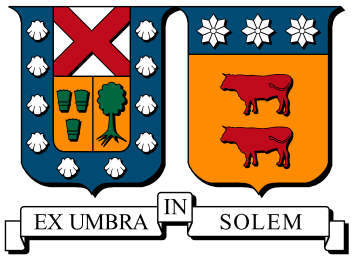
UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA

DEPARTAMENTO DE ELECTRÓNICA
VALPARAÍSO – CHILE



CLASIFICACIÓN DE ESCENARIOS DE OPERACIÓN DE LA RED ELÉCTRICA

JOSÉ MIGUEL BENAVENTE SILVA
ANGEL ANTONIO GUTIÉRREZ GAETE

MEMORIA DE TITULACIÓN PARA OPTAR AL TÍTULO DE INGENIERO CIVIL TELEMÁTICO

PROFESOR GUÍA: MAURICIO ARAYA
PROFESOR CORREFERENTE: MOHAMED ABDELHAMID

DICIEMBRE 2022

Agradecimientos Angel Gutiérrez

El camino de la formación no es fácil, y a lo largo de este aprendí que hubiera sido imposible realizarlo de manera individual , Quisiera agradecer a todos aquellos que me brindaron apoyo cuando fue necesario.

A mis amigos de la universidad y de afuera, por actuar siempre con solidaridad y empatía, a mis tíos *Vero* y *Ale* por enseñarme a creer en mi, a mi pareja *Natalia* por siempre encontrar la manera de apoyarme, y a mi padre y madre por hacerme la persona que soy hoy, por asegurarse de que nunca me faltase nada y por su amor incondicional.

Agradecimientos José Benavente

Al finalizar esta etapa de mi vida, solo queda agradecer a las personas que me acompañaron durante este proceso: al equipo de memoria que hizo posible todo este trabajo, a mis amigos y amigas que hicieron más linda mi experiencia universitaria. A mi pareja que me acompañó y apoyó desde el inicio de la carrera, compartiendo mis sufrimientos y alegrías en todo momento. A mis padres por darme soporte y la posibilidad de obtener la mejor educación posible. Finalmente a mis hermanas por todo el apoyo y ánimo que me han entregado durante toda mi vida.

Resumen

Los estados de la red eléctrica son datos constantemente analizados por parte de las empresas proveedoras de este tipo de servicio debido a que entrega relevante información que permite realizar tomas de decisiones logísticas mas precavidas. La empresa Chilquinta Energía S.A participó en el programa de Memorias Multidisciplinarias de la Universidad Técnica Federico Santa María para cambiar sus procesos de limpieza y clasificación manual de los datos. Estos procedimientos exponen un problema en los procesos de clasificación de los distintos estados de la red eléctrica de Valparaíso, ya que podrían encaminar a una mala resolución con consecuencias monetarias y sociales debido a que afectan la determinación de inversiones económicas por parte de la empresa a sectores mas vulnerables en cuanto disponibilidad y confiabilidad del servicio eléctrico.

El objetivo de esta memoria es lograr la automatización de este proceso con el fin de mejorar los tiempos trabajo y la toma de decisiones. Para ello, se propone una solución mediante el uso de inteligencia artificial, a través de redes neuronales capaces de extraer información y producir conclusiones a partir de los datos. El propósito es lograr un reconocimiento automático del estado de la red a través del aprendizaje de máquinas.

A partir de los resultados de los experimentos, se concluye que la información del conjunto de datos usado para el entrenamiento de los modelos de inteligencia artificial supervisados, no es confiable debido al voluble proceso de notificación de la existencia de una falla en la red eléctrica. Consecuentemente, se presenta un documento a la empresa Chilquinta Energía S.A, con una serie de recomendaciones que tienen como objetivo mejorar el flujo del registro de datos, para generar a futuro un mejor conjunto de datos que permita llegar a la solución automatizada de la clasificación de los estados de al red a través de la inteligencia artificial.

Abstract

The states of the electrical network are data constantly analyzed by the companies that provide this type of service because it provides relevant information that allows more cautious logistical decisions to be made. The company Chilquinta Energía S.A. participated in the Multidisciplinary Memories program of the Universidad Técnica Federico Santa María to change its data cleaning and manual classification processes. These procedures expose a problem in the classification processes of the different states of the electrical network of Valparaíso, since they could lead to a bad resolution with monetary and social consequences because they affect the determination of economic investments by the company to more vulnerable sectors in terms of availability and reliability of the electrical service.

The objective of this report is to achieve the automation of this process in order to improve work times and decision making. For this purpose, a solution is proposed through the use of artificial intelligence, by means of neural networks capable of extracting information and producing conclusions from the data. The purpose is to achieve an automatic recognition of the state of the network through machine learning.

From the results of the experiments, it is concluded that the information of the data set used for training the supervised artificial intelligence models is not reliable due to the fickle process of notification of the existence of a fault in the power grid. Consequently, a document is presented to the company Chilquinta Energía S.A., with a series of recommendations aimed at improving the data recording flow, in order to generate in the future a better dataset that allows reaching an automated solution for the classification of the network states through artificial intelligence.

Tabla de Contenidos

Lista de Tablas

Lista de Figuras

Capítulo 1

Introducción

Este trabajo se desarrolla dentro de la empresa Chilquinta Energía S.A. cuya especialización se centra en la distribución, transmisión y generación de energía eléctrica en su área de concesión, la cual comprende 11.496 km^2 de la quinta región de Chile, abasteciendo mayoritariamente con sus servicios a las provincias de Valparaíso, Marga Marga, Quillota, San Felipe, Los Andes y San Antonio [1].

Esta empresa esta formada por distintas áreas de trabajo, una de ellas es el área de planificación y expansión de la red eléctrica, con cual se trabajó directamente. Esta área se encarga de cumplir principalmente labores de análisis estratégico de mediano y largo plazo de la expansión de las redes de transmisión y distribución, estudios de alto impacto para grandes bloques de demanda o generación y proyecciones de demanda. Entre otras labores se encuentra la priorización y control del presupuesto de la compañía, cálculo de pérdida de energía y estudios generales.

Actualmente, el proceso que se realiza para la detección del estado de operación de la red eléctrica no se encuentra automatizado. Esta es una tarea que se realiza manualmente y no es precisa en términos cuantitativos, lo que afecta a diversas labores, entre ellas la proyección de demanda, donde se necesita identificar las fallas de las lecturas de los transformadores para que estas no sean consideradas en tal tarea.

1.1. Problemática planteada en las memorias multidisciplinarias

El área de "planificación y expansión de sistema" de la empresa Chilquinta, es la encargada de realizar una serie de labores, que tienen un alto impacto económico para la empresa ya que esta área es la encargada de tomar las decisiones de realizar ciertas acciones con respecto al resultado de sus estudios, por lo que la implementación de un sistema de automatización inteligente para el análisis de datos es fundamental para optimizar la toma de decisiones.

Un **alimentador** es un dispositivo conectado a una estación receptora y es el encargado de proveer suministro eléctrico por medio de las subestaciones. Actualmente Chilquinta obtiene datos de sus alimentadores cada 15 minutos con los valores de kW entregados, kW recibidos, kVAR entregado, kVAR recibido y tensión. Estos datos se almacenan en una estructura a la cual llamaremos **lectura** como se puede ver en la Figura 1.1 . El método que se utiliza a día de hoy para clasificar los escenarios de operación de la red eléctrica consiste en buscar valores anómalos dentro de las lecturas. Este proceso se realiza de forma manual y estimativa utilizando filtros de datos en Excel y el criterio del empleado a cargo de dicha operación. Los escenarios de operación de red asignados a los datos detectados como anómalos dentro de las lecturas pueden representarse como:

- **Respaldo:** Ocurre cuando otro alimentador está transfiriendo energía a otro mediante una operación de equipo.
- **Falla:** Puede significar que hay un desprendimiento de carga o un aumento de carga, ya que está haciendo respaldo a otro sector de la ciudad, en otras palabras se verá reflejado significativamente en la gráfica adoptando bruscamente el valor cero.

Ejemplos visuales para ambos casos se pueden encontrar en la Figura 1.1.

Además de los estados mencionados también se tiene en consideración que un dato pertenece a la categoría de "Operación normal" cuando no pertenece a los estados de **Respaldo** o **Falla**.

A raíz del tiempo limitado para la realización de esta memoria, y tras conversaciones

con la empresa se determino que el desarrollo estará enfocado en las **Fallas**, es decir determinar cuando un dato pertenece o no a este estado.

Además de las lecturas existe un documento que contiene los detalles de cuando ocurrió una falla o respaldo, este documento se llama **libro de fallas** y es gestionado por el área de operaciones.

Es importante reconocer los distintos estados de operación de la red eléctrica de manera automática ya que esto permitiría al área de planificación obtener, de manera efectiva, un ágil conocimiento de la situación actual de la red para realizar sus labores y toma de decisiones con un mejor juicio. Dicho esto se define el principal objetivo del desafío como la creación de un sistema inteligente capaz de clasificar automáticamente los distintos estados de operación de la red eléctrica. Este objetivo será explicado con mayor detalle en el punto 1.2.

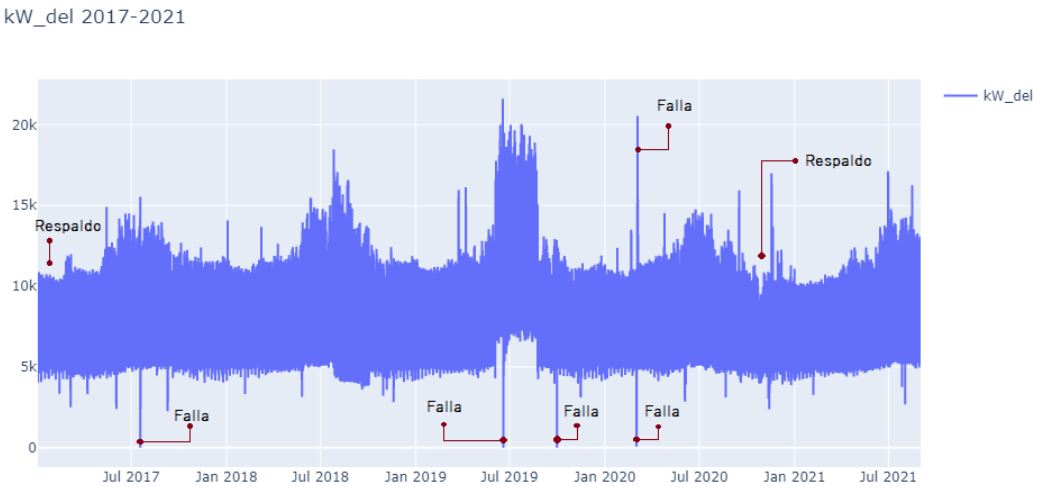


Figura 1.1: KW entregado por Transformador 2 de Miraflores, Viña del Mar.
Fuente: Elaboración propia

1.2. Objetivos de la Memoria

1.2.1. Objetivo General

Diseño de un modelo de datos y programación de un sistema que clasifique las lecturas de voltajes como fallas o no fallas dentro de la red eléctrica de los transformadores conectados a la subestación de Miraflores. Se busca automatizar el proceso actual utilizado por la empresa para obtener resultados más precisos acompañados de una mejor toma de decisiones.

1.2.2. Objetivos específicos

- **Investigación de técnicas de aprendizaje:** Exploración de técnicas y soluciones del estado del arte que se aplican actualmente para resolver problemas relacionados a clasificación, detección y pronóstico de estados.
- **Creación de modelo de clasificación:** Construir un modelo de inteligencia artificial que detecte y clasifique estados para series de tiempo.
- **Evaluación de modelo de clasificación:** Evaluar los diferentes modelos clasificatorios de machine learning mediante distintas métricas de desempeño.
- **Creación de documento de recomendaciones:** A través del análisis de los resultados de los modelos clasificatorios, se dispone a encontrar un margen de mejora dentro del flujo de datos del proceso de data mining que realiza la empresa, por lo que se genera un documento de recomendaciones para mejorar la calidad y confiabilidad de los datos.

1.3. Equipo de desarrollo

Dentro del contexto de memorias multidisciplinarias se formó un equipo, el cual esta compuesto por:

- José Benavente: Estudiante de Ingeniería Civil Telemática, encargado del desarrollo del área de Machine Learning.
- Angel Gutiérrez: Estudiante de Ingeniería Civil Telemática, encargado del desarrollo de selección de data, limpieza de los datos, y preprocesamiento.
- Omar Herrera: Estudiante de Ingeniería Civil Matemática, encargado del desarrollo de un algoritmo matemático, sin uso de redes neuronales, para la detección de fallas en el sistema eléctrico.
- Eduardo Díaz: Estudiante de Ingeniería Civil Industrial, encargado del desarrollo de un plan CAPEX para el proyecto.

Esta memoria esta enfocada a el trabajo relacionado a los procesos de selección de data, limpieza de los datos, preprocesamiento, ingeniería de características y machine learning. Es decir el desarrollo de ambos estudiantes de Ingeniería Civil Telemática.

1.4. Estructura del documento

A lo largo de este documento se presenta una propuesta de cómo abordar el problema de clasificación de estados de la red eléctrica de Valparaíso de la empresa Chilquinta Energía S.A. Esto es puesto en contexto en el Capitulo 2, en donde se explica como se enfrenta este tema en la actualidad y cuál es la tendencia que existe respecto a la implementación de este tipo de tecnologías en trabajos relacionados. En el Capítulo 1 se describe en detalle la definición del problema y los objetivos de esta memoria, los cuales tienen la finalidad de mejorar el proceso actual que realiza la empresa en cuanto a tiempo y toma de decisiones con ayuda de herramientas relacionadas con el uso de inteligencia artificial. El Capitulo 3 explica en detalle los modelos clasificadores utilizados durante el desarrollo de la investigación y las métricas de desempeño empleadas para evaluar los rendimientos de cada modelo de manera individual. En el Capitulo 4 se presenta la propuesta de solución en donde se aplicarán los conceptos explicados en el marco teórico a lo largo del Capitulo 3 con las limitaciones definidas por las restricciones del problema. El análisis de estos resultados son explicados en el Capitulo 5 en donde se toma en consideración elementos externos de la investigación cuantitativa, como procesos internos dentro de la empresa. Finalmente el capítulo 6 se detalla la conclusión y además se elabora una serie de consideraciones que ayudarán a la empresa mejorar la calidad y confiabilidad de sus datos.

Capítulo 2

Estado del Arte

La prognosis se define como el conocimiento anticipado de algún suceso. Este concepto es comúnmente visto en machine learning y deep learning debido a que estos sub-campos de la inteligencia artificial se suelen utilizar para obtener predicciones en áreas como en medicina para la estimación de vida en personas con enfermedades terminales o en las industrias para determinar la vida útil de herramientas o maquinas.

En el capítulo anterior se ha descrito el marco teórico, detallando las métricas de desempeño, los modelos de machine learning y deep learning que serán usados en esta memoria. En este capítulo esta divido en dos partes, en la primera se estudia la detección de fallas en sistemas de distribución. En la segunda parte se exponen distintos casos de uso de prognosis en diversas áreas de estudio con la finalidad de obtener una percepción actualizada del uso de la inteligencia artificial para obtener un conocimiento anticipado de algún suceso en particular.

2.1. Detección de fallas en sistemas de distribución

Al momento de recopilar información, mas precisamente datasets relacionados a la detección de fallas en sistemas de distribución eléctrica se encontraron datasets como los vistos en [2] y en [3] donde, si bien, están relacionados al tema contienen información muy especifica como para ser utilizados por nosotros. En [2] se tiene información resumida de 16000 transformadores, sin embargo no se tiene la información detallada en una serie temporal de alguno de ellos. Mientras que en [3] los datos están enfocados a problemas en sistemas de transmisión en vez de distribución.

Un proceso importante dentro de la realización de esta memoria realizada para una empresa de suministro electrico como es Chilquinta Energia S.A es comprender como funciona la detección y el análisis de fallos en los sistemas de distribución. Los autores de [4] señalan que la detección y el análisis de los transformadores son medidas clave para mejorar la seguridad de los sistemas de energía y la fiabilidad del suministro eléctrico, y que debido a la complejidad de la estructura del transformador y a las variaciones en las condiciones de funcionamiento, la aparición de un fallo dentro del transformador de potencia es incierta y aleatoria. En dicho articulo se propone un nuevo método de análisis del árbol de fallos basado en la teoría de conjuntos difusos para transformadores de potencia. En [5] se presenta una arquitectura para un sistema automatizado que permite monitorizar y realizar un seguimiento en tiempo real (online) de la posible aparición de fallos y transitorios electromagnéticos observados en las redes de distribución y de distribución de energía primaria.

En cuanto a los sistemas de energía a gran escala, los autores de este trabajo [6] investigaron un algoritmo de detección y aislamiento de fallos de sensores en tiempo real con de sensores en tiempo real con implementación de hardware basada en el sistema estándar IEEE 14-bus. Los resultados indican que este método es efectivo y puede ser utilizado en aplicaciones reales para aplicación en el mundo real para el control resistente en el sistema de energía.En [7] se desarrolla una técnica de máquinas de vectores de soporte multiclase modificada (MMC-SVM) para detectar y clasificar simultáneamente diferentes tipos de fallos de circuito abierto en sistemas de distribución de energía. Esta técnica es capaz de detectar e identificar fallos de circuito abierto teniendo en cuenta el impacto de las variaciones en la tensión de los diferentes nodos en los sistemas de distribución de energía. La tensión RMS (Root Mean Square) de la red eléctrica se utiliza

como señal de entrada para diagnosticar los fallos.

