dos outros – isto é, os volumes de dados requeridos para *machine learning* e a especificidade de cada sistema criam desafios para a construção de aplicações para detecção e identificação de eventos em uma dada rede. De tal maneira, o presente trabalho propõe uma forma de endereçar tais restrições e habilitar o uso de modelos de *machine learning* na vida real ao modelar um sistema real, simular uma grande quantia de eventos (como medição de PMU) e executar o processo de aprendizado de máquina com esses dados simulados. Tendo posse de qualquer conjunto de dados que contenha medições de evento da mesma PMU simulada, uma validação da aplicabilidade e performance do modelo obtido pode ser feita. Assim, um processo reprodutível e escalável foi definido pelo trabalho a partir de um estudo de caso no corredor Salto Caxias, um subsistema da rede elétrica paranaense operado pela COPEL, que forneceu três conjuntos de dados contendo eventos registrados em uma PMU dessa área. Alguns componentes como barras, linhas de transmissão, transformadores, geradores, PSSs, excitadores e controles de turbina foram modelados dentro da Power System Toolbox, embasada em MATLAB, para simulação de eventos. O algoritmo de *machine learning* selecionado para provar o conceito estabelecido foi rede neural artificial, definindo-se quatro classes possíveis para reconhecimento – “Curto-circuito”, “Perda de Carga”, “Perda de Linha” e “Normal”. Com o modelo de *machine learning* definido e treinado, se aplicaram os dados de eventos reais nele. Os resultados mostram que as métricas da rede neural no processo de aprendizado foram geralmente suficientes para aplicação em vida real, mas que sua performance nos conjuntos de dados de eventos reais foi abaixo da registrada com os dados simulados. Todavia, considerando-se que os dados reais providenciados são de eventos longínquos à PMU observada e ao próprio sistema modelado, distorções e atenuações de sinal são inerentes. Assim, pode-se dizer que o método proposto é aplicável, com mais etapas de pré-processamento de dados, a qualquer dado sistema – caso ele seja minuciosamente modelado e haja disponibilidade de conjuntos de dados de eventos internos ao sistema.

Palavras-chave: Sistemas de Medição de Grandes Áreas. Simulação Dinâmica de Sistemas de Potência. Aprendizado de Máquina. Identificação de Eventos.

ABSTRACT

Over the last ten years, the availability of WAMS (Wide Area Measurement Systems) has steadily increased and, with it, the need to optimize its usage concerning a large array of capabilities required at the operation centers. Concurrently, the Brazilian system has witnessed various events at multiple levels of criticality, and, thus, ways to quickly identify irregularities in the grid have been more and more requested by power transmission and distribution companies. The introduction of machine learning models and algorithms in such a context has been explored by the scientific community. However, even with such a diversity of events and their PMU (Phasor Measurement Unit) measurements, there is hardship in consolidating an event database and systems differ from each other – that is, the data volume required for machine learning and the specificity of each power system create challenges in constructing applications for detection and identification of events in a given grid. As such, the present work proposes a way to address those constraints and further enable the real-life application of machine learning models in a power system with WAMS through the modeling of a real-life system, simulating a large database of events as if they were registered through a PMU in said system and training machine learning models on this simulated data. If one has any dataset containing event measurements from the same PMU (which was simulated), a validation of model performance and applicability can be performed. A reproducible and scalable process was defined to achieve this through one case study for the Salto Caxias subsystem of the Paraná state grid, operated by COPEL, who provided the author with three event datasets captured from a PMU in the aforementioned system. Some components of the system were modeled in MATLAB-based Power System Toolbox for dynamic simulation, such as generators, PSSs, exciters, and turbine governors in addition to buses, transmission lines, and transformers. The selected algorithm for this proof-of-concept was artificial neural network, defining four distinct possible classes it can recognize – “Short-circuit”, “Load Loss”, “Line Loss” and “Normal”. With the machine learning model defined and trained, its application was executed on real event datasets. The results show that the metrics of the neural network model on the learning process were generally sufficient for real-life solutions, but its performance on the real event datasets was below that of the performance on simulated data. However, considering that the provided datasets were from events that happened far away from the selected PMU and its modeled system, signal distortions and attenuations are present. Thus, it can be stated that the proposed method is applicable, with further data preprocessing, to any given system – as long as it is thoroughly modeled and there is availability of datasets of events that happened within it.

Keywords: Wide Area Measurement Systems. Dynamic Power System Simulation. Machine Learning. Event Identification.

LIST OF ILLUSTRATIONS

Figure 1 – Number of disturbances registered in the Brazilian Core Grid in the last five years.	20
Figure 2 – Major elements of a modern PMU, assembled in a diagram.	29
Figure 3 – PDC architecture with OPC.	31
Figure 4 – Example of WAMS 5-layer architecture.	32
Figure 5 – Format for files transmitted to and from PMUs.	35
Figure 6 – Supervised Learning structure diagram.	38
Figure 7 – Unsupervised Learning structure diagram.	39
Figure 8 – Reinforcement Learning structure diagram.	40
Figure 9 – k -fold Cross Validation, where each block represents a subset of the whole dataset.	42
Figure 10 – Nonlinear model of a neuron k	44
Figure 11 – Example of threshold function ranging in $[0,1]$	45
Figure 12 – Example of sigmoid function ranging in $[0,1]$	45
Figure 13 – Example of Perceptron behavior for two inputs x_1 and x_2	46
Figure 14 – ANN architecture with n inputs, l layers with m possible neurons each and j output classes.	47
Figure 15 – Confusion Matrix for a multiclass classifier, with accuracy, precision and recall denoted in auxiliary rows and columns.	51
Figure 16 – System dynamic structure divided between mechanical (hydro/thermal) and electrical components, with an operation center controlling all aspects.	53
Figure 17 – Time ranges of dynamic phenomena commonly registered in electrical power systems.	54
Figure 18 – Structure for a power system model for dynamic transient analysis.	55
Figure 19 – Per-phase synchronous generator equivalent circuit.	56
Figure 20 – Block diagram for a synchronous generator's direct axis equations.	56
Figure 21 – Block diagram for a synchronous generator's quadrature axis equations.	56

Figure 22 – Block diagram for a synchronous generator’s electromechanical oscillation differential equations.	57
Figure 23 – Voltage and frequency plots for the first event.	80
Figure 24 – Voltage and frequency plots for the second event	81
Figure 25 – Voltage and frequency plots for the third event.	82
Figure 26 – Block diagram of the general process for data simulation, extraction and preprocessing together with machine learning model creation and application.	85
Figure 27 – Single-phase diagram of Salto Caxias system.	86
Figure 28 – Block diagram for the power system modelling process, with a MATLAB file as its output.	88
Figure 29 – Block diagram for the dynamic simulation file definition for batch event simulation.	90
Figure 30 – Block diagram for validating the constructed model using another proven model.	90
Figure 31 – Block diagram representing the iteration structure designed to simulate all possible events in a power system using PST, with a collection of event simulations as its output.	92
Figure 32 – Block diagram for the simulation and data grooming that happens within every iteration of the batch simulation process.....	93
Figure 33 – Block diagram for preprocessing a given simulated PMU dataset.....	95
Figure 34 – Block diagram for preprocessing a given real-life PMU dataset.	96
Figure 35 – Block diagram for the process of the ANN model and its best iteration.	98
Figure 36 – Block diagram for the exhaustive search for the most performant ANN model.....	99
Figure 37 – Block diagram for the exhaustive search for the most performant ANN model iteration.	100
Figure 38 – Block diagram for the application of real event data on a machine learning model trained with simulated data.	101
Figure 39 – Defined neural network model structure.	103
Figure 40 – Model’s best iteration confusion matrix for the training, testing and validation phases combined.	106

Figure 41 – First event detection graph.	108
Figure 42 – Second event identification graph.	109
Figure 43 – Third event identification graph.	111
Figure 44 – Air gap and open-circuit saturation curves for a given machine.	126

LIST OF TABLES

Table 1 – Initial search in Google Scholar via Publish or Perish, with “Phasor Measurement Unit” as the third axis’ keyword.	61
Table 2 – Articles and papers base filtering.	61
Table 3 – Metrics obtained for the selected papers.	63
Table 4 – Simulation time and sampling rate values for all simulation iterations.	89
Table 5 – Detailed view of the generated dataset for model definition and learning.	102
Table 6 – Metrics overview for the model’s cross-validation.	103
Table 7 – Metrics overview for the validation phases of the best iteration and all 50 different trained ANN model iterations.	105
Table 8 – Classification overview for the first event.	107
Table 9 – Classification overview for the second event.	109
Table 10 – Classification overview for the third event.	110
Table 11 – Required variables for the PST bus model.	124
Table 12 – Required variables for the PST transmission line and transformer models. ..	124
Table 13 – Required variables for the PST generator model with Salto Caxias system values.	125
Table 14 – Required variables for the PST ST5b exciter model with Salto Caxias system values.	127
Table 15 – Required variables for the PSS model with Salto Caxias system values.	127
Table 16 – Required variables for a hydro turbine governor with Salto Caxias system values.	128
Table 17 – Bus data specifications for the Salto Caxias system PST model.	129
Table 18 – Transmission line and transformer data specifications for the Salto Caxias system PST model.	130

LIST OF FRAMES

Frame 1 – Selected papers.	62
Frame 2 – “s_simu” script command matrix format.	77
Frame 3 – Learning (simulated and labeled) dataset structure.	95

LIST OF ABBREVIATIONS

A/D	Analog to Digital
AC	Alternating Current
ANEEL	Brazilian Electricity Regulatory Agency (Agência Nacional de Energia)
ANN	Artificial Neural Network
ASD	Asymptotical Spectral Distribution
AVR	Automatic Voltage Regulator
CFO	Cuttlefish Optimizer
CNN	Convolutional Neural Networks
COPEL	Paraná State Energy Company (Companhia Paranaense de Energia)
CSV	Comma-Separated Values
DC	Direct Current
DLA	Deep Learning Algorithm
ERAC	Regional Scheme for Load Relief (Esquema Regional de Alívio de Carga)
ESD	Empirical Spectral Distribution
FDE	Frequency Disturbance Events
FN	False Negatives
FP	False Positives
FRACSEC	Fraction of a second
GPS	Global Positioning System
GUI	Graphic User Interface
HIF	High Impedance Fault
HVDC	High Voltage Direct Current
IED	Intelligent Electronic Device
IEEE	Institute of Electrical and Electronics Engineers
IET	Institute of Engineering and Technology
kNN	k-Nearest Neighbors
LDA	Linear Discriminant Analysis
LSTM	Long-Short-Term Memory

MATLAB	Matrix Laboratory
ML	Machine Learning
MLP	Multilayered Perceptron
MSR	Mean Spectral Radius
ONS	National System Operator (Operador Nacional do System)
OPC	Open Platform Communications
PCA	Principal Component Analysis
PDC	Phasor Data Concentrator
PMU	Phasor Measurement Unit
PS	Power System
PSS	Power System Stabilizer
PST	Power System Toolbox
RF	Random Forest
RGB	Red-Green-Blue
RMT	Random Matrix Theory
RNN	Recurrent Neural Network
ROCOF	Rate of Change of Frequency
RTD	Research and Technological Development
SCADA	Supervisory Control and Data Acquisition
SDA	Spectral Distribution Analysis
SEA	Synchrophasor Estimation Algorithms
SGD	Stochastic Gradient Descent
SIN	Brazil National Integrated System (Sistema Integrado Nacional)
SNR	Signal-to-Noise Ratio
SOC	Second-of-century
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
UFPR	Federal University of Paraná (Universidade Federal do Paraná)

UFSC	Federal University of Santa Catarina (Universidade Federal de Santa Catarina)
UTC	Universal Time Coordinated
WACS	Wide Area Control Systems
WAMS	Wide Area Measurement Systems

LIST OF SYMBOLS

u_k	Sum of all weighted inputs at a given neuron
b_k	Bias of a given neuron
w_{kj}	Weights for a given neuron's input
x_j	Input for a given neuron
$\varphi(u_k)$	Activation function of a given neuron
y_k	Output of a given neuron
a	Slope parameter for an exponential curve
$\mathcal{C}$	Class of a classifier
ACC	Accuracy of a ML model for classification
TP	True Positive classifications of a ML model for classification
FP	False Positive classifications of a ML model for classification
TN	True Negative classifications of a ML model for classification
FN	False Negative classifications of a ML model for classification
PRC_i	Precision for a class i of a ML model for classification
REC_i	Recall for a class i of a ML model for classification
F_{1_i}	F_1 Score for a class i of a ML model for classification
F_G	Global F_1 Score of a ML model for classification
T_S	Simulation start time
T_{e_s}	Simulation event start time
T_{ef_1}	Simulation event bus clearance
T_{ef_2}	Simulation far bus clearance
$[T_c]$	Simulation time stamps for additional sampling rate changes
T_f	Simulation end time
b_e	Simulation event bus
b_f	Simulation far bus
Z_0	Simulation zero sequence impedance in pu
Z_n	Simulation negative sequence impedance in pu

ϵ	Simulation event type indicator
τ_s	Simulation pre-event sampling rate
$\tau_{e_s}, \tau_{e_{f_1}}$	Simulation event sampling rates
$[\tau_i]$	Simulation sampling rate increments
$\tau_{e_{f_2}}$	Simulation post-event sampling rate
$[S_1, S_2]$	Generator saturation curve parameters
v_{mag}	Voltage magnitude at PMU
v_{ang}	Voltage angle at PMU
i_{mag}	Current magnitude at PMU
i_{ang}	Current angle at PMU
f	Frequency at PMU
$ROCOF$	Rate of Change of Frequency at PMU

SUMMARY

1	INTRODUCTION	19
1.1	Context.....	19
1.2	Objectives.....	23
1.3	Justification.....	23
1.4	Dissertation Structure.....	25
2	THEORETICAL FOUNDATION	27
2.1	Wide Area Measurement Systems.....	27
2.1.1	Phasor Measurement Units	28
2.1.2	WAMS Architecture	30
2.1.3	Standards, Files, and Data Structures.....	33
2.2	Machine Learning.....	36
2.2.1	Machine Learning Concepts.....	37
2.2.2	Artificial Neural Networks	43
2.2.3	Performance Metrics for Machine Learning-based Classifiers	49
2.3	Power System Dynamics	52
2.3.1	Concepts of Power System Dynamics.....	53
2.3.2	Power System Dynamic Modelling	54
2.4	Chapter Final Considerations	58
3	STATE OF ART	60
3.1.	Bibliographic Portfolio Selection Process.....	60
3.2	Literature Review	63
3.3	Chapter Final Considerations	73
4	MATERIALS AND METHODS	76
4.1	Materials.....	76

4.1.1	Power System Toolbox.....	76
4.1.2	Salto Caxias System Documentation	79
4.1.3	Real Event Datasets.....	79
4.1.4	MATLAB Toolbox for Machine Learning and Statistics.....	83
4.2	Methods.....	84
4.2.1	Power System Modelling.....	86
4.2.2	Dynamic Simulation Definition.....	89
4.2.3	Batch Event Simulation	91
4.2.4	Data Processing	93
4.2.5	Neural Network Model Determination and Evaluation	97
4.2.6	Neural Network Model Application	100
5	RESULTS	102
5.1	Simulated Data-based Learning Results	102
5.2	Real Event Data Application Results	107
5.3	Discussion	111
6	CONCLUSIONS AND FUTURE WORKS	115
	REFERENCES	118
	ANNEX A – SALTO CAXIAS SYSTEM SPECIFICATIONS.....	124

1 INTRODUCTION

1.1 Context

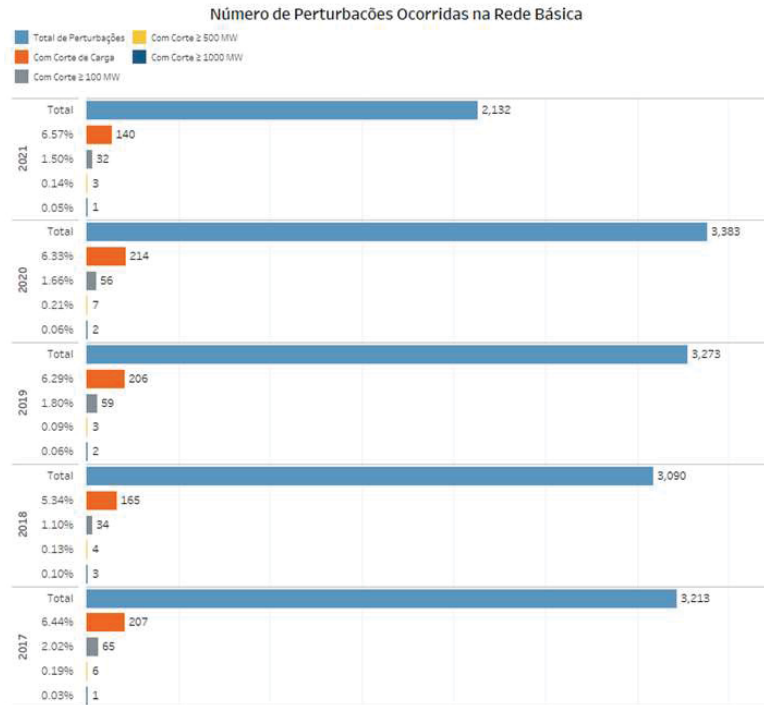
Electrical energy becomes progressively indispensable to humanity's development and survival. With an energetic demand that undergoes substantial increases in all sectors of society, electricity as a service starts demanding more requirements to be fulfilled. Therefore, its reliable, stable, and sustainable provision become paramount.

The faults observed worldwide in power grids during the 2000s evidenced this necessity, as perceives Atputharajah (2009) upon observing North America's blackout history and listed root causes, as well as Europe's and Asia's. The cascading nature of part of these events, many of which have origin in transmission line overloading, also may cause angular deviations on the generators and inadequate impedance matches between interconnected systems, mainly because of the synchronizing power lag. In addition, in case of transmission line outages, the power must be assumed by other lines – which, sometimes, may become overloaded and end up disconnected themselves. Thus, the aforementioned cascading effects can be felt throughout the whole system.

In the Brazilian case, blackouts have been registered in the South-Central region of the country and, more recently, Amapá state has suffered through an energetic crisis that stemmed from a failure in a substation transformer.

Moreover, not only blackouts cause undesired situations from society's perspective: dynamic events of lesser severity also can affect consumers and thousands of them occur yearly, as illustrated in Figure 1 (GARCIA; GIANI; BALDICK, 2015).

Figure 1 – Number of disturbances registered in the Brazilian Core Grid in the last five years.



SOURCE: Performance indicator availability system, ONS (2021)

Studying these electric events is in the interests of the power transmission and distribution companies and of the Operator of the National Electricity System (ONS, from Portuguese, *Operador Nacional do Sistema Elétrico*), since the former may be penalized for any deviations regarding operation performance and power quality.

The Grid Procedures, determined by ONS, are divided into modules that characterize all analyses, parameters, and standards that shall be upheld by power transmission and distribution companies during the most diverse stages and situations of electrical network operation. Tasks such as event occurrence and disturbance screening, load management, contingency operation, post-disturbance grid restoration, etc. are all thoroughly described and require decision-making based on studies that demand broad, real-time visibility of the system (ONS, 2021).

Furthermore, the task of operating modern electrical grids in itself has become progressively arduous, as power systems' complexity and interconnectivity increase. An agency such as ONS, for example, must coordinate several power transmission and

distribution companies, all of which have their operation control centers making decisions at all times in an individual fashion, but still in subordination to ONS.

All aforementioned factors transform real-time, synchronized, and geographically dependent monitoring of variables such as voltage, current, and frequency into a necessity of the power system operators (KOSTEREV; TAYLOR, 1999).

Traditionally, Supervisory Control and Data Acquisition (SCADA) system-based monitoring is used for various types of analyses. Despite it being widely used throughout the world, SCADA is incapable of capturing the real-time dynamics of the power systems with high accuracy and signal reconstruction due to its low measurement resolution, unsynchronized nature, and incomplete information concerning some grid behaviors.

As a partial solution to the lack of grand-scale system visibility, academia and the industry have been making efforts in the last two decades to give system operators a more comprehensive view. This perspective, currently, is commonly called Wide Area Measurement and Control Systems (WAMS/WACS) and seeks to enable power system operation at a local level, but with visibility for global impacts of actions taken in a given point of the network (CHAKRABORTTY; KHARGONEKAR, 2013).

This new paradigm facilitates the structuring and operation of a smarter electrical power system. The WAMS/WACS vision is reasoned in the utilization of synchronized phasor measurement units (PMU), which were introduced to the electrical engineering market in the early 2000s and have undergone a series of regulations and improvements ever since. This new measurement technology allows for a high capacity of real-time synchronous geographically independent system monitoring – thus, increasing the feasibility of analyses and applications that heretofore were complex.

As an example, one of the first big changes that came with the PMU-based WAMS/WACS efforts was the implementation of adaptive protection schemes in transmission networks, aiming to aid the power system operators and in system efficiency growth. Several studies have already been carried out on impedance-based, traveling wave-based, and data-driven methods using PMU data (BISWAL, 2016).

Other applications such as state estimation (paired with SCADA), detection and control of oscillations, monitoring and control of voltage stability, load modeling validation,

power system restoration, and post-event analysis have also been made possible with the usage of the WAMS/WACS vision.

As the technology makes use of specialized communication protocols to perform its data transactions, its transmission rates allow for real-time usage of a great number of applications – which is uncharted territory for the academic community and has been underutilized by the technical community (IEEE; IEC, 2018). Furthermore, with sampling rates of 60Hz, the data volume generated by a single PMU in a few minutes of measurement is high enough for all proposed problems to be faced under a big data analytics scope.

Therefore, in light of the previously exposed points, it's inviable for a human being to make effective observations on the dynamic behaviors of a power system, especially when it comes to transients. Moreover, often, the consequences of an electrical event are evidenced after a given time frame since its occurrence.

Thus, data-driven and machine learning (ML) techniques, both taking advantage of the high data volumes generated by a WAMS and having capabilities to provide operation engineers with valuable information regarding the grid's behaviors, become highly attractive and have been experimented with by researchers in problems such as detection and identification of events and faults in transmission systems.

In addition, the ability to host real-time applications in servers inside the phasor data concentrators (PDC) – which centralize all data stemming from all PMUs in a system or subsystem – and in clusters on the cloud has been leveraged and better utilized year after year. In other words, there is a possibility for establishing a new, additional layer of data analyses and predictions prior to the man-machine interfaces present in operation control centers.

All of these efforts from the technical-scientific community part from the idea that the time needed for decision-making at the operation control centers should be shortened, so as to mitigate or, at least, minimize the adverse effects that electrical events can generate throughout the system. Accordingly, being aware of the occurrence of an event in the shortest amount of time possible, with visibility of its nature and locale becomes a key-point of modern real-time power system operation.

1.2 Objectives

This dissertation has the development of machine learning applications designed for event detection and identification with the intent of aiding the operation of wide-area power systems as its main objective.

To do so, however, this work aims to validate a hypothesis: can one train, validate and test a classification machine learning model in simulated datasets and then apply it to real data?

Thus, to achieve this general goal and answer the presented research question, the following specific objectives must be fulfilled:

- Dynamically model the transmission subsystem of Governador José Richa area, commonly called Salto Caxias system, located in COPEL *Geração e Transmissão*'s electrical system to simulate PMU measurements during short circuits and line/load contingencies in all of its buses and lines;
- Elaborate a dataset with all possible dynamic simulations of the Salto Caxias subsystem, with operation state indicator labels for all timestamps;
- Apply techniques such as artificial neural networks using the labeled simulated dataset for training with the intent of detecting events and identifying event types;
- Validate the aforementioned ML models and their iterations against real event data registered in a PMU located within the Salto Caxias system.

1.3 Justification

Currently, the main data acquisition system in electrical grids is SCADA, which generates data of quantities such as bus voltages, active and reactive power, and system topology – hence, enabling system state estimation.

Periodically, it performs scans on all devices that collect this information with a duration that varies accordingly to system size. In normal operational conditions, where the grid is in a steady state, measurement resolution does not affect any sort of analysis.

However, when it comes to operation under dynamical events, said resolution will have a bigger influence on the feasibility of any system state analysis or estimation.

Therefore, given this nature, it was determined that SCADA systems have below-desired performances regarding the identification and characterization of dynamic events on a power system. Thus, the need arises for a new technology that can execute dynamic, geographically independent, and temporally synchronized measurements – that is, for PMUs, synchrophasor measurement devices that have synchronization done through the GPS technology and can do so with high resolution, generating accurate system operation data in form of electrical quantities.

With big investments in PMU and PDC installation and maintenance being made by power transmission and distribution companies worldwide in the last few years, it becomes essential to optimize its uses and ability to provide systemic visibility and situational awareness beyond what's already being offered by tools present in the market such as PhasorPoint, envisioning WAMS-assisted real-time operation.

For that, the methods of data analysis and algorithms that might constitute an application for real-time operation must make good use of the expressive data volumes generated by such a measurement system.

Considering all identified tendencies in the technical-scientific community, as presented in the previous section, the application of ML algorithms and data-driven techniques in datasets formed by PMU-collected data to aid power system operation engineers with assertive real-time information and predictions is in the greater interests of the actors involved in power system operation.

With a higher analysis velocity in the operation control centers, there is an expectation that power systems resiliency increase, since area restoration protocols can be deployed parting from a more assertive diagnostic. Furthermore, indicators can also alert operators on detected tendencies, behaviors, and threshold nonconformities of the electrical network.

Therefore, improvements are reflected in an increase in electrical energy quality: voltage sags and dips can be mitigated and events such as line contingencies, load disconnections/modulations, and short circuits can be more quickly identified.

However, the grid has many devices and stratagems that help it avoid collapses and critical situations in most scenarios. This means that examples of events happening within a given grid are scarce – which, then, poses a challenge to the application of machine learning methods aimed at classification problems, since they commonly require multiple datasets with high example diversity to learn.

Hence, through simulating a vast number of electrical events happening within the Salto Caxias system, using them to train a machine learning model, and subsequently applying it to real event measurements, the present work seeks to solve this example scarcity problem and help enable real-life machine learning-based classification applications.

1.4 Dissertation Structure

The present work is divided into six sections.

Section 1, “Introduction”, briefly illustrates the context under which the dissertation was proposed, along with its objectives to be fulfilled and justifications for execution.

Section 2, “Theoretical Foundation”, provides information about wide area measurement systems, machine learning, and power system dynamics; explaining concepts that are required to achieve full comprehension of the dissertation.

Section 3, “State of Art”, describes a process for procuring a bibliographic portfolio that contains the state of the art in regards to this dissertation’s fields of study and gave a basis to it. The materials composing the portfolio are also summarized.

Section 4, “Materials and Methods”, describes the computational tools alongside COPEL’s Salto Caxias system technical artifacts/documentation and PMU datasets – all of which are used in the processes that simulate events in a model of a real system to generate event-rich datasets to construct machine learning models that can be later applied to real-life datasets.

Section 5, “Results”, presents the machine learning model achieved through the simulated dataset, its training metrics, real-life dataset application metrics, and visual representations of the classifications. These results are also interpreted in this Section.

Section 6, “Conclusions and Future Works”, exhibits the main conclusions obtained through the execution of this work and also discusses possibilities of other works furthering the studies and discoveries of the dissertation.

2 THEORETICAL FOUNDATION

2.1 Wide Area Measurement Systems

Taylor et al. (2005) define Wide Area Measurement Systems (WAMS) as a strategic effort on part of the electrical power companies made to fulfill gaps and gather accurate critical information about the everchanging modern power systems and take educated decisions on any subsequent operational actions.

Such systems consist of advanced measurement technology, information tools, and operational infrastructure that facilitate the comprehension and management of the increasingly complex behaviors and scenarios that power system operation engineers face on a daily basis.

According to McBride et al. (2007), in its initial form, a WAMS could be used as a standalone framework that complements the grid's traditional Supervisory Control and Data Acquisition (SCADA) system, providing focused insight on particular geographical points of a given PS and being specifically designed to enhance the operator's real-time situational awareness.

As of the past decade, however, these infrastructures were further developed, as engineers and researchers across the world optimize their usage, ultimately leading the technology to reach a sufficient level of maturity by incorporating its capabilities into grid control systems – thus, creating Wide Area Control Systems (WACS), which enable the operators to more accurately and rapidly avoid or solve unwanted situations (CHAKRABORTTY; KHARGONEKAR, 2013).

Given the nature of the foundational technologies, the presence of WAMS/WACS in the grid entails large data volumes being generated across multiple geographic points at a high rate. Therefore, as noted by Alhelou (2019), for the applications to evolve at an evened pace with said technologies, the technical and academic communities have turned their attentions toward data-driven algorithms and machine learning models with low computational cost with the intent of taking the next step forward: to apply automation of all sorts by using estimations and predictions based on electrical variables.

2.1.1 Phasor Measurement Units

The bedrock of WAMS is the capability of having high-resolution and synchronized measurements across the grid. In a technological sense, phasor measurement units (PMU) are the enablers for that, given that they can measure the voltage and current phasors and the frequency of whichever bus they're allocated to with resolutions that typically vary from 1 to 120 measurements per second and synchronized through a common source in the grid such as GPS or IEEE 1588 Precision Time Protocol (PHADKE, 2002). Such synchronized phasor measurements are commonly referred to as "synchrophasors" throughout both academic and technical communities.

Furthermore, from the synchrophasors, other electrical variables such as frequency and rate of change of frequency (ROCOF) can also be calculated within the PMU's firmware – all of which are of great interest to the grid operators. However, one of the technology's main known peculiarities is that when the grid's frequency does not oscillate near or around 50/60 Hz, PMUs are not able to accurately reconstruct the waveforms and, consequentially, any other calculated points (PHADKE; THORP, 2008).

Commercial PMUs have been present in the world since the early 1990s, with the initial prototypes being constructed in the latter half of the 1980s in Virginia Tech, but its application only gained traction as a strategic resource for regional and national power grids by the mid-2000s.

It's noteworthy that they differ from each other in many important constructive aspects, varying by builder and intended usage – thus, it is difficult to discuss PMU hardware configurations in a universally applicable manner. Be that as it may, as noted by Phadke and Thorp in their 2008 book, it is possible to capture the essence of its main components in the form of a generic PMU, which is illustrated in Figure 2.

By taking three-phase currents and voltages from the secondary windings of the transformers as analog inputs and using them so that positive-sequence measurements can be carried out, the signals can be converted to voltages within a range of 10 V so that they are matched with the requirements of the analog-to-digital (A/D) converters through shunts or instrument transformers – thus, allowing the synchrophasor data to be processed.

The anti-aliasing filters have their parameters dictated by the sampling rates defined for a given PMU. In most cases, they are analog-type filters presenting a cut-off frequency that's less than half of the sampling frequency – thus, guaranteeing that the Nyquist Criterion is respected.

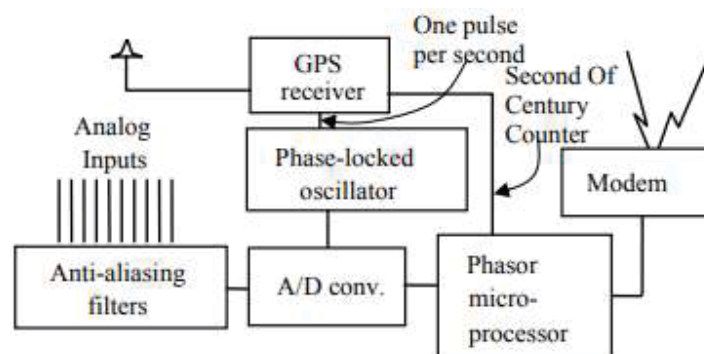
Oversampling with a corresponding high cut-off frequency is also of common practice in the industry, given that, by following it up with a digital decimation filter, one can achieve a structure that has an analog layer concatenated with a digital one. This helps ensure that phase angle differences and relative magnitudes of the different signals remain unchanged in the A/D converter's input layer.

The sampling clock is phase-locked with a GPS clock pulse. As faster A/D converters and microprocessors become commonplace, sampling rates have steadily and accordingly gone up over the years, with more modern models tailored for distribution networks being able to reach 128 samples per grid cycle.

The embedded microprocessor calculates positive-sequence for estimates of all the voltage and current signals, as well as the frequency and ROCOF measured locally, with a generic PMU's output consisting of the four aforementioned electric variables, spread across the phases in the substation which it has been allocated to.

One last output variable is the timestamp, which is created from two of the signals derived from the GPS receiver, identifying the identity of the Universal Time Coordinated (UTC) second.

Figure 2 – Major elements of a modern PMU, assembled in a diagram.



SOURCE: Phadke, A.G. & Thorp, J.S. (2008). Synchronized Phasor Measurements and Their Applications. Springer New York, NY.

2.1.2 WAMS Architecture

The PMUs are installed in the grids' substations, depending on the uses intended for a given set of measurements. Usually, however, they are placed in critical locations concerning the power system's visibility from an operational perspective. This is a topic of great interest to both the academic and technical communities, yielding various papers and studies on how to plan, execute and maintain the optimal placement of the units.

Generally, synchrophasor data is used at locations remote from the PMUs such as an operations control center of a given power company or even of the central operator of a wide area power system. Therefore, an architecture involving PMUs, communication networks, data concentrators, servers, databases, processing algorithms, and visualization dashboards must be in place so as to realize the full benefit of a WAMS/WACS solution.

The most commonly applied architecture pattern relies primarily on phasor data concentrators under which all synchrophasors stemming from all PMUs in a given region/area are centralized and matched by timestamp. There may be more than one layer of PDCs in a WAMS, but, at some point, all of them must converge to a single central PDC which has direct communication to the servers and databases that are accessed by the power system operator's codebase.

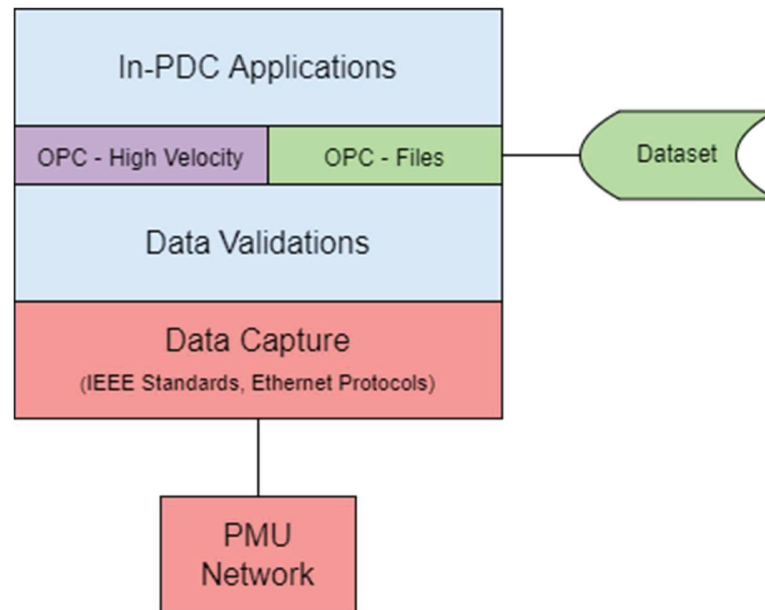
Communication between a PMU and a PDC happens in a serial form, as Martin et al. (2014) note, and the link between a PDC in one hierarchical layer and a PDC in another follows suit. However, communications across PDCs in the same hierarchical layer are also common – and those may not necessarily abide by a serial communication framework. Protocols such as TCP, IPv4 and IPv6 are used in the vast majority of WAMS architectures worldwide.

The usage of process control protocols such as Open Platform Communications (OPC) to handle data consisting of up to 16 entries (the general standard for PMUs and PDCs) is also a common sight across the landscape. With it, real-time monitoring, basic analysis, state flag, and data integrity validations can happen directly at the PDC hierarchy level, with no need of consolidating the data for primary checks. These, however, do not

substitute a full suite of algorithms running on a properly constructed database (KUMAR et al., 2015).

The architecture of a generic PDC can be seen in Figure 3.

Figure 3 – PDC architecture with OPC.



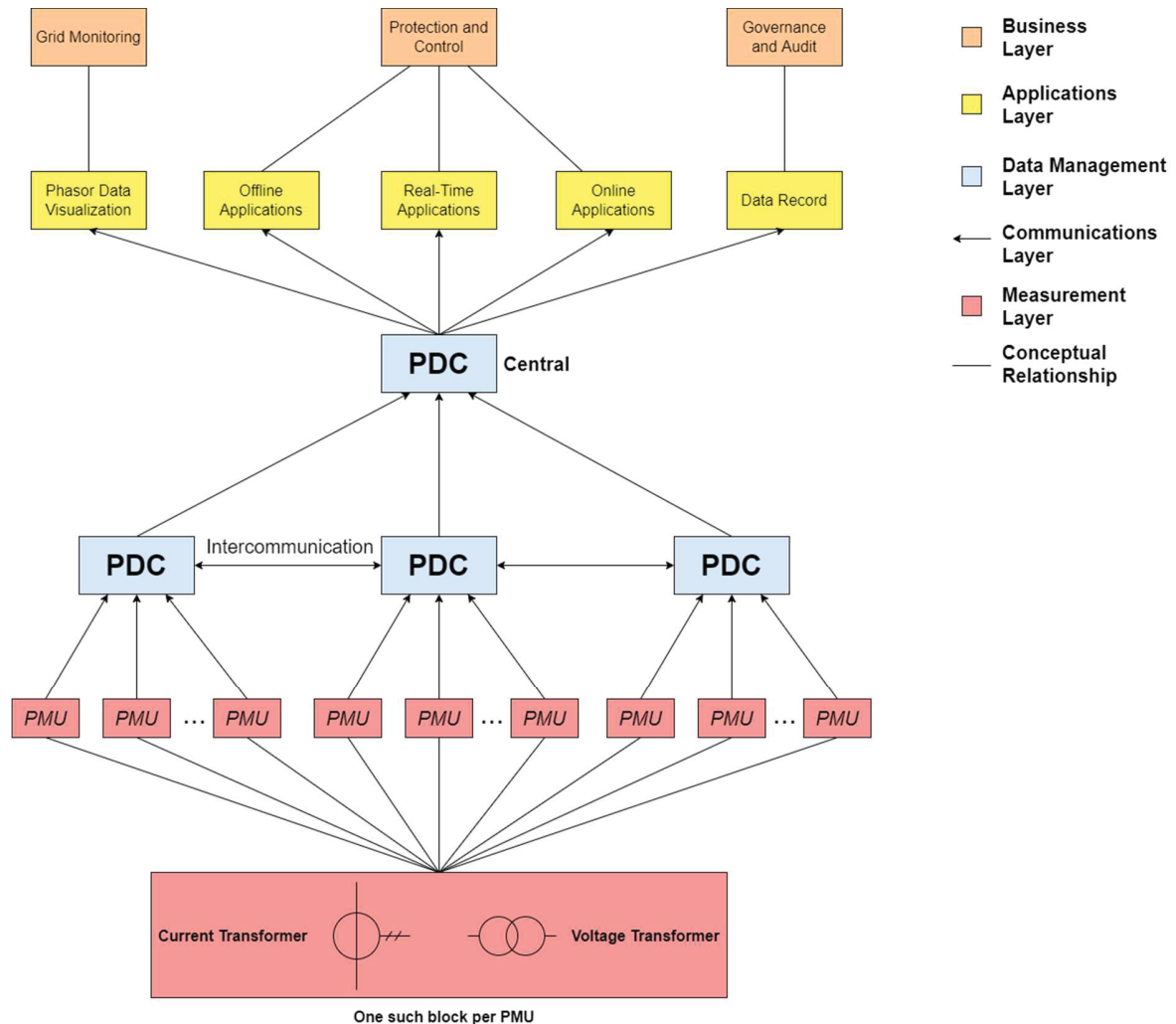
SOURCE: Adapted from Adamiak (2005).

Structurally speaking, a modern WAMS also has layers beyond the central PDC (sometimes called “Super PDC”), which are aimed towards processing all the data captured across the grid, performing analyses in real-time, offline, and online – ultimately, controlling certain aspects of the power system operation – and, sometimes, of the grid itself.

All of the aforementioned components and layers can be rearranged and streamlined into what is the architectural framework that is commonly reproduced across most power systems with WAMS (KHODADADI ARPANAHI et al., 2019) (GORE; KANDE, 2015).

It has five distinct layers, each catering to a specific need of WAMS, be it functional (e.g. end-user visualization dashboard and automated controls) or non-functional (e.g. communications and electrical equipment), as illustrated in Figure 4.

Figure 4 – Example of WAMS 5-layer architecture.



SOURCE: Adapted from Phadke and Thorp (2008), Gore and Kande (2015) and Khodadadi Arpanathi (2019).

The most basic layer of the WAMS architecture is the Measurement Layer, where the synchrophasors are generated. Sometimes, it is called “Component Layer” to include sensors, actuators, and PDCs.

Then, there is the Communications Layer, which is present throughout the WAMS and links other layers between themselves. It refers to all structures (be it hardware, firmware, or software) deployed to make sure that data travels in a secure, lossless, and fast fashion.

The Data Management Layer refers to all efforts and frameworks executed enterprise-wide to custody the data generated and transmitted through the previous

layers. It generally includes PDCs, service-oriented architectures, centralized data models, and enterprise service buses – all of which make the data available for all applications to consume for their particular needs without making changes to the source databases.

There is also the Applications Layer, which is sometimes referred to as “Function Layer”. It is the closest layer to the grid control and operation engineers, as it hosts all applications that make use of the data generated, transmitted, and organized in the previous layers. It includes software for system monitoring and applications that execute specific actions or analyses and may include vendor-owned parts such as user interfaces like Statnett SDK and SmarTS Lab.

This layer, however, is governed by the Business Layer, which reflects the models and policies through which WAMS is furthered and developed and also the capabilities that it caters to, such as system monitoring, control, and protection.

2.1.3 Standards, Files, and Data Structures

With multiple PMU and PDC manufacturers available today in the market, all of this equipment must operate and perform to a common standard, so that interoperability can be achieved among devices stemming from different builders in the same WAMS (PHADKE; THORP, 2008).

As such, the Institute of Electrical and Electronic Engineering (IEEE) first established a standard in 1995 for Synchrophasors, which enforced that all newly-manufactured PMUs needed to be tested for interoperability. Thus, this made manufacturers align on the sources of synchronizing signals, synchronization techniques, and data formats.

However, as WAMS applications became more expansive, it was soon discovered that, when operating in off-nominal frequencies, PMUs from different manufacturers had significantly different performances – which is not acceptable from the point of interoperability of the equipment. Thus, a revision of the 1995 standard was issued in 2005 – IEEE C37.118-2005.

This revision included performance requirements in off-nominal frequencies, in addition to being more specific in regards to the communication protocols that must be used, to the quality/accuracy of the synchrophasor estimations and it also updated the data formats and frames that should be complied with.

In 2011, another review of the synchrophasor technology standard came in two parts, IEEE C37-118.1-2011 and IEEE C37-118.2-2011. The former addressed new definitions and requirements for power system frequency and ROCOF under all operating conditions, as well as mandating stricter accuracy standards for time tagging and synchronization, whereas the latter defined new standards for synchrophasor data transfer, in lieu of the expansion of big data applications and the rapid rate of change of state-of-art communication protocols.

Furthermore, in 2013, two new standards were introduced to the academic and technical communities, regarding guidance for synchronization, calibration, testing, and installation of PMUs and guidance on PDC terminology and operation – in that particular order, standards IEEE C37.242-2013 and IEEE C37.244-2013.

The years 2018 and 2019 saw the last introductions and reviews of any standards regarding PMUs or PDCs.

IEC/IEEE 60255-118-1: 2018 IEC/IEEE 60255-118-1: 2018 IEC/IEEE 60255-118-1: 2018 IEC/IEEE 60255-118-1: 2018 superseded IEEE C37.118-2011 and brought over specifications on the anti-aliasing filters and A/D components of the PMUs when it comes to the synchrophasor estimations, the introduction of performance classes P and M to differentiate minimal requirement compliance based on the intended use and new testing procedures and standards for proofs of performance.

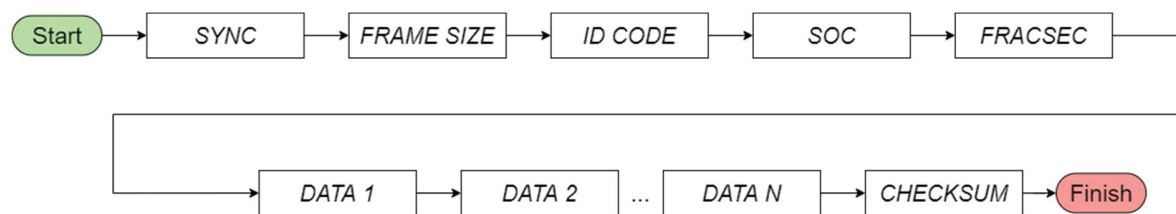
Lastly, standard IEEE C37.247-2019 introduced performance requirements for PDCs, including patterns for data aggregation, processing speed, and accuracy of synchrophasors and other synchronized data, data interfaces with other existing systems, handling of commands, configuration, and other metadata, general performance testing, latency testing, and environmental standards.

In general, the file and data structures around PMU measurements haven't seen considerable changes over the last standard versions. There are four file types for data transmission to and from PMUs, three of which are generated by the aforementioned

devices themselves: Header files, Configuration files, and Data files. The last type, the Command files, is generated upper in the hierarchy, typically at the PDC level.

All files have a common structure, as shown in Figure 5: the first word of 2 bytes is for synchronization of the data transfer, the second word conveys the size of the total record, the third word contains the unique identifier for the data source and the following two words describe the “second of century” (SOC) and the “fraction of a second” (FRACSEC) at which the data is being registered. Then, a sequence of words is provided, specified in the Configuration file. The last word in a file is the check sum to help validate any errors that might have been incurred during the data transmission.

Figure 5 – Format for files transmitted to and from PMUs.



SOURCE: Adapted from Phadke and Thorp (2008).

The Header file is human-readable, containing information that the producer of the file may wish or need to share with its end-user, such as synchrophasor and other synchronized data column names and PMU locations.

The Configuration file provides information about the interpretation of the data contained in the data files, such as the PMU state which indicates how trustworthy were the measurements at each given timestamp.

The Data file contains the actual synchrophasor data (along with other synchronized data), either in polar or rectangular form, divided by phase and, specifically in the current’s case, potentially by transmission line.

Finally, Command files are used basically for controlling PMU performance, passing through command codes to be executed in a timed fashion, usually stemming from the PDC or higher up the WAMS architecture/hierarchy.

2.2 Machine Learning

According to Joshi (2020), Machine Learning (ML) is a field of inquiry that devotes itself to understanding how computer programs can learn to produce behaviors that are not explicitly programmed by the author of the program, being capable to show properties that are unknown and unintelligible to the programmer. It is also considered to be a subfield of a broader subject, Artificial Intelligence.

A more formal definition of the algorithms studied in the field was provided by Tom M. Mitchell in 1997: “A computer program is said to learn from experience E concerning some class of tasks T and performance measure P if its performance as tasks in T , as measured by P , improved with experience E ”.

Initial studies on the subject date as far back as the 1950s, when IBM and Raytheon Company were experimenting and building rudimentary versions of human-driven and knowledge-based reinforcement learning for pattern recognition tasks. Interest in the subject remained throughout the following decades, although the financial feasibility and attractiveness of a highly scalable and reproducible application were ever in question.

With the advances of microelectronics pushing computer processors to be increasingly performant and, subsequently, with the advent of the Internet in the 1990s, researchers and technicians began having sufficient processing power and datasets to further their studies on the matter, switching from knowledge-based to data-driven learning.

Modern-day ML applications have two main objectives: to classify data based on models which have been developed and to make predictions for future outcomes based on these models.

On that note, a more contemporaneous definition for ML is that behaviors can be learned based on three factors: data that is consumed by the program, a set of metrics that can be used to quantify the distance between actual and expected behaviors, and a feedback mechanism that uses said metrics to guide the program towards better behaviors in subsequent iterations.

2.2.1 Machine Learning Concepts

With large datasets being available for all intents and purposes, modern ML applications are commonly designed to handle Big Data. Although there is no specific memory size threshold above which a dataset can be labeled as “big”, the generally accepted definition across the ML community is that when the size of the data is large enough such that it cannot be processed on a single machine, it may be called Big Data. Currently, the estimated threshold is 10 GB (JOSHI, 2020).

The reason for such size-based differentiation lies in the fact that the techniques and methods used for processing big datasets change drastically when compared to non-big ones, where one does not need to worry about the memory location where each bit of data is stored, as the whole dataset can be loaded within a single machine and, thus, accessed without the need for any additional overhead (CHEN et al., 2014).

However, in the realm of Big Data, where a cluster of machines (usually with varying memory and processing specifications) is needed to simply load the datasets, computational resource allocation becomes a problem on its own. For all numeric and string operations, a programmer must split the data across multiple machines, all of which shall perform the task at hand in a parallel fashion. Thus, differently than from its counterpart, Big Data operations are commonly iterative, which, by nature, are computation intensive.

Thus, when handling such datasets for Machine Learning purposes, one must carefully design not only the models but also the physical aspects of the application such as storage and processing power.

That being said, conceptually speaking, ML is applicable with no changes in its theory, methods, and algorithms on account of dataset sizes. As such, another one of its important aspects is the type of learning upon which a program can be built.

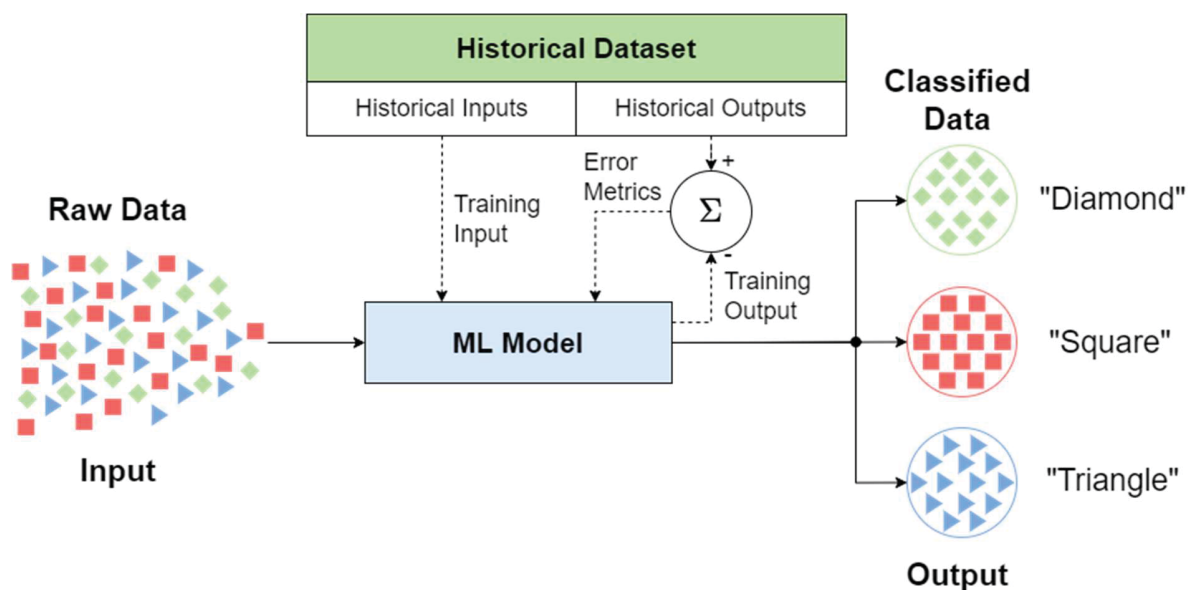
Three broader categories of ML algorithms shall be looked upon in detail within the current subsection – supervised learning, unsupervised learning, and reinforcement learning.

Considering an ML application as a black box that produces a given output for a given input, if the programmer is already in possession of a historical dataset that contains

the set of outputs for a set of inputs, then the learning based on this data is categorized as supervised learning. This is illustrated in Figure 6.

One of the main problems solved by ML algorithms based on this learning type are classification problems such as traffic lights and signs recognition, where a set of images (historical inputs) are provided along with human-made classifications (historical outputs) and those data are used to train a suitable ML model which, after training, can be used to classify any traffic light or sign (between what's provided to it) based on an image.

Figure 6 – Supervised Learning structure diagram.

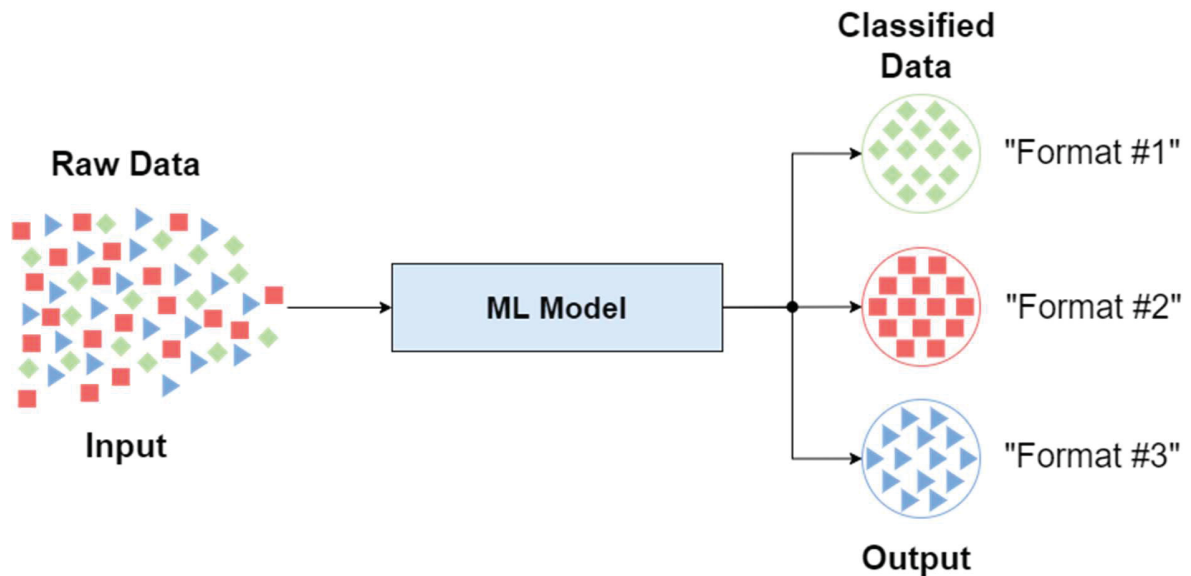


SOURCE: The author (2022).

In the unsupervised learning paradigm, a historical dataset is not present and, thus, the ML models that leverage it cannot tell specifically what a given data sample is – just to which group it belongs to, where the group is also something defined by the model itself, with no meaning defined by it. Additionally, there are instances where the programmer can also input how many output classes are expected. This structure can be seen in Figure 7.

One classic example of unsupervised learning is “clustering”. Considering the same traffic lights and signs example, only the input images would be fed to the ML model, which would then call each cluster “traffic signaling-1”, “traffic signaling-2”, ..., “traffic signaling-n”. New images, for example, would then be categorized by their similarity to any of the existing clusters, which were defined during the learning procedure.

Figure 7 – Unsupervised Learning structure diagram.



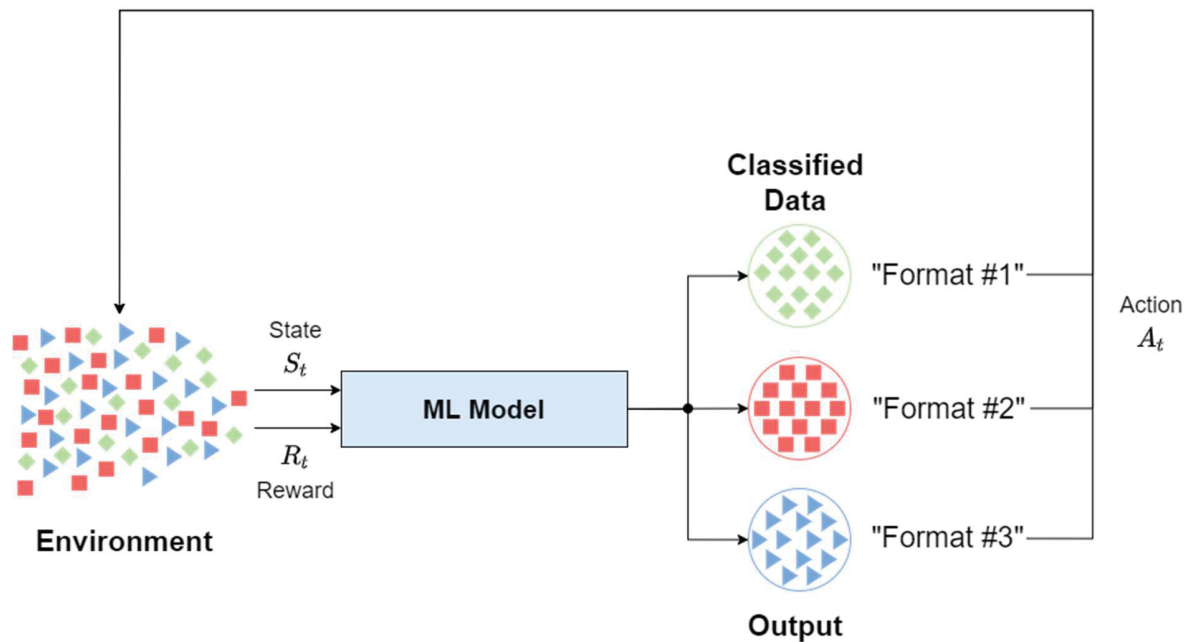
SOURCE: The author (2022).

The last of the three main types, reinforcement learning, is considered to be vastly different than the other two, as it involves feedback from the environment – hence, it's not unsupervised, but it also does not have a historical dataset, preventing it from being considered supervised. In models that leverage this learning type, the application is continuously interacting with the environment in search of producing desired behavior and, as such, is getting feedback on its performance.

Sometimes, reinforcement learning is also described as a feedback system where an initial "state" of the environment is set, along with an initial "reward", which guides the model concerning the quality of its predictions. Then, the model takes an "action" upon the environment – which informs the model of its new "state" and the "reward" it got as a result of the "actions" taken. This particular description can be seen as illustrated in Figure 8.

Considering the traffic lights and signs example, the model would be fed with the images, make its categorizations and predictions, to then receive feedback from the environment (possibly a human), calculate its deviation from the expected behavior, and then try again – repeating the cycles until a satisfactory performance is reached.

Figure 8 – Reinforcement Learning structure diagram.



SOURCE: The author (2022).

There are other ways of categorizing ML methods, such as based on the type of data the models will interact with, but for the purposes of the present dissertation, focus will be given to supervised learning.

According to Joshi (2020), another concept that should be considered when using ML models is dimensionality. As opposed to the common understanding of the idea, which stems from our three-dimensional reality (width, length, and height), most datasets have tens, if not thousands, of dimensions to consider.

Going back to the traffic lights and signs example, when handling images, it is very common to split them down to their indivisible units (pixels), all of which may be a combination of red, green, or blue with a given saturation and brightness and a bidimensional cartesian position – thus, totaling seven dimensions to describe a pixel, which constructs an image.

Dimensions are sometimes called “features” within the ML community and they follow the same principle as described by linear algebra – a dataset’s features must be independent of each other, that is, the dimensions must be orthogonal. As such, a topic of discussion for many papers is dimensionality reduction, which ultimately helps “simplify” the space for a ML method to perform better.

Joshi (2020) also notes that linearity and nonlinearity are also topics of interest for ML, as a model or dataset may be called linear if the relationship between the input and output is linear or composed solely of linear equations. Thus, performing mathematical manipulations to simplify complex models and datasets down to a set of linear equations is commonplace among the academic and technical community.

Linear models are the simplest to understand, build and interpret and, accordingly, all models in the theory of ML can handle linear data. Some examples of purely linear models are support vector machines without nonlinear kernels and linear regressions.

There are occasions, however, when the relationship between the inputs and outputs of a model or dataset is purely nonlinear – that is, it cannot be simplified down to a set of linear equations. Linear approximations are valid, but sometimes do not suffice in terms of performance or computational burden.

Consequently, some models inherently use nonlinear functions to approximate the nonlinear characteristics of the data. Some examples of this model type are neural networks.