Los autores de [8] se dedicaron a investigar el problema de la detección de fallas en líneas monofásicas a tierra en sistemas de potencia con conexión a tierra no directa. En este documento se formulan algoritmos de medición de fallos para la magnitud en estado transitorio, en estado estacionario y el ángulo de fase en estado estacionario, respectivamente, Además se introduce la teoría de la evidencia D-S para tratar la fusión de criterios múltiples.

A continuación se presenta una tabla resumen-comparativa entre los distintos documentos mencionados en esta sección y esta memoria.

Título	Ref	Tipo	Uso de IA	Funcionamiento
Fuzzy Set Theory and Fault Tree Analysis based Method Suitable for Fault Diagnosis of Power Transformer	[4]	Analisis de fallas	No	Analisis probabilístico
Fault Detection in Power Distribution Systems Using Automated Integration of Computational Intelligence Tools	[5]	Deteccion de fallas	No	Monitoreo en tiempo real
Power System with Hardware Implementation	[6]	Deteccion de fallas	No	Monitoreo en tiempo real
A Fast Fault Detection and Identification Approach in Power Distribution Systems	[7]	Deteccion de fallas	No	Monitoreo en tiempo real
Multi-criteria Relaying Strategy for Single Phase to Ground Fault in MV Power Systems	[8]	Detección e identificación de fallas	No	Monitoreo en tiempo real
Clasificacion de escenarios de operacion de la red electrica	**-**	**Detección de fallas**	**Si**	**Clasificación a partir de datos**

Tabla 2.1: Comparación entre distintos trabajos relacionados a la detección de fallas

2.2. Prognosis

2.2.1. Prognosis en el ámbito industrial

Para lograr un adecuado desarrollo de este proyecto se debe tener en cuenta el uso de prognosis en sectores industriales. En PHM (Prognostics Health Management o Pronósticos de la gestión de la Salud) es utilizada para la estimación de vida útil de un sistema en particular (RUL, por sus siglas en inglés) tomando en cuenta su trayectoria de degradación y el futuro plan de uso [9]. Desde una perspectiva práctica, es importante tener una estimación precisa de RUL, porque una predicción temprana puede dar como resultado un mantenimiento excesivo y una predicción tardía podría conducir a fallas catastróficas.

Por ejemplo en el campo de la energía eólica, esta problema de la formación de hielo en las cuchillas de las turbinas de viento. Actualmente, la data del funcionamiento en tiempo real de las turbinas son almacenados por el sistema SCADA (control de supervisión y adquisición de datos) [10], con la finalidad de activar alertas y apagar las turbinas en base a la comparación de la potencia real con la potencia teórica cuando hay una cierta desviación. A pesar de esto las cuchillas de la turbina de viento ya se han congelado cuando la alarma es activada. El mayor problema es que si la turbina eólica sigue funcionando en tales condiciones, el riesgo de rotura y daño de las cuchillas aumenta considerablemente. La dificultad se encuentra en encontrar una relación explícita entre los datos recompilados y el proceso temprano del problema de formación de hielo en las cuchillas.

En la mayoría de este tipo de problemas, relacionados a la industria, los datos son series de tiempo con múltiples características obtenidas a través de múltiples sensores. Por esta misma razón es que se propone el uso de un método de extremo a extremo basado en una red CNN-LSTM, en donde la red neuronal convolucional extraerá automáticamente las características del dataset y la red neuronal recurrente LSTM se encargará de analizar la nueva secuencia de características entregadas por la CNN. Los autores realizaron una comparación con otros clasificadores de machine learning en distintos tipos de arquitecturas, en donde el mejor desempeño fue obtenido por el modelo propuesto, obteniendo los valores mas bajos de RMSE y mas altos de Accuracy y Score [11].

2.2.2. Prognosis en salud

El área de la salud ha buscado ayudarse de la tecnología para intentar pronosticar distintos tipos de enfermedades, en el documento [12] se propone un modelo de deep learning híbrido basado en LSTM y CNN para el pronóstico automático y preciso de las arritmias cardíacas a partir de big data. Se utilizó un conjunto de datos de 123998 ECG de una combinación de conjuntos de datos de "MIT-BIH Arrhytmias database" y "PTB diagnosis database". Tras medir el rendimiento del modelo utilizando seis métricas distintas y compararlo con otras técnicas actuales se calculó que el porcentaje de precisión global y media es del 99 % y 99.7 %

Análogamente en [13] se habla de una de las principales enfermedades que causan discapacidad en el mundo, el ictus. Los autores desarrollaron un modelo de prognosis basado en CNN para predecir los resultados de la recuperación de once pacientes con ictus tras un entrenamiento de rehabilitación utilizando una interacción cerebro-ordenador(BCI). El rendimiento del modelo se evaluó mediante la validación cruzada con exclusión de la información. En general, el modelo propuesto predijo la recuperación de los pacientes con un R2 de 0,98 y un MSE de 0,89. Otro estudio se encarga de analizar imágenes de la lengua humana para identificar condiciones de los órganos internos del cuerpo humano. En este articulo [14] se utiliza el clasificador SVM y CNN para clasificación de imágenes de la lengua con el fin de pronosticar diabetes de tipo 2. Las tres características cuantitativas correspondientes a la geometría, el color y la textura se miden utilizando MATLAB, luego el clasificador SVM y CNN se utilizan para las imágenes de clasificación del conjunto de datos.

En el estudio [15] se trata el tema del cáncer de mama, el cual es un tipo de enfermedad en la cual urge la capacidad de predicción de pronóstico. En los últimos años, se han puesto a disposición conjuntos de datos multimodales sobre el cáncer (expresión génica, alteración del número de copias y clínica). Motivados por la mejora de los modelos basados en el aprendizaje profundo, los autores propusieron utilizar algunos modelos predictivos basados en el aprendizaje profundo para mejorar la predicción del pronóstico del cáncer de mama a partir de conjuntos de datos multimodales disponibles.

Por ultimo en [16] se emplea el reconocimiento de Actividad Humana(HAR) para el pronóstico de la depresión, uno los principales trastornos mentales, usando LSTM. Los datos son registrados desde los sensores de smartphones y se usan para el reconocimiento

de 13 actividades humanas como caminar, subir escaleras, sentarse, comer, etc. Luego, para el calculo de factor de riesgo de la depresión se utiliza una red neuronal LSTM. Para medir el rendimiento del método se eligieron cinco sujetos, dos de ellos deprimidos y tres sanos. A partir de las actividades diarias de los sujetos deprimidos se obtuvo un factor de riesgo de 67.44 % y 74.92 %. Mientras que los sujetos sanos obtuvieron un 29,86 %, 27,91 % y 29,87 % de factor de riesgo.

Como se ha podido analizar tras los diferentes estudios mencionados anteriormente, para las aplicaciones relacionadas a la detección de diferentes estados o fallas se han utilizado tecnologías relacionadas como CNN o LSTM en deep learning y otros modelos clásicos de machine learning. Por esta razón se ha tomado la decisión de utilizar estos modelos mencionados en las investigaciones, debido a su gran éxito para llegar a resultados concluyentes.

2.2.3. Prognosis en redes Eléctricas

Los sistemas de generación y distribución a menudo están expuestos a fallas y errores. Dentro de los más comunes se encuentran el fallo de los componentes del sistema, el error humano y el envejecimiento de los equipos. La ocurrencia de estos eventos afecta negativamente a la fiabilidad del sistema eléctrico e implica costos de reparación, disminución en la productividad y clientes insatisfechos. Dado que los fallos son imprevisibles, es necesario localizar y aislar rápidamente los fallos para minimizar su impacto en los sistemas de distribución [17]. Por ello, los investigadores han desarrollado métodos para localizar y detectar estos fallos en los sistemas de generación y distribución.

En [18] se menciona un esquema inteligente de pronostico de congestión de lineas basado en mediciones de amplia área para identificar precisamente una congestión y el causante de esta. En este trabajo se utilizaron métodos de redes neuronales debido a su excelente rendimiento en la predicción del comportamiento no lineal del sistema eléctrico.

En [19] Se propone un método basado en datos para la predicción del riesgo de fallos en la red de distribución. En primer lugar, se lleva a cabo un análisis de datos para determinar la clasificación objetivo y el conjunto de características de correlación de los fallos de la red de distribución. Se presenta el algoritmo Adaboost-SVM para pronosticar los fallos de la red de distribución y excavar la relación entre el fallo y los factores de

influencia.

En [20] se revisa el estado del arte en la detección, localización y diagnóstico de fallos en los sistemas de interconexión de cables eléctricos (EWIS), incluyendo en la red eléctrica y en los vehículos y máquinas. La mayoría de los métodos de prueba eléctrica se basan en mediciones de corrientes y tensiones o en reflexiones de alta frecuencia a partir de discontinuidades de impedancia

En [21] Se propone un nuevo método de análisis del árbol de fallos basado en la teoría de conjuntos difusos para el transformador de potencia. Utilizando este método, el índice de la tasa de fallos puede convertirse en un número difuso de la tasa de fallos. El método de clasificación de expertos puede utilizarse para realizar la estimación de la probabilidad de fallo sin necesidad de toda la información estadística correspondiente. En el artículo se describen los detalles del diseño de los números difusos y también se ofrece un ejemplo de aplicación del método.

En [22] se presenta una arquitectura de un sistema automatizado que permite monitorear y rastrear en tiempo real la posible ocurrencia de fallas y transitorios electromagnéticos observados en las redes de distribución primaria de energía.

Capítulo 3

Marco Teórico

3.1. Smart grid

Smart Grid es una red en desarrollo de nuevas tecnologías, equipos y controles que trabajan juntos para responder inmediatamente a nuestra demanda de electricidad del siglo XXI. La Red Inteligente representa un avance sin precedentes oportunidad de llevar la industria de la energía a una nueva era de Confiabilidad, Disponibilidad y Eficiencia que contribuir a nuestra salud Económica y Ambiental [23].

Los beneficios asociados con smart grid incluyen:

- Transmisión de electricidad más eficiente.
- Restauración más rápida de la electricidad después de interrupciones en el suministro eléctrico.
- Reducción de los costos de operación y administración para los servicios públicos y menores costos de energía para los consumidores.
- Reducción del "peak"de demanda, lo que también ayudará a reducir las tarifas eléctricas.
- Mayor integración de sistemas de energía renovable a gran escala.

- Mejor integración de los sistemas de generación de energía del propietario del cliente, incluidos los sistemas de energía renovable.
- Mejor seguridad.

A continuación se presenta un diagrama con la arquitectura de Smart grid.

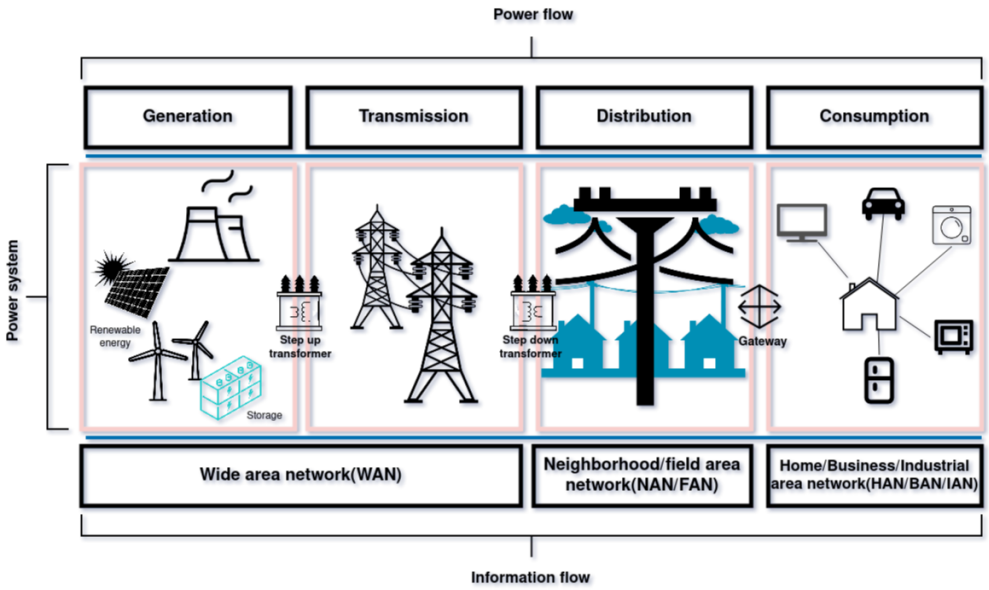


Figura 3.1: Arquitectura de Smart Grids.

En la figura 3.1 se presentan las cuatro capas de dentro de la distribución de poder en las arquitecturas de smart grids:

- **Generación:** Centrales eléctricas generan energía de manera convencional o renovable.
- **Transmisión:** La electricidad se transmite desde las centrales eléctricas a centros de carga remotos a través de líneas de transmisión de alta tensión.
- **Distribución:** Los sistemas de distribución distribuyen energía eléctrica a los consumidores finales.
- **Utilización de energía:** Aquí es donde los hogares o industrias utilizan la energía para alimentar refrigeradores, televisores, lavadoras, maquinarias, lineas de producción, etc.

Esta memoria se centrará principalmente en la capa de **distribución**.

3.2. Sistema de distribución eléctrica

En la siguiente sección se especifica el funcionamiento de un sistema de distribución eléctrica, sus principales componentes, y los principales motivos por los que se generan fallas.

3.2.1. Componentes

Dentro de los dispositivos de distribución podemos encontrar los siguientes:

- **Interruptor:** Se usan para desconectar partes del sistema del alimentador.
- **Disyuntor:** Al igual que los switches se usan para desconectar partes del sistema.
- **Reconector:** Es un tipo especial de interruptor diseñado para reducir los tiempos de interrupción del sistema en caso de fallos momentáneos.
- **Fusibles:** Dispositivos que se utilizan para proteger partes del circuito.

Luego el esquema de automatización FDIR(fault detection, localization, isolation, restoration) consiste en controladores montados en:

- Interruptores de montaje en panel
- Interruptores de montaje en poste
- Subestaciones

Figura 3.2: Interruptores de montaje en panel.

Figura 3.3: Controles de reenganche montados en poste.

3.2.2. Funcionamiento

Dentro del sistema de suministro eléctrico el sistema de distribución de energía eléctrica (o red de distribución eléctrica) se encarga del suministro de energía eléctrica desde las subestaciones a los usuarios finales. El siguiente diagrama ilustra su funcionamiento y como interactúa cada uno de sus componentes.

En la figura se pueden observar cuatro subestaciones(A,B,C,D), cada una con sus transformadores principales en azul. Además se detallan los interruptores, disyuntores y reconectores en color rojo cuando se encuentran cerrados y verde cuando están abiertos.

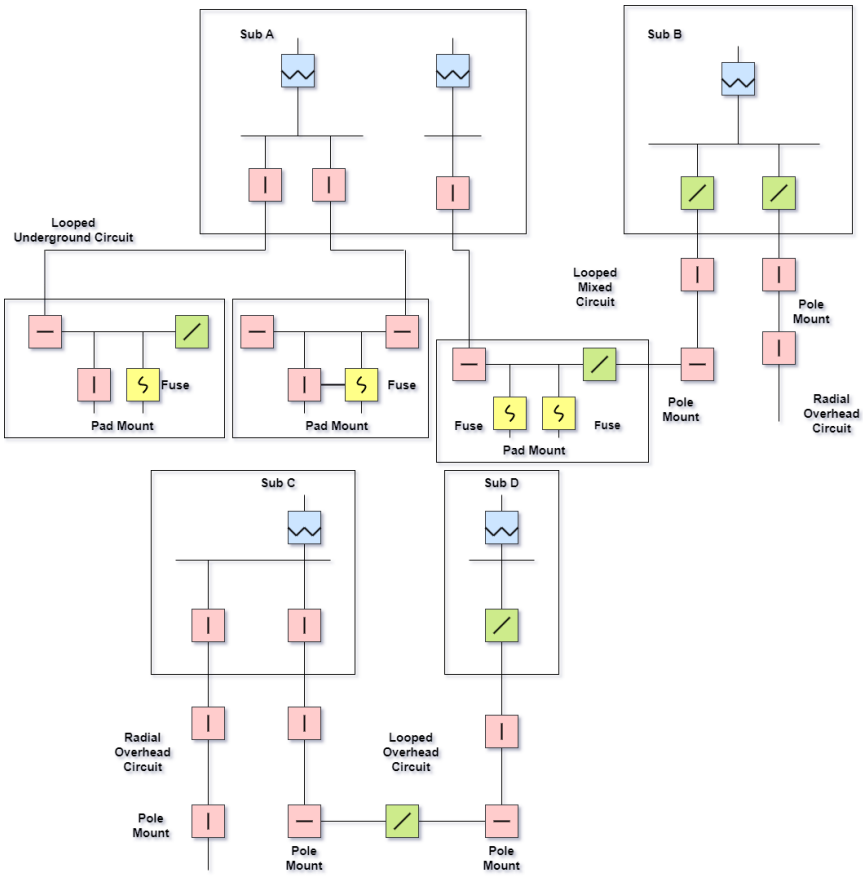


Figura 3.4: Diagrama de sistema de distribución simple. Fuente: elaboración propia

Cada una de estas subestaciones puede distribuir de manera subterránea o aérea y con una configuración radial o en bucle.

Configuración radial

La configuración radial tiene solo una fuente de energía para un grupo de clientes. Debido a esto cualquier cortocircuito, fallo de alimentación o línea eléctrica caída interrumpirían la energía en toda la línea, que debe ser reparada antes de poder restablecer el suministro. Esta configuración se ve del lado derecho de la figura 3.4 debajo del switch de montage en poste(pole mount).