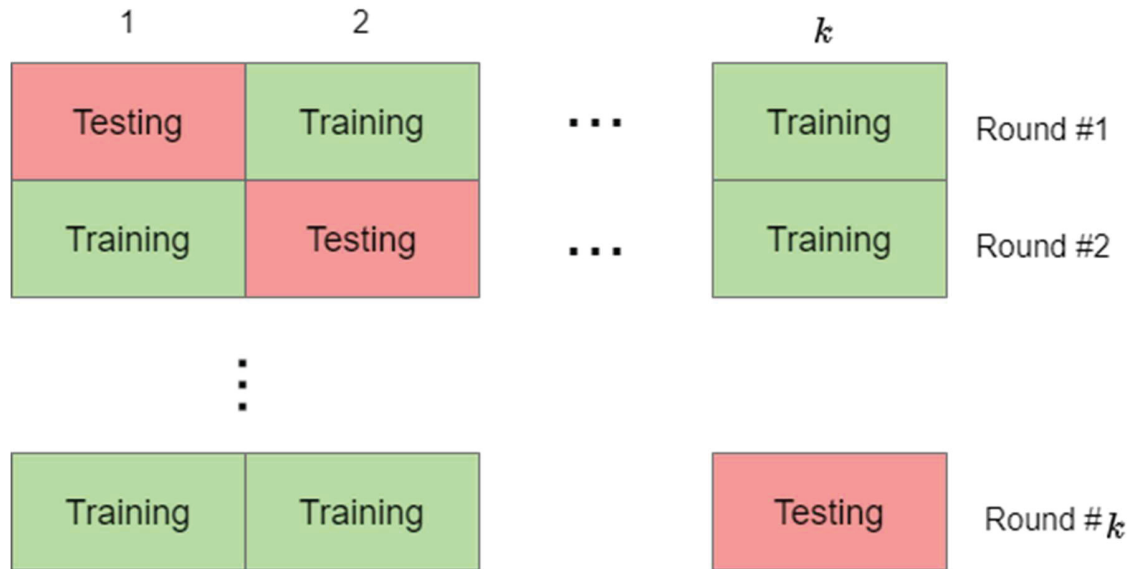
Model robustness is also a topic of great interest in the ML community, given that any classifier or pattern identifier can become overfitted in relation to its training dataset, that the data themselves sometimes create biases in the application and that model parameters have random initial values during training, which may lead the configuration towards a local optimum in terms of model performance (SCHER; TRÜGLER, 2022).

Forms of preventing this include dataset manipulation techniques such as standardization, downsampling, and index randomization in blocks. In order to diminish the model's parameters' random initialization, the training process is executed multiple times, to then choose the instance that presents the best results in terms of performance.

A prevalent way of attesting model robustness is to perform k -fold Cross Validations. This procedure consists of dividing a dataset into k subsets that shall then be used for training in $k - 1$ instances and for testing in the remaining instance. Thus, the ML model is applied to every element of the dataset at least one time.

Figure 9 provides a visual explanation of the aforementioned procedure.

Figure 9 – k -fold Cross Validation, where each block represents a subset of the whole dataset.



SOURCE: The author (2022).

Additionally, some concepts are useful to a programmer when using ML models which come in a qualitative fashion, as guidelines and principles to be followed in order to assure the robustness of the design for a problem's solution.

One of them is the No Free Lunch theorem, which comes from the field of optimization. Essentially, if an algorithm performs better on a certain class of problems, then it has degraded performance in other classes – that is, there is no single optimal solution for all classes of problems, each one must be solutioned for accordingly (WOLPERT; MACREADY, 1997).

There are other concepts widely used when applying ML in specific classes of problems, however, in light of the foregoing, in order to generally solve a problem with ML algorithms, one must take the nature of the data and the solution requirements into consideration in order to determine which approach best suits it. One of the critical starting steps is understanding the data's linearity, to then consider its dimensionality (and perhaps reduce it) and choose the learning method, while considering model performance tradeoffs and computational resources, time, and burden – to finally implement it and collect results.

2.2.2 Artificial Neural Networks

There is a multitude of algorithms that can be categorized as supervised learning, all with varying foundational concepts that allow for further segmentation of this type of ML algorithm into smaller groups.

According to Joshi (2020), most – but not necessarily all – modern supervised learning algorithms can be categorized as linear methods, support vector machines, probabilistic models, decision trees, perceptrons, and neural networks. In particular, the last category has historically been one of the foremost examples of ML, has been used as a solution to various problem types, and, as of late, has seen the most breakthroughs in the form of deep learning – which, on its own, is considered to be a subfield of ML. Furthermore, the category hosts the algorithm that shall be used through the course of this dissertation and, henceforth, focus shall be given to it.

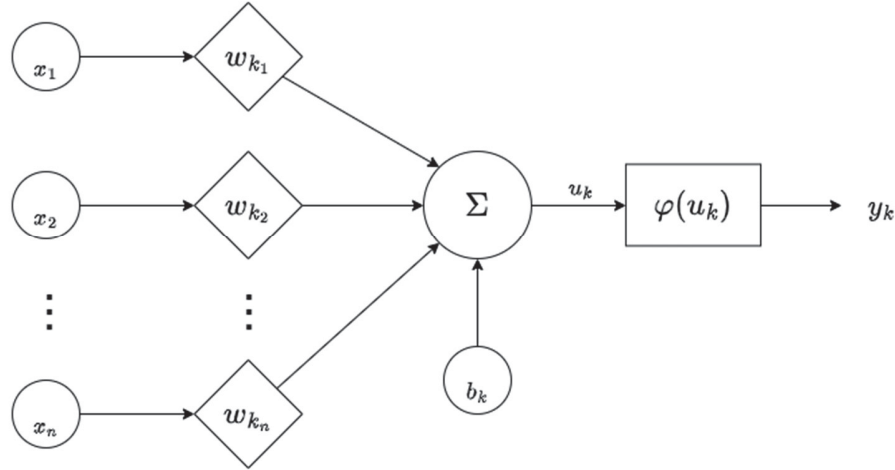
The subject of Artificial Neural Networks (ANN) was opened by McCulloch and Pitts (1943), by creating a computational model based on biological neural networks, inspired by hypotheses around recreating certain behaviors of the human brain such as synapses with the intent to make machines learn and infer. They also provided a formal design for Turing-complete “artificial neurons” – the most basic component of any type of ANN up to contemporaneity.

According to Haykin (2009), a neuron k is an information-processing unit that can accept n inputs (x_n), outputs a single output y_k and contains four basic elements:

- A set of synapses or connecting links, each of which is characterized by a weight of its own, generally denoted as w_{kj} – k referring to the neuron itself and j to the input end of the synapse;
- A bias, commonly denoted by b_k , which has the effect of modulating the net input of the activation function.
- An adder for summing the synapse-weighted input signals and the bias (also can be called a linear combiner), whose output is denoted by u_k ;
- An activation function, commonly denoted by $\varphi(u_k)$, for limiting the amplitude of the output of a neuron. It is usual to define functions that have intervals such as $[0,1]$ or $[-1,1]$.

A visual representation of a neuron can be seen in Figure 10.

Figure 10 – Nonlinear model of a neuron k .



SOURCE: Adapted from Haykin (2009).

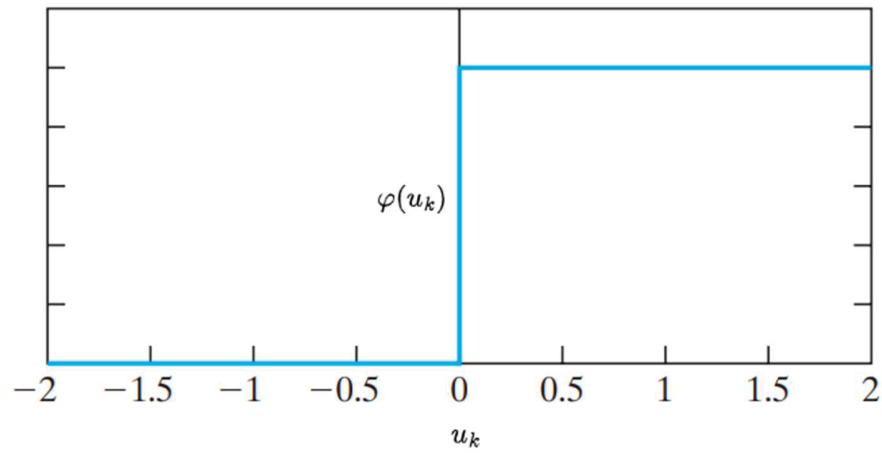
Mathematically, a neuron can be described through Equations 1 and 2.

$$u_k = b_k + \sum_{j=1}^n w_{kj} x_j \quad (1)$$

$$y_k = \varphi(u_k) \quad (2)$$

There are multiple types of activation functions, as they are the elements responsible for making neurons and, consequentially, neural networks considered to be nonlinear models (JOSHI, 2020). In the original models, linear functions were used to perform this task, but nowadays the most common ones are the threshold and sigmoid functions, which can be described by Equations 3 and 4 and depicted in Figures 11 and 12 respectively for a $[0,1]$ interval.

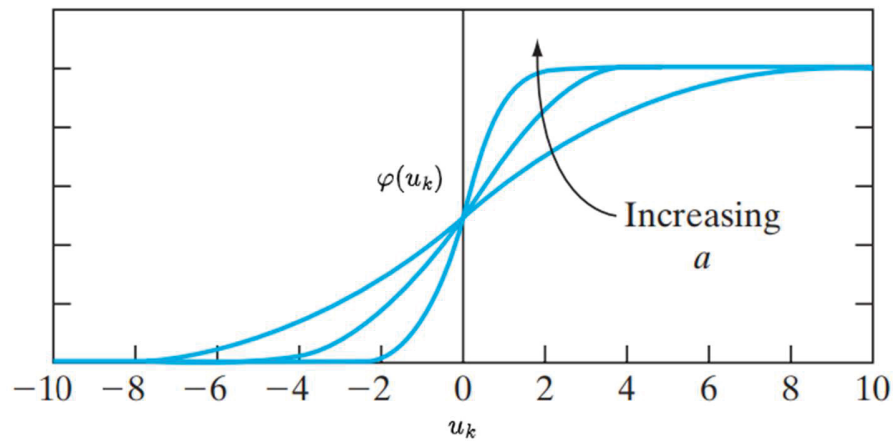
$$\varphi(u_k) = \begin{cases} u_k \geq 0 \Rightarrow 1 \\ u_k < 0 \Rightarrow 0 \end{cases} \quad (3)$$

Figure 11 – Example of threshold function ranging in $[0,1]$.

SOURCE: Adapted from Haykin (2009).

$$\varphi(u_k) = \frac{1}{1 + \exp(-a \cdot u_k)} \quad (4)$$

Where a is a slope parameter.

Figure 12 – Example of sigmoid function ranging in $[0,1]$.

SOURCE: Adapted from Haykin (2009).

It is also usual to have activation functions range within $[-1,1]$. For such cases, the threshold and sigmoid functions can be described by Equations 5 and 6, respectively.

$$\varphi(u_k) = \begin{cases} u_k > 0 \Rightarrow 1 \\ u_k = 0 \Rightarrow 0 \\ u_k < 0 \Rightarrow -1 \end{cases} \quad (5)$$

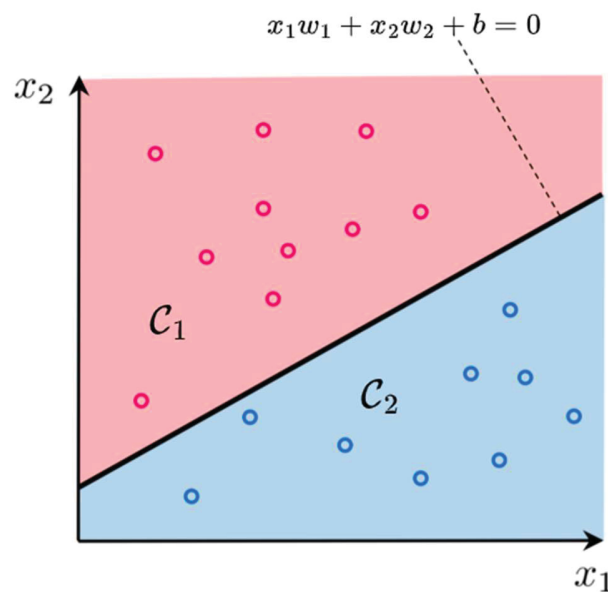
$$\varphi(u_k) = \tanh(u_k) \quad (6)$$

Rosenblatt (1957) introduced the Perceptron as a generalized computation framework for solving linear problems by seeking to design, fabricate and evaluate an electronic brain model. He proposed a pilot model capable of “learning” responses to given inputs, identifying patterns, and classifying stimuli given to it.

It's considered to be the simplest form of a neural network used for the classification of patterns said to be linearly separable and, also, the first practical implementation of it. The Perceptron consists of a single neuron with a hard limiter activation function, adjustable synaptic weights, and a bias summed linearly – from a mathematical perspective, it can be said to be a linear plane.

Its main goal is to correctly classify a set of externally applied stimuli (inputs) into one of two classes, $\mathcal{C}_1$ or $\mathcal{C}_2$. A visual description of such behavior can be seen in Figure 13.

Figure 13 – Example of Perceptron behavior for two inputs x_1 and x_2 .



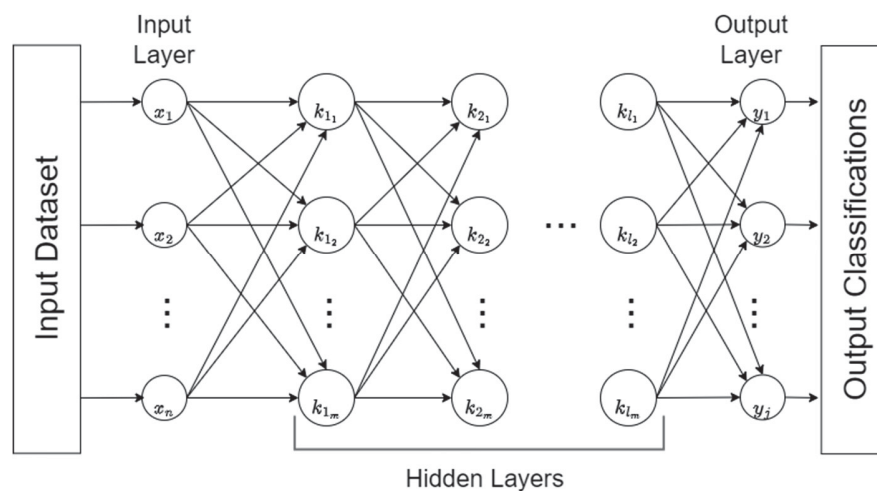
SOURCE: The author (2022).

Soon after Rosenblatt's initial publication, however, fundamental flaws in the theory and technological impediments were discovered, limiting the single-layer Perceptron's scope significantly. As a logical extension of it, a Multilayered Perceptron (MLP) was proposed, where instead of a single layer with a neuron, one could define multiple layers with multiple neurons and full connectedness – that is, each node in a layer interacts with all of the next layer's nodes. This is an important aspect of MLP which was then incorporated into future neural networks, normally referred to as “feedforward operation”: the information that is entered from the input into it propagates through each layer towards the output with no feedback, closely resembling the human brain's operation (HAYKIN, 2009).

However, with the mappings remaining linear, a single-layered Perceptron and an MLP can be proven to be mathematically equivalent. Thus, the proposition also included the introduction of nonlinear functions (such as the ones presented in this section) to be used as activation functions instead of only a scalar product (ROSENBLATT, 1962).

Nowadays, the term MLP is ambiguously used to refer to any feedforward ANN, as it was the first architecture that featured multiple “hidden” layers whose neuron properties are not directly defined by the programmer. Eventually, applications that can classify between more than 2 classes were developed, using this framework. Such an architecture is illustrated in Figure 14.

Figure 14 – ANN architecture with n inputs, l layers with m possible neurons each and j output classes.



SOURCE: Adapted from Haykin (2009).

The number of hidden layers and neurons in each layer are the only characteristics of a classical feedforward ANN and are commonly referred to as its hyperparameters. Each layer in a neural network transforms the dimensionality of the input based on the number of neurons in it, which may lead to improved performance at the cost of computational burden and increased training times. Furthermore, exceedingly large values for the hyperparameters are proven to lead the ANN towards low generalization and overfitting – that is, it may memorize the training data and not learn structures or be capable of inferring upon unseen data (HAYKIN, 2009).

It's noteworthy that there are no theoretical guidelines around how to set the hyperparameters up in an optimal manner for a given application and, therefore, one must perform an open-ended exploration using a grid of possible values for them and then choose the best possible results within the constraints of the training resources.

A common practice in the academic and technical communities is to use optimization algorithms to ascertain the best possible values for the hyperparameters, using them as variables and, thus, setting the search space boundaries based on the problem's dimensionality. As noted by Castillo et al. (2003), the use of evolutionary algorithms for ANN design has great benefits and is commonplace, given that they have been proved to not tend towards local optima as opposed to other more traditional methods.

Regarding the training process of a neural network, considering the supervised nature of the algorithm's learning structure, the synaptic weights are learned from the historical dataset's labeled data, considering that they are randomly initialized. It also has an iterative nature to it. One can refer back to Figure 6 for a visualization of the structure.

Cömert and Kocamaz (2017) summarize the main modern training algorithms within the neural network subspace, noting that most variations to the standard supervised learning type are grounded in manipulating the data volumes that are inputted into the neural network and in changing the optimization method for synaptic weight updates.

One notable method is stochastic learning (or online learning), where samples are randomly chosen within the dataset, one per training iteration, with no regard for the actual sequence of indexes and then used as input to the network. Based on the output error, the weights are updated through the Stochastic Gradient Descent (SGD) optimization

method, which is beneficial when the expected learning path can contain multiple local minima. Considering that each iteration relates to one sample within a dataset, the learning path for this method is considered to be noisy but of fast convergence (BOTTOU, 2004).

One other notable aspect of neural networks is the Universal Approximation Theorem, which states that any continuous function can be approximated well by a neural network with at least one hidden layer and a finite number of neurons (HORNIK, 1989). However, it has also been stated by LeCun et al. (2015) that the approximation capabilities of a neural network are dependent on the number of available parameters in the model – that is, the addition of more layers may be more effective at modelling more complex functions and, hence, problems.

2.2.3 Performance Metrics for Machine Learning-based Classifiers

Although ML methods and algorithms may differ from one another in regards to how the data gets processed, ultimately, they all serve the same purpose when being used as classifiers and, as such, the same set of metrics apply to all of them.

The metrics that shall be presented can be used to describe behavioral qualities and illustrate the performance of any ML-based classifier from a multitude of perspectives, ultimately helping programmers evaluate their models and decide on which approach to take next to achieve an instance with high generalization capabilities within a problem to be solved (THARWAT, 2021).

For binary classification tasks, the terms “true positives” (TP), “true negatives” (TN), “false positives” (FP), and “false negatives” (FN) compare the results of the classifier under test with trusted external judgments – that is, the labeled data in the historical datasets. In this context, “positive” and “negative” refer to a prediction, while “true” and “false” refer to the prediction’s correctness regarding the external judgments. Based on this set of concepts, a multitude of metrics can be defined.

It’s notable that for a multiclass classifier, by treating each class separately as a binary problem, one can use the same set of metrics as the ones defined for binary classification and, henceforth, all equations and descriptions relating to metrics for ML-

based classifiers shall be discussed in terms of multiclass classifiers, as binary classifiers are a subset of theirs.

Accuracy, as a metric, can be defined as the proportion of correct predictions among the total population. It provides a general overview of performance, albeit it is not complete in the sense that it is susceptible to the balance of classes in a dataset – and, thus, it must be used together with other metrics that are class-specific. Mathematically speaking, it is defined by Equation 7.

$$ACC = \frac{\sum(TP + TN)}{\sum(TP + FP + TN + FN)} \quad (7)$$

Precision, in the current context, can be the estimation of the probability that a positive prediction is true for a given class, out of all positive predictions. It can also be considered the measure of the ability of a classifier to identify all relevant data points within a dataset. Equation 8 defines it for a class i .

$$PRC_i = \frac{\sum TP_i}{\sum(TP_i + FP_i)} \quad (8)$$

Recall, also referred to as “sensitivity”, complements precision in the sense that it can be the estimated probability that out of all positives available in the dataset, a prediction will be true for a given class. It’s also sometimes defined as the ability of a classifier to identify only relevant data points, hence being sometimes called as the “true positive rate”. It can be described by Equation 9.

$$REC_i = \frac{\sum TP_i}{\sum(TP_i + FN_i)} \quad (9)$$

A classifier can be considered to be perfect if both precision and recall are unitary, meaning that all of its classified positives are real positives from the historical dataset. A model that presents high precision and low recall in regards to a class behaves in a way

where it rarely classifies a sample as positive that is not positive (FP), but it may classify the actual negatives as positives (FN) – that is, and vice-versa.

A widely adopted form of visualizing the three aforementioned metrics is through the confusion matrix. Each one of its columns represents the instances in a “target class” (labels in the historical dataset, for example), while each one of its rows represents the instances in an “output class”. Thus, it is a medium through which one can visualize whether a prediction is a true/false positive/negative and calculate accuracy, precision, and recall (STEHRMAN, 1997). An example of a multiclass confusion matrix with c classes is available in Figure 15.

Figure 15 – Confusion Matrix for a multiclass classifier, with accuracy, precision and recall denoted in auxiliary rows and columns.

Classifier Outputs	Class 1	TP_1	$FP_{1,2} = FN_{2,1}$	• • •	$FP_{1,c} = FN_{c,1}$	PRC_1
	Class 2	$FP_{2,1} = FN_{1,2}$	TP_2	• • •	$FP_{2,c} = FN_{c,2}$	PRC_2
		• • •	• • •	• • •	• • •	
	Class c	$FP_{c,1} = FN_{1,c}$	$FP_{c,2} = FN_{2,c}$	• • •	TP_c	PRC_c
		REC_1	REC_2	• • •	REC_c	ACC
		Class 1	Class 2		Class c	
		Learning Labels				

SOURCE: The author (2022).

Another metric that is prevalent in both academia and the industry is the F -measure or, specifically for binary classifiers, F_1 Score. While it is one of the premier indicators of

ML classifier generalization, being able to summarize other metrics, it has no particular meaning on its own.

Originally, it was defined as the harmonic mean of the precision and recall of a binary classifier. According to Grandini et al. (2020), for multiclass classifiers, an F -measure can be achieved through the arithmetic mean of the original F_1 Score. Some studies also use a weighted mean instead of an arithmetic mean, should the dataset be severely unbalanced in relation to a specific class. Nonetheless, it can be said that a model whose F -measure, here noted by F_G , tends towards 1 has high generalization capabilities. Equations 10 and 11 illustrate the arithmetic mean scenario, with c being the total number of classes.

$$F_1 = 2 \cdot \frac{PRC \cdot REC}{PRC + REC} \quad (10)$$

$$F_G = \frac{\sum_{i=1}^c F_{1i}}{c} \quad (11)$$

2.3 Power System Dynamics

According to Sauer and Pai (2006), electrical power systems (PS) across the world vary in size, architecture, infrastructure, and load – thus making each one unique. In contrast, all of them possess basic characteristics which allow for the usage of scientific and generalized methodologies in order to study and develop them.

As such, a PS can be comprehended as a set of three-phasic machinery and circuitry designed with the intent of alternate current operation (although sometimes in high voltage scenarios, direct current transmission is used) with structural divisions in the form of generation, transmission, and distribution.

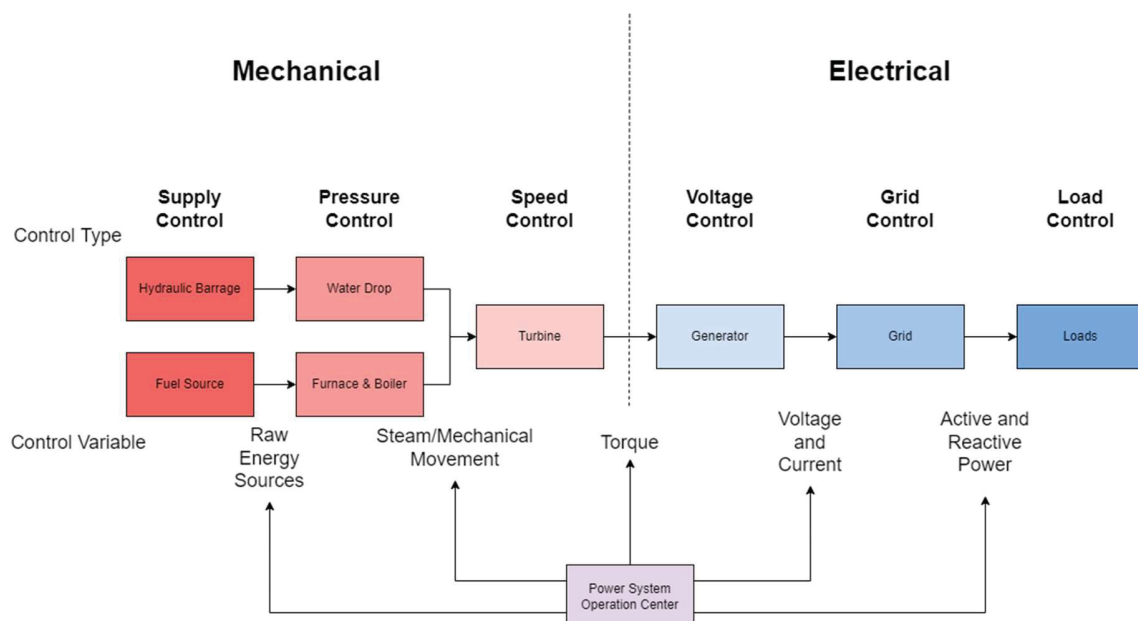
The present dissertation shall provide more detail on the dynamic aspects of the transmission sector, as they are one of its main objects of study.

2.3.1 Concepts of Power System Dynamics

Various studies have been made over the past century on the nature of power systems, some of them concluding that they are not in steady state and that, therefore, models of their components must be detailed in this regard so as to allow for efficient and practical analysis. In other words, it can be said that PSs are dynamic and that such a characteristic originates in its building blocks – with greater expression being given to electrical machines and all the control apparatus designed to control them (SHENKMAN, 2005).

Historically, there has been a major division into the mechanical and electrical components of a PS. This, however, is not absolute because electrical components such as generators and transformers have important mechanical aspects and, on the other hand, mechanical components such as process controls involve electrical properties such as PMU measurements. Thus, for a model to sufficiently describe the dynamics of a power system, it must take both its mechanical and electrical aspects and components into consideration, describing it in some form or another. Figure 16's block diagram summarizes these components.

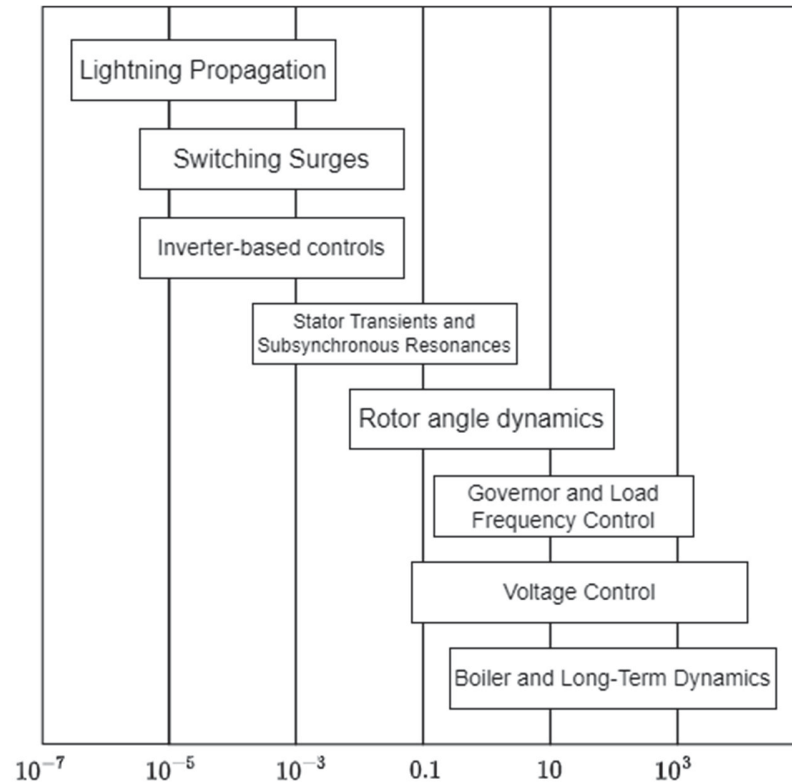
Figure 16 – System dynamic structure divided between mechanical (hydro/thermal) and electrical components, with an operation center controlling all aspects.



SOURCE: Adapted from Sauer and Pai (2006).

One usual form of classifying dynamic phenomena is by their natural time range of response. Different analysis methods and technologies may be best suited to events of distinct types, as resolution requirements may change accordingly (HATZIARGYRIOU et al., 2021). A summary of such classifications can be found in Figure 17.

Figure 17 – Time ranges of dynamic phenomena commonly registered in electrical power systems.



SOURCE: Adapted from Hatziaargyriou et al., 2021.

2.3.2 Power System Dynamic Modelling

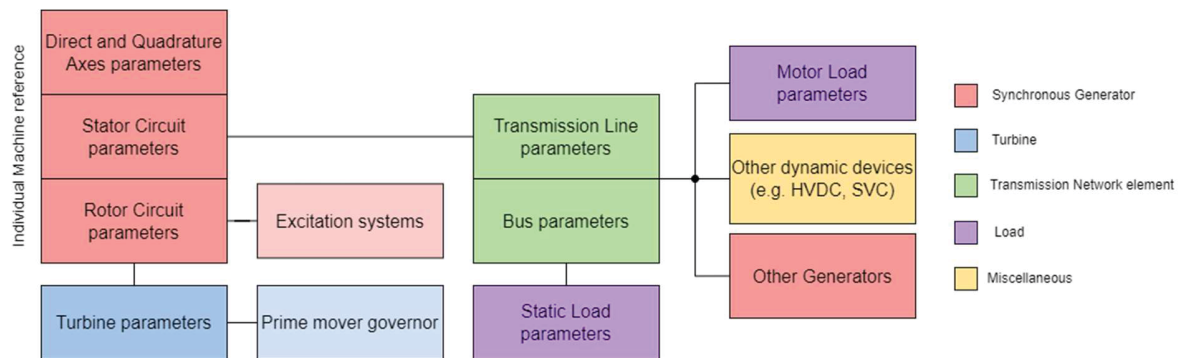
In the context of power transmission, although high-speed transients such as lightning propagation and switching surges are impactful, the main drivers of instability and anomalous operation are stator transients and subsynchronous resonances, all of which intimately relate to constructive and operational aspects of the power system under study.

Hence, in order to understand and study the peculiarities of such events, one must understand the structure of a power system model and the computational efforts taken to

represent its nonlinear dynamic responses to disturbances in the aforementioned components (KUNDUR, 1994).

A model must, thus, include mathematical representations of interconnecting transmission network elements such as lines, buses, and static loads in addition to synchronous generators and turbines and, respectively, their associated excitation and government systems. Induction and synchronous motor loads and devices such as HVDC converters and SVCs also must be considered. Figure 18 provides a visual representation of such a high-level model.

Figure 18 – Structure for a power system model for dynamic transient analysis.



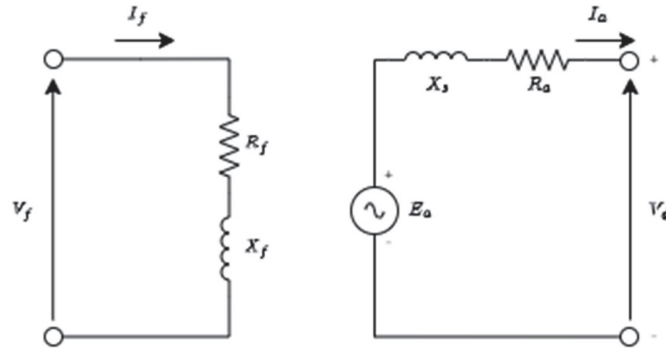
SOURCE: Adapted from Kundur (1994).

Notably, some components need to be mathematically modeled through differential equations to illustrate their electromechanical characteristics and others can be modeled through algebraic equations. This is a fact of importance for the iterative algorithms that are used to approximate the system variables during dynamic simulations.

Synchronous machines need to be represented, for dynamic simulation purposes, by their electromechanical dynamics' differential equations, which aim to model the relationship between machine acceleration and power unbalances, and their equivalent circuits – mathematical representations of subcomponents such as rotor circuits, stator circuits and direct and quadrature axes.

A high-level equivalent circuit for a synchronous generator is depicted in Figure 19, whereas block diagrams for direct and quadrature axes are available in Figures 20 and 21 respectively. Figure 22 also illustrates the electromechanical oscillation differential equation through a block diagram.

Figure 19 – Per-phase synchronous generator equivalent circuit.



SOURCE: Adapted from Fitzgerald, Kingsley and Umans (2003).

Figure 20 – Block diagram for a synchronous generator's direct axis equations.

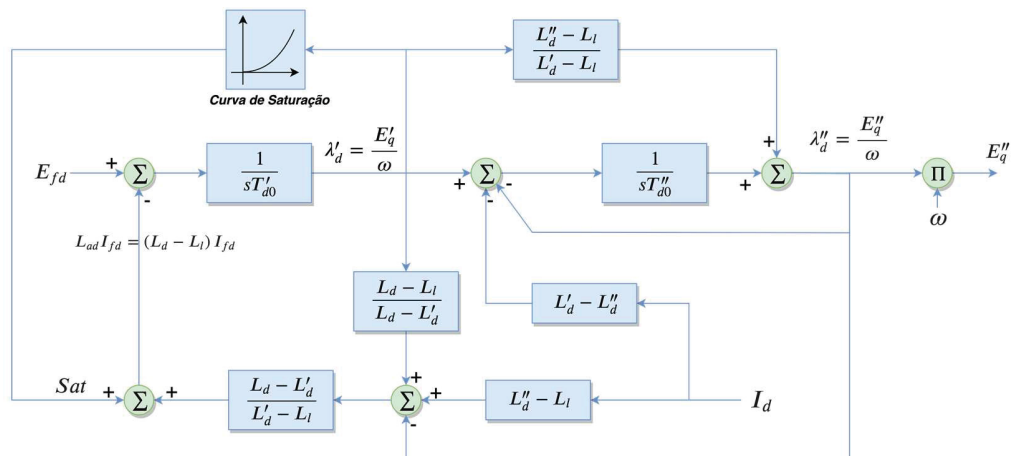
SOURCE: [CEPEL's ANATEM Manual](#), accessed in October 9th, 2022.

Figure 21 – Block diagram for a synchronous generator's quadrature axis equations.

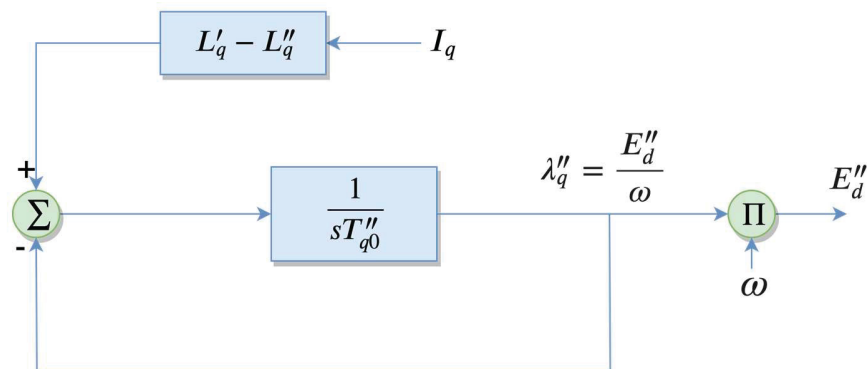
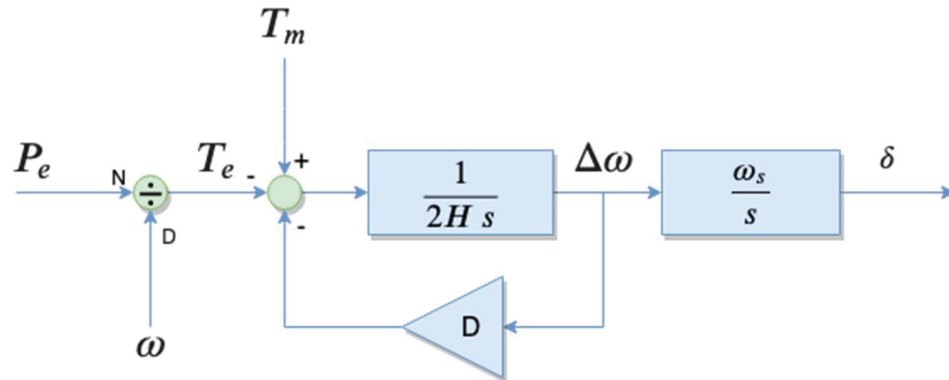
SOURCE: [CEPEL's ANATEM Manual](#), accessed in October 9th, 2022.

Figure 22 – Block diagram for a synchronous generator's electromechanical oscillation differential equations.



SOURCE: [CEPEL's ANATEM Manual](#), accessed in January 3rd, 2023.

It is generally considered that these machines are the main components of dynamism in power systems. Furthermore, their mechanical transients are directly transformed into electrical ones – and all the control apparatus associated with synchronous machines are also designed to take time as a variable.

Excitation systems for synchronous generators also play an important role, as they are responsible for creating and, sometimes, allowing an electromagnetic field to be created or maintained – thus, aiding in the creation of electricity parting from mechanical moment (IEEE, 2016). There are multiple models acknowledged by IEEE, with more modern models having features such as under and overexcitation controls and Power System Stabilizer (PSS) connectivity.

PSSs also have a significant impact on the power system dynamics, as their main use is to enhance the damping of oscillations through excitation control by taking variables such as shaft speed, terminal frequency, and power as inputs (IEEE, 2016). As it occurs with AC exciters, IEEE denotes multiple models for a PSS.

Another set of components with important dynamic behaviors are turbines and their governors. Although their nature tends towards the mechanical specter of Figure 16, they are directly connected to the generators – any variation in a power plant's electricity output is tied to an increase or decrease in turbine rotation speed through an increase or decrease in steam pressure or water flow. One can refer to Chow and Sanchez-Gasca's (2020) publication for more detail on the matter.

The dynamic aspects of elements such as transmission lines and static load buses also can be modeled through differential equations and, for the purpose of studying electromechanical modes and behaviors, cannot be represented simply through their steady-state models.

However, for event identification in the context of power system operation they can be generalized to their static models without significant loss of information (KUNDUR, 1994). Usually, an initial set point for voltages, currents, and frequency is provided for these components.

2.4 Chapter Final Considerations

The presence of PMU-based WAMS/WACS in power grids across the world is now a long-established fact. The level of detail on power system dynamics and the augmented ease of issue tracking provided by these structures, along with continually smaller acquisition, implementation, and maintenance costs, turns the technology into a necessary component of every wide area system.

With advances in the technology architecture in this space, recent breakthroughs were made possible from a data volume, diversity, retention, and quality perspective – which, in turn, enable innovations in the WAMS/WACS applications space and drive business and societal benefits for the grid operators and consumers.

Thus, researchers from both academic and technical spaces have been spending their efforts uncovering what can be found or extracted from these unprecedented data volumes generated by PMUs and organized in a standardized, inter-operational form.

The usage and application of data analytics and machine learning methods on the PMU datasets have only recently become feasible on account of faster internet connection speeds, improved big data processing apparatus (both on a hardware and software front), and wider availability of the aforementioned methods.

However, it is noted that specifically when it comes to machine learning, a wide variety of examples must be provided to a model so it can learn and perform whichever activity it is meant to do. On the other hand, WAMS/WACS, its predecessor frameworks

(many of which are still operational) and other electromechanical devices were built, enhanced and acquired with the intent of improving grid reliability.

This means that a PMU-based WAMS is not prone to provide a wide array of examples of the various possible dynamics of the power system it's located at – which, in turn, reduces the feasibility of creating and applying machine learning models to these datasets and data streams.

Therefore, in order to partake in the benefits of big data and machine learning, more and more studies and research must be executed about ways to circumvent or address this lack of example diversity and enable the usage of ML-based classifiers in the day-to-day grid operation activities.

3 STATE OF ART

3.1. Bibliographic Portfolio Selection Process

The process of literature survey regarding the project's themes was performed using a systematic method of bibliographic portfolio construction developed by researchers from the Laboratory of Decision-Making Multicriteria Methodologies (LabMCDA) from UFSC called Knowledge Development Process – Constructivist (ProKnow-C).

This method, described by (ENSSLIN et al., 2012), presents a process divided into stages that aim to return, at its end, a right, adapted, and applied base of papers centered around the proposed themes.

ProKnow-C suggests that the first stage of the process is the definition of keywords related to the subject of a given research. Thus, one must define the main axes of search, which exert the role of guiding it in order to make it more objective. Once established, their crossing is performed in a search.

Consequently, for this work, “problem”, “method” and “technology” axes were defined – where the first axis refers to the problem the study seeks to remedy, the second to the methods utilized in a possible solution, and the third to the technologies involving the former axes.

The addition of a third axis – deviating from what's defined in (ENSSLIN et al., 2012) – occurred on account of the need of retrieving the search-specific papers where the usage of PMUs is noted. Therefore, “Phasor Measurement Unit” is the only keyword used in this third axis. The crossings and their general results can be seen in Table 1.

Afterward, the obtained base of papers is inserted in the Mendeley Reference Manager tool, which updates and corrects the papers' data that was gathered via Publish or Perish. After this stage, the base is filtered through a series of criteria that aim to reduce the number of papers, in order to increase the reading's quality and feasibility. This process can be visualized in Table 2.

Table 1 – Initial search in Google Scholar via Publish or Perish, with “Phasor Measurement Unit” as the third axis’ keyword.

Problem Method		
	Fault Detection	Fault Identification
Machine Learning	583	324
Data-Driven Modelling	409	169
Data Analytics	252	140
Feature Extraction	331	209

SOURCE: The author (2022).

A total of 2417 papers were found in the keyword crossing, thus creating a base of papers to be filtered.

Table 2 – Articles and papers base filtering.

Filter	Number of withdrawn papers
Duplicates	902
Books	138
Patents	108
Age > 10 years AND Cites/Year < 3	211
Conferences/Workshops/Symposiums	488
Source is not among the major ones	480
Title reading	61
Abstract reading	11

SOURCE: The author (2022).

A total of 2399 papers were discarded. The filters presented in Table 2 ensure the relevance of the remaining works in regard to the research themes to be explored during the present dissertation, as well as their pertinency and contemporaneity in academia. Given the similarities between the two keywords in the problem axis, the number of duplicates is high. Furthermore, all materials that were not scientific articles – such as books and patents – were withdrawn from the base.

Concerning the filters that address Conferences/Workshops/Symposiums and sources, their application is made to consider only papers published in journals of organizations/institutions such as IEEE, IET, Elsevier, Springer, Scopus, Science Direct,

MDPI, and Wiley – all of which are major entities in the electrical engineering research community.

In the penultimate filter, papers whose titles raise doubts in the researchers about their relevance to the research area that encompasses their work should be discarded. In turn, the last filter included reading the abstract of 29 articles. This step is necessary in order to guarantee greater proximity to the problems and techniques addressed in the remaining papers in relation to the themes that the dissertation seeks to address.

Thus, 18 papers were selected at the end of ProKnow-C's implementation, as presented in Frame 1.

Frame 1 – Selected papers.

Year	Title	Author
2014	Dimensionality reduction of synchrophasor data for early event detection: Linearized analysis	<i>Xie, L.; Chen, Y.; Kumar, P.R,</i>
2015	Principal Component Analysis of Wide-Area Phasor Measurements for Islanding Detection - A Geometric View	<i>Liu, X. et al.</i>
2017	Rule-Based Data-Driven Analytics for Wide-Area Fault Detection Using Synchrophasor Data	<i>Liang, X.; Wallace, S.A.; Nguyen, D.</i>
2018	Real-Time Event Identification Through Low-Dimensional Subspace Characterization of High-Dimensional Synchrophasor Data	<i>Li, W.; Wang, M.; Chow, J.H.</i>
2019	A Feature Selection Method for High Impedance Fault Detection	<i>Cui et al.</i>
2019	Identifying Overlapping Successive Events Using a Shallow Convolutional Neural Network	<i>Li, W.; Wang, M.</i>
2020	A review of fault diagnosing methods in power transmission systems	<i>Raza, A. et al.</i>
2020	Power grid online surveillance through PMU-embedded convolutional neural networks	<i>Wang, S. et al.</i>
2020	A data-driven approach for fault time determination and fault area location using random matrix theory	<i>Xiong, Y. et al.</i>
2020	Development of a Transmission Network Fault Location Platform Based on Cloud Computing and Synchrophasors	<i>Lin, T.C. et al.</i>
2020	A new intelligent scheme for power system faults detection and classification: A hybrid technique	<i>Balakrishnan, P.; Gopinath S.</i>
2021	A Hierarchical Event Detection Method Based on Spectral Theory of Multidimensional Matrix for Power System	<i>Ma, D. et al.</i>
2021	Real-Time Synchrophasor Data Anomaly Detection and Classification Using Isolation Forest, KMeans, and LoOP	<i>Khaledian, E. et al.</i>
2021	Brown Measure Based Spectral Distribution Analysis for Spatial-Temporal Localization of Cascading Events in Power Grids	<i>Yang, F. et al.</i>
2021	A novel three-dimensional deep learning algorithm for classification of power system faults	<i>Srikanth, P.; Koley, C.</i>
2021	Artificial Neural Network Approach for Fault Detection and Identification in Power Systems with WAMS	<i>Barreto, N.E.M. et al.</i>
2021	Fault Classification Comparison in Power System Using Nearest Neighbor and Decision Tree	<i>Kumar, D.</i>
2022	A real-time hierarchical framework for fault detection, classification, and location in power systems using PMUs data and deep learning	<i>Shadi et al.</i>

FONTE: The author (2022).

The metrics for the selected base of papers indicate its relevance as the state of art with respect to ML algorithm and data-driven technique applications in dynamic electrical event detection and identification in WAMS-permeated systems. A summary is presented in Table 3.

Table 3 – Metrics obtained for the selected papers.

Metric	Value
Papers	18
Cites	510
Publication years	8 (2014 – 2022)
Cites/Year	72,86
Cites/Paper	28,33
Cites/Author	8,54
Authors/Paper	3,33
h-index	10

SOURCE: The author (2022).

In addition to the papers stemming from the bibliographic portfolio survey, correlate articles to the proposed themes, which were of previous knowledge of the author and served as the basis for the dissertation's proposition, were also read.

3.2 Literature Review

The paper of Xie et al. (2014) is considered to be the first step in the direction of understanding and utilizing dimensionality reduction methods in real-time PMU data. In its first part, the authors study the fundamental dimensionality of synchronized phasor measurement data under normal and anomalous operational conditions seeking to extract information about the principal subspaces of PMU data using Principal Component Analysis (PCA). In light of the results obtained in its first half, the article proposes a real-time application for early detection of dynamic events on the network, parting from an algorithm based on the change of the aforementioned subspaces called Robust Online Monitoring and based on the theory of dynamic linear systems. Additionally, the paper proves that the technique requires less processing time than that of algorithms that are constructed based on raw voltage and/or frequency measurement data.

During PCA's application, the authors perceive low dimensionality in PMU data, even though its volumes are high and it has a multivariate nature. The proposed real-time application is validated through simulations using synthetic (extracted from simulation in the PSS/E 23-bus system) and real event data.

Still in the same context of applications of dimensionality reduction algorithms in PMU data, Liu et al. (2015)'s work aims to study real-time detection of islanding conditions in a large system. In the article, it can be seen that such a problem is especially predominant in electrical grids with significant penetrations of renewable distributed generation. The technique bases itself on the application of PCA in frequency datasets extracted from PMU and in the monitoring of the T^2 and Q statistics of the algorithm to define a space of detection for islanding conditions and subsequent load unbalances and generator trippings. To do so, however, historical system data is needed to model all algorithm parameters and control limits for a desired confidence level.

In a discussion section, the authors note that the algorithm's response time is sufficient for the real-scenario application. Furthermore, its scalability was also validated, verifying that the proposed technique can be applied in smaller and/or distribution systems. Nevertheless, it's also perceived that the algorithm's accuracy and precision depend on the existence of a broad historical data ambient. Moreover, it is noted that if the frequency of an islanded section of a grid has similar characteristics in relation to that of the whole system, the algorithm fails in detecting the islanding conditions.

Islanding, load modulation, generator tripping, and normal operation datasets were extracted from the British interconnected power grid and used for the proposed method's validations and tests. The authors also compare their algorithm's performance against conventional techniques based on ROCOF, frequency difference between PMUs, and voltage angle variation analysis – obtaining better accuracy and reliability measures than those of the latter.

The work of Liang et al. (2017) also possesses an approach more focused on exploring the nature of PMU data, further exploring data-driven and rule-based algorithm applications for event detection, identification, and location problems. In it, the authors

propose real-time data analysis through a data-driven approach that is substantiated by event detection rules that do not have any sort of dependency on the existence of the system's historic data or knowledge of its topology. Events such as phase-to-ground, phase-to-phase, and three-phase-to-ground short circuits are studied in this paper.

The developed method begins by capturing real-time data and subsequently determining the bus where the biggest voltage deviation is detected, among the monitored buses. Afterward, a series of base criteria is applied to the data, aiming to ascertain if the registered voltage sags and dips are, respectively, above or below any pre-defined operation thresholds that indicate what sort of event is taking place in the system at a given point in time. In case no thresholds are exceeded, the data is classified as normal operation – otherwise, more rules are applied in order to distinguish between the three event types considered in the paper and which PMUs are geographically closer to the event.

To validate the proposed algorithm, the authors apply it to real event data stemming from the interconnected system of the Canadian west, obtaining accuracy values above 96% for identification and location in a dataset containing PMU measurements of 60 distinct events.

The paper of Li et al. (2018) builds on the aforementioned ones, proposing to address event identification through data dimensionality reduction methods with lesser associated computational costs such as Singular Value Decomposition (SVD). Its main idea is to characterize an event using the low dimensional subspace generated through the singular vectors of a matrix containing blocks of space-temporal PMU data. Such a representation is robust to the system's initial conditions and also describes its dynamics.

In this work, a dictionary of subspaces is established by inserting historical system data in SVD, with each entry corresponding to different event types. Subsequently, a classification algorithm is applied to the data in real-time, using the dictionary entries as its possible classes. The predictions are based on output parameters and the calculation of the angular difference between subspaces obtained in a second SVD application. Additionally, the subspace dictionary can be updated at the end of each detected event.

Numerical experimentations are performed by the authors on both simulated (IEEE 68-bus system) and real event (New England Power System, in the USA) datasets. Events such as line outages, three-phase, and two-phase short circuits, load modulations, and generator trippings were tested, all being classified with an accuracy above 95%.

Considering the high impedance fault (HIF) detection and identification problem and the general insufficient performance identified in real-world applications, Cui et. al (2019) propose a systematic design of feature extraction and ranking in a PMU database, aiming to give PMU data a more intelligible format in regards to machine learning algorithms. The main motivation for the authors in this paper resides in the fact that although there are several HIF models, there are still no verisimilar waveforms for real events and real-world applications. Furthermore, the work also observes tendencies in the academia leading to their article, such as the usage of methodologies that theretofore parted from an investigation in a limited group of features – all of which failed to provide a practical and implementable solution.

In the proposed methodology, for example, the features are extracted accordingly to the magnitude of the event, as well as its first instant of detection and time duration. Other features, specialized for a given system where the method is applied and stemming from offline analyses are also included. Furthermore, there is a feature ranking procedure executed in order to balance the complexity of the problem as a whole and avoid overfitting to a given event type or operational condition. In their tests, the authors notice high reliability and performance for the proposed methodology when the extracted features were used in typical machine learning algorithms and, also, when compared to more traditional approaches. Real event data in well-documented USA systems were used.

The work of Li; Wang (2019) explores overlapping and cascading events in power systems, remarking that the better part of algorithms developed by the technical-scientific community address solely singular and isolated events, not accurately detecting subsequent events that may take place when a system is under the influence of other disturbances. Considering, however, that overlapping and cascading events do not

happen frequently, there is not enough data for any sort of application of conventional machine learning techniques.

Therefore, the authors suggest the usage of convolutional neural networks (CNN). More specifically, they propose a two-layer CNN trained offline, using a set of eigenvalues and singular values extracted as features instead of using time-series PMU data – thus, reducing the number and size of datasets needed for training and augmenting the CNN model's robustness in regards to PMU measurement imprecisions. In real-time, the proposed method predicts events separately, subtracting the impacts of already-detected ones and, afterward, extracting the dominating features of the residual measurements. Finally, CNN is applied to the measurements.

So as to validate their hypotheses and developments, the authors apply them to measurements simulated in the IEEE 68-bus system. Through this, it is identified that using a set of features when training a CNN model instead of a direct time series of PMU data results in more precise and stable classifications. Additionally, the model's robustness in regard to measurement delays and noise is validated.

Seeking to understand the directions taken by the academic community in using PMU data with dynamic applications for real-time operation, Raza et al. (2020) summarize the most usual event diagnostic methods being used and their variations. Noticing all needs that system operators around the world have, the authors perform an investigative effort regarding most recent developments that address dynamic events.

It is perceived that the scientific community historically has been using time series PMU data for the majority of analyses and methods – however, a tendency in using feature extraction engineering has been identified, since they attenuate the necessity that most machine learning algorithms have with respect to historical data volumes. Furthermore, several papers also include the usage of dimensionality reduction methods to circumvent the aforementioned necessities or, even, to leverage the usage of data-driven methods.

The paper also discusses the benefits, impediments, strengths, and weaknesses of machine learning algorithms in the WAMS context. It also compares the most commonly used methods for event detection, identification, and location, establishing a guide for researchers to understand the most relevant applications done up to publication date.

The article of Wang et al. (2020) studies the state transition through which power system pass, caused by internal and external uncertainties, from a perspective guided by the nature of PMU data and by the capabilities presented by embedded Intelligent Electronic Devices (IED) in the measurement units.

The authors' main proposition is the development of a machine learning application that can be executed in an IED so that more accurate real-time monitoring can be leveraged, in comparison to what's currently available in real-world systems with Synchrophasor Estimation Algorithms (SEA). In detail, the framework discussed by the paper has two stages: the first, containing a continuous quadrature wavelet transform; and the second, involving the application of a CNN for event classification based on the features extracted from the scalograms generated in the first stage.

Experiments with a real IED fed with normal operation and computationally simulated events were executed, achieving results that demonstrate that the proposed framework is capable of reaching high accuracy for multiple event types that commonly take place in power grids. Thus, enhanced situational awareness can be attained.

In Xiong et al. (2020), PMU data imprecision and corruption, commonly named "bad data", is studied alongside the dynamic event start instant and location approximation problems. The authors apply Random Matrix Theory (RMT), seeking to address all three issues at the same time.

In this study, measurements retrieved from a PDC are used in order to structure a raw data matrix, from which information regarding its Mean Spectral Radius (MSR) index and correlation can be extracted, with the addition of the calculation of the integrals of both measures. With this information in hand, RMT is applied so that the event's start time and location approximation are determined upon observing the differences in the processed MSR and correlation values. Furthermore, a decision-making support tool is programmed, making use of the outputs of the aforementioned method, aiming to aid system operators.

Validations were performed in a simulated environment using the IEEE 39-bus system and, posteriorly, in real event data captured in a power system located in China's

countryside. All cases fed to the algorithm were correctly identified, both in start time and location approximation.

Investigations around the possibilities, current benefits, and impediments of cloud computing in PMU data storage and processing are discussed by Lin et al. (2020), with a few case studies involving pre-existing fault identification and location data-driven applications developed for the Taiwanese transmission system being migrated from on-premise servers to clusters in the cloud. The algorithms detect at which geographical point of a hybrid transmission line an event has taken place.

An application for mobile devices was also developed by the authors, in which the Google Map API is used to indicate the location of the event in a map through the detection algorithms hosted on the cloud clusters. This application aims to aid engineers and power system resiliency agents in the execution of remediation actions upon event occurrence.

A comparison of algorithm performance when run in on-premise servers and cloud clusters shows that processing times and accuracies are equal between both infrastructures. Additionally, the mobile application was applied to all 345kV and 161kV transmission lines of the Taiwan Power Company. Its acceptance among employees of the said company was ample, according to a survey run before the paper's publication.

In Balakrishnan; Gopinath (2020)'s work, a hybrid technique between Linear Discriminant Analysis (LDA) and Random Forest Algorithm, using Cuttlefish Optimizer (CFO) for the learning process, is proposed to address event detection and identification in transmission lines in distinct stages, through distinct modules.

In the first step, LDA is applied in order to categorize PMU data as either "normal condition" or "anomalous condition" – thus, detecting the event and creating a subset of event samples. Afterward, CFO-based RFA is applied to said subset, seeking to discern between event types within the samples.

As a whole, the technique is tested in simulated single-phase-to-ground, phase-phase, phase-phase-ground, and three-phase-ground events in the IEEE 9-bus system. The authors conclude that approaching event detection and identification separately

results in lesser architectural complexity, increased solution scalability, and, also, in higher computational performances.

The work of Ma et al. (2021) surveys situational awareness problems in power systems with massive measurement data volumes. In order to address it, in the first moment, a grid partitioning algorithm based on graph theory is proposed, seeking to detect events in a de-centralized form to obtain better response times and prove feature independency between different groups.

Furthermore, a hierarchical method of event detection and location is presented, aiming to evaluate voltage variations and identify the closes PMU to the event according to the variations on the spectral distribution of a multidimensional matrix of systemic variables. Additionally, in the partitioned model proposed by the authors, detecting weak events that generate mild disturbances on the grid is also possible, enabling operators to address them in time for quick remediation. This method can also involve other data-driven techniques that have been presented in the present section, such as MSR and SVD.

The method was validated with data simulated in the IEEE 118-bus system, partitioning it into 6 distinct groups of buses and creating events such as load modulation and line contingency. In addition, actions to deal with these events were also programmed. The mean time for normal operational condition recovery (parting from the event's start) was around 5s.

The paper written by Khaledian et al. (2021) aims to approach the problem of anomaly detection in electrical systems, considering bad data existence within the PMU datasets of a given WAMS. The authors address the two themes, both of which are in recurrent discussion within academia, in a three-step approach by proposing a detection framework using two non-supervised machine learning techniques with the addition of a clustering one.

Taking the need for PMU measurement bad data correction together with that of event detection into consideration, the paper presents a hybrid methodology where data is inserted in the Isolation Forest in a time-series format with the aim of categorizing

anomalous and normal measurements, generating scores that indicate that for each sample within a dataset/window. Subsequently, k-Means and LoOP are applied to perform, respectively, sample clustering and association of probabilities of each sample being an outlier. In a post-processing step, a correlation between all PMUs is calculated for each anomalous measurement, in order to identify whether it refers to an event (high correlation) or bad data (low correlation).

The framework is validated using a simulated dataset in the IEEE 14-bus and 68-bus systems. The classifier as a whole obtained precision values of approximately 96% and accuracy values of 98% for 10 randomly generated simulations, indicating good algorithm generalization and performance.

Yang et al. (2021) seek to handle cascaded events since those are the usual root-cause of great blackouts at a transmission system level. Based on a real-time data Spectral Distribution Analysis (SDA), the researchers propose an approach to spatial-temporal cascaded event location using PMU datasets.

The anomaly detection problem is constructed parting from hypothesis tests using the Empirical Spectral Distribution (ESD), through which the assignment of a test statistic happens. This statistic, then, is compared to its theoretical counterpart, in the Asymptotical Spectral Distribution (ASD) that can be attained through the Brown Measure. Thus, the proposed approach is sensitive and capable of simultaneously locating cascaded events throughout space and time registered within PMU data.

The developed structure's robustness to normal operating point fluctuations, oscillations, and bad data is experimentally argued for. Furthermore, it was proven to be applicable in systems of small and large scale by using a tensor product method. Case studies are performed, using simulated datasets from an ACTIVSg500 system and real event data from a Chinese grid.

In the article of Srikanth; Koley (2021), a tridimensional Deep Learning Algorithm (DLA) using a CNN is presented with the intent of addressing event classification in WAMS-present power systems. Such structure allows the authors to overcome usual problems of machine learning algorithms – as well as overfitting stemming from a low

number of CNN layers. DLA's input is a 4D image divided into RGB channels that is obtained by transforming event current measurements in a time-frequency domain.

The convolution filters transform this 4D image into several 3D kernels, which are used for the tridimensional CNN's training process – which performs the classification of all event types present in the original time-series dataset. All data originated through simulations in the IEEE 9-bus system. The algorithm, as a whole, has an accuracy of 94%.

The paper of Barreto et al. (2021) presents a real-time electrical event detection and identification system based on ANN application, discussing its advantages and disadvantages as a classifier algorithm. The authors construct a dataset to be used for training and validation purposes parting from computational simulations performed in the IEEE 39-bus system that considers three-phase-to-ground, phase-phase, two-phase-to-ground, phase-to-ground, and load and line contingencies as its possible events. The proposed algorithm needs a sliding window of a single sample to execute its predictions – thus, once trained, the ANN has no need for historical system operation data, even within the real-time operation scope.

Once the algorithm's model is defined and validated through a cross-validation process, the authors apply it to a dataset of 28800 randomized samples each. An F_1 Score of roughly 0.8 was obtained, indicating sufficient classifier generalization. Since the predictions are taken parting from a single sample, the time-wise performance indicates that each classification takes less than a grid cycle – thus, presenting a possibility for preventive and remediation actions in a real-time operation context.

In Kumar (2021), a comparison between two distinct non-parametrical methods for electrical event classification – namely, kNN and decision tree – is established using simulated PMU samples from the IEEE 9-bus system. The data pass through a dimensional reduction done through PCA so that the dataset as a whole has a bidimensional nature.