Configuración en bucle

La configuración en bucle, como su nombre indica hace un bucle a través del área de servicio y vuelve al punto original. El bucle suele estar conectado a una fuente de energía alternativa. Colocando interruptores en lugares estratégicos para que la empresa pueda asegurar el suministro de energía desde distintas direcciones. Este sistema es más utilizado desde pequeñas plantas industriales hasta medianas o grandes construcciones comerciales donde es de gran importancia dar continuidad en el servicio. Esta configuración se puede ver del lado izquierdo de la salida de la subestación A en la figura 3.4 en el cuadro de "Pad mount" que corresponden a paneles como los mostrados en la figura 3.2.

Sistemas aéreos y subterráneos

El sistema aéreo es el mas barato debido a que el aire actúa como principal método de aislamiento, aquí el cableado viaja a través de aisladores instalados en crucetas, en postes de madera o de concreto. Este sistema ofrece ventajas en los costos y tiempos de construcción, así como también en el ámbito del mantenimiento y localización de fallas.

Los sistemas subterráneos son empleados generalmente en zonas urbanas céntricas, donde por razones de urbanismo, estética, congestión o condiciones de seguridad no es aconsejable el sistema aéreo. Este sistema ofrece ventajas ya que al no encontrarse expuestas son mucho mas seguras y robustas frente al vandalismo e inclemencias climáticas.

3.2.3. Fallas en redes eléctricas de energía

Las razones principales de falla en las redes eléctricas o EPG's (Electric Power Grids), se pueden clusterizar en tres principales causas de falla, las cuáles serán explicadas a continuación:

Causas Naturales: Existen muchos tipos de desastres naturales que pueden originar una falla en las redes de energía eléctrica, entre ellas se encuentra las inundaciones, terremotos, tormentas, olas de calor, entre otras.

Errores: Causas relacionadas a fallas humanas o mal funcionamiento del equipo técnico.

Ataques: Ciberataques como la denegación del servicio eléctrico o ataques humanos como lo es el terrorismo.

En la siguiente tabla se presentan diferentes fallas separadas entre los tres cluster principales explicados anteriormente:

Tabla 3.1: Fallas reportadas en la literatura. Fuente: A Survey on Power Grid Faults and Their Origins: A Contribution to Improving Power Grid Resilience. Energies [24]

Causas	Fallas
Causas Naturales	- Apagón - Falla en cascada - Colapso de torres de transmisión - Daños y averías en subestaciones - Cables caídos - Líneas desconectadas - Corrientes de falla - Falla de líneas de distribución y transmisión - Falla de transformadores - Fallas y daños en líneas aéreas de transmisión y distribución - Descarga disruptiva de líneas de transmisión - Aumento de corriente
Continúa en página siguiente	

Tabla 3.1 – Continuación de página anterior

Causas	Fallas
	- Fallas de línea - Pérdida de potencia - Sobrecargas de línea - Apagones localizados e interrupciones momentáneas - Corto circuitos - Límites de estabilidad excedidos - Inundación de subestación - Sobrecargas térmicas - Capacidad de transferencia limitada - Deslizamiento del transformador en la cimentación y caída o colapso total de la cimentación - Cargas de cables subterráneos afectadas - Inestabilidades de voltaje y frecuencia
Errores	- Apagón - Cortes en cascada - Corrientes de falla - Falla de transformadores - Desviación de frecuencia - Hidden faults of protection - Fallos ocultos de protección - Sobrecargas de línea - Inestabilidades de voltaje y frecuencia
Ataques	- Apagón - Fallos en cascada - Infraestructuras de control de redes inteligentes afectadas - Retraso, bloqueo o corrupción - Cables caídos - Perturbaciones económicas y sociales - Fallas de línea - Apagones localizados e interrupciones momentáneas - Power loss
Continúa en página siguiente	

Tabla 3.1 – Continuación de página anterior

Causas	Fallas
	- Daño generalizado

3.3. Métricas de desempeño

A continuación se presentaran las distintas métricas de desempeño usadas en esta memoria para la evaluación de los diferentes modelos de inteligencia artificial desarrollados a lo largo de este proyecto.

3.3.1. Matriz de confusión

La matriz de confusión es un método de evaluación de rendimiento de un modelo de clasificación [25]. En ella las diagonales contienen los elementos clasificados correctamente y los valores fuera de la diagonal contienen el número de confusiones, es decir, los errores debidos a omisiones o comisiones [26]. Se utilizara esta métrica ya que normalmente, los errores de un clasificador se muestran en la matriz de confusión. [27][28].

		PREDICCIÓN	
		0	1
REALIDAD	0	TN	FP
	1	FN	TP

Figura 3.5: Matriz de Confusión. Fuente: Elaboración propia

- **TN:** True Negative son los valores de predicción negativa que fueron clasificados correctamente por el modelo.

- **FN:** False Negative son los valores de predicción negativa que fueron clasificados incorrectamente por el modelo.
- **FP:** False Positive son los valores de predicción positiva que fueron clasificados incorrectamente por el modelo.
- **TP:** True Positive son los valores de predicción positiva que fueron clasificados correctamente por el modelo.

3.3.2. Accuracy:

De todas las clases cuantas se predijeron correctamente.

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \tag{3.1}$$

Precision:

De todas las positivas que se han predicho correctamente, cuantas son realmente positivas.

$$Precision = \frac{TP}{TP + FP} \tag{3.2}$$

3.3.3. Recall:

De todas las clases positivas, cuantas se predijo correctamente.

$$Recall = \frac{TP}{TP + FN} \tag{3.3}$$

3.3.4. F1-measure:

Permite comparar dos modelos de baja precisión y alta exhaustividad (recall) utiliza la medida armónica para castigar los valores extremos.

$$F_1 = 2 \cdot \frac{Recall \cdot Precision}{Recall + Precision} \tag{3.4}$$

3.4. Conceptos de Inteligencia Artificial

3.4.1. ¿Qué es la inteligencia artificial?

La inteligencia artificial es un concepto bastante debatido entre científicos desde sus inicios, debido a sus múltiples y diferentes opiniones sobre su definición. Este término ha ido evolucionado con el pasar de los años, teniendo en 1956 una de sus primeras definiciones, dada por el profesor John McCarthy de la Stanford University en la conferencia de Dartmouth en donde se declara que la inteligencia artificial es la ciencia y la ingeniería para crear máquinas inteligentes, especialmente programas informáticos inteligentes. Además está relacionada con la tarea similar de utilizar ordenadores para comprender la inteligencia humana, pero la inteligencia artificial no tiene por qué limitarse a métodos biológicamente observables [29]. Actualmente IA se refiere a los sistemas o máquinas que imitan la inteligencia humana para desarrollar tareas y pueden iterativamente mejorase ellos mismos basado en la información que recolectan realizando tales tareas [30].

3.4.2. Machine Learning

Podemos encontrar al aprendizaje automático o machine learning dentro de las ramas de la inteligencia artificial, este se define como el campo que se encarga de la capacidad de una IA para aprender a emular el comportamiento humano basado en datos observados. Esto permite a la inteligencia artificial realizar tareas complejas de forma similar a como lo haría un humano [31].

Machine learning se puede clasificar en 3 tipos de aprendizaje: Supervised Learning, Unsupervised Learning y Reinforcement Learning.

Supervised Learning

En este tipo de aprendizaje se utilizan datos etiquetados para entrenar los algoritmos, donde se conocen la entrada y la salida. El conjunto de entradas ingresadas se denomina características, indicadas por X y las salidas correspondientes, son indicadas por Y. El algoritmo aprende comparando su producción real con las salidas Y para disminuir el error. En consecuencia a esto los parámetros del modelo son modificados, lo que permite a los modelos aprender y volverse más precisos con el tiempo. Generalmente los datos son divididos en dos secciones. La primera parte es para entrenar el algoritmo y la otra región se usa para evaluar el algoritmo entrenado.

Unsupervised Learning

El aprendizaje no supervisado es el tipo de aprendizaje automático en el que se usan datos sin etiquetar para entrenar el algoritmo. El propósito de esto es explorar la información ingresada y encontrar patrones o tendencias que las personas no buscan explícitamente. En este tipo de aprendizaje la entrada de datos se ingresa al algoritmo directamente sin procesamiento previo y sin conocer la salida de ellos. Además, el algoritmo deberá descifra la información procesada y, de acuerdo con los segmentos de datos, crea grupos de ellos con nuevas etiquetas. En este caso, los datos no pueden dividirse en un secciones de entrenamiento o de prueba.

Reinforcement Learning

El aprendizaje por refuerzo es el tipo aprendizaje automático en el que no se proporcionan datos sin procesar como entrada, sino que el algoritmo de aprendizaje por refuerzo tiene que resolver la situación por sí mismo. Con el aprendizaje por refuerzo, el algoritmo descubre a través de prueba y error qué acciones producen las recompensas acumuladas más significativas. Este tipo de entrenamiento tiene tres componentes principales, que son el agente, que se puede describir como el aprendiz o el tomador de decisiones, el entorno, que se describe como todo con lo que interactúa el agente, y las acciones, que se representan como lo que el agente puede hacer. El objetivo es que el agente realice acciones que maximicen la recompensa esperada en un período de tiempo determinado.

El agente alcanzará la meta mucho más rápido siguiendo una buena política. Entonces, el propósito del aprendizaje por refuerzo es aprender este mejor plan condicional.

3.4.3. Deep Learning

Antes de conocer el concepto del aprendizaje profundo es necesario destacar lo que son las redes neuronales. Una red neuronal, o tambien conocidas como redes neuronales artificiales, esta compuesta por capas de nodos o neuronas artificiales conectadas entre si. Una neurona artificial intenta imitar el comportamiento de las neuronas del cerebro humano. En una red neuronal artificial se procesa el valor de las entradas mediante una función de activación no lineal dando así una salida. Luego estos resultados son usados como entradas por otras neuronas de la red.

Las redes de aprendizaje profundo o Deep Learning son redes neuronales con la capacidad de procesar grandes cantidades de datos utilizando muchas capas. Estas pueden determinar parámetros que se pueden entrenar como pesos y umbrales de cada enlace de la red. Algunos usos comunes del aprendizaje profundo son el reconocimiento de habla, clasificación de imágenes y detección de objetos[31].

3.5. Modelos clasificadores de deep learning y machine learning clásico

Un clasificador es capaz de categorizar automáticamente los datos de un conjunto en distintas “clases”. Para llevar esto a cabo se utiliza el reconocimiento de las características que diferencian a los elementos de una clase u otra [32].

3.5.1. Convolutional Neural Network (CNN)

¿Que son?

Las redes neuronales convolucionales pertenecen a los métodos de aprendizaje profundo. Estas se componen de múltiples bloques de construcción llamados **capas**, y están diseñadas para aprender automáticamente, y de forma adaptativa, jerarquías espaciales de características mediante un algoritmo de backpropagation. Entre las capas mas comunes de una red neuronal convolucional podemos encontrar las siguientes:

- **Capa de Convolución:** procesará la salida de neuronas que están conectadas en “regiones locales” de entrada, calculando el producto escalar entre sus pesos y una pequeña región a la que están conectados en el volumen de entrada
- **Capa de pooling:** se encarga de reducir las dimensiones de la capa oculta combinando las salidas de los grupos de neuronas de la capa anterior en una sola neurona de la capa siguiente
- **Capa fully connected:** conecta cada neurona de entrada con cada neurona de salida

Las CNN son usadas habitualmente en imágenes, ya que es posible explotar la estructura que de por si tiene una imagen. A diferencia de otros tipos de redes neuronales donde todos los nodos están conectados entre sí ("fully connected"), este opta por una red “sparse”. Esto quiere decir que la red tendrá muchos menos enlaces que el número máximo posible y solo conectará por áreas o sectores en la imagen con el fin de reducir la cantidad de parámetros que se tiene que buscar u optimizar.

¿Por que utilizar CNN en este proyecto?

El objetivo de este desafío es crear un software que sea capaz de identificar las fallas de la red eléctrica y clasificarlas, en otras palabras, verificar si existe una falla o no y tener al mismo tiempo otra salida distinta que indique de qué tipo de falla se trata.

Si bien las CNN son usadas ampliamente en imágenes, no se escapa del rango de uso que se le puede dar para una señal de una dimensión. En este caso se cuenta con una serie de tiempo.

Debido a la naturaleza de los datos existe una **dependencia temporal** entre un dato y otro, por ejemplo, el valor actual de los kW entregados por la red eléctrica de la región se vera mas relacionado a un valor inmediatamente anterior que a uno lejano en el tiempo.

La idea de utilizar estas redes neuronales en este proyecto va ligada a que hay una fuerte evidencia a que genera buenos resultados cuando se trata de lidiar con problemas multitask y gracias a la capacidad de aprender jerarquías espaciales mencionada anteriormente.

En resumen

A través de las propiedades de los datos que ya se conocen, ya sea una imagen o una señal, localmente solo algunas variables efectivamente estarán conectadas entre si (no todas las variables estarán conectadas con todas). Esto reafirma el argumento planteado, aplicando lo anteriormente explicado a través de varias capas convolucionales, la reducción de la cantidad de parámetros que se debe buscar u optimizar es significativa, lo que permite entrenar efectivamente una máquina que funciona. Por eso tiene sentido las CNN en una dimensión y en series de tiempo.

Un ejemplo de reducción es el análisis de componentes principales (PCA), método estadístico que permite simplificar la complejidad de espacios muestrales con muchas dimensiones a la vez que conserva su información. Método muy útil de aplicar previo a la utilización de otras técnicas estadísticas tales como regresión o clustering.

3.5.2. Long short-term memory (LSTM)

Las redes neuronales recurrentes (RNN) son un tipo de red neuronal artificial que se diferencia de las demás por la capacidad para considerar información de datos anteriores, esto les otorga un sentido de "memoria". Las RNN utilizan datos de entrenamiento para aprender (como las redes neuronales convolucionales). Este tipo de red se utiliza en problemas de tipo ordinal o temporal y se caracteriza por utilizar datos de entrada secuenciales o de series temporales. [33]

Este tipo de redes suelen sufrir el problema de memoria de corto plazo (short-term memory). Si una secuencia es lo suficientemente larga, tendrán problemas procesando esa información de pasos anteriores a los futuros. Por lo que si se quiere procesar señales de 1 dimensión de las lecturas de los transformadores de Miraflores con el suficiente largo de información como para que hayan las suficiente cantidad de fallas de manera que se pueda entrenar de manera eficiente el modelo de clasificación RNN's podría dejar afuera mucha información importante desde el inicio.

Back propagation es un algoritmo sirve para detectar errores en procesos que implican el uso de redes neuronales. Un inconveniente relevante que las RNN's sufren durante este proceso es el problema del desvanecimiento de la gradiente ("vanishing gradient problem"). Las gradientes son valores usados para actualizar los pesos generados por las redes neuronales. El problema del desvanecimiento de la gradiente es cuando la gradiente se encoge a medida que ocurre el back propagation en el tiempo. Por lo que en RNN's, las capas que obtienen pequeñas gradientes dejan de aprender, estas suelen ser capas iniciales, esto indica que las RNN's pueden olvidar lo que "ven" en secuencias muy largas.

Para solucionar este problema se usan redes neuronales LSTM, las cuales tienen un mecanismo interno, llamado puertas que regulan el flujo de información. Estas puertas pueden aprender que secuencia de datos es importante guardar o eliminar, por lo que hay una mayor probabilidad de pasar información relevante por la cadena de secuencias para hacer una predicción en la clasificación.

3.5.3. Random Forest (RF)

Los árboles de decisión son un método de aprendizaje supervisado utilizado en problemas de clasificación y regresión. La finalidad es crear un modelo que prediga el valor de una variable objetivo, mediante el aprendizaje de reglas de decisión simples, deducidas de las características de los datos. Se realiza una secuencia de decisiones sobre los datos disponibles que tenemos hasta que llegamos a una clasificación.

El clasificador Random Forest, consiste de un largo número de árboles de decisión individuales que operan como un conjunto. Cada árbol de decisión individualmente genera como output la clase de predicción y la clase con la mayor cantidad de "votos"termina siendo la clase predicha.

Aquí la baja correlación entre los modelos es fundamental, ya que los modelos no correlacionados pueden producir predicciones de conjunto que son más precisas que cualquiera de las predicciones individuales. Si bien puede ocurrir que algunos árboles estén equivocados, es mas probable que muchos más árboles estén correctos, por lo que, como grupo, los árboles pueden orientar sus respuestas a una clasificación correcta.