An analysis regarding execution times was performed, concluding that decision trees are generally faster than kNN. As for accuracy, kNN has a 4% marginal advantage

in regards to decision tree, which allows for feature interaction – while the former, being a clustering algorithm, does not.

In Shadi et al.'s (2022) paper, a study on frequency disturbance events (FDE) is performed. FDEs encompass several event types such as line outages, generator trippings, and load contingencies. The article seeks to establish a novel hierarchical structure that aims to address event detection, classification, and location of FDE-causing events using PMU datasets and deep learning algorithms.

A recurrent neural network (RNN), alongside a long-short-term memory (LSTM), are modeled using more recent developments in DLA, seeking to distinguish and locate FDEs with significative accuracy values. The authors only use data in the format of ROCOF time series as the input for the hierarchical structure, which applies LSTMs and RNNs in parallel multiple times, with distinct purposes in each loop (detection, classification, and location in that particular order).

The method as a whole was tested in the IEEE 14-bus and 39-bus systems and compared CNN applications for separate problems, obtaining F_1 Scores of approximately 0.8 in all levels of the hierarchy – which still has a lower performance than that of the convolutional algorithms, but with a broader scope, as the hierarchical structure tackles all three event-related problems at once.

3.3 Chapter Final Considerations

The systematic literature survey process not only aids in the foundation of studies but also confers researchers with a clear view of the state-of-art regarding a research theme that one may desire to study in a given scientific work.

As such, after the reading of 18 papers selected in the process described in section 3.1, it is notable that applications involving WAMS for event detection, classification, and/or location in a power system are of great interest to power transmission and distribution companies and can be divided into two major groups: one which bases itself in applying data-driven methods and the other that focuses on machine learning

algorithms. There is, however, a third minor and still novel group that aims to conciliate the two proposals in a single one.

In regards to data-driven methods, applications that use dimensionality reductions to decrease the problem complexity that stems from the great data quantities and diversity generated by a WAMS are notable. Various approaches to uses and interpretations of reduced dimensionalities have been studied and applied in real-world situations in the past few years. Furthermore, the addition of stochastic processes to already existing frameworks has been considered in some recent studies.

When it comes to the application of machine learning techniques in the aforementioned power system problems, deep learning algorithms have been enabling researchers to use resources such as CNNs and RNNs (and other families of algorithms) and to address topics such as overlapping and cascading events. In addition, a concern for more holistic approaches to the three main problems related to electrical events (detection, identification, and location) has been identified in academia. Hierarchical approaches using different algorithms in its stages are also being explored with this same intent.

The interest in methods that reduce/mitigate the effects of bad data (and, in some cases, to restore lost measurements) and a focus – especially in works with power transmission and distribution company financial foundation – on determining event start times as also been identified as a common feature of some of the selected papers.

Furthermore, some articles also investigate and develop methods for electrical event remediation parting from algorithm classifications, with the most recent ones even studying behaviors of area regeneration actions based on said predictions.

One can infer, as well, that it is usual for works on the application of classification methods in electrical event PMU data to perform its trainings, validations, and tests in simulated datasets based on the various IEEE benchmarking systems. In fewer cases, when available, researchers use real event data collected by engineers in power transmission and distribution companies for case studies and algorithm validations.

In light of the above, therefore, the present dissertation aims to contribute with a hypothesis validation on the applicability of machine learning models trained with simulated data stemming from a model of a real non-benchmarking system on PMU data

originating from events that took place in that particular system. This is aimed at solving the problem of example data scarcity which is a widely stated blocker for real-life application of machine learning models in the power system context across the academic community.

4 MATERIALS AND METHODS

4.1 Materials

Given the present dissertation's topics and main body of work, the artifacts and tools described in this subsection have a mixture of computational and physical nature, since the simulations and machine learning methods are run using MATLAB (Matrix Laboratory) but the system used in them is a model of a real one in the Paraná State grid and the real event measurements stem from PMUs placed in said system.

The usage of MATLAB over other platforms with similar capabilities such as the Python language and its stack of open-source libraries for the simulations is rooted in how the latter has few packages aimed at power system simulation – and even fewer that include dynamic simulations –, whereas the former has been historically adopted by power system researchers and engineers alike for its multiple toolboxes that allow for most of the simulations required for studies in this field of knowledge.

As for the machine learning part, Python actually has more assets and options than MATLAB and, for the past 10 years, has been the tool of choice for most industries, as it has libraries that natively perform parallel computing with big datasets such as Pandas, Dask and PySpark. However, for the intents and purposes of this work, MATLAB's native toolboxes present sufficient performance and capabilities – and, thus, considering the structures and processes implemented for the simulations, it was also used for ML applications.

4.1.1 Power System Toolbox

In a period spanning from 1991 to 2008, researchers from the Rensselaer Polytechnic Institute's Electric Power Department developed a MATLAB-based power system dynamics simulation and control design package, envisioning to provide a flexible platform for studies on said subjects. They opted to call it Power System Toolbox (PST)

and, over time, several papers have been published mentioning it as their simulation tool of choice (CHOW; CHEUNG, 1992).

The toolbox contemplates a wide array of simulation types and scenarios, having been reviewed and modernized over the course of 17 years. It has various modules that allow steady-state and dynamic simulations of HVDC-present grids, stability analysis with small signals, steady-state stability, and dynamic events.

Specifically for this dissertation's purposes, the "s_simu" script was chosen to be the basis for the data generation algorithm. It enables dynamic simulation on a user-defined grid of a chosen event type on a chosen bus/transmission line, with chosen sampling rates and simulation times with which pre-event, event, and post-event periods can be defined.

The script requires, at a minimum, data relating to the system buses, transmission lines, transformers, and generators – in addition to a command matrix containing all definable parameters relating to the simulation at the end of the file. The model for this matrix can be visualized in Frame 2.

Other components that can be simulated for further clarity on grid behavior are machine exciters, power system stabilizers (PSS), automatic voltage regulators (AVR), active loads, HVDC links, and turbine governors. It has to be noted, however, that given that PST's last review was in 2008, it cannot simulate equipment designed under posterior IEEE or IEC standards. Thus, model equivalences and approximations may be required in these scenarios.

Frame 2 – "s_simu" script command matrix format.

Simulation Time Periods	Event Bus	Far Bus	Zero Sequence Impedance (pu)	Negative Sequence Impedance (pu)	Event Type	Sampling Rate for the Periods
T_s	0	0	0	0	0	τ_s
T_{es}	b_e	b_d	Z_0	Z_n	ϵ	τ_{es}
T_{ef_1}	0	0	0	0	0	τ_{ef_1}
T_{ef_2}	0	0	0	0	0	τ_{ef_2}
$[T_c]$	0	0	0	0	0	$[\tau_i]$
T_f	0	0	0	0	0	0

SOURCE: Chow, Joe. Rogers, Graham. (2008). Power System Toolbox Manual.

The variables that compose Frame 2 represent the aforementioned definable parameters and, according to PST's Manual, are as follows:

- T_S – Simulation start time;
- T_{e_s} – Event start time;
- T_{ef_1} – Event bus clearance;
- T_{ef_2} – Far bus clearance;
- $[T_c]$ – Time stamps for additional sampling rate changes;
- T_f – Simulation end time;
- b_e – Event bus;
- b_f – Far bus;
- Z_0 – Zero sequence impedance in pu;
- Z_n – Negative sequence impedance in pu;
- ϵ – Event type indicator;
 - 0 – Three-phase short-circuit
 - 1 – Phase-to-Ground short-circuit
 - 2 – Phase-to-Phase short-circuit
 - 3 – Phase-to-Phase-to-Ground short-circuit
 - 4 – Line contingency
 - 5 – Load contingency
 - 6 – Steady-state
- τ_S – Pre-event sampling rate;
- τ_{e_s}, τ_{ef_1} – Event sampling rates;
- $[\tau_i]$ – Sampling rate increments;
- τ_{ef_2} – Post-event sampling rate.

The script's execution relies on the MATLAB vector calculation facility, solving linear and non-linear equations at each time step in an iterative manner to gather the states of each variable. The final output is a bundle of variables, calculated for every bus in the modeled grid, relating to the internal workings of the grid's equipment and their effects on its overall operational state.

4.1.2 Salto Caxias System Documentation

The *Companhia Paranaense de Energia* (COPEL), in a joint venture with *Universidade Federal do Paraná* (UFPR) through a research and technological development project (PD-06491-0531/2019, regulated by ANEEL) has provided the author with material on the hydro power plant Governador José Richa machinery and auxiliary equipment to enable the construction of a model per PST requirements.

In regards to the generator, the nature of the documentation is of original manufacturer datasheets and other reports stemming from recent COPEL-driven studies on their conditions. The exciter model had all of its parameters set as per IEEE Std 421.5-2016.

As for the PSS, a functional description document from the original manufacturer and an Excel sheet containing the parameter values set for the Salto Caxias subsystem were provided.

The turbine governor system was also made available to the author. It contains information on the mechanical specifications of the turbines, the complete control loop, and other data regarding the equipment apparatus used to implement it all such as servo motors, power electronics circuitry, gate specifications, and others.

The *Operador Nacional do Sistema Elétrico* (ONS) also made contributions in this regard, providing an ANAREDE simplified case for power flow dispatch calculations which contains bus, transmission line, shunt components, transformer, and load data.

4.1.3 Real Event Datasets

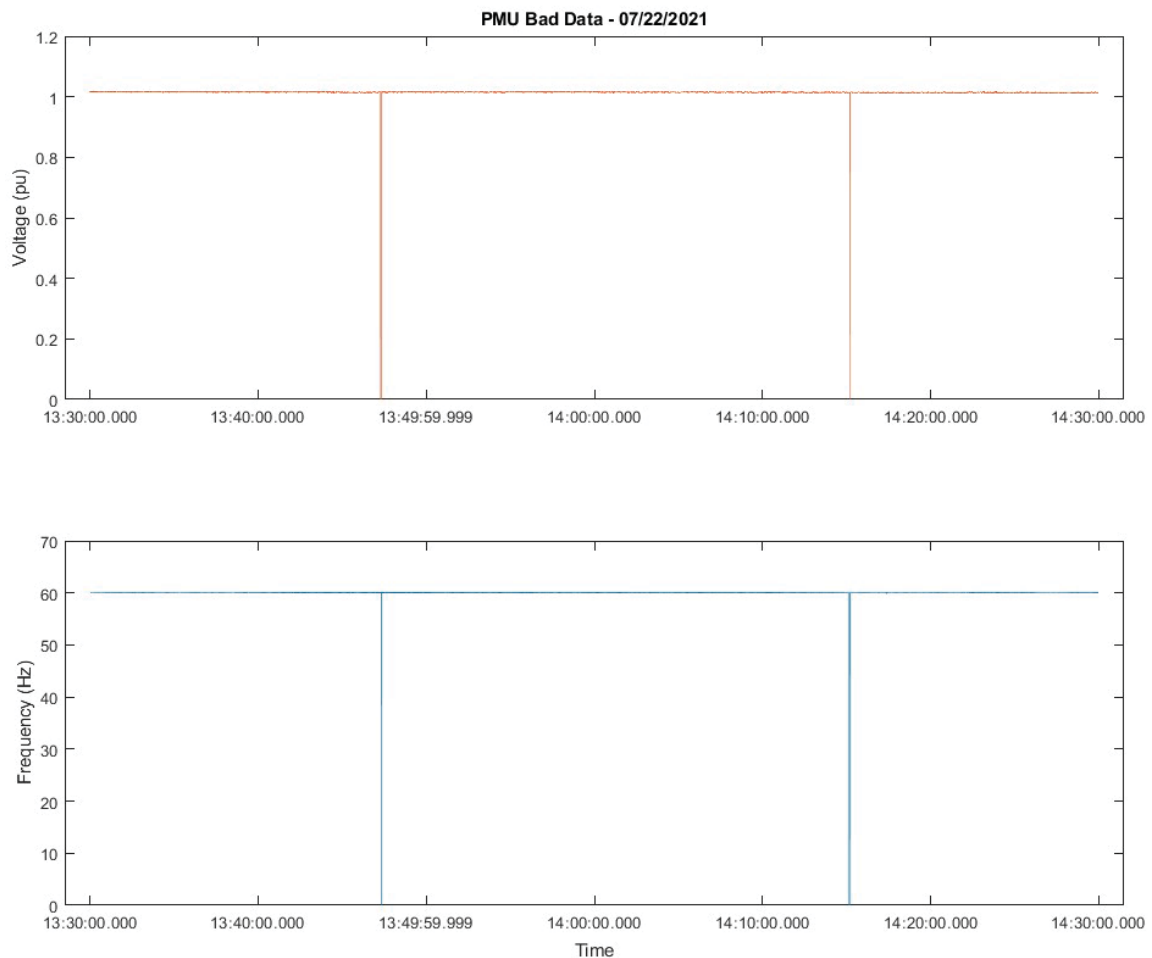
COPEL has provided three distinct datasets, in Comma Separated Values (CSV) format, which refer to events registered at a PMU in the Salto Caxias system – namely, that of *Cascavel Oeste* (Cascavel East) in a substation rated at 230 kV (CVO-230kV), within the Paraná state grid.

It's noteworthy that none of these events occurred within the aforementioned system – rather, they were registered at CVO-230kV with loss of signal power and

distortions stemming from all the transmission lines and circuitry present between the PMU and the events' epicenters.

The first event in consideration for this dissertation is a synchronous malfunction at a national level in Brazil, leading to all PMUs registering two instantaneous spikes on 07/22/2021 between 1:00 PM and 1:30 PM. The voltage (phase A) and frequency graphs are available in Figure 23.

Figure 23 – Voltage and frequency plots for the first event.

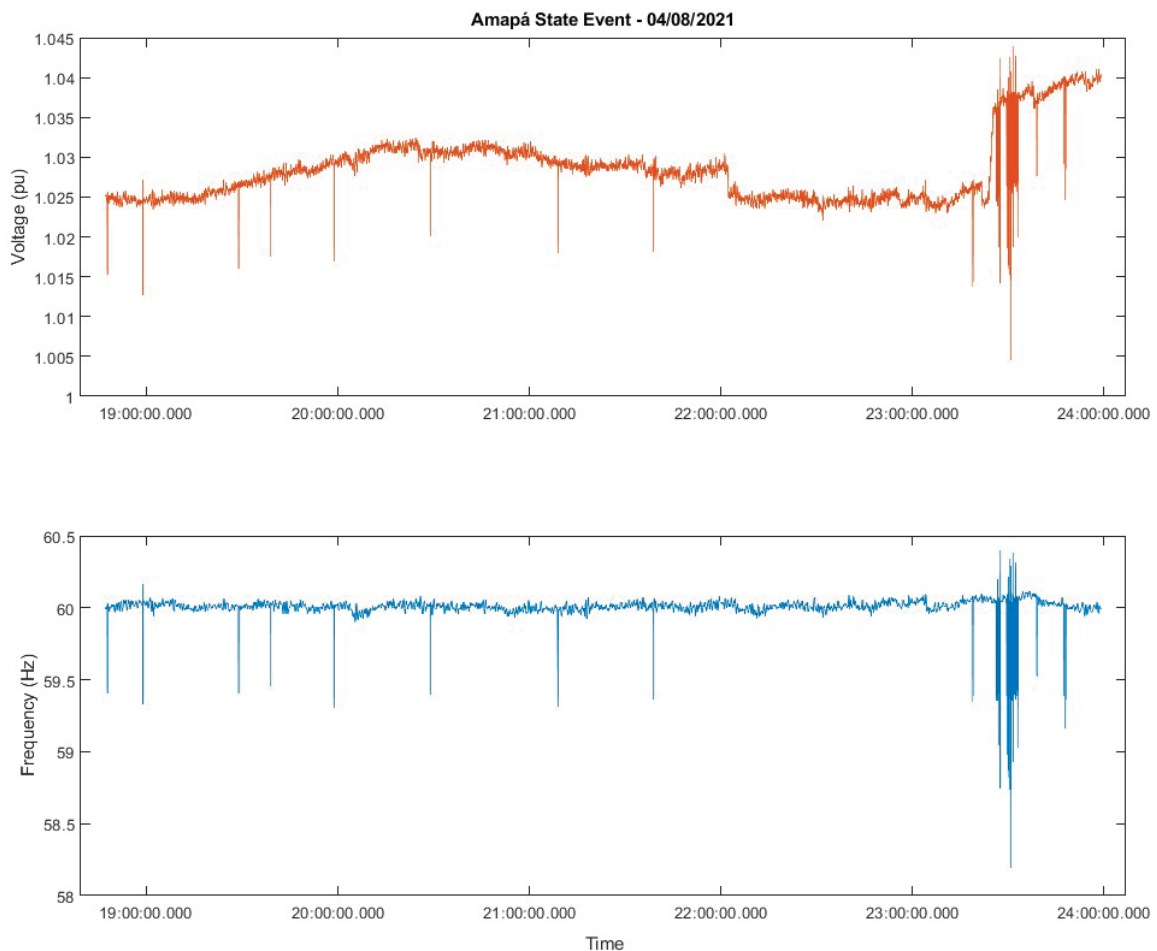


SOURCE: The author (2022)

The second event refers to an automatic shutdown of 230 kV transmission lines Jurupari/Laranjal C1 and C2 and of hydropower plants Ferreira Gomes, Cachoeira Caldeirão, Coaracy Nunes and Santo Antônio do Jari, which totalized a loss of a generation of 800 MW – thus, leading to the complete shutdown of the Amapá state system in 04/08/2021 at 6:34 PM.

As a consequence, a load of 200 MW was interrupted at a distribution level, affecting the totality of Amapá state municipalities. At 7:10 PM of the event's occurrence day, load reestablishment was started and by 9:40 PM the procedure was completed. Other systematic oscillations due to weather conditions were also registered afterward. The voltage (phase A) and frequency graphs are available in Figure 24.

Figure 24 – Voltage and frequency plots for the second event.



SOURCE: The author (2022)

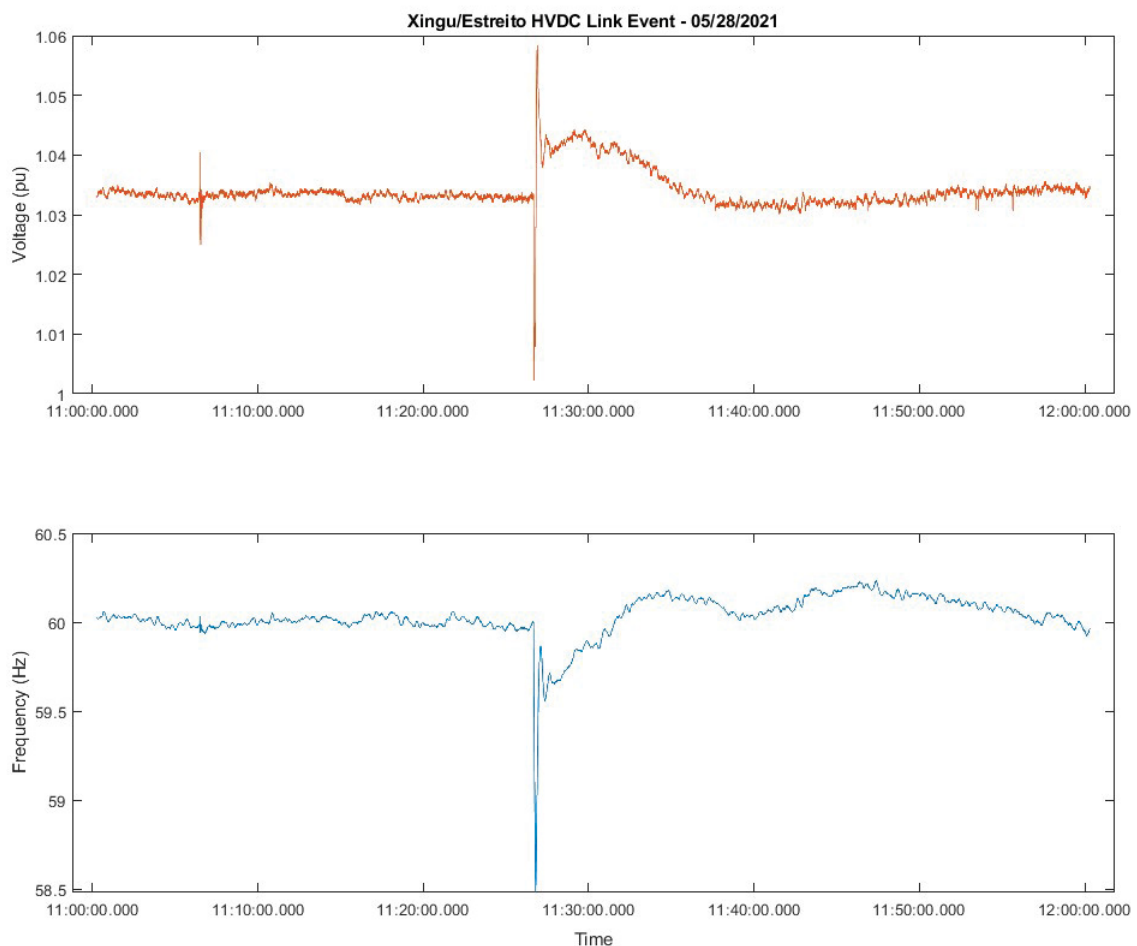
The third and last event provided by COPEL happened on 05/28/2021 between 11:06 AM and 11:45 AM. It took place in the 2076 km long Xingu-Estreito 800kV DC link, which connects Brazil's North and Southeast regions.

Initially, its first pole suffered automatic shutdown and roughly 2 GW were assumed by the remaining poles with no bigger consequences to the National Integrated System (SIN, from Portuguese, *Sistema Integrado Nacional*).

By 11:26 AM, however, pole 2 of the DC link was also shut down, and 7 generation units in the hydropower plant Belo Monte were disconnected from the grid, resulting in a loss of 4.05 GW. Throughout the country, frequencies were noted to fall as down as 58.44 Hz and, thus, ERAC (Load Relief Regional Scheme, from Portuguese, *Esquema Regional de Alívio de Carga*) had its first stage triggered and interrupted 3.4 GW across the SIN.

By 11:31 AM, ONS lifted the load relief procedure and ordered load reestablishment, which ended at 11:45 AM. The voltage (phase A) and frequency graphs relating to this event are available in Figure 25.

Figure 25 – Voltage and frequency plots for the third event.



SOURCE: The author (2022)

It's noteworthy that none of the events had their stamps labeled by COPEL or ONS, a fact that has implications that shall be discussed in the next subsection.

4.1.4 MATLAB Toolbox for Machine Learning and Statistics

MATLAB, out of the box, provides its users with a package that enables training and application of multiple ML and statistics techniques aimed at purposes that may vary from performing descriptive statistics and data visualizations to generating classification models with dimensionality reduction and feature extraction.

For the intents and purposes of the present dissertation, henceforth, the Neural Network for Pattern Recognition app (commonly called `nprtool`) within the toolbox shall be discussed in detail.

The aforementioned app allows one to create, visualize and train two-layer feedforward neural networks to solve data classification problems by simply importing data into MATLAB and referencing it in the app's graphic user interface (GUI), which guides users through the construction of a model.

The first step in doing so is by providing `nprtool` with a dataset containing inputs and classifications relating to each sample in separate variables. Afterward, the percentage split for training, validation, and test subsets is requested (although usually and in this work the split is 70:15:15 in this particular order) so that a model whose hyperparameters are defined by the user can be trained through cross-entropy error and misclassification error evaluation.

As its output, `nprtool` provides users with confusion matrix plots for each one of the three model definition stages and an overall one, along with receiver operation characteristic curves. The app also allows for the generation of MATLAB scripts and functions for result reproduction and training process customization, thus enabling the deployment of the model and its iterations.

It's noteworthy, however, that `nprtool` is of limited nature and that most scientific and technical works that reference it make use of the functions and classes built to support it, so that models with more layers can be included and algorithms for hyperparameter definition can be programmatically applied.

4.2 Methods

As observed in the recent literature which defines the state-of-art on machine learning applications for WAMS/WACS, the number of synchrophasor measurement datasets containing events or faults in a specific power grid is inherently small – especially when taking into account all strategic efforts and investments performed by the system actors so that final consumers are not affected by extreme weather conditions and systemic variations.

In contrast, the need for a large number of examples illustrating all classes one aims to predict is a shared characteristic of the majority of machine learning algorithms. Therefore, the existing data volume and diversity describing the behavior of a given power grid in event situations is, generally speaking, insufficient for designing and training ML models.

Thus, the scientific community has been leveraging computational simulations in order to generate multiple situations where anomalous behaviors are presented in benchmarking models such as IEEE 39-bus (New England Power System) – thus, creating a sufficient quantity of examples with the appropriate diversity for machine learning usage.

The usage of stochastic methods for data preprocessing with the intent of making it more compatible with the characteristics of machine learning models is notable, as a higher computational burden and increased model complexity with the risk of loss of generalization capabilities are required to handle raw data.

In light of the above, the present work aims to leverage the existing, proven, and established processes for the dynamic simulation of power systems and further training of machine learning methods with this generated data for its first stage. The system in contemplation, however, is not an IEEE benchmarking model – rather, it is a model of the Salto Caxias system, which encompasses the Paraná state West region and parts of Mato Grosso do Sul state, constructed by the author through references to COPEL's machine and equipment state reports and studies.

This subsystem of the Brazil national grid possesses one bus that contains a PMU which has registered all three events presented in the previous subsection, with signal attenuation and distortion stemming from the distance to the epicenters of said events.

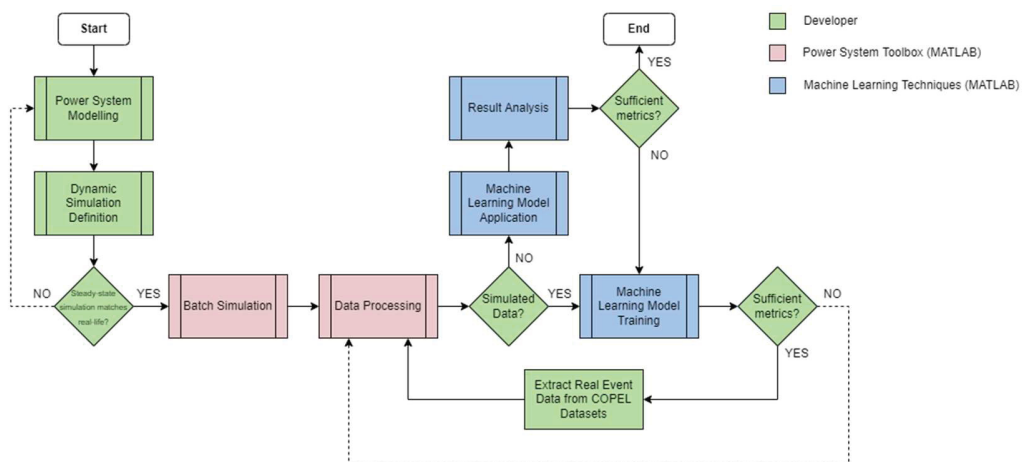
The dissertation's second stage, then, consists of validating the suitability of the ML models trained using the simulated datasets containing measurements of internal events to the Salto Caxias system (together with the related preprocessing methods) for real situations by applying them to the three real event datasets.

It's noteworthy, however, that the natures of the training process events and the real-life validation events are different, given the geographic distances of the event epicenters to the observed PMU. However, the scarcity of real event datasets in the power grids must be highlighted.

Nonetheless, there are multiple substages relating to the definition of simulation parameters and model evaluation that must be executed or prepared by the application's developer. In the same way, MATLAB scripts were created in order to automate simulated data generation and ML model training with the simple provisioning of data formatted in a way described in the following subsubsections.

Figure 26 illustrates the procedure defined for the technical development of the present dissertation, performed to have a reusable, scalable, and reproducible codebase and processes. In the next subsections, each one of the bordered boxes in it shall be discussed at length.

Figure 26 – Block diagram of the general process for data simulation, extraction and preprocessing together with machine learning model creation and application.



SOURCE: The author (2022)

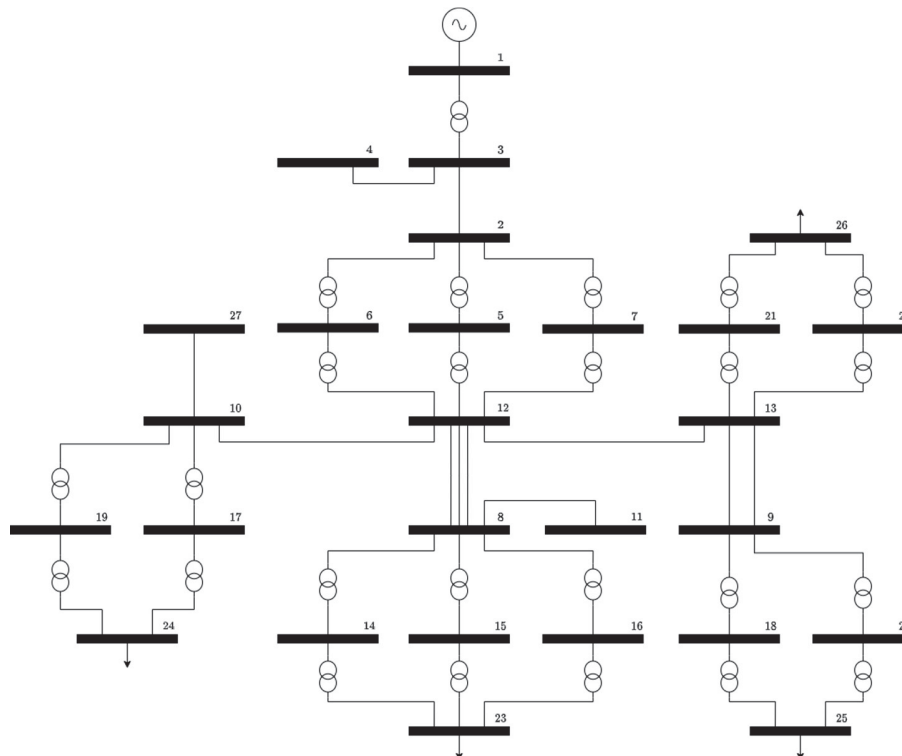
4.2.1 Power System Modelling

The execution of a dynamic system simulation in PST demands one file containing parameters for the machines, equipment, and other constructive components of a given power system – as described in subsubsection 4.1.1.

In regards to the materials provided by COPEL and ONS for the modeling of the Salto Caxias system, given their technical natures and need for detail, the information gathered is more than what is required by PST – therefore, it was possible to fully map the toolbox's matrices to what was available in technical documentation such as case studies on steady-state operation and equipment parameter definition, calculation and test reports.