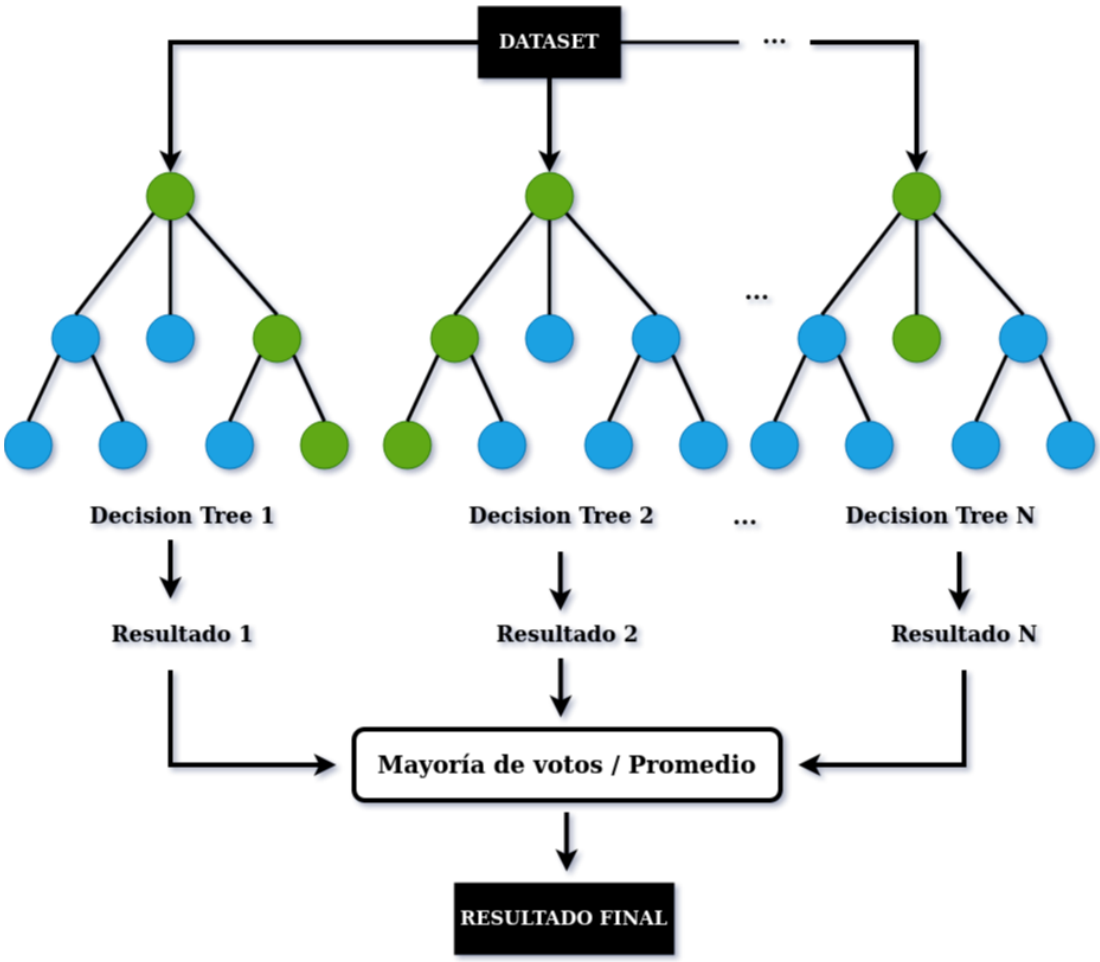


Figura 3.6: Diagrama de Random Forest. Fuente: Elaboración propia

3.6. Herramientas tecnológicas

Debido a que la solución de este proyecto fue creada a partir del lenguaje de programación Python es que se ha decidido usar las siguientes herramientas técnologicas que facilitarán la manipulación, el análisis y la visualización de los datos.

- **Numpy:** Biblioteca para Python que ofrece herramientas de computación numérica como funciones matemáticas, transformaciones, vectores, matrices, etc.
- **Pandas:** Biblioteca para Python que se especializa en la manipulación y el análisis de datos. Ofrece estructuras de datos y operaciones para manipular tablas numéricas y series temporales.
- **Matplotlib:** Biblioteca para Python que permite crear visualizaciones estáticas, animadas e interactivas en Python
- **Scikit-Learn:** biblioteca de aprendizaje automático para Python diseñada para algoritmos de clasificación, regresión, clustering, entre otros.
- **Keras:** Biblioteca de redes neuronales para Python que ofrece la posibilidad de experimentar con redes de aprendizaje profundo.

Numpy y Pandas serán utilizadas para la manipulación de los datos, Matplotlib para su visualización, Scikit-Learn para su análisis y finalmente Keras sera utilizada para la creación de las arquitecturas de los modelos y su posterior entrenamiento.

Capítulo 4

Propuesta de solución

Se busca generar una clasificación del estado de la red a través de las lecturas de los transformadores entregados por el área de planificación de Chilquinta. Esta solución se inicia a través de un preprocesamiento de la data en donde se etiquetan los datos de lectura de los transformadores T1 y T2 conectados a la subestación de Miraflores, para luego ser usados como entrada a un modulo de machine learning o deep learning en donde a través de un aprendizaje supervisado será capaz de clasificar si el dato de lectura de la red eléctrica se encuentra en estado de falla o no. Finalmente se entrega un dataset en formato excel con la clasificación hecha por el modelo de inteligencia artificial implementado.

Para llevar a cabo lo explicado anteriormente, se deberá realizar una serie experimentos en base a distintas hipótesis. Dependiendo de los resultados obtenidos, los distintos supuestos planteados inicialmente se validarán o se refutarán. Esto tiene la finalidad de lograr converger a una conclusión sobre que modelo es el mas apto para entregar la mejor solución posible al problema planteado.

4.1. Proceso de minería de datos

Antes de explicar las restricciones del problema, es necesario comprender el proceso del trabajo común para un proyecto de las características mencionadas a lo largo de esta memoria. A continuación en la figura 4.1 se detallan los pasos del flujo de datos en los procesos de minería de datos.

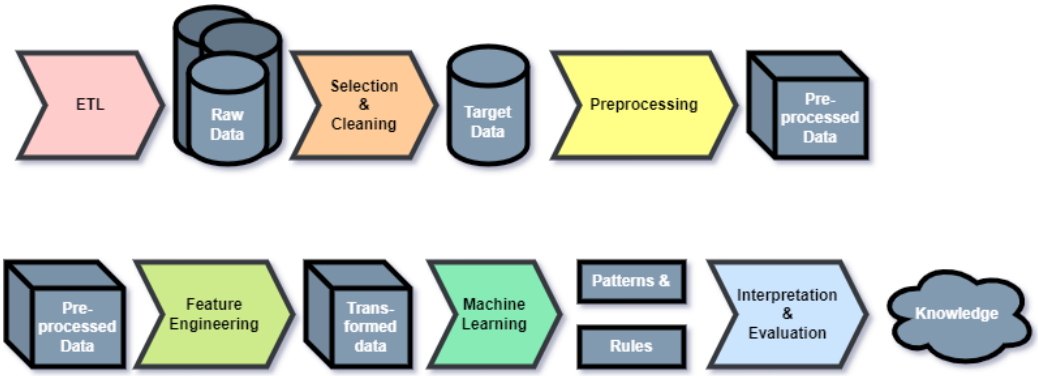


Figura 4.1: Proceso de minería de datos. Fuente: Elaboración propia

Se detalla cada uno de los pasos del diagrama:

- **ETL:** Para obtener la raw data o data cruda se debe realizar un proceso llamado ETL, compuesto de tres pasos: Extract, Transform y Load. El primer paso llamado **extracción** corresponde a cuando se adquieren datos desde múltiples bases de datos. Esto con el objetivo de construir un dataset en particular para minería de datos.

  Luego, el siguiente paso es **transformar**, es decir, se procesa cada uno de estos datos y se ponen dentro de un mismo dataset y finalmente se consolida su caso de uso analítico. En este paso se incluye medidas como dar formato a los datos en tablas o tablas unidas para que coincidan con el esquema del almacén de datos de destino, realización de auditorías para garantizar la calidad de los datos y el cumplimiento, entre otros.

  Finalmente, el proceso de **carga** corresponde a llevar ese gran conjunto de datos que se han extraído y transformado anteriormente en la base de datos de destino.

- **Selección & Cleaning:** Proceso de selección y limpieza que siguen los datos. Por ejemplo, se tienen los datos crudos que podrían venir de sensores. Es necesario

limpiar esos datos, revisar que todos se encuentren en la misma unidad y eliminar las variables que no estén entregando ninguna información. En este proceso también se elijen los datos objetivos dentro de toda la totalidad de datos.

- **Preprocessing:** Proceso que consiste en revisar los datos, imputarlos y seleccionar no solamente las columnas que van a servir sino que además verificar si los datos están balanceados o no. Esto quiere decir que si se encuentren datos de distintas clases en cantidades similares ocurrirá un sesgo. Por ejemplo si se quiere clasificar entre perros y gatos pero se tiene una cantidad mucho menor de datos de gatos, es posible que al intentar clasificar un gato, que no se parezca a los que se tienen en los datos de entrenamiento, termine siendo clasificado como un perro.

- **Feature Engineering:** En esta etapa se toman las características y se pueden transforman a otro espacio o se puede convertir de un conjunto de datos a otro muy similar realizando transformaciones. Además es posible eliminar algunas columnas que no sirven desde una transformación como una reducción de dimensionalidad. Esto se hace para que lo que tenga que aprender la maquina lo experimente de la mejor forma posible.

  En resumen, este proceso se pueden reducir las dimensiones de un conjunto de datos proveniente de una base de datos a dimensiones mas homogéneas y se pueden transformar esos datos a distintos valores para que se mantengan bajo ciertos limites numéricos (a menudo tomando valores entre 0 y 1).

- **Machine Learning:** Este paso fue explicado con mayor detalle en la sección 3.4.2(Machine Learning) de esta memoria.

- **Interpretation & Evaluation:** Una vez que se obtienen los resultados que pueden ser columnas, probabilidades o clasificaciones, se procede a interpretarlos y evaluarlos para generar el conocimiento.

4.2. Restricciones del problema

4.2.1. Área de Estudio

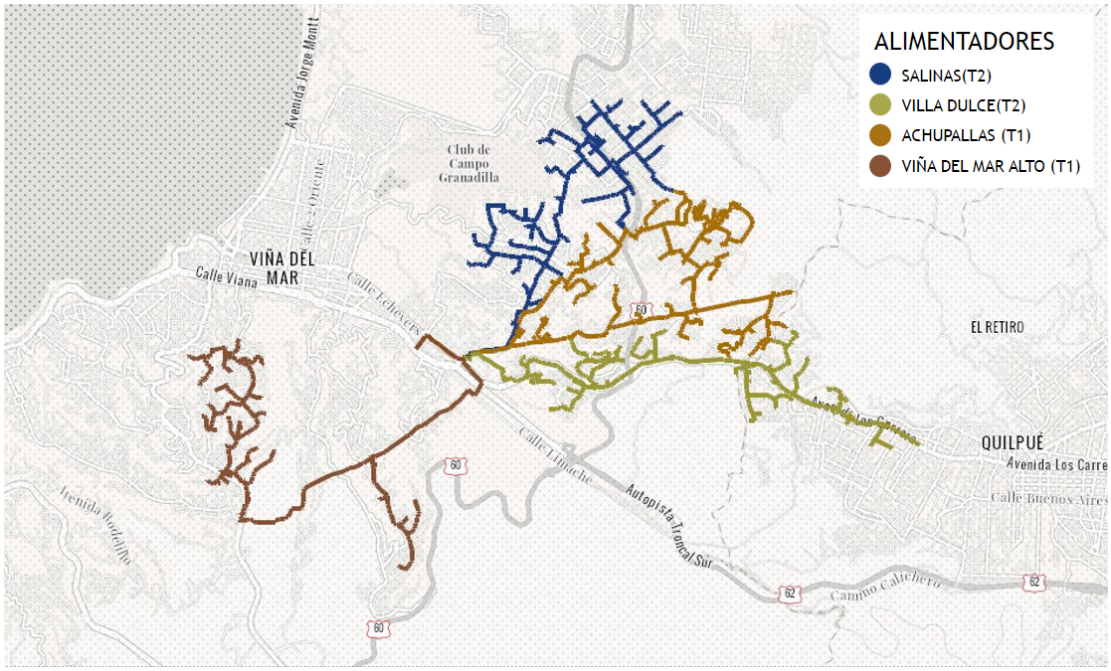


Figura 4.2: Mapa de alimentadorers conectados a subestación Miraflores. Fuente: Chilquinta Energía S.A

Si bien este proyecto busca ser implementado en toda la región de Valparaíso, primero se debe realizar una investigación en una área acotada para evaluar su factibilidad. Esta zona de estudio escogida fue el sector de Miraflores ubicada en la ciudad de Viña del Mar.

La razón detrás de esta selección, se centra en que Miraflores cuenta con una subestación que suministra a cuatro distintos y apartados sectores de la ciudad. Ellos cuentan con unas significativas diferencias como la cantidad de habitantes, factores ambientales y cantidad de accidentes, entre otros. Estas disimilitudes son relevantes en cuanto a las causas de fallas en el sistema eléctrico, permitiendo así, contar con una variedad de fallas lo que enriquecerá el dataset utilizado para el estudio.

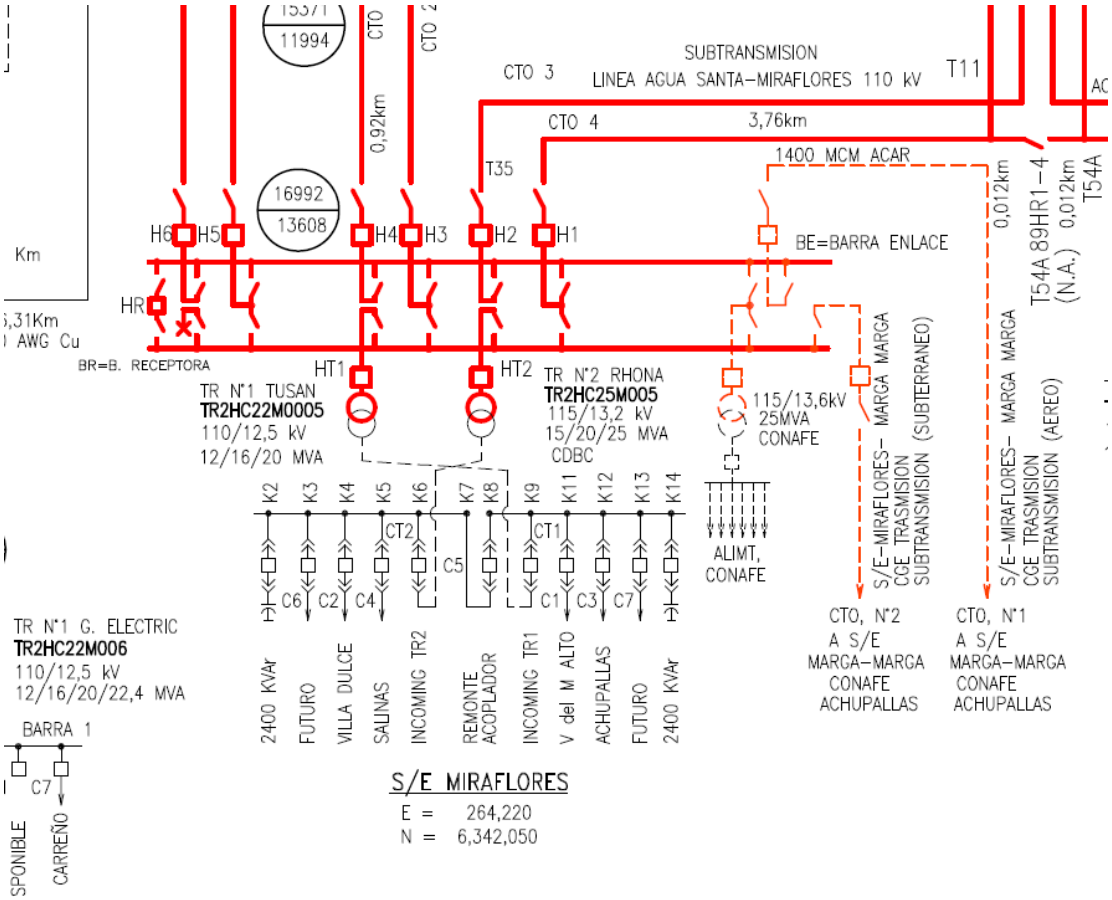


Figura 4.3: Plano de subestación Miraflores, al centro se pueden ver los trasnformadores T1 y T2 conectadas a los distintos sectores de la subestación Miraflores. Fuente: Chilquinta Energía S.A

En la subestación de Miraflores, se encuentran las conexiones del transformador 1 (T1) y el transformador 2 (T2). Como se puede apreciar en la figura 4.3, en T1 se encuentran conectadas los sectores de Viña del mar alto y Achupallas, mientras que en T2 están los sectores de Villa dulce y Salinas.

4.2.2. Libro de fallas

El área de operaciones de la empresa Chilquinta es la encargada de planificar y desarrollar las actividades en terreno requeridas para hacer el mantenimiento de la red eléctrica en las distintas zonas de concesión.

Dentro de sus funciones cuenta con la de recopilar la información extraída desde las distintas fallas que ocurren a lo largo de la red eléctrica y así generar el ”libro de fallas” el cual contiene la información tanto cuantitativa como cualitativa de las fallas ocurridas desde el año 2017 hasta el 2021.

4.2.3. Restricciones

Dado lo explicado en los últimos dos puntos anteriores, se consideran las siguientes restricciones:

- Solo se cuenta con las lecturas de los transformadores 1 y 2 conectados a la subestación de Miraflores, Viña del Mar.
- Se consideraran solo los datos del libro de fallas correspondientes a los sectores de Viña del mar alto, Achupallas, Villa dulce y Salinas para el estudio. Estos son los datos que pertenecen a las lecturas de T1 y T2.
- Dada la delimitación del período de tiempo almacenado en libro de fallas, solo se considerará las lecturas de los transformadores que se encuentren dentro de los años 2017 al 2021.

4.3. Entrenamiento y testeo

Para los entrenamientos y testeos a realizar durante los experimentos se procede a dividir el dataset construido durante el preprocesamiento en una proporción de la data en un 80 % para el entrenamiento y un 20 % para el testeo. Además se recalca que al momento de realizar esta división del dataset, no se mezclarán o desordenarán los datos con el fin de mantener y preservar la importancia de la temporalidad que tienen los datos durante el entrenamiento del modelo de clasificación.

4.4. Experimentos

En esta sección se procede a realizar distintos experimentos, los cuales irán evolucionando de acuerdo a los resultados obtenidos en cada uno de ellos. La finalidad de este proceso será determinar al mejor candidato entre los modelos propuestos y el preprocesamiento ideal en base al rendimiento y resultado que tengan en las métricas determinadas anteriormente en el marco teórico.

4.4.1. Redes Neuronales Convolucionales

Como primera instancia se propone el uso de redes neuronales convolucionales (CNN), debido a que hay una fuerte evidencia a que genera buenos resultados cuando se trata de hacer esquemas multitask[34], en este caso se busca una red neuronal que otorgue como output la probabilidad de que el dato leído sea una falla y al mismo tiempo que indique que tipo de falla es.