The buses, lines, shunt components, transmission lines, and transformers were modeled afterward in an ANAREDE file provided by ONS, which contains a simplified 27-bus version of the Salto Caxias system for power flow execution and other steady-state studies. The system topology can be found in Figure 27.

Figure 27 – Single-phase diagram of Salto Caxias system.



FONTE: The author (2022).

This system has multiple PMUs, allocated across its major consumption centers – namely, the municipalities of Cascavel, Foz do Iguaçu, Guaíra, and Medianeira (buses 8, 9, 10, and 13 respectively) – which total an active load of 300 MW and a reactive load of 98.6 MVar that is supplied by one single generator in the hydropower plant *Usina Hidrelétrica Governador José Richa* (bus 1) that has a nominal active power of 345 MW.

However, the only PMU present in the COPEL-provided datasets is that of Cascavel. Thus, the machine learning model that shall be constructed through the methods described in the next subsections needs to take that into consideration – that is, only the aforementioned PMU can be considered/observed henceforth.

This system model also includes 12 transmission lines and 25 transformers, spanning across four different voltage bases – 991 kV, 525 kV, 230 kV, and 138 kV – and comprehending the westernmost parts of the Paraná state and southernmost parts of the Mato Grosso do Sul state in Brazil. It is connected to the eastern Paraná state grid and is, at a macro scale, also part of the Brazilian interconnected system.

All other materials, relating to the generator specifications, PSS models and parameters, exciter models, and turbine governors were provided by COPEL in files that range from spreadsheets with parameter values to architectural diagrams and field engineering reports.

Specifically, however, when it comes to the machine exciters and turbine governors, considering that PST's last release was in 2008, the simulable models do not fully represent the complexity of the real-life equipment.

In the Salto Caxias system, the exciter's type is ST10C which was introduced in Std IEEE 421.5-2016 as a variation on ST5C. The closest PST-available type to it is ST5B, which is considered to be a predecessor of ST5C in design and architecture, given that it lacks some features such as over-excitation and under-excitation limitations.

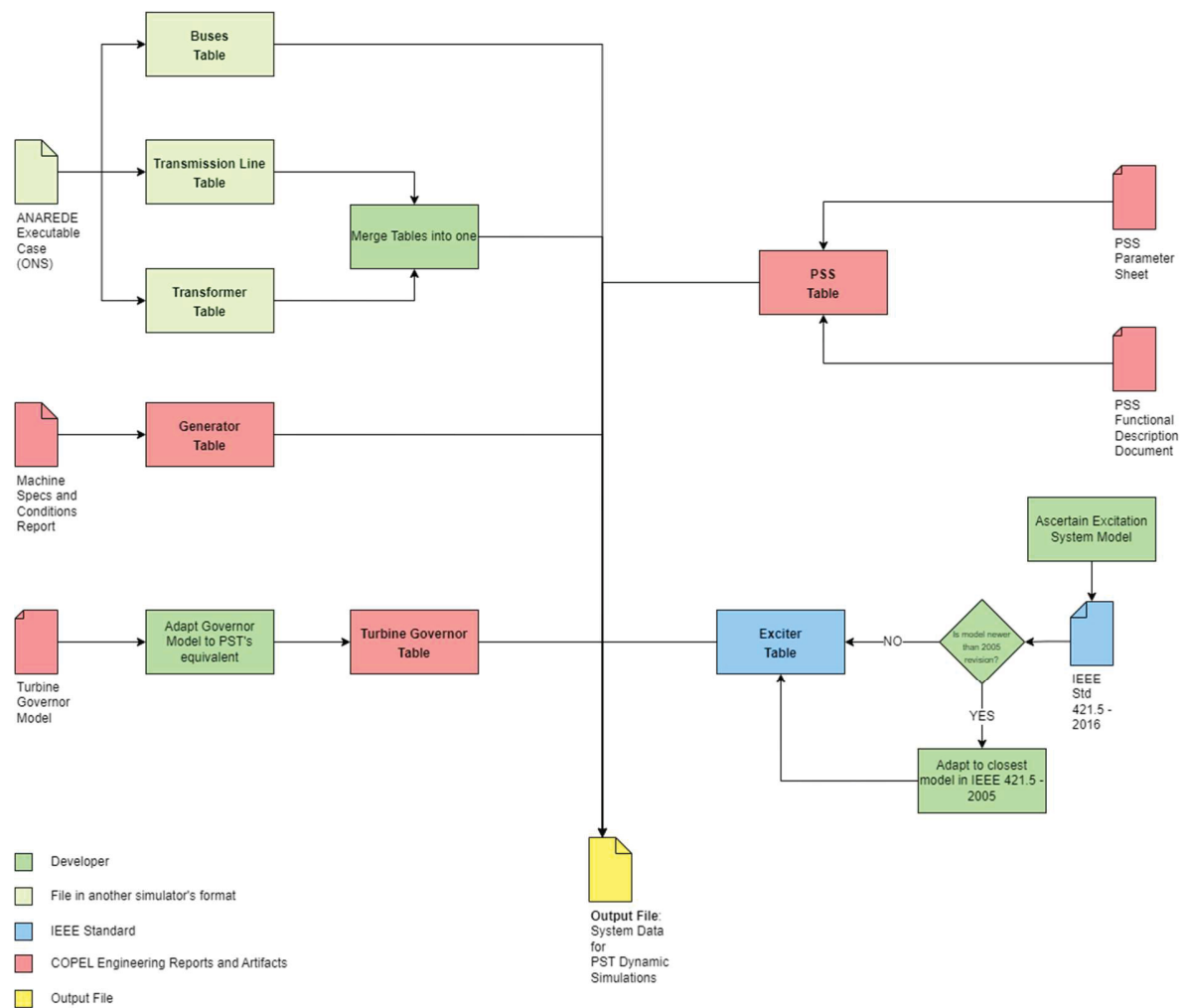
The turbine governor model provided by COPEL has a level of detail that goes down to its components' data sheets and specifications. PST, given its original academic intents, cannot simulate this component at such a degree of complexity. Instead, it models turbine governors using a theoretical structure and, therefore, COPEL's model was

simplified through grouping gains and lead-lag blocks, and omitting unrelated parts to what PST simulates so as to make it conform to the toolbox.

Nonetheless, all information contained in the aforementioned sources provided by COPEL and ONS – adapted or not – were compiled in a single MATLAB script file. All of the matrix structures and variable values for the construction of the Salto Caxias model for dynamic simulation in PST are available in Annex A, together with an explanation of handling multiple formats when it comes to modeling the generator saturation curve parameters.

The data flow and process designed through the construction of the file containing a PST-based version of the Salto Caxias model are available in Figure 28.

Figure 28– Block diagram for the power system modelling process, with a MATLAB file as its output.



SOURCE: The author (2022).

4.2.2 Dynamic Simulation Definition

Through leveraging Power System Toolbox's "s_simu" script for dynamic simulation, one can execute as many simulations as the number of events multiplied by the number of buses/lines/transformers in an automated manner by adding a matrix using Frame 2 as a reference with its variables as generic parameters that are defined at each iteration of a batch process.

As such, several modifications on the base code were performed, since some of the resources used in the toolbox's last release (2008) were incompatible or deprecated as of MATLAB R2022 with new equivalent methods being made available.

Additionally, the original version of the script had a user interface through which one could select what types of graphs would be plotted – which, for batch simulation, needed to be omitted (commented code).

More specifically, for batch simulation purposes, the event simulation control matrix is defined in such a way where b_e , b_f and ϵ are not pre-defined. In contrast, other variables that control simulation times and sampling rates have pre-established values for all iterations of the simulation algorithm. Additionally, no changes in sampling rate can happen – thus, $[T_c]$ and $[\tau_i]$ are omitted.

Table 4 contains the values for each one of the aforementioned pre-established parameters.

Table 4 – Simulation time and sampling rate values for all simulation iterations.

Symbol	Variable	Value	Unit
T_s	Simulation start time	0	seconds
T_{e_s}	Event start time	15	seconds
$T_{e_{f_1}}$	Event bus clearance	16	seconds
$T_{e_{f_2}}$	Far bus clearance	17	seconds
T_f	Simulation end time	45	seconds
$\tau_s = \tau_{e_s} = \tau_{e_{f_1}} = \tau_{e_{f_2}}$	Sampling rates for all event periods	$\frac{1}{60}$	seconds

SOURCE: The author (2022).

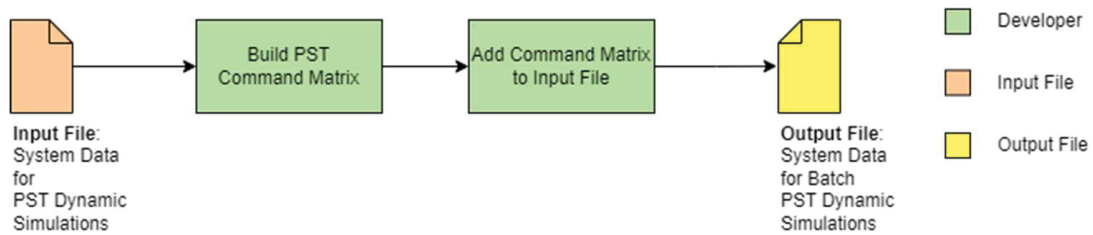
The reason for triggering an event 15 seconds into the simulation iteration is that PST simulates model initialization – thus, in the initial moments, it simulates the generator

curve. Empirically, through observation of 30 iterations, it was noted that after 10 seconds, this behavior is already over and the system is in steady-state. Therefore, at later steps in the overall process, these initial 10 seconds of system initialization are discarded.

Furthermore, considering that the volume of the simulated data must be that of a real PMU's, one of the first parameters that need to be defined is the sampling rate for all event periods. That can be achieved by setting all sampling rates with the same value – that which can be found in a real PMU (in the present work, $\frac{1}{60}$ s, one measurement per grid cycle).

Figure 29 illustrates the complete process for setting up the system model with the simulation control matrix for batch simulation.

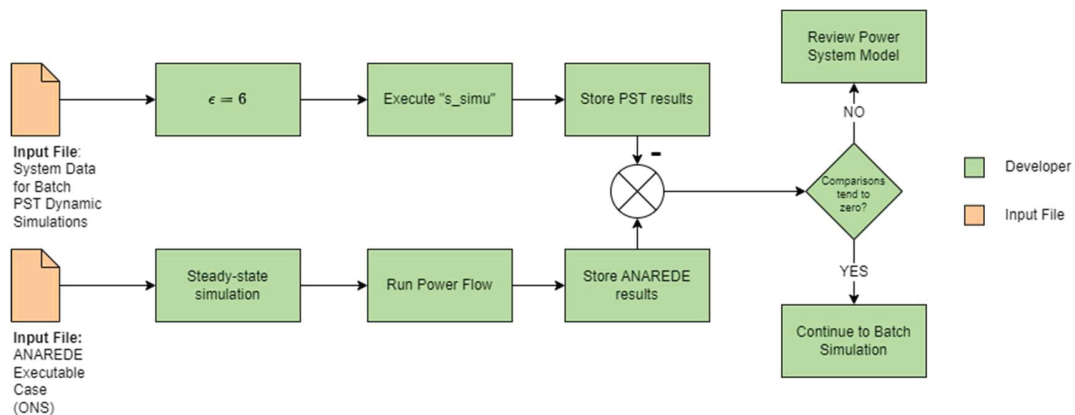
Figure 29 – Block diagram for the dynamic simulation file definition for batch event simulation.



SOURCE: The author (2022).

In order to validate the constructed model, prior to entering the batch simulation step, one can execute a steady-state simulation ($\epsilon = 6$) in any bus and verify the consistency of the model concerning a real-life condition or other proved model, as shown in Figure 29. For the Salto Caxias system, this was performed using ANAREDE's power flow capabilities. Figure 30 provides visibility on how this is performed.

Figure 30 – Block diagram for the validating the constructed model using another proven model.



SOURCE: The author (2022).

4.2.3 Batch Event Simulation

In order to construct a dataset with enough volume and diversity for a machine learning model to be successfully trained and applied, a large number of events must be simulated. In order to increase process scalability and reusability, a batch class was created to automatically and iteratively simulate all possible scenarios (bounded by PST's capabilities and limitations) in a given power grid whose model was defined and provided.

The only bus in the Salto Caxias system that was included in the datasets provided by COPEL is that of Cascavel West 230 kV (CASCAV-230kV). Thus, even though variables for all buses in the system model are simulated by PST, only those in the aforementioned bus will be kept in the process' output files.

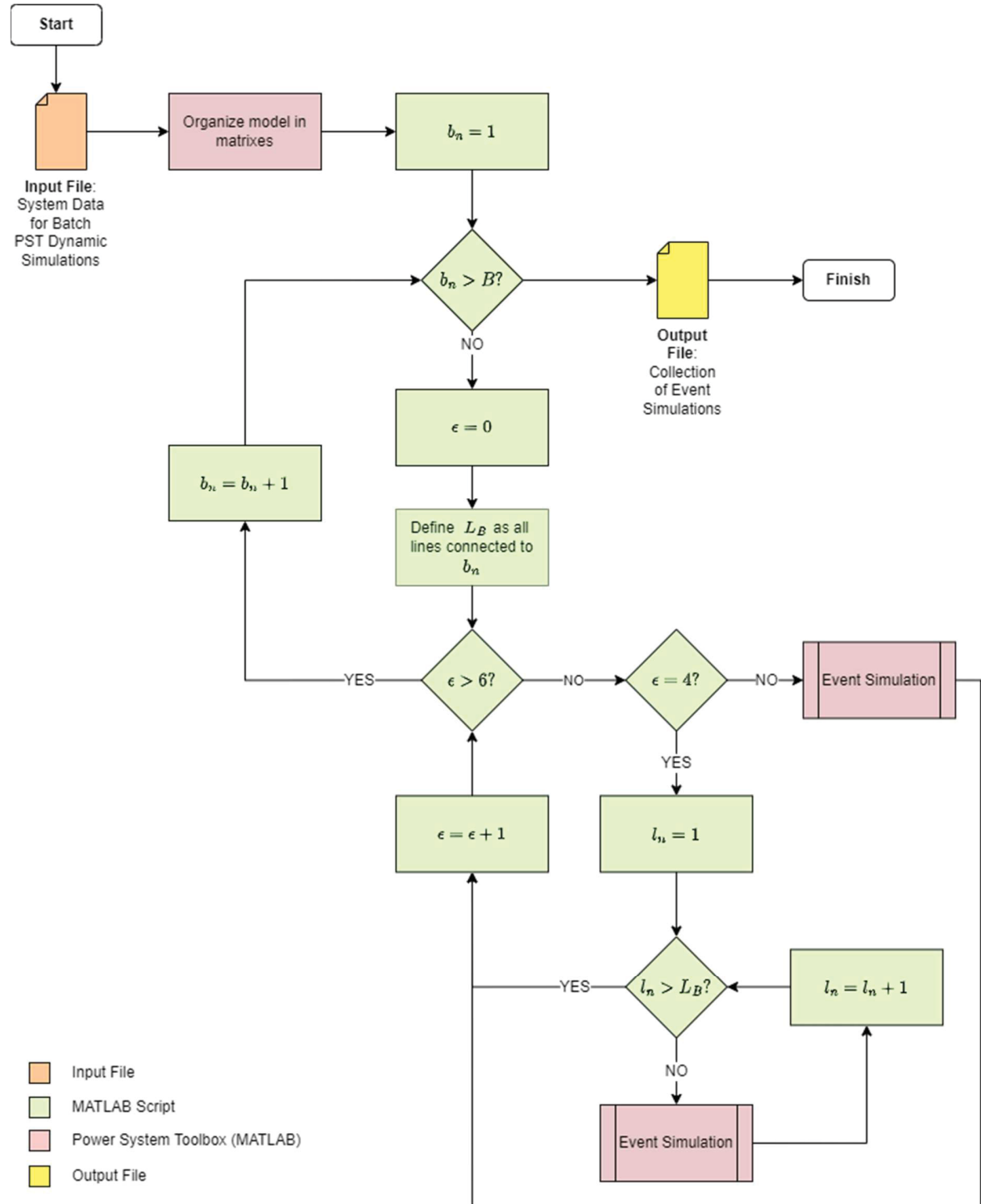
Aiming to achieve simulation of all possible and feasible scenarios within the Salto Caxias system, the class iterates over the number of buses B and lines L_B (including transformers) and over the event type indicator ϵ . A particular iteration's bus or transmission line is thereby denoted by b_n or l_n , respectively.

In each one of the iterations, thus, a simulation is executed. Therefore, with 27 buses, 12 transmission lines, 25 transformers, and 7 simulation types, at the end of the class execution a collection totaling 236 MATLAB data files (representing 236 simulations) containing 45s of simulation with PMU sampling rates is obtained.

Furthermore, in order to more accurately represent a PMU's behavior, white noise with 45dB SNR is added. The datasets are also converted from cartesian form (real and imaginary values) to polar form (phasor) to better approximate them to what is found at a PDC.

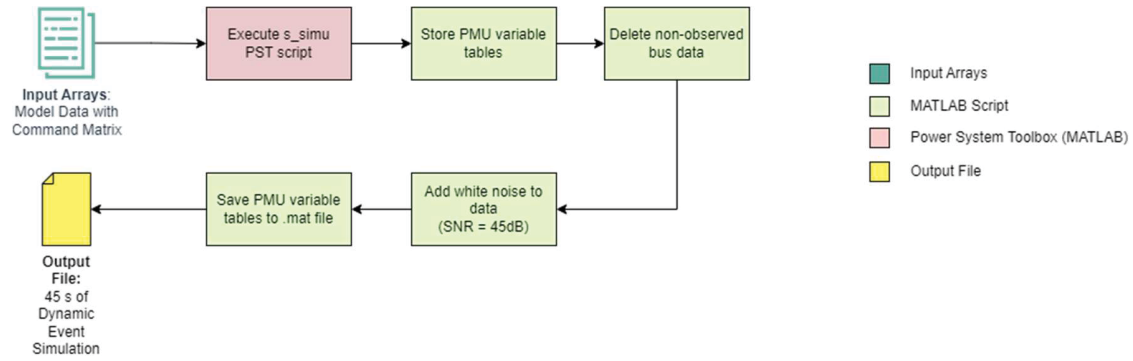
The iteration structure is available in Figure 31 with a denotation for the event simulations that ties into Figure 32, which illustrates the programmatic actions to collect simulation data, add white noise and save it.

Figure 31 – Block diagram representing the iteration structure designed to simulate all possible events in a power system using PST, with a collection of event simulations as its output.



SOURCE: The author (2022).

Figure 32 – Block diagram for the simulation and data grooming that happens within every iteration of the batch simulation process.



SOURCE: The author (2022).

4.2.4 Data Processing

In order to improve machine learning model performance in this context where they are trained with simulated data and applied over real event PMU data provided by COPEL, a series of data manipulation techniques, which do not insert any biases and therefore do not invalidate the model construction and application, must be used.

It's noteworthy that these processing techniques must be equally applied to both the simulated and real event datasets so that whatever is fed to the machine learning models is in the same standard and format.

The constructed framework processes each file provided to it, segmenting them between the real-life application or simulation-based training files. This is done because the simulated dataset needs to be labeled ("short-circuit", "load loss", "line loss" or "normal") prior to being processed, as the supervised learning-based training requires it.

In regards to the simulated dataset, the data are initially divided into pre-event, event, and post-event subsets – and are labeled through different procedures. It's noteworthy that for files where event type $\epsilon = 6$ (steady-state), all samples are labeled as "normal".

From the first subset, specifically in the Salto Caxias system scenario, the first 10 seconds of measurements are removed, as they include simulation initialization and systemic stability is reached a few seconds before the 10 s mark in every simulation. The

5 remaining seconds in the pre-event subset are, then, used to extract the means and standard deviations of steady-state operation and are, thus, labeled as “normal”.

The event subset, which in the specific scenario relative to this dissertation is represented by 2 seconds worth of measurements, is labeled as either “short-circuit”, “load loss” or “line loss” in its entirety, depending on which event type ϵ was defined.

Lastly, the last subset of the three, which relates to post-event data, is constituted of the remaining measures until the simulation’s end (18 seconds worth of measurements). In case the simulated event is in the “short-circuit” category, the samples are automatically labeled as such. As for “load loss” and “line loss”, the ONS operation limits (frequency: $59.5 \leq f \leq 60.5 \text{ Hz}$; voltage: $0.95 \leq v_{mag} \leq 1.05 \text{ pu}$) are consulted for each sample and, in case they are not respected, then they are labeled according to the ϵ value for the simulation. If the limits are respected, then the sample is labeled as “normal”.

The training data have their order randomized in blocks of W samples (usually and in the present work, $W = 1$) after categorization. This procedure is applied to mitigate the biasing effects on the machine learning models stemming from the fact that all simulations have the same size and structure – that is, it is ensured that the algorithms’ learning is dictated by electric variables and not by the form in which the samples were organized.

Afterward, using the mean and standard deviation values collected in the pre-event data subset, the whole dataset can undergo standardization, which allows all variables to be treated under the same scale with no distortion of the difference between two distinct points – in other words, removal of any effects the grid’s operation point might have on the machine learning model.

The usage of these particular values for the mean and standard deviation also stems from a need of making outliers and anomalous behaviors more evident in relation to normal, steady-state ones.

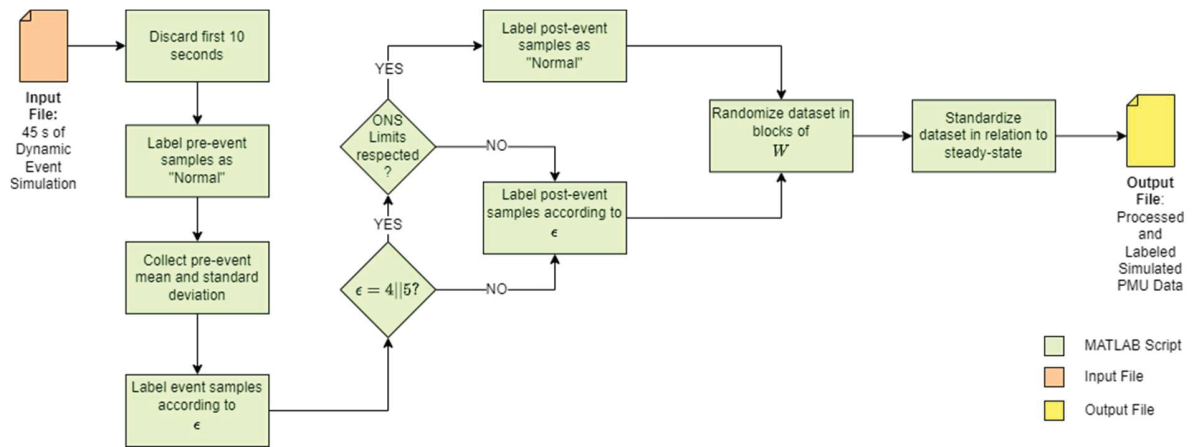
This concept is mathematically described by Equation 12, where $[x_s]$ refers to a standardized dataset originating from $[x]$.

$$[x_s] = \frac{[x] - \bar{x}}{\sigma_x} \quad (12)$$

Once standardized, the data are saved to a temporary MATLAB data file. Upon the processing of the last file in the queue, all samples from all files are loaded and consolidated into one single file in a supervised machine learning-compatible format.

This whole procedure is performed iteratively, so as to assure all simulation files in the directory undergo it. Figure 33 illustrates one single iteration – that is, one file being processed – of it.

Figure 33 – Block diagram for preprocessing a given simulated PMU dataset.



SOURCE: The author (2022).

In addition, the simulated and labeled dataset's structure is one where each sample is a row and each variable is a column, as per Frame 3. The labels are, then, appended to the simulated samples and operate as a n -bit “one-hot” array (only one bit can be active at a time) to indicate the target classification. For this dissertation, $n = 4$, as short-circuits are grouped for enhanced model learning as suggested by Barreto et al. (2021).

Frame 3 – Learning (simulated and labeled) dataset structure.

Simulated Samples						Learning Targets			
v_{mag}	v_{ang}	i_{mag}	i_{ang}	f	$ROCOF$	“Short-Circuit”	“Load Loss”	“Line Loss”	“Normal”
2100 samples in randomized order per simulation. Each sample is a row.						0 1	0 1	0 1	0 1
						0 – Negative 1 – Positive Only 1 bit can be active for a given sample.			

SOURCE: The author (2022).

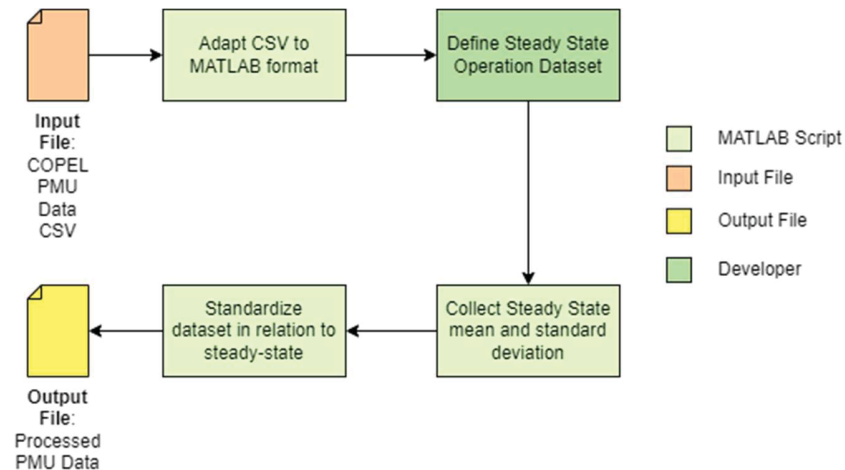
When it comes to real event data, it is notable that the files must be adapted to MATLAB-compatible formats prior to anything and that dataset randomization is not a requirement. Afterward, for the dissertation's scenario, upon ONS and agent studies on the event, a period of steady-state operation can be captured so as to extract the means and standard deviations from it but, for result analysis purposes, it cannot be accurately labeled since there was no event in the vicinities of CVO-230kV and, thus, such labels shall not be present in the next section.

The dataset is, as it happens with simulated data, standardized using the mean and standard deviation collected for the steady-state periods – thus making it compatible with the machine learning model that can be constructed using the procedures described in the following subsections.

Figure 34 illustrates the procedures for data processing being applied to one single real event data. It is notable that the classes and methods produced to enable this act on all files in a directory through an iterative method.

It is also important to restate that the same codebase processes both simulated and real event data, with the appropriate logic supporting each distinct scenario.

Figure 34 – Block diagram for preprocessing a given real-life PMU dataset.



SOURCE: The author (2022).

4.2.5 Neural Network Model Determination and Evaluation

Neural Networks historically have demonstrated sufficient results for real-life application, given their computational structure, ease of use, and pattern recognition capabilities. Thus, for proof-of-concept purposes, in this stage of the dissertation, they were the first choice of machine learning algorithm, for application. Considering the supervised nature of the learning process for the technique, the dataset format with samples and labels established in the previous subsubsection is leveraged.

An iterative process for defining the ANN's hyperparameters is, thus, initialized. With a designer-defined κ maxima numbers of hidden layers and χ neurons per layer, an exhaustive search can be defined. Specific to the dissertation's scenarios, the boundaries for the search variables are $K = 2$ and $[X] = [20, 20]$, respectively.

The usage of genetic algorithms to find the best combination of hyperparameter values is noted throughout the academic community. Such an approach, however, requires the provisioning of an estimative in the vicinities of the global optimal one.

The input layer has the same number of entry points as there are variables – which means that for each PMU under observation, six ANN entries relating to voltage magnitude and angle, current magnitude and angle, frequency, and ROCOF are created. Specific to the Salto Caxias system scenario, the ANN was defined to have 5 input nodes.

As for the output layer, it must have as many output points as there are identifiable classes. For the methods developed and applied through the present work, there are four possible classes: “short-circuit”, “line loss”, “load loss” and “normal”.

With respect to the exhaustive search itself, each iteration's model is trained only once. Performance metrics such as loss of cross-entropy M and F_1 Score can be used to orient and automate this process. Specifically for this dissertation, the former is used so as to avoid the computational cost of calculating metrics outside of the optimization algorithm used for training.

At the end of the search, within the defined space, the optimal number of hidden layers and neurons in each one of them – that is, the model – is defined. Pursuant to this, however, attesting model robustness is of vital importance, since ANNs can develop categorization biases through overfitting – which may be generated by the usage of

specific subsets of samples for the training, validation, and testing stages of the learning process.

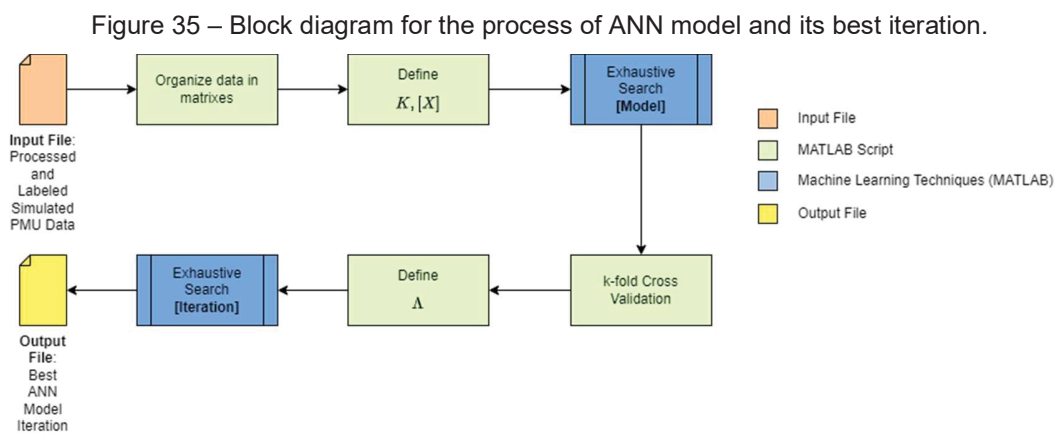
To do so, Figure 9's k -fold cross-validation can be applied for the model, ensuring each sample is used at least once for testing purposes and, thus, upon consistent performance metrics across all folds, model robustness can be attested. Specific to the Salto Caxias system scenario, $k = 10$.

However, even with a robust and performant model, the random initialization of neuron synaptic weight is still in effect. That is, although the optimal ANN structure for a given problem was defined, it is still not tuned to achieve its potential optimal performance.

To do so, another exhaustive search must be executed, with its space being defined by a number of iterations large enough to diminish the aforementioned effect. Thus, the learning process for the defined model is repeated Λ times and, by its end, the model iteration with the best metrics is selected and permanently saved for further application. Specifically for the Salto Caxias scenario, $\Lambda = 50$.