4.5. Preprocesamiento Para CNN

4.5.1. Material inicial

Como punto de partida se presentan dos datasets, el primero corresponde al de las lecturas de los alimentadores, la cual cuenta con la información del transformador, es por esto que se cuenta con dos de estos archivos ya que Miraflores tiene dos transformadores a los cuales llamaremos T1 y T2. Dentro del dataset podemos encontrar 6 columnas las que dan la información de los kilovatios entregados, recibidos, Kilo Volt Amper Reactivo entregados y recibidos y tensión. Como se puede observar en la figura 4.5 algunos datos de la columna de tensión toman como valor "NaN", el cual impedía la realización de algunos cálculos numéricos. Por esto se cambio este valor por 0. En la sección 4.7.2 se replantea este cambio a partir de una observación.

	medidor	fecha	kW_del	kVAr_del	kW_rec	kVAr_rec	tension
0	Trf Miraflores T1	2011-01-01 00:15:00	7782.243000	928.896	0.0	0.000000	NaN
1	Trf Miraflores T1	2011-01-01 00:30:00	7875.647000	938.860	0.0	0.000000	NaN
2	Trf Miraflores T1	2011-01-01 00:45:00	7629.507000	659.056	0.0	0.000000	NaN
3	Trf Miraflores T1	2011-01-01 01:00:00	7863.619000	773.697	0.0	0.000000	NaN
4	Trf Miraflores T1	2011-01-01 01:15:00	7953.123000	803.134	0.0	0.000000	NaN
...	...	...	...	...	...	...	...
365111	Trf Miraflores T1	2021-04-30 23:00:00	8532.578125	0.000	0.0	558.630371	NaN
365112	Trf Miraflores T1	2021-04-30 23:15:00	8303.352539	0.000	0.0	563.043518	NaN
365113	Trf Miraflores T1	2021-04-30 23:30:00	7994.871582	0.000	0.0	586.591919	NaN
365114	Trf Miraflores T1	2021-04-30 23:45:00	7681.770020	0.000	0.0	663.821289	NaN
365115	Trf Miraflores T1	2021-05-01 00:00:00	7403.301270	0.000	0.0	727.333801	NaN

Figura 4.4: Archivo de Lectura de un alimentador. Fuente: Elaboración propia

Interrupcion Id	Bloqu	Califica	Causa	Fh Inicio Interrupci	Fh Termino Bloq	Duraci	Alimentador	Nombre Alimentador	Periodo S
006302201701019001	1	3	1706	01-01-2017 4:27	01-01-2017 5:05	2280	90000006	CARTAGENA	12017
006302201701019003	1	2	2304	01-01-2017 6:54	01-01-2017 8:39	6300	90000006	CARTAGENA	12017
006302201701019004	1	2	2404	01-01-2017 11:41	01-01-2017 13:15	5640	60000087	PLACILLA	12017
006302201701019005	1	2	2401	01-01-2017 8:53	01-01-2017 10:00	4020	60000008	PEÃ‘ABLANCA	12017
006302201701019006	1	2	1802	01-01-2017 16:35	01-01-2017 17:35	3600	60000012	MARGA MARGA	12017
006302201701019007	1	2	2304	01-01-2017 9:08	01-01-2017 10:26	4680	90000006	CARTAGENA	12017
006302201701019008	1	2	2401	01-01-2017 14:33	01-01-2017 16:25	6720	60000034	URUGUAY	12017
006302201701019009	1	2	1802	01-01-2017 17:20	01-01-2017 18:15	3300	60000005	SALINAS	12017
006302201701019010	1	2	1802	01-01-2017 14:34	01-01-2017 16:05	5460	60000070	SAN ESTEBAN	12017
006302201701019011	1	2	2401	01-01-2017 17:35	01-01-2017 18:30	3300	60000026	TOMAS RAMOS	12017
006302201701019012	1	2	1101	01-01-2017 16:51	01-01-2017 17:45	3240	60000026	TOMAS RAMOS	12017
006302201701019013	1	2	1807	01-01-2017 17:21	01-01-2017 18:08	2820	60000025	PLAYA ANCHA	12017
006302201701019014	1	2	2401	01-01-2017 17:16	01-01-2017 18:40	5040	60000101	LLIU LLIU	12017
006302201701019015	1	2	2401	01-01-2017 17:30	01-01-2017 20:20	10200	60000023	POLANCO	12017
006302201701019016	1	2	2304	01-01-2017 19:05	01-01-2017 19:55	3000	90000006	CARTAGENA	12017
006302201701019017	1	2	2404	01-01-2017 19:21	01-01-2017 21:22	7260	60000014	QUINTERO	12017
006302201701019018	1	2	2204	01-01-2017 18:56	01-01-2017 21:10	8040	60000012	MARGA MARGA	12017
006302201701019019	1	2	1807	01-01-2017 17:05	01-01-2017 21:26	15660	60000034	URUGUAY	12017

Figura 4.5: Libro de fallas. Fuente: Chilquinta Energía SA

4.5.2. Filtrado de alimentadores

Los parámetros de que se consideraron para comenzar con la limpieza de datos fueron los siguientes:

- Se consideraron solo los alimentadores que corresponden a la zona de estudio es decir el área de Miraflores, estos alimentadores se muestran en la figura 4.3 donde se detalla que los alimentadores de Villa Dulce y Salinas corresponden al ya mencionado T2 y los de Achupallas y Viña del Mar Alto corresponden al transformador T1.
- La duración y la fecha, ya que servirán como enlace con los datos de la lectura para unificar ambos archivos.

4.5.3. Etiquetado de los datos

Para el preprocesamiento de los datos utilizados como entrada para la CNN, se utiliza la información contenida en el libro de fallas, en el cual detalla la duración de cada falla, con el fin de etiquetar las lecturas del transformador 1 añadiendo una columna extra que indica de manera binaria la presencia de una falla con el número 1 o la ausencia de ella con el número 0.

Además, como se mencionó anteriormente en las restricciones, el libro de fallas solo

cuenta con la información del periodo de enero del año 2017 hasta julio del 2021, por lo que también se deberá realizar un filtro de las lecturas del transformador que calzen dentro del periodo señalado.

Luego de este proceso se consigue un dataset como el que se puede ver en la figura 4.6 que contiene la información del dataset de las lecturas, pero con la columna añadida de las fallas.

	utc	medidor	fecha	kW_del	kVAr_del	kW_rec	kVAr_rec	falla
219343	2017-04-03 23:00:00	Trf Miraflores T2	2017-04-03 20:00:00	9554.534180	258.573395	0.0	0.0	1
219344	2017-04-03 23:15:00	Trf Miraflores T2	2017-04-03 20:15:00	10506.958984	281.073456	0.0	0.0	1
219345	2017-04-03 23:30:00	Trf Miraflores T2	2017-04-03 20:30:00	10720.787109	269.900024	0.0	0.0	1
219346	2017-04-03 23:45:00	Trf Miraflores T2	2017-04-03 20:45:00	10888.494141	256.999603	0.0	0.0	1
219347	2017-04-04 00:00:00	Trf Miraflores T2	2017-04-03 21:00:00	10831.455078	244.494278	0.0	0.0	1
...	...	...	...	...	...	...	...	...
374015	2021-09-01 03:00:00	Trf Miraflores T2	2021-08-31 23:00:00	10451.503906	1794.065063	0.0	0.0	0
374016	2021-09-01 03:15:00	Trf Miraflores T2	2021-08-31 23:15:00	9940.391602	1671.242065	0.0	0.0	0
374017	2021-09-01 03:30:00	Trf Miraflores T2	2021-08-31 23:30:00	9574.870117	1613.777710	0.0	0.0	0
374018	2021-09-01 03:45:00	Trf Miraflores T2	2021-08-31 23:45:00	9189.042969	1571.922852	0.0	0.0	0
374019	2021-09-01 04:00:00	Trf Miraflores T2	2021-09-01 00:00:00	8785.914062	1512.587402	0.0	0.0	0

Figura 4.6: Lecturas con la etiqueta de datos correspondiente a su estado de operación registrado en el libro de fallas. Fuente: Elaboración propia

4.5.4. Ventana deslizante

Luego de realizar el etiquetado de las lecturas del transformador se conduce la data a un nuevo preprocesamiento llamado "ventana deslizante". Comúnmente ocupado en problema que incluyen una CNN en su arquitectura, en donde es necesario tomar en cuenta una porción de un conjunto de datos, en este caso datos de series de tiempo multivariables. Para ello esta información debe convertirse en un conjunto de entradas de vector o matriz de longitud fija antes de ingresar a la CNN para obtener una mejor implementación [35].

Para aplicar este método se necesita un conjunto de datos de series temporales. Estos valores a través de los pasos de tiempo se pueden agrupar en un vector de entrada, mientras que la salida se les asigna una etiqueta especifica.

La imposición de ventanas deslizantes en los flujos de datos es un método natural de aproximación que tiene varias propiedades atractivas como en este caso conservar la temporalidad de los datos en un periodo de tiempo {[36], [37]}.

Este ultimo proceso se realiza debido a que no se puede hacer una detección instantánea con el dataset actual, por lo que se necesita un cambio de tendencia para poder detectar una falla en las lecturas. Este cambio de tendencia permite que cada muestra sea comparable temporalmente respecto al centro, en otras palabras le agrega una especie de temporalidad a la data con el fin de tener mejores resultados en el entrenamiento de la CNN.

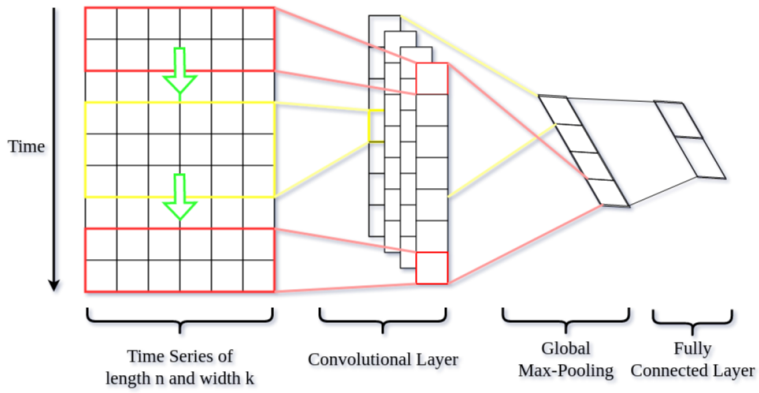
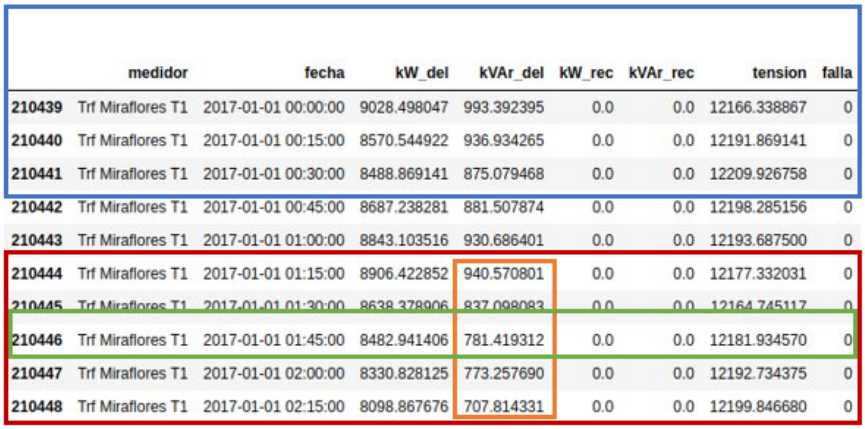


Figura 4.7: Diagrama representativo de los datos al entrar a la capa convolucional. Fuente: Elaboración propia

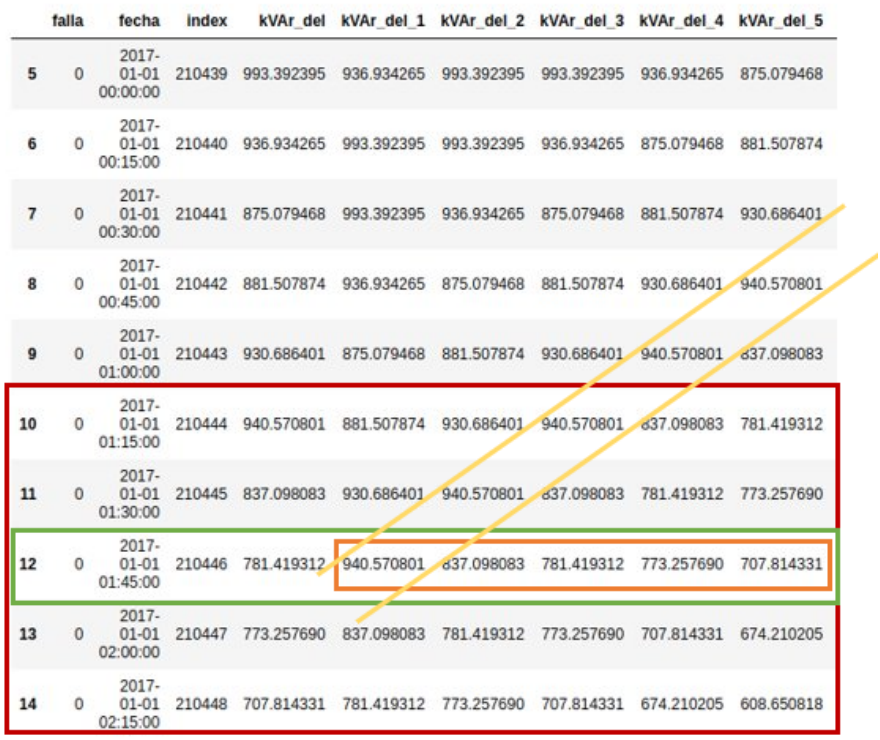
Para explicar el procedimiento realizado se tomará como ejemplo un tamaño de 5 para la ventana deslizante. Como se puede apreciar en la figura 4.8 , el cuadro inferior rojo indica la ventana, en donde se toma el centro de esta ventana (indicada en el cuadro verde) y por cada característica o columna (marcada por el cuadro naranjo) se desplegará de manera horizontal en este nuevo dataset tal como se puede ver en la 4.9. Naturalmente esto se realiza por cada característica o columna del dato propiamente-tal a la vez por cada fila de la data.



	medidor	fecha	kW_del	kVAr_del	kW_rec	kVAr_rec	tension	falla
210439	Trf Miraflores T1	2017-01-01 00:00:00	9028.498047	993.392395	0.0	0.0	12166.338867	0
210440	Trf Miraflores T1	2017-01-01 00:15:00	8570.544922	936.934265	0.0	0.0	12191.869141	0
210441	Trf Miraflores T1	2017-01-01 00:30:00	8488.869141	875.079468	0.0	0.0	12209.926758	0
210442	Trf Miraflores T1	2017-01-01 00:45:00	8687.238281	881.507874	0.0	0.0	12198.285156	0
210443	Trf Miraflores T1	2017-01-01 01:00:00	8843.103516	930.686401	0.0	0.0	12193.687500	0
210444	Trf Miraflores T1	2017-01-01 01:15:00	8906.422852	940.570801	0.0	0.0	12177.332031	0
210445	Trf Miraflores T1	2017-01-01 01:30:00	8638.378906	837.098083	0.0	0.0	12164.745117	0
210446	Trf Miraflores T1	2017-01-01 01:45:00	8482.941406	781.419312	0.0	0.0	12181.934570	0
210447	Trf Miraflores T1	2017-01-01 02:00:00	8330.828125	773.257690	0.0	0.0	12192.734375	0
210448	Trf Miraflores T1	2017-01-01 02:15:00	8098.867676	707.814331	0.0	0.0	12199.846680	0

Figura 4.8: Bosquejo de ventana deslizante de tamaño 5. Fuente: Elaboración propia

Lo interesante que se debe notar aquí, es lo que ocurre con los datos al realizar este método. Por ejemplo, en el sector marcado en amarillo, Figure 5.7, el dato 940 va "deslizándose" a medida que las filas avanzan, así se genera este input que será entregado posteriormente a la red neuronal con el fin de entrenarla y que pueda detectar si hay una falla o no en la red eléctrica.



	falla	fecha	index	kVAr_del	kVAr_del_1	kVAr_del_2	kVAr_del_3	kVAr_del_4	kVAr_del_5
5	0	2017-01-01 00:00:00	210439	993.392395	936.934265	993.392395	993.392395	936.934265	875.079468
6	0	2017-01-01 00:15:00	210440	936.934265	993.392395	993.392395	936.934265	875.079468	881.507874
7	0	2017-01-01 00:30:00	210441	875.079468	993.392395	936.934265	875.079468	881.507874	930.686401
8	0	2017-01-01 00:45:00	210442	881.507874	936.934265	875.079468	881.507874	930.686401	940.570801
9	0	2017-01-01 01:00:00	210443	930.686401	875.079468	881.507874	930.686401	940.570801	837.098083
10	0	2017-01-01 01:15:00	210444	940.570801	881.507874	930.686401	940.570801	837.098083	781.419312
11	0	2017-01-01 01:30:00	210445	837.098083	930.686401	940.570801	837.098083	781.419312	773.257690
12	0	2017-01-01 01:45:00	210446	781.419312	940.570801	837.098083	781.419312	773.257690	707.814331
13	0	2017-01-01 02:00:00	210447	773.257690	837.098083	781.419312	773.257690	707.814331	674.210205
14	0	2017-01-01 02:15:00	210448	707.814331	781.419312	773.257690	707.814331	674.210205	608.650818

Figura 4.9: Bosquejo de ventana deslizante de tamaño 5. Fuente: Elaboración propia

Se presento un problema para los primeros y últimos datos, algunas filas de la ventana deslizante quedaban fuera del dominio, por lo que no se podían rellenar ciertas filas del dataset final a causa de esto.