A set of scripts using MATLAB's toolbox for machine learning (nprtool) was developed to perform the procedure described above with a split of 70:15:15 for training, validation, and testing datasets.

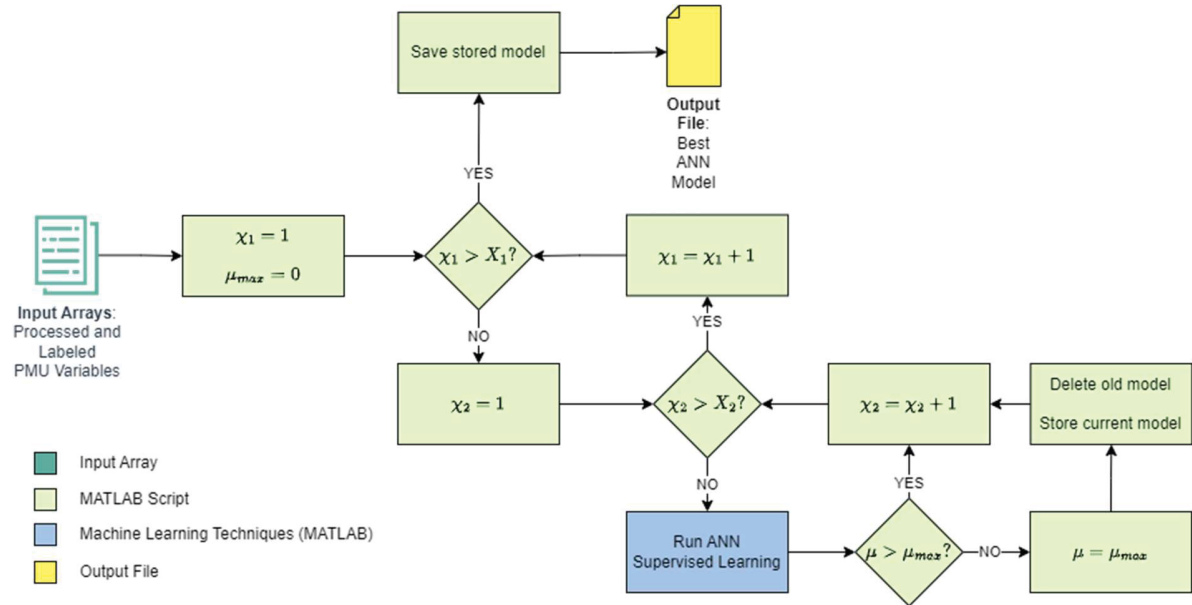
Figure 35 illustrates the actions taken to reach the final ANN with the parameters that had an optimal performance for event identification. It is supported by Figures 36 and 37, which picture the iteration structures described in the present subsection to determine the ANN model and its best iteration respectively.



SOURCE: The author (2022).

Specifically in Figure 36, X_1 denotes the maximum number of neurons in the first hidden layer, whereas X_2 does the same for the second hidden layer. χ_1 and χ_2 represent the iteration variables for this stage. μ_{max} and μ are the current maximum and iteration loss of cross-entropy, used for comparisons to determine whether a model was better than the previous best one.

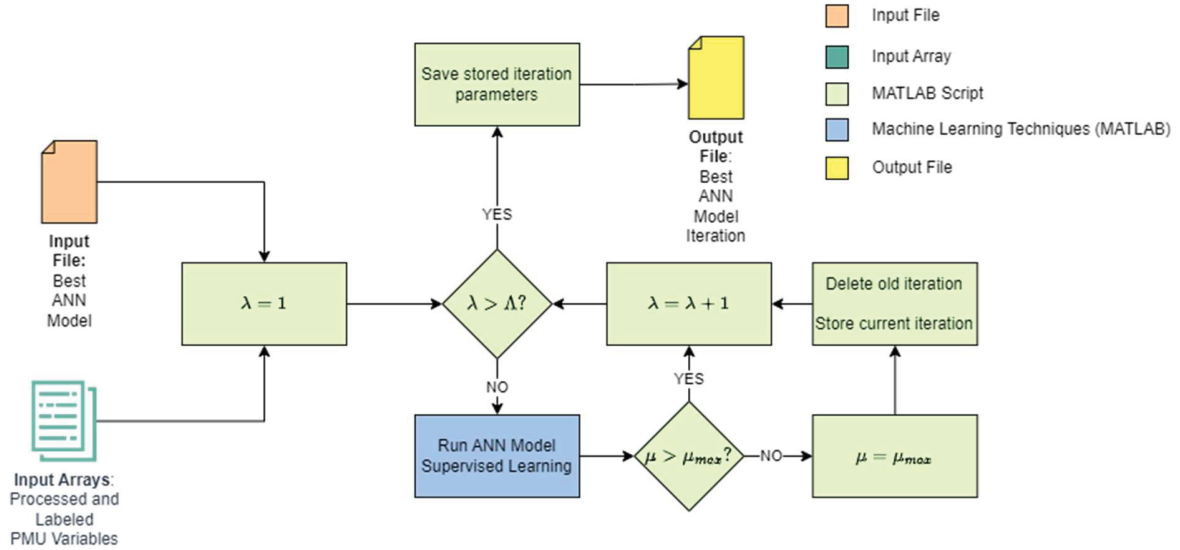
Figure 36 – Block diagram for the exhaustive search for the most performant ANN model.



SOURCE: The author (2022).

For Figure 37, Λ is the maximum number of iterations that the exhaustive search can perform and λ is the iteration variable. μ_{max} and μ , in its context, are the current maximum and iteration loss of cross-entropy, used for comparisons to determine whether a model's iteration was better than the previous best one.

Figure 37 – Block diagram for the exhaustive search for the most performant ANN model iteration.



SOURCE: The author (2022).

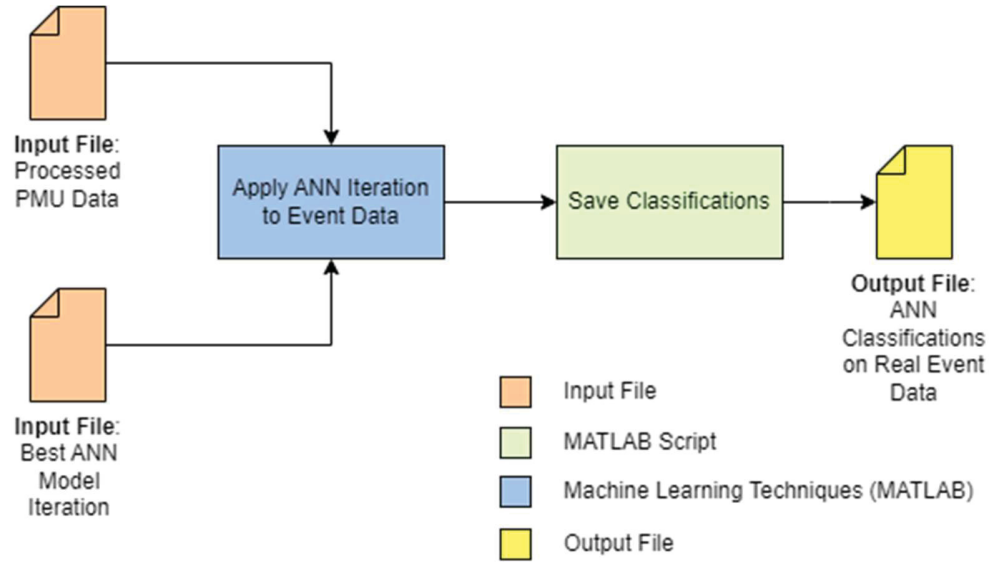
4.2.6 Neural Network Model Application

With the real event data already processed, in a compatible format to the methods described in the previous subsections and with the best possible and feasible iteration of the machine learning (neural network) model defined and trained in the simulated datasets, simply applying the aforementioned real event data to the model is sufficient for result capture.

Upon algorithm execution, it is possible to visualize its predictions against the data. In case one desires to obtain confusion matrix metrics such as accuracy, precision, recall, and F1 Score, labeling the samples prior to application is required. This may be of interest when trying to ascertain model performance in relation to the reality identified by the human grid operators.

The procedure and required files for it are illustrated in Figure 38.

Figure 38 – Block diagram for the application of real event data on a machine learning model trained with simulated data.



SOURCE: The author (2022).

To apply the models achieved through this process in a real-time, online application, several new steps and perhaps new subprocesses must be included so as to have a performant, minimal viable product – such as code optimization and adaption to a PDC-embedded architecture, sliding windows, daily extraction of means and standard deviations, periodic model retraining/redefinition and more.

5 RESULTS

5.1 Simulated Data-based Learning Results

The total number of simulations performed to construct the learning dataset through the processes described in the previous section is 236, with each simulation file containing 35 s worth of measurements (considering the initial 10 s from each file that are discarded). The total number of generated samples per variable is, then, roughly 500,000. A detailed view of these numbers can be seen in Table 5.

Table 5 – Detailed view of the generated dataset for model definition and learning.

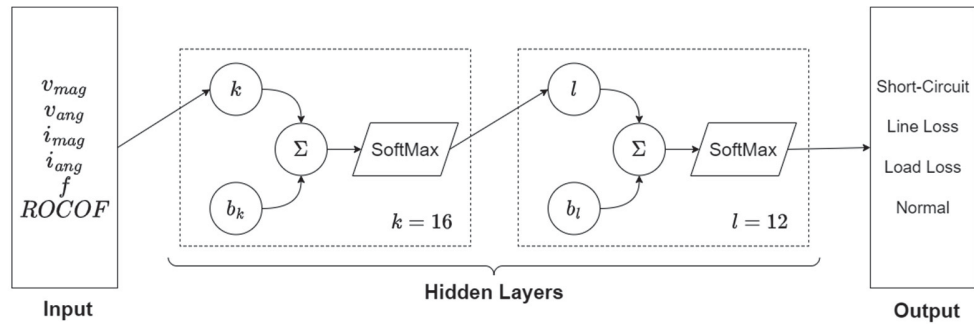
Simulation Type	Number of Simulations	Samples per Variable
Three-phase to ground Short Circuit	27	56,700
Two-phase Short Circuit	27	56,700
Two-phase to ground Short Circuit	27	56,700
Single-phase to ground Short Circuit	27	56,700
Line Loss	74	155,400
Load Loss	27	56,700
Steady-state	27	56,700
Total	236	495,600

SOURCE: The author (2022).

Through the execution of the learning process described in the previous section, where the model is trained, tested, and validated using the simulated dataset, a neural network model with six inputs, and two hidden layers where the first has 16 neurons and the second has 12, and four outputs was defined. Both hidden layers use SoftMax functions. Its cross-entropy loss, during the exhaustive search, was 0.2247 through 311 epochs

The model structure can be found in Figure 39.

Figure 39 – Defined neural network model structure.



SOURCE: The author (2022).

The 10-fold cross-validation results for the model are available in Table 6.

Table 6 – Metrics overview for the model's cross-validation.

Class	Metric	Value (mean of all folds)
Short-Circuit	Precision	0.867
	Recall	0.842
	F_1	0.854
Line Loss	Precision	0.952
	Recall	0.917
	F_1	0.934
Load Loss	Precision	0.969
	Recall	0.912
	F_1	0.940
Normal	Precision	0.892
	Recall	0.922
	F_1	0.910
Accuracy		0.901
F_G		0.910

SOURCE: The author (2022).

This table presents the mean of all 10 folds of the cross-validation process, which is described in Figure 9. Through it, the average behavior for each metric across all classes can be described – and, thus, the model’s capacity of classifying yet-unseen data and the balance of its performance across the classes can be analyzed.

Using the short-circuit class as an example, one can see that the precision is 0.867 and that the recall is 0.842 on average across the 10 folds. This means that out of all samples categorized as short-circuit by the model, 86.7% are actually labeled short-circuits in the fold’s testing dataset – and that out of all samples labeled as short-circuit in the fold’s testing dataset, 84.2% are categorized accordingly. F_1 Score, being the harmonic mean of the two aforementioned metrics, allows one to have a generalist view on the model generalization on each class.

In regards to the exhaustive search for the model’s best iteration (where the training, testing, and validation process is repeated 50 times) and its resulting optimal neural network, the metrics are presented in Table 7.

Through analysis of the mean and standard deviation values, one can verify the spread of the metrics that describe the ANN’s performance through all 50 iterations. The best iteration, as described in the previous section, is the one chosen for the application of real event data, given that it has the best generalization capabilities and performance.

Once more using the short-circuit class as an example, one can see that across all 50 runs of the training, testing and validation process, the minimal value perceived for the precision metric was 0.8653 and the maximum value was 0.8721 with the average of all runs being 0.8688. The standard deviation for this metric on the short-circuit class is on the scale of 10^{-3} , which illustrates a small spread across all runs.

It is important to note that the minimal and maximum value columns do not necessarily refer to the same run across metrics. That is, the run that presented the minimal short-circuit precision may not be the run that presented the minimal short-circuit recall and so forth.

The best iteration column refers to the values captures for the model iteration that had the best combination of precision and recall across all classes – that is, the best F_G Score, which is the mean of the F_1 Scores of each one of the four classes.

Table 7 – Metrics overview for the validation phases of the best iteration and of all 50 different trained ANN model iterations.

Condition	Metric	Minimal Value	Mean Value	Maximum Value	Standard Deviation	Best Iteration
Short-Circuit	Precision	0.8653	0.8688	0.8721	0.0017	0.8703
	Recall	0.8284	0.8376	0.8403	0.0030	0.8398
	F_1	0.8450	0.8512	0.8557	0.0027	0.8548
Line Loss	Precision	0.8812	0.9036	0.9615	0.0402	0.9541
	Recall	0.8573	0.8984	0.9219	0.0162	0.9213
	F_1	0.8693	0.8976	0.9403	0.0178	0.9374
Load Loss	Precision	0.9183	0.9674	0.9798	0.0154	0.9771
	Recall	0.9491	0.9712	0.9763	0.0068	0.9752
	F_1	0.9327	0.9689	0.9778	0.0113	0.9762
Normal	Precision	0.8937	0.8944	0.8975	0.0010	0.8973
	Recall	0.9279	0.9286	0.9312	0.0008	0.9292
	F_1	0.9077	0.9086	0.9138	0.0016	0.9130
F_G		0.8947	0.9059	0.9204	0.0067	0.9204
Accuracy		0.8774	0.8946	0.9033	0.0065	0.9033

SOURCE: The author (2022).

The confusion matrix relating to the training, testing, and validation phases of the learning process is available in Figure 40. Through it, all metrics used for machine learning model evaluation in this dissertation are calculated.

In addition, the accurate classifications can be viewed together with the inaccurate ones, which allows for an analysis on what are the most common mistakes the model might make in the face of yet-unseen data – that is, which classes it mixes up the most.

Furthermore, the confusion matrix can also provide one with key performance indicators for possible improvements in the preprocessing algorithms and reliability across classes, as it contains information about the classes with higher misclassification rates and the boundaries across them.

Another observable fact is that the most common confusion is between normal and short-circuit, having mistaken the two (classifying short-circuit samples as normal or vice-versa) in 45,032 instances.

5.2 Real Event Data Application Results

Since no labels could be established for the real events, that means that a confusion matrix cannot be constructed for the classifications that regard them. A quantitative overview of classifications can be constructed, however, since each one of them is associated with a timestamp of the PMU dataset.

This allows for the construction of a table where the volume of observations can be denoted for a given class across the whole dataset provided to the network and, additionally, of a graph where all classifications can be plotted together with one of the PMU variables.

Specifically for the plots, all non-“Normal” classes are grouped so as to have only two graphs to compare the event and the classifications. In that sense, the graphs describe where did the ANN detect an event (output equal 1) and where it did not (output equal 0).

For the first event, relating to the PMU bad data registered across the Brazilian integrated system, the neural network classification overview is available in Table 8.

Table 8 – Classification overview for the first event.

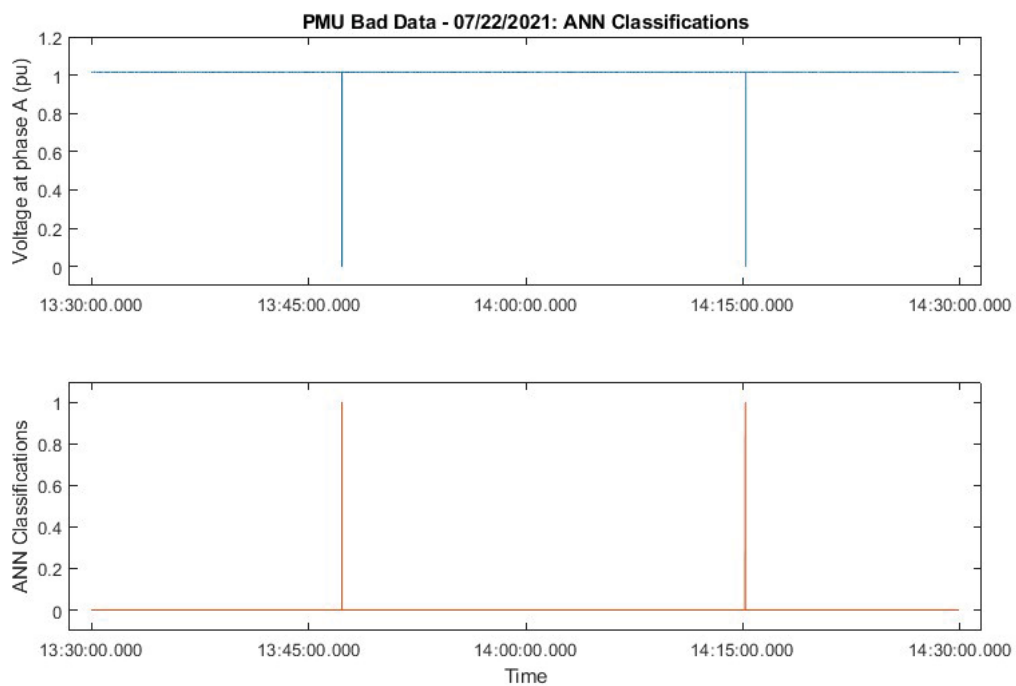
Class	Number of Observations
Short-Circuit	491
Line Loss	0
Load Loss	0
Normal	215509
Total	216000

SOURCE: The author (2022).

All ANN observations denote that the occurrence is of one single type – albeit short-circuit and not “bad data”, as PST does not include such a scenario.

The event itself is abrupt and does not take place for an extended period of time, with two sudden drops to zero in under an hour. Figure 41 illustrates the voltage magnitude for phase A of the CASCAV-230kV PMU and the detection results.

Figure 41 – First event detection graph.



SOURCE: The author (2022).

The perfect overlap between the drops across the PMU variables and the ANN detections indicates that the events are being detected at the correct timestamps every time, for this type of occurrence.

As for the second event, relating to the Amapá state grid temporary shutdown, the neural network classification overview is available in Table 9.

From it, it can be stated that less than 3% of measurements in the file are detected as events in the Salto Caxias system, but that out of all classifications, 91.76% of them indicate short-circuit or load loss – both of which are plausible classes when considering the event root cause.

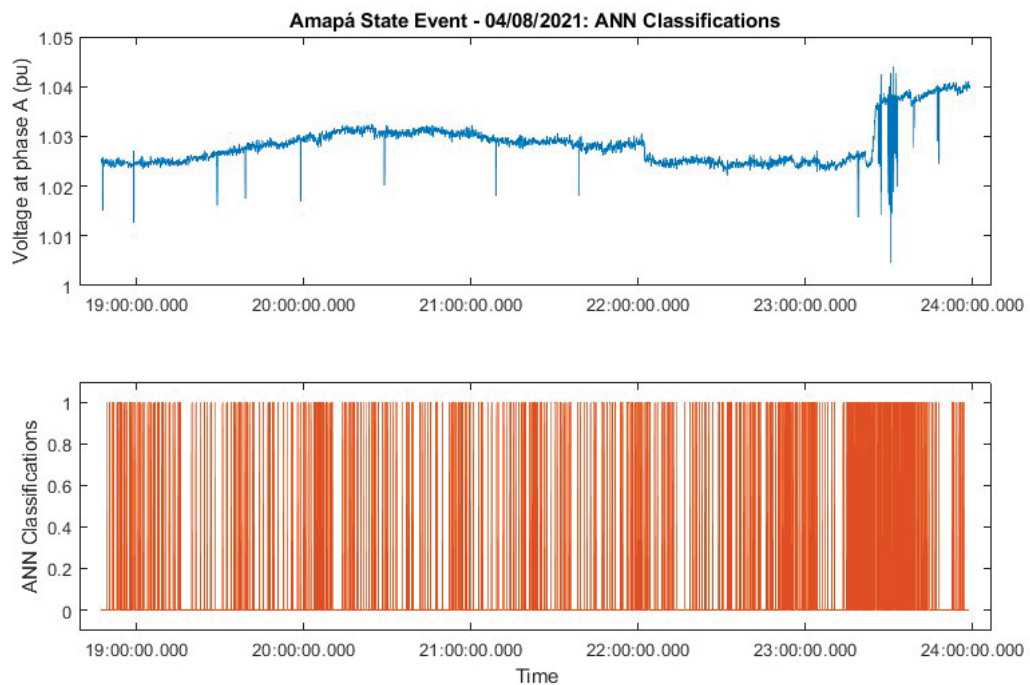
Table 9 – Classification overview for the second event.

Class	Number of Observations
Short-Circuit	26990
Line Loss	3459
Load Loss	11547
Normal	1080222
Total	1122218

SOURCE: The author (2022).

Figure 42 illustrates the voltage magnitude for phase A of the CASCAN-230kV PMU and the detection results. There are multiple spikes on the PMU variables, which get translated into a higher classification/detection density around them.

Figure 42 – Second event detection graph.



SOURCE: The author (2022).

Lastly, for the third event, relating to the Xingu-Estreito HVDC link shutdown, the neural network classification overview is available in Table 10.

Through inspection of it, it can be seen that around 1.7% in the file are detected as events in the Salto Caxias system, but that out of all classifications, around 69.91% indicate a line loss (which is, overall, the root cause of the event).

Table 10 – Classification overview for the third event.

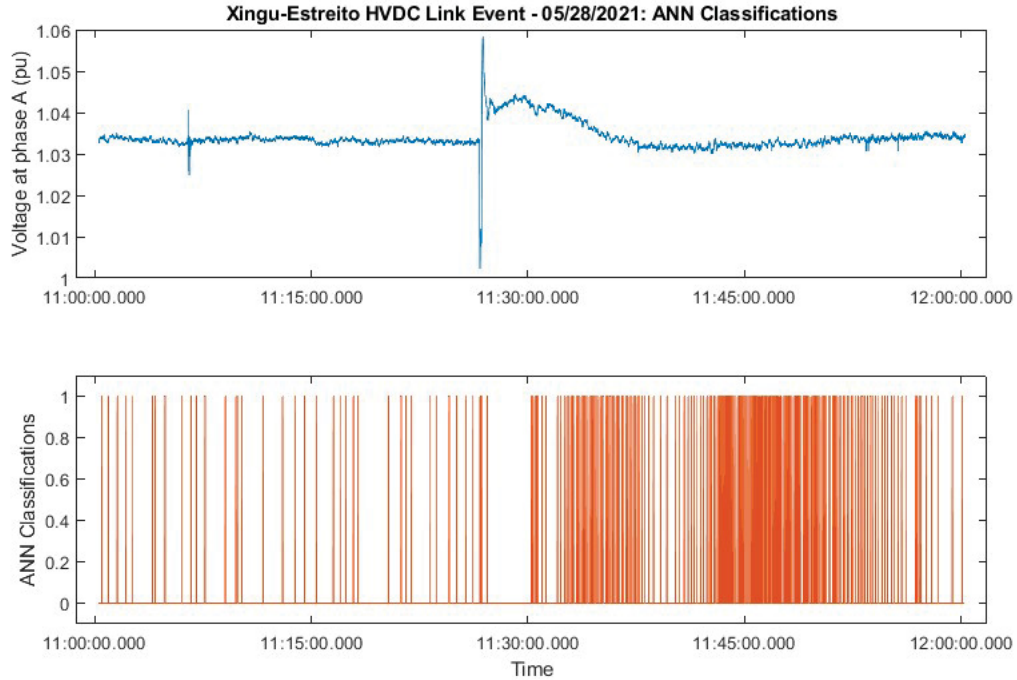
Class	Number of Observations
Short-Circuit	1126
Line Loss	2617
Load Loss	0
Normal	212317
Total	216060

SOURCE: The author (2022).

Figure 43 illustrates the voltage magnitude for phase A of the CVO-230kV PMU and the identification results.

The two main spikes in the PMU variables are detected, together with some other fluctuations due to the oscillations that stem from one of the HVDC link's poles being shut down and then having its power accommodated in the remaining poles.

Figure 43 – Third event identification graph.



SOURCE: The author (2022).

5.3 Discussion

The structure and format of the dataset constructed through event simulations within the Salto Caxias system assure similarity to real PMU data, which is one of the first enablers in the execution of the work. Through 236 simulations spanning seven different types with 35s worth of emulated PMU measurements, the diversity and data volume of the dataset also allows for the execution of the model learning and definition stages.

The choice of using PST for these simulations comes from its capabilities of simulating multiple event types across all buses and lines of a system model in a batch fashion, alongside having a customizable sample rate that allows for PMU emulation. Furthermore, PST allows the programmer to set simulation times, which facilitates labeling the samples when constructing the learning datasets.

It's noteworthy that the usage of a single PMU (CVO-230kV) for both model learning on simulated data and application on real data is due to the non-availability of

COPEL PMU event files containing measurements collected in any other PMU located within the Salto Caxias system.

The ANN achieved through the processes described in Section 4, with five inputs, two hidden layers with 16 and 12 neurons and four outputs presented consistent results and behavior upon 10-fold cross-validation, with the means of its metrics being above 0.8 with low standard deviations. More specifically, the mean of F_G Score being 0.910 indicates that the model is capable of generalization and has a low probability of presenting overfitting behaviors stemming from native biases in the learning datasets.

Repeating the learning process for the defined model 50 times also assures that the best possible iteration of the model is being selected for further applications. The small range (maximum to minimum) paired with mean values above 0.8 and low standard deviations of the metrics across all iterations attests to that, with the model presenting consistent behavior in the face of multiple.

The designed ANN has acceptable values of recall, precision and F_1 in relation to all classes alongside a global F_G score of 0.9204 and an accuracy of 0.9033, when considering the intent of using it on real event data. Considering the confusion matrix of Figure 39, the bulk of its errors are concentrated in differentiating “Short-Circuit” and “Normal”, 4.4% of the time confusing the former with the latter and 4.6% of the time mistaking “Normal” with “Short-Circuit” – meaning that although errors are unlikely when it comes to simulated data, most errors are between the “Normal” and “Short-Circuit” classes.

As for the real PMU datasets, in the first event, where a WAMS failure (bad data) is registered across Brazil’s PMUs, the model classifications a short circuit. This happens because no “bad data” examples could be provided to the model since the simulator of choice does not contemplate these scenarios. Given its simplicity, this dataset was used for the first validations of the model. Nonetheless, the ANN flawlessly observes anomalies at the correct timestamps for this dataset where the event is present within the Salto Caxias system.

When it comes to the other two events, one being a shutdown of the Amapá state grid and the other the momentary loss of pole 1 of the Xingu-Estreiro HVDC link, there are two possible interpretations, depending on the geographic context of the analysis.

If any anomaly in the Brazilian integrated system, captured at any PMU, is considered to be an event for the machine learning model, the detection characteristic of the ANN model does not fully match the transient behaviors presented by the data. It momentarily recognizes behaviors such as short-circuits, line losses, and load losses – all of which are plausible, given the nature of the events – but the classifications are few and far between, scattered across the timeline provided in the CSV file and not grouped in any sense.

That is, specifically at a national level, the approach leveraging one single PMU simulated in the Salto Caxias system does not suffice for real-life application.

In the observed PMU's local context, however, considering the distance of the events' epicenters to it, no anomalies happen – that is to say, there are no load intermittences, supply shortages, or damage to the Salto Caxias system infrastructure or its consumers' equipment. Through that perspective, the ANN's behavior is consistent with the results obtained in the learning stages of the model, where the variables are generally within ONS' operation limits and the classifications in Tables 17 and 18 both denote steady-state operation for more than 95% of the time on average.

In short, at a local level, for the Salto Caxias system, the single PMU simulation approach can be considered for real-life application.

The hierarchized structure noted in (BALAKRISHNAN; GOPINATH, 2020) among other works has a similar effect to that of the one achieved by this dissertation's ANN model. The simpler, single black-box design, when put in contrast with a multi-layered hierarchical design presents more manageability for real-life scenarios, as the retraining processes become more streamlined over time. However, the gains in architecture and the reduction of pre-processing algorithms found in hierarchized structures are also determinant factors.

The introduction of DLA methods in the event classification problem, noted in works such as (SRIKANTH; KOLEY, 2021) and (SHADI et. al, 2022) as opposed to the usage of more traditional methods such as the ANN used throughout this dissertation, brings the benefit of higher generalization capabilities with less pre-processing stages. On the other hand, computational processing power is a concern when dealing with DLA – whereas

the simpler structure of a common shallow ANN is, nowadays, more suitable to run on a PDC's server-side given its lower processing requirements.

Furthermore, none of the aforementioned papers include real event data in their results and discussions – they all use either the IEEE 9-bus or 39-bus benchmarking models to prove their concepts, whereas the present dissertation's main efforts are aimed toward validations with measurements captured at real PMUs.

6 CONCLUSIONS AND FUTURE WORKS

The volume and diversity found in PMU-based WAMS datasets are unprecedented in the energy transmission space and, thus, the exploration and usage of data analytics and machine learning have been one of the main themes studied by both academic and technical communities over the last few years.

Constraints around the scarceness and low availability of electromechanical event data, however, are one of the main blockers for further optimizing the capabilities of WAMS/WACS when it comes to machine learning. Power systems are built prioritizing robustness and reliability for socioeconomic reasons and, therefore, such occurrences are few and far between even when considering wide area systems.

Thus, the present dissertation aimed to address this example scarcity problem by modeling real systems and simulating events on them. That way, one can achieve the construction of a labeled dataset large enough to enable the design of machine learning models with examples of various anomalous behaviors in the system.

Such a hypothesis was validated by modeling the Salto Caxias system (one of the areas of COPEL's power grid), using system reports and documentation provided by the aforementioned power company, simulating events in all of its buses, transmission lines, and transformers via MATLAB-based Power System Toolbox and constructing a neural network model based on the generated data to identify event types.

The training, testing, and validation results for the ANN model constructed through this method attested to its robustness. Even though the largest number of the errors presented by it is in mistaking short-circuits with normal operation measurements (and vice-versa), with an aggregated percentage of 9% of all provided samples, its accuracy and F_G score indicate sufficient performance across all four classes when considering real-life scenario application.