Para solucionar este problema de los casos bordes, lo realizado fue una clonación de los datos de manera espejo. Por ejemplo, en la figura 4.10 se puede percibir el cuadro verde el cual contiene los mismos datos del cuadro rojo de manera inversa, exactamente como si hubiera un espejo entre ellos. Luego de rellenar las filas faltantes del dataset final, los datos extras agregados fueron eliminados. De esta manera se logró obtener el dataset final.

	medidor	fecha	kW_del	kVAr_del	kW_rec	kVAr_rec	tension	falla
210443	Trf Miraflores T1	2017-01-01 01:00:00	8843.103516	930.686401	0.0	0.0	12193.687500	0
210442	Trf Miraflores T1	2017-01-01 00:45:00	8687.238281	881.507874	0.0	0.0	12198.285156	0
210441	Trf Miraflores T1	2017-01-01 00:30:00	8488.869141	875.079468	0.0	0.0	12209.926758	0
210440	Trf Miraflores T1	2017-01-01 00:15:00	8570.544922	936.934265	0.0	0.0	12191.869141	0
210439	Trf Miraflores T1	2017-01-01 00:00:00	9028.498047	993.392395	0.0	0.0	12166.338867	0
210439	Trf Miraflores T1	2017-01-01 00:00:00	9028.498047	993.392395	0.0	0.0	12166.338867	0
210440	Trf Miraflores T1	2017-01-01 00:15:00	8570.544922	936.934265	0.0	0.0	12191.869141	0
210441	Trf Miraflores T1	2017-01-01 00:30:00	8488.869141	875.079468	0.0	0.0	12209.926758	0
210442	Trf Miraflores T1	2017-01-01 00:45:00	8687.238281	881.507874	0.0	0.0	12198.285156	0
210443	Trf Miraflores T1	2017-01-01 01:00:00	8843.103516	930.686401	0.0	0.0	12193.687500	0

Figura 4.10: Caso borde cabecera. Fuente: Elaboración propia

	medidor	fecha	kW_del	kVAr_del	kW_rec	kVAr_rec	tension	falla
365111	Trf Miraflores T1	2021-04-30 23:00:00	8532.578125	0.0	0.0	558.630371	NaN	0
365112	Trf Miraflores T1	2021-04-30 23:15:00	8303.352539	0.0	0.0	563.043518	NaN	0
365113	Trf Miraflores T1	2021-04-30 23:30:00	7994.871582	0.0	0.0	586.591919	NaN	0
365114	Trf Miraflores T1	2021-04-30 23:45:00	7681.770020	0.0	0.0	663.821289	NaN	0
365115	Trf Miraflores T1	2021-05-01 00:00:00	7403.301270	0.0	0.0	727.333801	NaN	0
365115	Trf Miraflores T1	2021-05-01 00:00:00	7403.301270	0.0	0.0	727.333801	NaN	0
365114	Trf Miraflores T1	2021-04-30 23:45:00	7681.770020	0.0	0.0	663.821289	NaN	0
365113	Trf Miraflores T1	2021-04-30 23:30:00	7994.871582	0.0	0.0	586.591919	NaN	0
365112	Trf Miraflores T1	2021-04-30 23:15:00	8303.352539	0.0	0.0	563.043518	NaN	0
365111	Trf Miraflores T1	2021-04-30 23:00:00	8532.578125	0.0	0.0	558.630371	NaN	0

Figura 4.11: Caso borde cola. Fuente: Elaboración propia

Una manera mas visual y representativa del proceso explicado anteriormente, se pude ver en la figura 4.12. En donde una cantidad N de columnas del dataset inicial (lado izquierdo) son replicadas, tanto en la cabecera como en la cola, de manera espejo para dar solución al problema de los casos bordes.

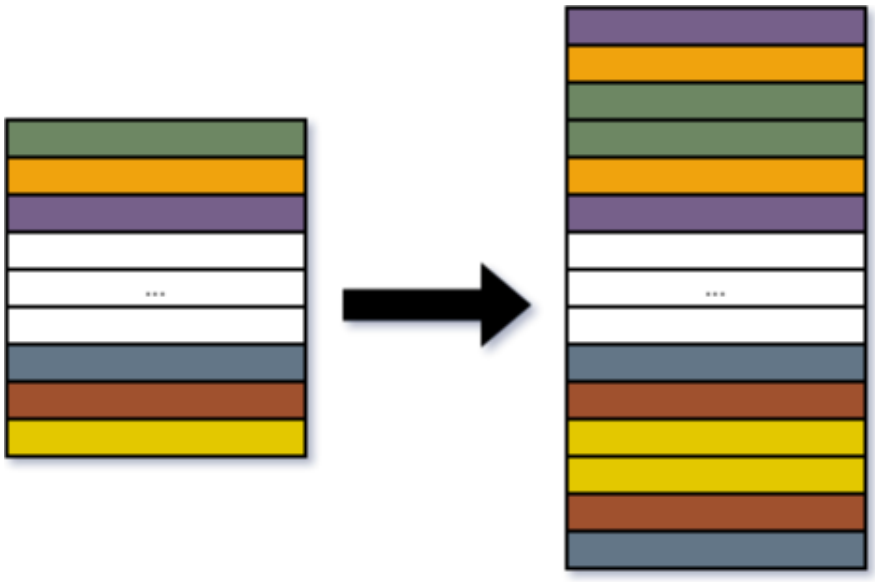


Figura 4.12: Caso borde cabecera y cola. Fuente: Elaboración propia

Finalmente se obtiene el dataset mostrado en la figura 4.13 en el cual podemos encontrar la información en nuevas columnas que representan la información capturada por la ventana deslizante añadida a cada medición, es decir que a cada medición se le añadirá por cada característica un número de columnas con las características de las mediciones vecinas dependiendo del tamaño de la ventana.

	falla	fecha	index	kVAr_del	kVAr_del_1	kVAr_del_2	kVAr_del_3	kVAr_del_4	kVAr_del_5
5	0	2017-01-01 00:00:00	210439	993.392395	936.934265	993.392395	993.392395	936.934265	875.079468
6	0	2017-01-01 00:15:00	210440	936.934265	993.392395	993.392395	936.934265	875.079468	881.507874
7	0	2017-01-01 00:30:00	210441	875.079468	993.392395	936.934265	875.079468	881.507874	930.686401
8	0	2017-01-01 00:45:00	210442	881.507874	936.934265	875.079468	881.507874	930.686401	940.570801
9	0	2017-01-01 01:00:00	210443	930.686401	875.079468	881.507874	930.686401	940.570801	837.098083
...	...	...	...	...	...	...	...	...	...
154677	0	2021-04-30 23:00:00	365111	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
154678	0	2021-04-30 23:15:00	365112	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
154679	0	2021-04-30 23:30:00	365113	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
154680	0	2021-04-30 23:45:00	365114	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
154681	0	2021-05-01 00:00:00	365115	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000

Figura 4.13: Ejemplo de resultado de ventana deslizante de tamaño 5. Fuente: Elaboración propia

4.6. Experimentos CNN

Se probó una arquitectura de red neuronal convolucional sin el efecto bottleneck. Esta técnica en una red neuronal es solo una capa con menos neuronas que la capa inferior (o superior). Tener una capa de este tipo incentiva a la red a comprimir las representaciones de características para ajustar y reducir la dimensión de la entrada. Las mejoras en la compresión ocurren debido a la reducción de la función de costo, como para todas las actualizaciones de peso. El motivo de esto es para posteriormente realizar una comparativa con una red que si lo posea.

La arquitectura inicial utilizada fue la siguiente:

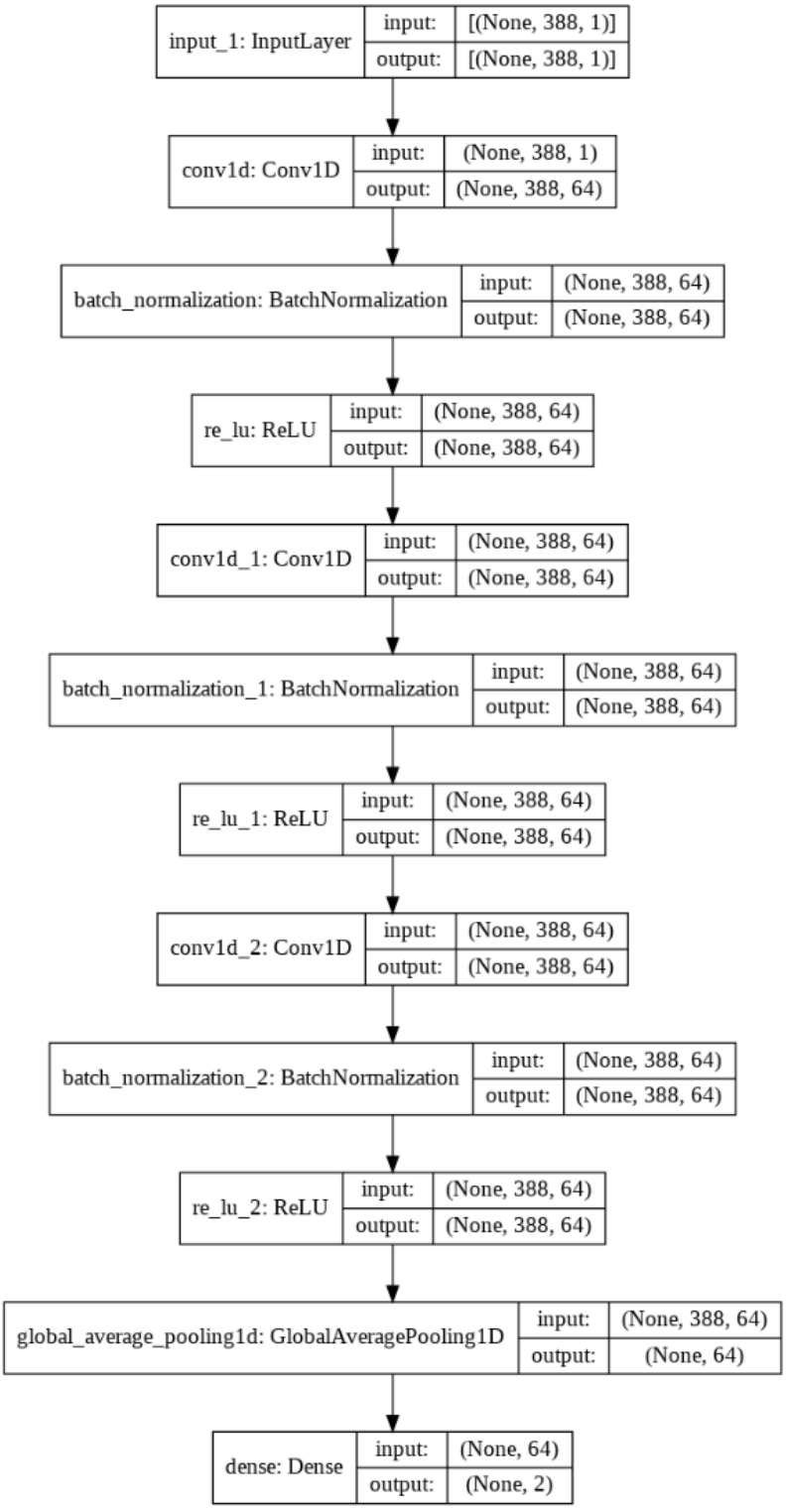


Figura 4.14: Arquitectura inicial CNN. Fuente: Elaboración propia

Experimento 1

El primer experimento consistió en el uso de la arquitectura explicada en el apartado anterior con una ventana deslizante de tamaño 5 y 500 épocas para el entrenamiento del modelo. La razón del uso de una ventana deslizante tan pequeña es debido a que se deseaba tener un punto base de comparación al momento de incrementar el tamaño de la ventana y obtener conclusiones precisas respecto de cómo afecta el tamaño de esta en los resultados obtenidos.

Como se puede ver en la figura 4.15 se utilizo la métrica de rendimiento "sparse categorical accuracy" para medir el rendimiento de red neuronal. Las lineas naranja y azul representan los resultados para los conjuntos de validación y entrenamiento respectivamente.

Para la realización del experimento se utilizo un método de early stopping que permite detener el algoritmo una vez que se detecta que el rendimiento del modelo deja de mejorar. En este caso se detuvo a las 346 épocas.A pesar del pequeño tamaño de la ventana y la baja cantidad de épocas se nota un buen indicio en los valores de accuracy obtenidos, el crecimiento continuo y la convergencia de ambas curvas de training y validation.

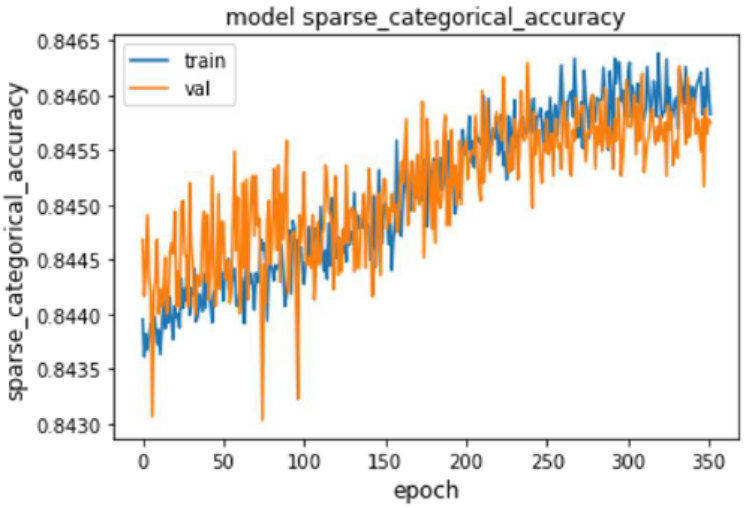


Figura 4.15: Accuracy CNN Experimento N°1. Fuente: Elaboración propia

Con el fin de mejorar la calidad de estos resultados, se deberá tomar una ventana deslizante de 97, ya que como mencionamos en secciones anteriores, la data es recolectada cada 15 [min], por lo que una ventana de tamaño 97 contendrá la información de al menos un día. Esta decisión se justifica con el tiempo máximo que puede llegar a estar la red en estado de falla de manera consecutiva.

Además, se deberá tener presente que al ejecutar el método de la ventana deslizante no se está considerando la dependencia temporal que existe entre los datos lo cual es relevante para tener una mayor calidad en los resultados. Esto es solucionable ya que si bien se están aplicando las capas convolucionales al principio para las cinco características (KW_del, KW_rec, KVAr_rec, Kvar_del, tensión), se puede agregar la variable indicadora de tiempo en las capas densas de la CNN.

Luego de aumentar el tamaño de la ventana deslizante se agrega la característica de la tensión que en el primer experimento no se estaba considerando y se propuso aumentar la cantidad de épocas a 700 para entrenar el modelo. Al igual que en el experimento se usó la métrica "sparse_categorical_accuracy" para la estimación del "accuracy" y sparse_categorical_crossentropy para el loss.

En esta última métrica las etiquetas de verdad están codificadas en números enteros, por ejemplo, [1], [2] y [3] para un problema de 3 clases. Además utiliza una función de pérdida donde cada probabilidad de clase predicha se compara con la salida deseada tomando un valor de 0 o 1 de la clase real y se calcula una puntuación/pérdida que penaliza la probabilidad en función de qué tan lejos está del valor esperado real. La pérdida de cross-entropy se usa cuando se ajustan los pesos de los modelos durante el entrenamiento.

En 4.16 se puede ver como lentamente las curvas se van atenuando tanto para la curva de entrenamiento como para la de validación a lo largo del avance de las épocas, llegando a un valor cercano a 0.86 de accuracy.

En cuanto a las curvas de loss mostradas en 4.18 se puede observar que todavía no hay un "optimal fit", y esto tiene que ver con la calidad del modelo que se está usando y la cantidad de épocas para este modelo en particular, que como se puede concluir requiere mucho más que 700, además si bien se está formando un gap entre las curvas.

Aun no es concluyente si habrá una deficiencia en la presición del aprendizaje relacionada

al sobreajuste o no. Mientras que por lo que se puede observar no habrá una de subajuste.

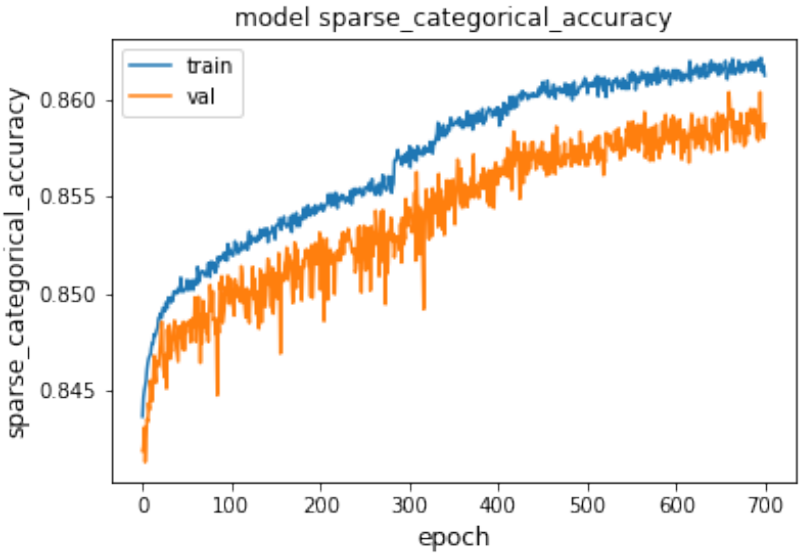


Figura 4.16: Accuracy CNN ventana tamaño 97. Fuente: Elaboración propia

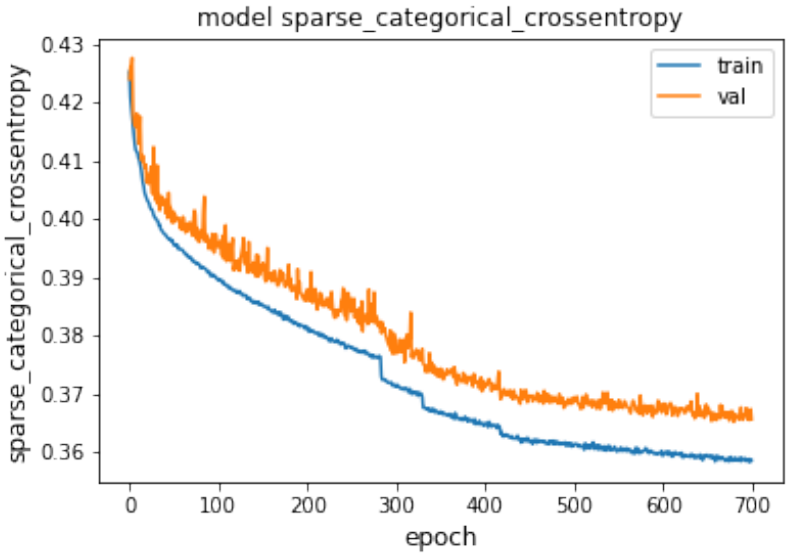


Figura 4.17: Loss CNN ventana tamaño 97. Fuente: Elaboración propia

Se puede apreciar que aún hay un desbalance de esta matriz al haber una considerable cantidad de falsos negativos, los cuales se esperan reducir con modificaciones en la arquitectura.

		PREDICCIÓN	
		0	1
REALIDAD	0	127030	3347
	1	23693	607

Figura 4.18: Matriz de Confusión de CNN ventana tamaño 97. Fuente: Elaboración propia

$$Accuracy = \frac{607 + 127030}{607 + 127030 + 23693 + 3347} = \frac{127637}{154677} = 0{,}825 \tag{4.1}$$

$$Precision = \frac{607}{607 + 3347} = \frac{607}{3954} = 0{,}154 \tag{4.2}$$

$$Recall = \frac{607}{607 + 23693} = \frac{607}{24300} = 0{,}025 \tag{4.3}$$

$$F_1 - measure = 2 \cdot \frac{0{,}025 \cdot 0{,}154}{0{,}025 + 0{,}154} = 2 \cdot \frac{0{,}00385}{0{,}404} = 0{,}02 \tag{4.4}$$

Experimento 2

Se prueba cambiar la arquitectura del modelo a una de estilo VGG. La razón de este cambio es la implementación de una arquitectura con “bottleneck” que , en otras palabras, va reduciendo las representaciones de características hasta llegar a una sola característica que represente el estado de operación de la red eléctrica. La finalidad de este cambio va ligado a que no solo el entrenamiento en es mas rápido, sino que también más efectivo ya que, la cantidad de épocas necesarias para el entrenamiento del modelo se ve disminuida y la calidad de los resultados tiende a mejorar.

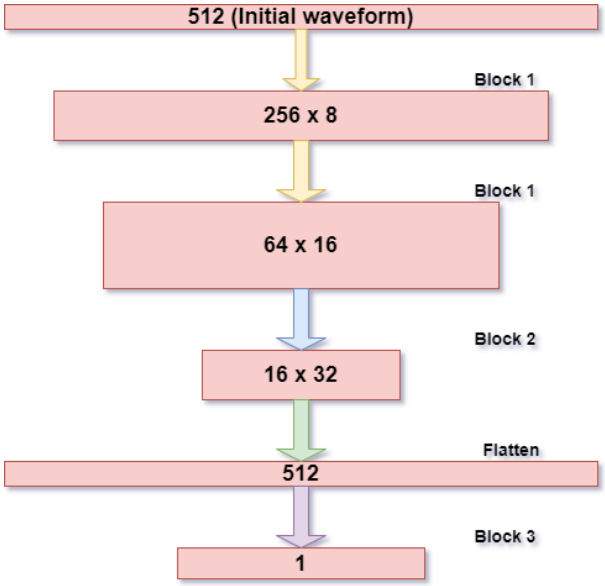


Figura 4.19: Arquitectura CNN estilo VGG. Fuente: Elaboración propia

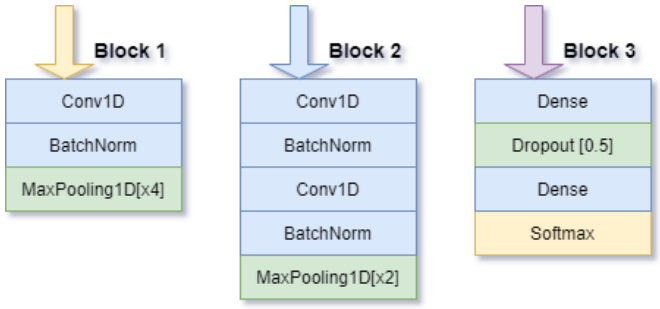


Figura 4.20: Descripción de bloques de arquitectura CNN estilo VGG. Fuente: Elaboración propia

Se procede a un nuevo experimento con el mismo tamaño de la ventana deslizante y cantidad de épocas que el experimento 2 pero esta vez, dando uso a la nueva arquitectura diseñada. Este experimento se detuvo entorno a los 263 “epoch” debido a un “early stopping” y obtuvo un 85.22 % de accuracy, al observar el grafico del accuracy se puede identificar que la curva de validación pasa por encima de la de entrenamiento.

De la gráfica del loss puede identificarse que existe un menor loss para el conjunto de validación que para el entrenamiento. En este caso, indica que el conjunto de datos de validación puede ser más fácil de predecir para el modelo que el conjunto de datos de entrenamiento.

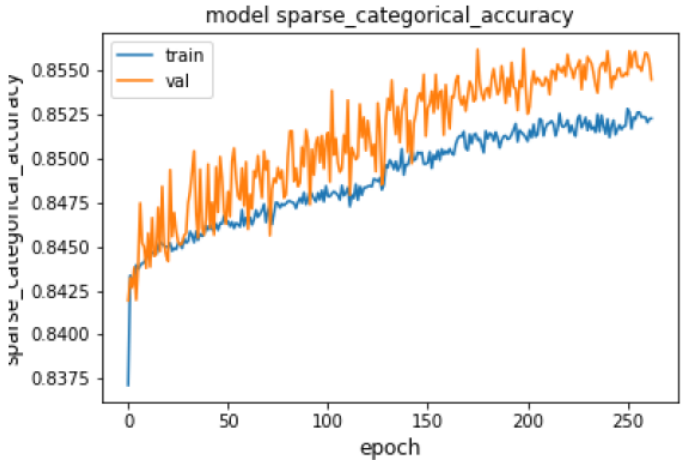


Figura 4.21: Accuracy CNN arquitectura VGG. Fuente: Elaboración propia

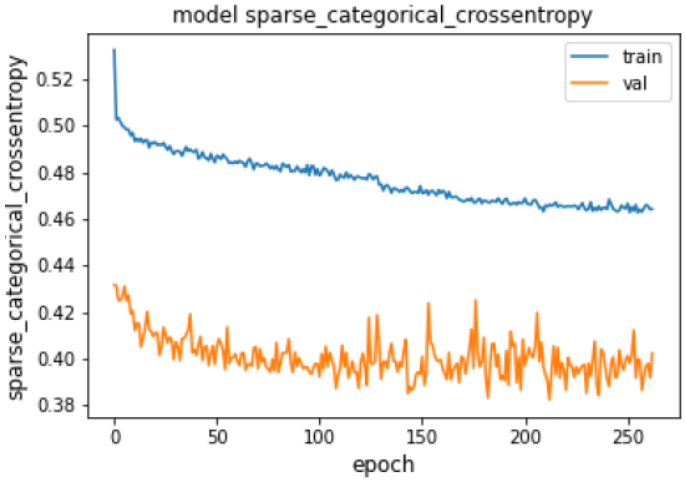


Figura 4.22: Loss CNN arquitectura VGG. Fuente: Elaboración propia

Al observar los resultados de la matriz de confusión se puede notar una leve mejora

debido a la disminución de falsos positivos (lo que implica un aumento en los valores de true positive). Además, estos resultados se obtuvieron en una menor cantidad de épocas, por lo que el tiempo de entrenamiento disminuyo considerablemente.

		PREDICCIÓN	
		0	1
REALIDAD	0	128297	2080
	1	23895	405

Figura 4.23: Matriz de Confusión CNN arquitectura VGG. Fuente: Elaboración propia

$$Accuracy = \frac{405 + 128297}{405 + 128297 + 2080 + 23895} = \frac{128702}{154677} = 0{,}832 \tag{4.5}$$

$$Precision = \frac{405}{405 + 2080} = \frac{405}{2485} = 0{,}163 \tag{4.6}$$

$$Recall = \frac{405}{405 + 23895} = \frac{405}{24300} = 0{,}017 \tag{4.7}$$

$$F1 - measure = 2 \cdot \frac{0{,}017 \cdot 0{,}163}{0{,}017 + 0{,}163} = 2 \cdot \frac{0{,}002771}{0{,}18} = 0{,}031 \tag{4.8}$$

A pesar de obtener un valor de accuracy bastante alto, $F1 - measure$ nos indica que los resultados no fueron buenos. Esta métrica de desempeño se utiliza para combinar las medidas de $Precision$ y $Recall$ en un sólo valor, lo que nos facilita el trabajo de comparar el rendimiento combinado de la precisión y la exhaustividad entre varias soluciones. El mejor valor que podemos obtener a partir de $F1 - measure$ es 1 y el peor es 0. En este caso el valor obtenido fue 0,031, lo cual es bastante cercano a 0. Esto nos indica que tanto la calidad del modelo entrenado como la cantidad de fallas que fue capaz de identificar, son bajas.

4.7. Long Short-Term Memory (LSTM)

Tras una serie de experimentos en donde se probaron distintas arquitecturas, diferentes optimizadores y distintos funciones de loss y de accuracy, se decide cambiar el tipo de red neuronal por una LSTM.

Estas redes tienen ventaja en clasificación de series de tiempo, ya que al ser de tipo recurrentes permiten que la información pueda persistir introduciendo bucles en el diagrama de la red, por lo que, básicamente, pueden «recordar» estados previos y utilizar esta información para decidir cuál será el siguiente estado. Esta característica permite manejar redes cronológicas de mejor manera y nos ahorra el hecho de tener que utilizar una ventana deslizante.

En particular esto encaja con el desarrollo del proyecto porque dentro de este se trabaja con estados de operación de la red eléctrica y cada dato se encuentra espaciado cada 15 minutos, y una falla puede llegar a durar varias horas.

4.7.1. Pre-procesamiento para LSTM

Para el pre-procesamiento en este caso ya que la temporalidad de los datos es una tarea que ya realiza de por sí la LSTM, no será necesario proceder a una técnica como la ventana deslizante donde se encargue de enfatizar el cambio de tendencia para el entrenamiento de la red neuronal, por lo que en este caso se utilizará las lecturas de los transformadores con la columna extra 'falla' encargada de indicar el etiquetado de los datos.

	kW_del	kVAr_del	kW_rec	kVAr_rec	tension	falla
0	9028.498047	993.392395	0.0	0.000000	12166.338867	0
1	8570.544922	936.934265	0.0	0.000000	12191.869141	0
2	8488.869141	875.079468	0.0	0.000000	12209.926758	0
3	8687.238281	881.507874	0.0	0.000000	12198.285156	0
4	8843.103516	930.686401	0.0	0.000000	12193.687500	0
...	...	...	...	...	...	...
154672	8532.578125	0.000000	0.0	558.630371	0.000000	0
154673	8303.352539	0.000000	0.0	563.043518	0.000000	0
154674	7994.871582	0.000000	0.0	586.591919	0.000000	0
154675	7681.770020	0.000000	0.0	663.821289	0.000000	0
154676	7403.301270	0.000000	0.0	727.333801	0.000000	0

Figura 4.24: Etiquetado de lecturas de Transformador 1. Fuente: Elaboración propia

	kW_del	kVAr_del	kW_rec	kVAr_rec	falla
0	5984.583984	707.117981	0.0	0.0	1
1	5961.703613	673.666931	0.0	0.0	1
2	5908.790039	662.464294	0.0	0.0	1
3	5937.251465	649.111389	0.0	0.0	1
4	5948.978516	634.345276	0.0	0.0	1
...	...	...	...	...	...
163515	10451.503906	1794.065063	0.0	0.0	0
163516	9940.391602	1671.242065	0.0	0.0	0
163517	9574.870117	1613.777710	0.0	0.0	0
163518	9189.042969	1571.922852	0.0	0.0	0
163519	8785.914062	1512.587402	0.0	0.0	0

Figura 4.25: Etiquetado de lecturas de Transformador 2. Fuente: Elaboración propia

4.7.2. Experimentos LSTM

A en la figura 4.26 se muestra la arquitectura inicial que se utilizó para probar este tipo de red. Se puede observar que contiene una capa de Dropout. Esta es una técnica de regularización que se usa para reducir el sobreajuste omitiendo neuronas ocultas o visibles de manera aleatoria durante el proceso de entrenamiento de la red neuronal.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm (LSTM)                  (None, 5, 128)            66560
_________________________________________________________________
dropout (Dropout)            (None, 5, 128)            0
_________________________________________________________________
lstm_1 (LSTM)                (None, 128)               131584
_________________________________________________________________
dense (Dense)                (None, 64)                8256
_________________________________________________________________
dropout_1 (Dropout)          (None, 64)                0
_________________________________________________________________
dense_1 (Dense)              (None, 10)                650
=================================================================
Total params: 207,050
Trainable params: 207,050
Non-trainable params: 0
_________________________________________________________________
None
```

Figura 4.26: Arquitectura inicial LSTM. Fuente: Elaboración propia

Como resultado del entrenamiento se generaron los siguientes gráficos:

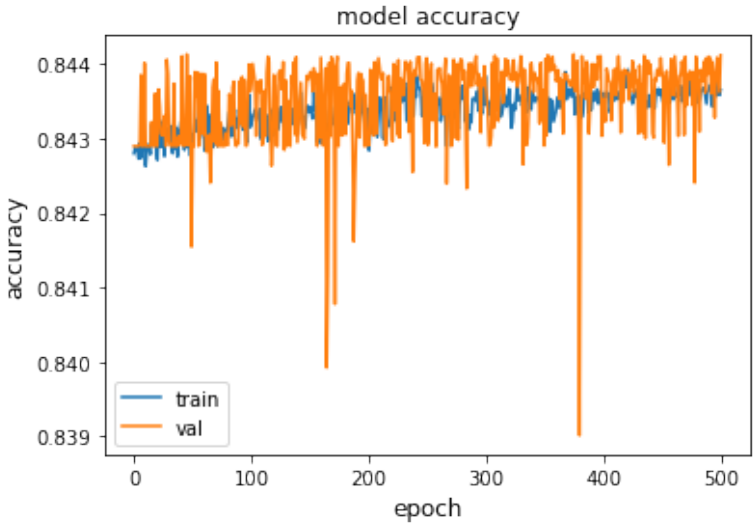


Figura 4.27: Accuracy LSTM Experimento N°1. Fuente: Elaboración propia

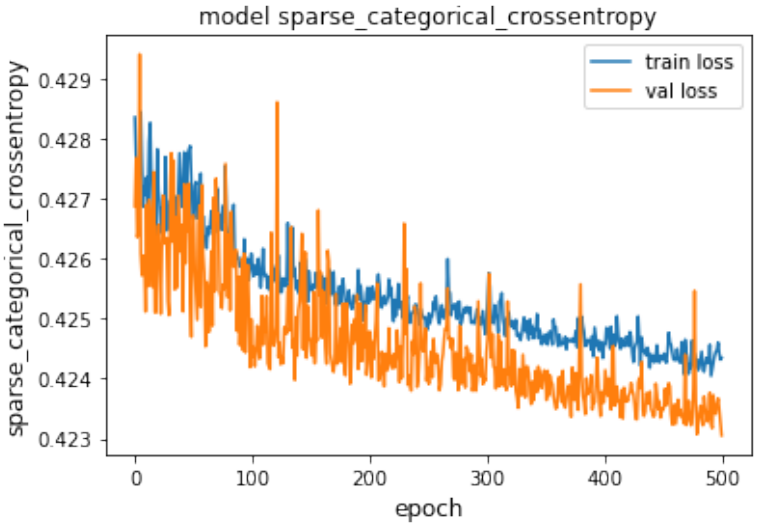


Figura 4.28: Loss LSTM Experimento N°1. Fuente: Elaboración propia

Al observar la curva del loss se puede ver que, la curva de validación pasa un poco por encima de la de entrenamiento lo que normalmente indica que los datos de entrenamiento son más difíciles de modelar que los de validación, por lo tanto, este modelo puede no ser lo suficientemente bueno. Como mejora se plantea un cambio en los valores de los "dropout", más específicamente aumentarlo a un valor entre 0,5 y 0,9 lo que hace que, durante el entrenamiento, un porcentaje de las funciones se establece en cero. Durante la prueba, se utilizan todas las neuronas lo que hace que el modelo durante la prueba sea

más robusto con el fin de intentar evitar el “overfitting”.

Luego de obtener los primeros resultados, surge la observación de que reemplazar los valores de NaN por 0´s (como se vio en la sección 4.5.1) es un error debido a que la no existencia de esta no es lo mismo a que su valor sea 0, por lo que se procede a eliminar la columna de tensión del dataset para los futuros experimentos.

Además, el hecho de que las fallas sean una clase minoritaria en el dataset claramente esta afectando a los resultados obtenidos por el modelo, ya que al haber una cantidad tan minúscula de fallas el modelo tiende a reconocer la mayoría de los datos como no fallas, lo que genera un accuracy muy alto pero la cantidad de fallas identificadas correctamente son muy bajas, lo que implica que estamos presentes ante un problema de data desbalanceada, por lo que se agregan pesos a las clases para aplicar una ayuda extra al entrenamiento.

Por ultimo se modifica la arquitectura del modelo con el fin de darle una mayor profundidad agregando un par de capas en la LSTM.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm_2 (LSTM)                (None, 4, 64)             16896
_________________________________________________________________
dropout_2 (Dropout)          (None, 4, 64)             0
_________________________________________________________________
lstm_3 (LSTM)                (None, 32)                12416
_________________________________________________________________
dense_2 (Dense)              (None, 16)                528
_________________________________________________________________
dropout_3 (Dropout)          (None, 16)                0
_________________________________________________________________
dense_3 (Dense)              (None, 2)                 34
=================================================================
Total params: 29,874
Trainable params: 29,874
Non-trainable params: 0
_________________________________________________________________
None
```