Another aim of the presented work was to verify the performance of a machine learning model constructed through the usage of simulated data in real event data captured by a PMU in the real-life version of the modeled system through two perspectives: a national-level and a local-level one.

The datasets containing events registered by the CVO-230kV PMU (which is within the Salto Caxias system) relate to occurrences that took place far from its geographical location. Thus, even though they were registered, they are situationally different in regard to the simulated data. However, these datasets were the only available ones where the aforementioned PMU registered any sort of anomalous behavior and, therefore, were used for validation purposes.

No labels for the events were provided by COPEL, and, given the signal distortion and attenuation stemming from the distance between the PMU and the event epicenters, accurately labeling them is a conceptually complex task. On one hand, the national interconnected grid is impacted on varying levels of criticality – but on the other, seldom did the Salto Caxias system violate the ONS limitations that guardrail “normal operation”.

Notably, the simulated behavior for to-ground short-circuits is similar to bad data (with some variables falling as down as zero), but no further correlation can be affirmed in that regard, as the machine learning model interpretations in the hidden layers are not intelligible to humans.

One way to address this, in future works, is to simulate occurrences of PMU bad data in the same way as electromechanical events and retrain the model adding it as a new, fifth class.

Given that the simulated events are much higher in intensity than what’s presented in the real event measurements, it is a fact that the ANN model did not receive any examples of similar events. Even though data standardization aids in removing the effects of the grid’s point of operation, it does not change the distances between two different points.

As such, to address this for future works, other data processing techniques that help in evidencing outliers and that have relativizing effects can be applied. One such example is Z-Score normalization and Pearson’s Correlation, which have been noted to help identify anomalies in relation to the dataset’s characteristics (KHALEDIAN et al., 2021).

The present work proves the concept that a machine learning model trained with PMU data simulated in a modeled power grid can identify events that take place within that particular power grid. However, it is noted that the efforts of this dissertation do not

include event identification across interconnected systems – this would be a scaled approach to the one proposed throughout this work.

With that in mind, the procedures and codebase produced as an outcome of this dissertation can be leveraged and perhaps enhanced for future works and even the construction of real-life applications, as instead of modeling the Salto Caxias system, one would need to model the Brazilian interconnected system in order to identify events on a national scale.

Aside from the previous indications of adjustments and improvements for possible new works, considering the nature of PMU data and of machine learning models, future developments based on this dissertation can also involve:

- Feature engineering and extraction to improve the presentation of data to the models;
- Application of other commonplace classification techniques such as k-Nearest Neighbors (kNN), Support Vector Machine (SVM), Random Forest (RF), and Convolutional Neural Networks (CNN);
- Hierarchical identification structures, where different binary identifications are performed at each layer using the most performant model available for a particular problem;
- Real-time application of machine learning models on-premise within the PDC servers;
- Machine learning-based event location and event start/end detection solutions;
- Assistance for system operators during the execution of grid recovery procedures through machine learning classifications and predictions, providing engineers with initial estimates to increase efficiency and decrease recovery times.

REFERENCES

ALHELOU, H. H. An Overview of Wide Area Measurement System and Its Application in Modern Power Systems. Em: [s.l: s.n.]. p. 289–307.

ATPUTHARAJAH, A.; SAHA, T. K. **Power system blackouts - literature review**. 2009 International Conference on Industrial and Information Systems (ICIIS). **Anais...IEEE**, dez. 2009.

BALAKRISHNAN, P.; GOPINATH, S. A new intelligent scheme for power system faults detection and classification: A hybrid technique. **International Journal of Numerical Modelling: Electronic Networks, Devices and Fields**, v. 33, n. 5, 2020.

BARRETO, N. E. M. et al. Artificial Neural Network Approach for Fault Detection and Identification in Power Systems with Wide Area Measurement Systems. **Journal of Control, Automation and Electrical Systems**, 2021a.

BARRETO, N. E. M. et al. Artificial Neural Network Approach for Fault Detection and Identification in Power Systems with Wide Area Measurement Systems. **Journal of Control, Automation and Electrical Systems**, 2021b.

BISWAL, M.; BRAHMA, S. M.; CAO, H. Supervisory Protection and Automated Event Diagnosis Using PMU Data. **IEEE Transactions on Power Delivery**, v. 31, n. 4, p. 1855–1863, 1 ago. 2016.

BOTTOU, L. Stochastic Learning. Em: [s.l: s.n.]. p. 146–168.

CASTILLO, P. A. et al. Artificial Neural Networks Design using Evolutionary Algorithms. Em: **Advances in Soft Computing**. London: Springer London, 2003. p. 43–52.

CHAKRABORTTY, A.; KHARGONEKAR, P. P. **Introduction to wide-area control of power systems**. 2013 American Control Conference. **Anais...IEEE**, jun. 2013.

CHEN, M. et al. **Big Data**. Cham: Springer International Publishing, 2014.

CHOW, J. H.; CHEUNG, K. W. A toolbox for power system dynamics and control engineering education and research. **IEEE Transactions on Power Systems**, v. 7, n. 4, 1992.

CHOW, J. H.; SANCHEZ-GASCA, J. J. **Turbine-Governor Models and Frequency Control**. [s.l: s.n.]. Disponível em: <www.wiley.com/go/chow/power-system-modeling>.

CÖMERT, Z.; KOCAMAZ, A. F. Journal of Science and Technology A study of artificial neural network training algorithms for classification of cardiocography signals.

JOURNAL OF SCIENCE AND TECHNOLOGY, v. 7, n. 2, p. 93–103, 2017.

CUI, Q.; EL-ARROUDI, K.; WENG, Y. A Feature Selection Method for High Impedance Fault Detection. **IEEE Transactions on Power Delivery**, v. 34, n. 3, p. 1203–1215, 2019.

ENSSLIN, L. et al. Environmental disclosure management: a constructivist case. **Management Decision**, v. 50, n. 6, 22 jun. 2012.

FITZGERALD, A. E. (ARTHUR E.; KINGSLEY, C.; UMANS, S. D. **Electric machinery**. [s.l.] McGraw-Hill, 2003.

GARCIA, M.; GIANI, A.; BALDICK, R. **Smart grid data integrity attacks: Observable islands**. 2015 IEEE Power & Energy Society General Meeting. **Anais...IEEE**, jul. 2015.

GORE, R.; KANDE, M. **Analysis of Wide Area Monitoring System architectures**. 2015 IEEE International Conference on Industrial Technology (ICIT). **Anais...IEEE**, mar. 2015.

GRANDINI, M.; BAGLI, E.; VISANI, G. Metrics for Multi-Class Classification: an Overview. 13 ago. 2020.

HATZIARGYRIOU, N. et al. Definition and Classification of Power System Stability - Revisited & Extended. **IEEE Transactions on Power Systems**, v. 36, n. 4, p. 3271–3281, 1 jul. 2021.

HAYKIN, S. S. **Neural networks and learning machines**. [s.l.] Prentice Hall/Pearson, 2009.

HORNIK, K. Multilayer Feedforward Networks are Universal Approximators. **Neural Networks**, v. 2, p. 359–366, 1989.

IEEE. **IEEE Std 1344-1995(R2001): IEEE Standard for Synchrophasors for Power Systems**. [s.l.] IEEE, 1995.

IEEE. **IEEE Std C37.118-2005 (Revision of IEEE Std 1344-1995): IEEE Standard for Synchrophasors for Power Systems**. [s.l.] IEEE, 2005.

IEEE. **IEEE Standard for Synchrophasor Measurements for Power Systems Sponsored by the Power System Relaying Committee IEEE Power & Energy Society.** [s.l: s.n.].

IEEE. **IEEE Standard for Synchrophasor Data Transfer for Power Systems Sponsored by the Power System Relaying Committee IEEE Power & Energy Society.** [s.l: s.n.].

IEEE. **IEEE Guide for Synchronization, Calibration, Testing, and Installation of Phasor Measurement Units (PMUs) for Power System Protection and Control Sponsored by the Power System Relaying Committee.** [s.l: s.n.].

IEEE. **IEEE Guide for Phasor Data Concentrator Requirements for Power System Protection, Control, and Monitoring Sponsored by the Power System Relaying Committee IEEE Power and Energy Society.** [s.l: s.n.].

IEEE. **IEEE Std 421.5-2016 (Revision of IEEE Std 421.5-2005): IEEE Recommended Practice for Excitation System Models for Power System Stability Studies.** [s.l.] IEEE, 2016.

IEEE. **C37.247-2019 - IEEE Standard for Phasor Data Concentrators for Power Systems.** [s.l.] IEEE, 2019.

IEEE; IEC. **INTERNATIONAL STANDARD Measuring relays and protection equipment-Part 118-1: Synchrophasor for power systems-Measurements.** [s.l: s.n.].

IEEE; IEC. **INTERNATIONAL STANDARD Measuring relays and protection equipment-Part 118-1: Synchrophasor for power systems-Measurements.** [s.l: s.n.].

JOSHI, A. V. **Machine Learning and Artificial Intelligence.** [s.l.] Springer International Publishing, 2020.

KHALEDIAN, E. et al. Real-Time Synchrophasor Data Anomaly Detection and Classification Using Isolation Forest, KMeans, and LoOP. **IEEE Transactions on Smart Grid**, v. 12, n. 3, p. 2378–2388, 2021.

KHODADADI ARPANAHI, M. et al. An Augmented Prony Method for Power System Oscillation Analysis Using Synchrophasor Data. **Energies**, v. 12, n. 7, p. 1267, 2 abr. 2019.

KOSTEREV, D. N.; TAYLOR, C. W. **WSCC System OutageIEEE Transactions on Power Systems.** [s.l: s.n.].

KUMAR, D. Fault Classification Comparison in Power System Using Nearest Neighbor and Decision Tree. **Lecture Notes in Electrical Engineering**, v. 693, p. 581–589, 2021.

KUMAR, S.; SONI, M. K.; JAIN, D. K. Monitoring of Wide Area Power System Network with Phasor Data Concentrator (PDC). **International Journal of Information Engineering and Electronic Business**, v. 7, n. 5, p. 20–26, 8 set. 2015.

KUNDUR, P. Power System Stability and Control. 1994.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **Nature**, v. 521, n. 7553, p. 436–444, 28 maio 2015.

LI, W.; WANG, M. Identifying Overlapping Successive Events Using a Shallow Convolutional Neural Network. **IEEE Transactions on Power Systems**, v. 34, n. 6, p. 4762–4772, 2019.

LI, W.; WANG, M.; CHOW, J. H. Real-Time Event Identification Through Low-Dimensional Subspace Characterization of High-Dimensional Synchrophasor Data. **IEEE Transactions on Power Systems**, v. 33, n. 5, p. 4937–4947, 2018.

LIANG, X.; WALLACE, S. A.; NGUYEN, D. Rule-Based Data-Driven Analytics for Wide-Area Fault Detection Using Synchrophasor Data. **IEEE Transactions on Industry Applications**, v. 53, n. 3, p. 1789–1798, 2017.

LIN, T. C. et al. Development of a Transmission Network Fault Location Platform Based on Cloud Computing and Synchrophasors. **IEEE Transactions on Power Delivery**, v. 35, n. 1, p. 84–94, 2020.

LIU, X. et al. Principal Component Analysis of Wide-Area Phasor Measurements for Islanding Detection - A Geometric View. **IEEE Transactions on Power Delivery**, v. 30, n. 2, p. 976–985, 2015.

MA, D. et al. A Hierarchical Event Detection Method Based on Spectral Theory of Multidimensional Matrix for Power System. **IEEE Transactions on Systems, Man, and Cybernetics: Systems**, v. 51, n. 4, p. 2173–2186, 2021.

MARTIN, K. E. et al. An overview of the IEEE standard C37.118.2 - Synchrophasor data transfer for power systems. **IEEE Transactions on Smart Grid**, v. 5, n. 4, p. 1980–1984, 2014.

MCBRIDE TW EDGAR LR O, H. J.; JOHNSON, N. J. **Securing Wide Area Measurement Systems**. [s.l: s.n.].

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. **The Bulletin of Mathematical Biophysics**, v. 5, n. 4, p. 115–133, dez. 1943.

PHADKE, A. G. **Synchronized phasor measurements-a historical overview**. IEEE/PES Transmission and Distribution Conference and Exhibition. **Anais...IEEE**, [s.d.].

PHADKE, A. G.; THORP, J. S. **Synchronized Phasor Measurements and Their Applications**. Boston, MA: Springer US, 2008.

RAZA, A. et al. A review of fault diagnosing methods in power transmission systems. **Applied Sciences (Switzerland)**, v. 10, n. 4, 2020.

ROSENBLATT, F. **The Perceptron - A perceiving and recognizing automaton**. Buffalo, New York, United States: [s.n.].

ROSENBLATT, F. **Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms**. Washington, DC: Spartan Books, 1962.

SAUER, P. W.; PAI, M. A. **Power System Dynamics and Stability**. [s.l: s.n.].

SCHER, S.; TRÜGLER, A. Robustness of Machine Learning Models Beyond Adversarial Attacks. 21 abr. 2022.

SHADI, M. R.; AMELI, M. T.; AZAD, S. A real-time hierarchical framework for fault detection, classification, and location in power systems using PMUs data and deep learning. **International Journal of Electrical Power and Energy Systems**, v. 134, 2022.

SHENKMAN, A. L. **Transient analysis of electric power circuits handbook**. [s.l.] Springer, 2005.

SRIKANTH, P.; KOLEY, C. A novel three-dimensional deep learning algorithm for classification of power system faults. **Computers and Electrical Engineering**, v. 91, 2021.

STEHMAN, S. V. Selecting and interpreting measures of thematic classification accuracy. **Remote Sensing of Environment**, v. 62, n. 1, p. 77–89, out. 1997.

TAYLOR, C. W. et al. WACS-Wide-Area Stability and Voltage Control System: R&D and Online Demonstration. **Proceedings of the IEEE**, v. 93, n. 5, p. 892–906, maio 2005.

THARWAT, A. Classification assessment methods. **Applied Computing and Informatics**, v. 17, n. 1, p. 168–192, 4 jan. 2021.

WANG, S.; DEHGHANIAN, P.; LI, L. Power grid online surveillance through PMU-embedded convolutional neural networks. **IEEE Transactions on Industry Applications**, v. 56, n. 2, p. 1146–1155, 2020.

WOLPERT, D. H.; MACREADY, W. G. No free lunch theorems for optimization. **IEEE Transactions on Evolutionary Computation**, v. 1, n. 1, p. 67–82, abr. 1997.

XIE, L.; CHEN, Y.; KUMAR, P. R. Dimensionality reduction of synchrophasor data for early event detection: Linearized analysis. **IEEE Transactions on Power Systems**, v. 29, n. 6, p. 2784–2794, 2014.

XIONG, Y. et al. A data-driven approach for fault time determination and fault area location using random matrix theory. **International Journal of Electrical Power and Energy Systems**, v. 116, 2020.

YANG, F. et al. Brown Measure Based Spectral Distribution Analysis for Spatial-Temporal Localization of Cascading Events in Power Grids. **IEEE Transactions on Smart Grid**, v. 12, n. 2, p. 1805–1820, 2021.

ANNEX A – SALTO CAXIAS SYSTEM SPECIFICATIONS

Table 11 – Required variables for the PST bus model.

Symbol	Variable	Unit
n_b	Bus number	-
v_{bus}	Voltage magnitude at operating point	pu
θ_{bus}	Voltage angle at operating point	degrees
P_g	Active power generation	pu
Q_g	Reactive power generation	pu
P_{load}	Active power load	pu
Q_{load}	Reactive power load	pu
G_{sh}	Shunt conductance	pu
B_{sh}	Shunt susceptance	pu
b_t	Bus type ($PV, PQ, V\theta$)	-
Q_{gmax}	Maximum reactive power generation	pu
Q_{gmin}	Minimum reactive power generation	pu
V_r	Area nominal voltage	kV
v_{max}	Maximum voltage magnitude	pu
v_{min}	Minimum voltage magnitude	pu

SOURCE: Power System Toolbox Version 3.0 Manual

Table 12 – Required variables for the PST transmission line and transformer models.

Symbol	Variable	Unit
n_f	“From” bus number	-
n_t	“To” bus number	-
r_l	Line resistance	pu
x_l	Line reactance	pu
b_l	Line susceptance	pu
ρ	Tap ratio	%
ρ_θ	Tap phase	rad
ρ_{max}	Maximum tap ratio	%
ρ_{min}	Minimum tap ratio	%
ρ_s	Tap size	%

SOURCE: Power System Toolbox Version 3.0 Manual

Table 13 – Required variables for the PST generator model with Salto Caxias system values.

Symbol	Variable	Value	Unit
m	Machine number	1	-
n_m	Machine bus number	1	-
S_{base}	Power base	345	MVa
x_l	Leakage reactance	0.1635	pu
r_a	Armature resistance	0.0000	pu
x_d	Direct axis synchronous reactance	0.9300	pu
x'_d	Direct axis transient reactance	0.3000	pu
x''_d	Direct axis subtransient reactance	0.2300	pu
T'_{do}	Direct axis open-circuit time constant	7.6100	seconds
T''_{do}	Direct axis open-circuit subtransient time constant	0.0600	seconds
x_q	Quadrature axis synchronous reactance	0.6300	pu
x'_q	Quadrature axis transient reactance	0.6300	pu
x''_q	Quadrature axis subtransient reactance	0.0000	pu
T'_{qo}	Quadrature axis open-circuit time constant	0.1700	seconds
T''_{qo}	Quadrature axis open-circuit subtransient time constant	0.0170	seconds
H	Inertia constant	3.6300	seconds
d_o	Local damping coefficient	0	pu
d_1	System damping coefficient	0	pu
S_1	Saturation curve parameter 1	0.2525	-
S_2	Saturation curve parameter 2	0.5317	-
g_{type}	Machine type	21	-

SOURCE: The author (2022)

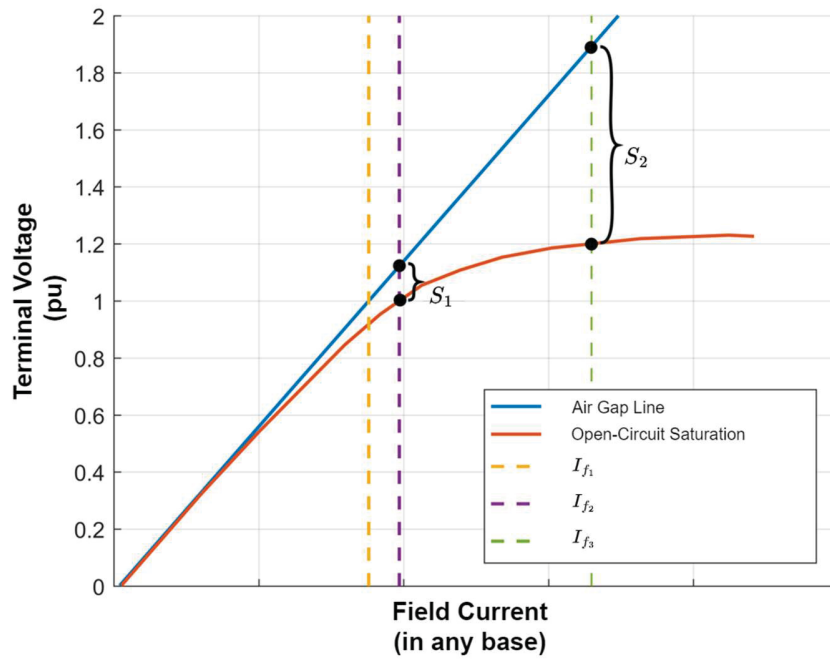
For g_{type} , 21 indicates a model of a subtransient salient pole machine – which is appropriate to represent the generation unit located within the Salxo Caxias system, in the Governador José Richa hydropower plant.

As for saturation curve parameters S_1 and S_2 , COPEL's documentation provided them in the format of exponential curve parameters, described by Equation 13.

$$y(x) = A \cdot e^{B(x-C)} \quad (13)$$

These parameters refer to the distance between the air gap line and open-circuit saturation at two distinct points: when the field current achieves a voltage of $V_1 = 1$ pu and $V_2 = 1.2$ pu in open-circuit saturation (FITZGERALD; KINGSLEY; UMANS, 2003). Figure 44 illustrates this concept.

Figure 44 – Air gap and open-circuit saturation curves for a given machine.



SOURCE: The author (2022).

Equations 14 and 15 describe the relationship between the parameters in Equation 13 and S_1 and S_2 , considering $A = 0.474867$, $B = 3.7236$ and $C = 0.8$.

$$B = \frac{\ln\left(\frac{S_1}{S_2}\right)}{V_1 - V_2} \quad (14)$$

$$A = \frac{S_1}{e^{B(V_1 - C)}} \quad (15)$$

Plugging in the variables and solving for S_1 and S_2 yields the values in Table 13.

Table 14 – Required variables for the PST ST5b exciter model with Salto Caxias system values.

Symbol	Variable	Value	Unit
m	Machine number	1	-
R_c	Compensation resistance	0	pu (generator)
X_c	Compensation reactance	0	pu (generator)
T_r	Transducer time constant	0	seconds
K_a	Voltage regulator gain	500	-
T_a	Voltage regulator time constant	0.01	seconds
$[T_b, T_{b_1}]$	Lag time constants	[21.429, 0.1]	seconds
$[T_c, T_{c_1}]$	Lead time constants	[3, 0.1]	seconds
V_{Rmax}	Maximum voltage regulator output	9.3636	pu
V_{Rmin}	Minimum voltage regulator output	-8.234	pu
K_c	Rectifier regulation factor	0.01	-

SOURCE: The author (2022).

Table 15 – Required variables for the PSS model with Salto Caxias system values.

Symbol	Variable	Value	Unit
τ_{PSS}	1. Speed input 2. Bus frequency input 3. Power input 4. dpwf with speed input 5. dpwf with bus frequency input	1	-
m	Generator number	1	-
G_{PSS}	PSS gain	30	-
T_W	PSS washout time constant	5	seconds
T_{n1}	PSS lead time first constant	0.3	seconds
T_{n2}	PSS lead time second constant	0.33	seconds
T_{d1}	PSS lag first time constant	0.03	seconds
T_{d2}	PSS lag second time constant	0.2	seconds
y_{max}	PSS maximum output	0.1	pu
y_{min}	PSS minimum output	-0.1	pu

SOURCE: The author (2022).

Table 16 – Required variables for a hydro turbine governor with Salto Caxias system values.

Symbol	Variable	Value	Unit
m	Machine number	1	-
d_p	Permanent droop	5	%
d_t	Temporary droop	4	%
T_r	Reset time	5	s
T_p	Pilot servo time constant	0.1	s
K_s	Servo gain	0.3	-
T_g	Gate servo time constant	0.15	s
g_{max}	Maximum rate of change of gate position	0.01	pu/s
g_{min}	Minimum rate of change of gate position	-0.01	pu/s
DB	Dead Band	0	-
P_t	Turbine rating	345	MW
T_{wr}	Rated water time constant	2	s
p_0	No-load gate position	0.06	pu
p_{total}	Full-load gate position	0.94	pu
h	Initial head	1	pu

SOURCE: The author (2022).

Tables 17 and 18 illustrate the actual values for the bus, transmission line, and transformer specifications of the model respectively.

Table 17 – Bus data specifications for the Salto Caxias system PST model.

Number (Model)	Number (ONS)	Name (ONS)	Type	Voltage		Generation		Load		Voltage Base (kV)
				Magnitude	Angle	Active [MW]	Reactive [Mvar]	Active [MW]	Reactive [Mvar]	
Area 1 - PR 525 kV										
1	808	GJRICUHE002	$V\theta$	0.969	0	303	-185.4	0	0	992
2	896	CASCOE-PR525	PQ	1,021	-3.7	0	0	0	0	525
3	897	SSCAXIA-PR525	PQ	1,020	-2.4	0	0	0	0	525
4	1060	SSANTI-PR525	PQ	1,027	-2.5	0	0	0	0	525
5	43265	CVO--A-PR000	PQ	1,021	-4.5	0	0	0	0	991
6	43266	CVO--B-PR000	PQ	1,021	-4.5	0	0	0	0	991
7	43268	CVO--C-PR000	PQ	1,021	-4.4	0	0	0	0	991
Area 2 - PR 230 kV										
8	839	CASCAV-PR230	PQ	1,019	-4.7	0	0	0	0	230
9	843	FIGUAN-PR230	PQ	0.990	-8.7	0	0	0	0	230
10	978	GUAIRA-PR230	PQ	1,038	-5.9	0	0	0	0	230
11	1047	SOSORI-PR230	PQ	1,024	-4.8	0	0	0	0	230
12	2458	CASCOE-PR230	PQ	1,021	-4.4	0	0	0	0	230
13	10005	MEDNOR-PR230	PQ	1,000	-7.5	0	0	0	0	230
14	43267	CASC-A-PR000	PQ	1,011	-5.9	0	0	0	0	991
15	43268	CASC-B-PR000	PQ	1,010	-6.0	0	0	0	0	991
16	43269	CASC-C-PR000	PQ	1,009	-6.2	0	0	0	0	991
17	43278	GUA--A-PR000	PQ	1,033	-6.6	0	0	0	0	991
18	43310	FIN--A-PR000	PQ	0.977	-10.8	0	0	0	0	991
19	43313	GUA--B-PR000	PQ	1,034	-6.5	0	0	0	0	991
20	43320	FIN--B-PR000	PQ	0.977	-10.8	0	0	0	0	991
21	43342	MED--B-PR000	PQ	0.990	-9.0	0	0	0	0	991
22	43356	MED--A-PR000	PQ	0.991	-8.8	0	0	0	0	991
Area 3 - PR DIT/DIST										
23	840	CASCAV-PR138	PQ	1,010	-5.9	0	0	100.0	32.9	138
24	847	GUAIRA-PR138	PQ	1,034	-6.5	0	0	35.0	11.5	138
25	2447	FIGUAN-PR138	PQ	0.980	-10.3	0	0	85.0	27.9	138
26	10253	MEDNOR-PR138	PQ	0.990	-9.0	0	0	80.0	26.3	138
Area 4 - MS 230 kV										
27	1086	DOURAD-MS230	PQ	1,029	-5.9	0	0	0	0	230

SOURCE: The author (2022).

A shunt capacitance of -27 MVar in bus 27 must be considered as well.

Table 18 – Transmission line and transformer data specifications for the Salto Caxias system PST model.

FROM Bus		TO Bus		Resistance (%)	Reactance (%)	Susceptance (MVar)	Tap Ratio	Minimum Tap Ratio	Maximum Tap Ratio
Number (Model)	Name (ONS)	Number (Model)	Name (ONS)						
3	SCAXIA-PR525	1	GJRICUHE002	0.000	1,360	0	1,024	-	-
12	CASCOE-PR230	5	CVO--A-PR000	0.010	-0.120	0	1,000	0.900	1,100
2	CASCOE-PR525	5	CVO--A-PR000	0.010	1,390	0	1,000	-	-
23	CASCAV-PR138	14	CASC-A-PR000	0.160	0.330	0	1,000	-	-
8	CASCAV-PR230	14	CASC-A-PR000	0.280	6,310	0	1,000	0.869	1,043
24	GUAIRA-PR138	17	GUA--A-PR000	0.040	-1.190	0	1,000	-	-
10	GUAIRA-PR230	17	GUA--A-PR000	0.000	7,550	0	1,000	0.869	1,043
26	MEDNOR-PR138	22	MED--A-PR000	0.160	0.810	0	1,000	-	-
13	MEDNOR-PR230	22	MED--A-PR000	0.010	5,860	0	1,000	0.869	1,043
9	FIGUAN-PR230	18	FIN--A-PR000	0.165	8,267	0	1,000	-	-
25	FIGUAN-PR138	18	FIN--A-PR000	0.005	-1.733	0	1,000	0.900	1,100
12	CASCOE-PR230	6	CVO--B-PR000	0.000	-0.130	0	1,000	0.900	1,100
2	CASCOE-PR525	6	CVO--B-PR000	0.010	1,360	0	1,000	-	-
23	CASCAV-PR138	15	CASC-B-PR000	0.080	-0.170	0	1,000	-	-
8	CASCAV-PR230	16	CASC-C-PR000	0.100	9,030	0	1,000	0.869	1,043
25	FIGUAN-PR138	20	FIN--B-PR000	0.005	-1.704	0	1,000	0.900	1,100
24	GUAIRA-PR138	19	GUA--B-PR000	0.100	0.200	0	1,000	-	-
9	FIGUAN-PR230	20	FIN--B-PR000	0.162	8,131	0	1,000	-	-
8	CASCAV-PR230	15	CASC-B-PR000	0.070	6,460	0	1,000	0.869	1,043
26	MEDNOR-PR138	21	MED--B-PR000	0.130	0.010	0	1,000	-	-
13	MEDNOR-PR230	21	MED--B-PR000	0.090	6,650	0	1,000	0.869	1,043
12	CASCOE-PR230	7	CVO--C-PR000	0.010	-0.030	0	1,000	0.900	1,100
2	CASCOE-PR525	7	CVO--C-PR000	0.010	1,300	0	1,000	-	-
10	GUAIRA-PR230	19	GUA--B-PR000	0.000	6,310	0	1,000	0.869	1,043
23	CASCAV-PR138	16	CASC-C-PR000	0.260	-1.700	0	1,000	-	-
2	CASCOE-PR525	3	SSCAXIA-PR525	0.050	0.740	79,518	0	0	0
12	CASCOE-PR230	8	CASCAV-PR230	0.200	1,030	1,861	0	0	0
10	GUAIRA-PR230	12	CASCOE-PR230	0.610	7,650	34,460	0	0	0
13	MEDNOR-PR230	12	CASCOE-PR230	1,062	5,998	11,467	0	0	0
9	FIGUAN-PR230	13	MEDNOR-PR230	0.878	5,542	10,056	0	0	0
8	CASCAV-PR230	12	CASCOE-PR230	0.180	1,070	2,141	0	0	0
13	MEDNOR-PR230	8	CASCAV-PR230	1,133	7,055	12,846	0	0	0
9	FIGUAN-PR230	13	MEDNOR-PR230	0.892	5,033	9,610	0	0	0
8	CASCAV-PR230	11	SOSORI-PR230	1,220	7,690	13,810	0	0	0
10	GUAIRA-PR230	27	DOURAD-MS230	2,772	18,924	45,385	0	0	0
3	SSCAXIA-PR525	4	SSANTI-PR525	0.080	1,170	124,580	0	0	0
8	CASCAV-PR230	12	CASCOE-PR230	0.170	1,030	2,053	0	0	0

SOURCE: The author (2022).