Figura 4.29: Arquitectura LSTM Mejor Experimento

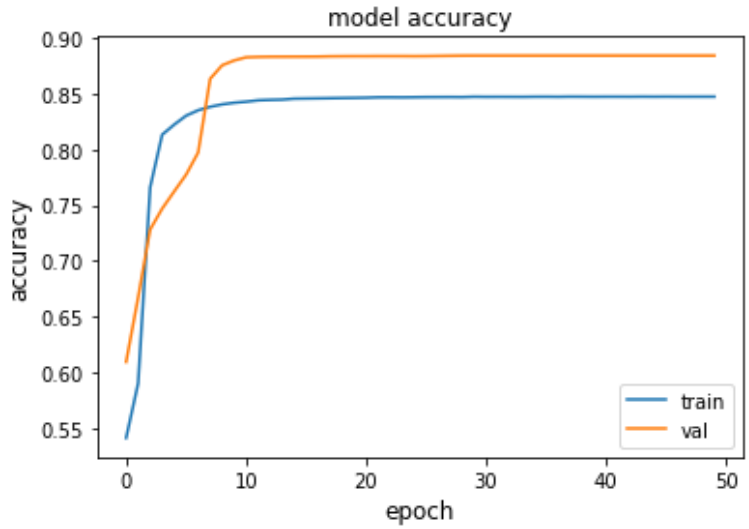


Figura 4.30: Accuracy LSTM Mejor Experimento. Fuente: Elaboración propia

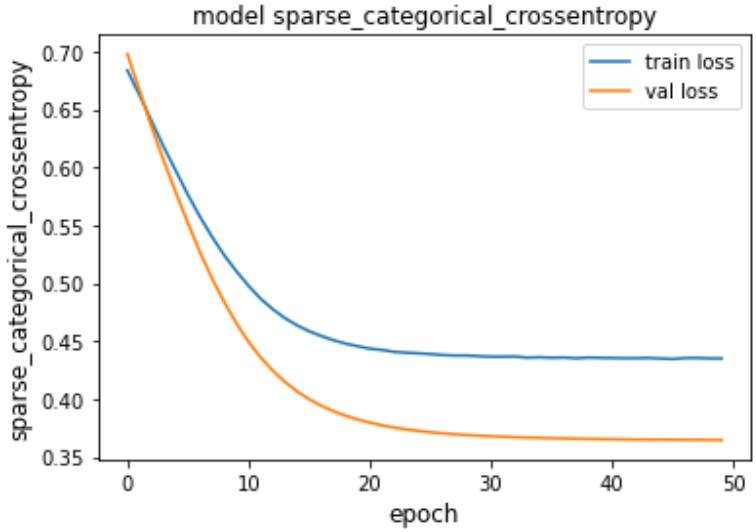


Figura 4.31: Loss LSTM Mejor Experimento. Fuente: Elaboración propia

$$Accuracy = \frac{53 + 28837}{53 + 28837 + 77 + 3737} = \frac{28890}{32704} = 0{,}883 \tag{4.9}$$

$$Precision = \frac{53}{53 + 77} = \frac{53}{130} = 0{,}41 \tag{4.10}$$

		PREDICCIÓN	
		0	1
REALIDAD	0	28837	77
	1	3737	53

Figura 4.32: Matriz de Confusión LSTM Mejor Experimento. Fuente: Elaboración propia

$$Recall = \frac{53}{53 + 3737} = \frac{53}{3790} = 0{,}014 \tag{4.11}$$

$$F1 - measure = 2 \cdot \frac{0{,}014 \cdot 0{,}41}{0{,}014 + 0{,}41} = 2 \cdot \frac{0{,}0574}{0{,}55} = 0{,}21 \tag{4.12}$$

Este último experimento se repitió varias veces, en donde se probo cambiando las siguientes variables:

- Los valores de Dropout.
- El optimizador utilizado.
- Las funciones de activación distintas a ReLu(como Sigmoid).
- Distintas formas de balancear los pesos del par de clases presentes en el dataset.

Lamentablemente los resultados anteriores fueron los mejores obtenidos, donde queda claro que no satisface los requerimientos mínimos de detección de fallas necesarios para ser implementados en un sistema automatizado.

4.8. Random Forest (RF)

Debido a la baja calidad de resultados obtenidos por las redes neuronales en el uso de deep learning, se opta por dar uso a machine learning clásico, específicamente el algoritmo de clasificación Random Forest, donde si bien los valores de accuracy y precisión probablemente sean buenos, no se podrán mejorar mucho mas de lo que obtenga en primera instancia. Debido a que la correlación entre las características de la serie de tiempo de naturaleza continua afecta negativamente al rendimiento del modelo.

Para el uso de Random Forest con el datset del transformador T1 se obtuvo los siguientes resultados:

- Recall Baseline: 1.0 Test: 0.1 Train: 0.94
- Precision Baseline: 0.16 Test: 0.34 Train: 1.0
- Roc Baseline: 0.5 Test: 0.62 Train: 1.0

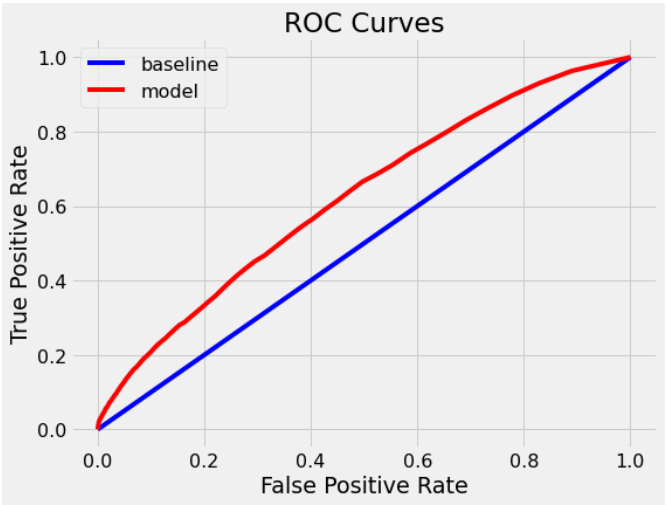


Figura 4.33: Curva ROC para T1. Fuente: Elaboración propia

		PREDICCIÓN	
		0	1
REALIDAD	0	37660	1454
	1	6545	745

Figura 4.34: Matriz de confusion para Random forest con T1. Fuente: Elaboración propia

Para el uso de Random Forest con el datset del transformador T2 se obtuvo los siguientes resultados:

- Recall Baseline: 1.0 Test: 0.08 Train: 1.0
- Precision Baseline: 0.15 Test: 0.31 Train: 1.0
- Roc Baseline: 0.5 Test: 0.61 Train: 1.0

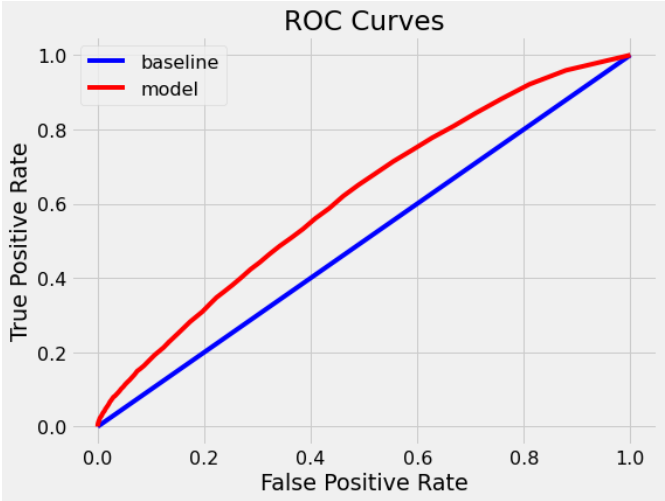


Figura 4.35: curva ROC para T2. Fuente: Elaboración propia

		PREDICCIÓN	
		0	1
REALIDAD	0	40689	1238
	1	6545	566

Figura 4.36: Matriz de confusion para Random forest con T2. Fuente: Elaboración propia

Como se podrá apreciar en ambos resultados la calidad del reconocimiento de las fallas no es muy bueno, obteniendo un 34 % de reconocimiento correcto de fallas como el mejor resultado.

Capítulo 5

Alcance del proyecto y trabajo a futuro

En este capítulo se explica el alcance que se tuvo en este proyecto en relación a el proceso completo de minería de datos, detallando cada una de sus secciones y como el desarrollo de secciones anteriores afectaron el resultado de esta memoria. Finalmente se detallan las opciones del trabajo a futuro que permiten darle continuidad a este proyecto.

5.1. Alcance del proyecto

De los pasos especificados anteriormente en la sección 4.1, esta memoria se centró en **Selection & Cleaning**, **Preprocessing**, **Feature Engineer** y **Machine learning**, mientras que la empresa Chilquinta realiza los procesos de **Interpretation & Evaluation** como del **ETL**. Esta última sección mencionada condujo los experimentos desarrollados de machine learning y deep learning a resultados no esperados.

En la siguiente sección se explica el proceso realizado por la empresa y como esto produce una reducción en la fiabilidad de los datos del libro de fallas, afectando directamente al proceso **ETL** y así a la cadena completa del flujo de minería de datos.

5.1.1. Proceso de reporte de fallas actual

El proceso de reporte de fallas actualmente en Chilquinta es manual. En otras palabras, al ocurrir una falla de tipo 2 específicamente, un funcionario es enviado para verificar visualmente la falla. En ese momento se reporta a la central el estado de la red, ingresando de esta manera, el inicio del estado de falla de la red eléctrica en el libro de fallas, creado por el área de operaciones.

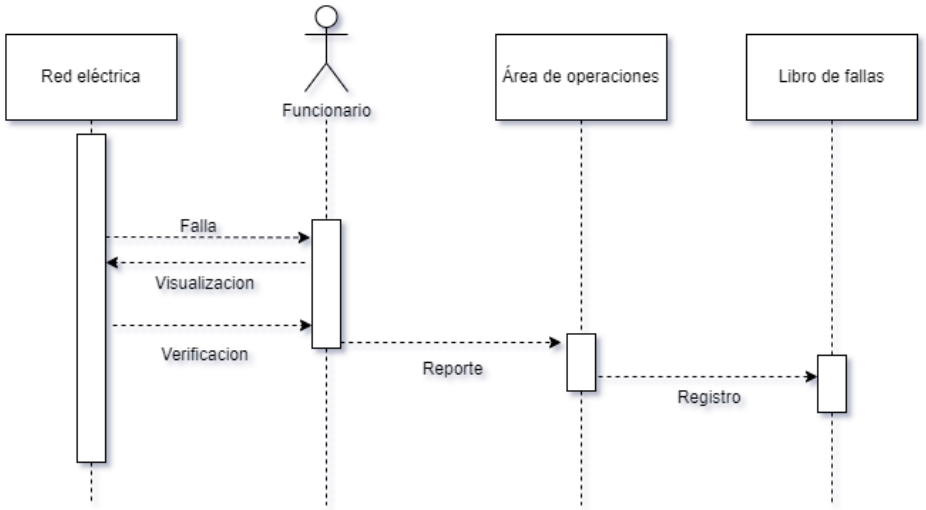


Figura 5.1: Diagrama de interacciones de reporte de falla. Fuente: Elaboración propia

5.2. Recomendaciones

Debido a problemas como los explicados en la sección 5.1.1 y otros inconvenientes encontrados durante el progreso de esta memoria, a continuación se muestra una serie de recomendaciones hacia la empresa para fortalecer la base de la construcción de sus proyectos relacionados a manejo de datos e inteligencia artificial.

Estandarización de los registros: es necesario que los registros usados cuenten con estructuras repetibles entre las distintas áreas de la empresa y con su respectiva documentación en dónde se explique el tipo de información que debe ser ingresada y la forma en que se ingresará.

Ejemplo de esto es que para el entrenamiento y uso de la red neuronal, es necesario ingresar un dataset con la misma estructura para que reconozca los patrones existentes en el entrenamiento y que luego pueda aplicarlos a los nuevos registros.

Área TI interna o subcontratada: El desarrollo de una red neuronal, dependiendo de su complejidad, siempre va asociado a un tiempo desarrollo extenso, ya que se debe conseguir un modelamiento en base a las necesidades del cliente, y en el caso de iniciar sin una previa base, se extiende más el tiempo. Por lo anterior es requerido un equipo de desarrollo que haga seguimiento constante al entrenamiento y las problemáticas que surjan en este, volviendo necesaria la existencia de esta área en la empresa o en su defecto la externalización de esta.

Esto se ve reflejado en que si bien no se pudo acercar a una predicción de falla del 100 %, el inconveniente de contar con información con desfases son el tipo de problemas que ocurren en los desarrollos informáticos y extienden los tiempos de trabajo.

Registros temporales exactos: El uso de registros temporales es una herramienta que permite analizar la ciclicidad de hechos ocurridos y la relación con los valores previos, los cual es necesario para la incorporación específica de inteligencias artificiales, y en la mayoría de herramientas predictivas, por lo que la existencia de desfases en los registros imposibilita su uso.

Stack de tecnlogías: Durante el período de incersión que se tuvo con la empresa, se indicó que los datos de alimentadores eran almacenados en una base de datos relacional, específicamente PostgreSQL. Este tipo de base de datos tiene la característica de ser relacional, lo cual no es lo indicado, ya que el alimentador entregará información de manera continua a una base de datos que no esta diseñada para este tipo de problemas. Esto puede generar, a largo plazo, una sobre carga en tiempos de computo y una baja en cuanto a su rendimiento. Lo anteriormente explicado, era evidenciado al momento de solicitar data histórica sobre los alimentadores conectados a la subestación Miraflores ya que, los computadores de los empleados se ”congelaban” en el momento que revisaban data mas antigua que 5 años.

Por otro lado al haber una falta de comunicación entre las distintas áreas de la empresa se produce una cierta independencia de cada una de ellas. Esto genera que cada una establezca sus propios estándares en sus registros, definan sus propios stacks de tecnología y sus propias maneras de entregar sus resultados, transformándose así cada una de estas áreas en una ”caja negra” donde solo se conoce el resultado final entregado por esta caja y si se desea utilizar el mismo, se debe modificar para ser usado por otra área que lo necesite.

Capítulo 6

Conclusiones

El exhaustivo proceso de experimentación, en donde se probaron diferentes arquitecturas, hiperparámetros, funciones de activación, funciones de pérdida y optimizadores para los distintos modelos de deep learning, además de distintos algoritmos de machine learning, resultó en un poco alentador resultado en cuanto a la efectiva clasificación de las fallas en la red eléctrica de la quinta región. Sin embargo, esto permitió detectar una incongruencia de los datos entregados por la empresa en el libro de fallas.

Este problema tiene origen en el proceso actual que tiene la empresa en ingresar y reportar las fallas de la red. Al reportar las fallas de la red eléctrica al sistema con un tiempo de desfase del momento en que ocurrió la misma, se pierde un fragmento de información relevante, lo que se traduce a que la gran mayoría de las fallas se encuentran incompletas en cuanto al registro de su duración, lo que afecta al preprocesamiento de la data, mas específicamente, al proceso de etiquetado de las lecturas de los transformadores, ya que produce que lecturas que están efectivamente en un estado de falla, no se etiqueten como una. Esto afecta directamente a la calidad del entrenamiento del modelo y, por ende, en los resultados entregados.

Referencias

[1] Chilquinta Energia S.A chilquintaenergia.cl https://www.chilquintaenergia.cl/informacion-corporativa

[2] Bravo, Diego Lozano, Carlos. (2021). Dataset of Distribution Transformers for Predictive Maintenance. Data in Brief. 38. 1-4. 10.1016/j.dib.2021.107454.

[3] Ogar, V.N.; Hussain, S.; Gamage, K.A.A. Transmission Line Fault Classification of Multi-Dataset Using CatBoost Classifier. Signals 2022, 3, 468–482. https://doi.org/10.3390/signals3030027

[4] Wu, T., Tu, G., Bo, Z. Q., Klimek, A. (2007). Fuzzy Set Theory and Fault Tree Analysis based Method Suitable for Fault Diagnosis of Power Transformer. 2007 International Conference on Intelligent Systems Applications to Power Systems. doi:10.1109/isap.2007.4441664

[5] Zamboni, L., Nunes da Silva, I., Nascimento Soares, L., Souza Fernandes, R. A. (2011). Fault Detection in Power Distribution Systems Using Automated Integration of Computational Intelligence Tools. IEEE Latin America Transactions, 9(4), 522–527. doi:10.1109/tla.2011.5993738

[6] Yang, H., Touria, E.-M., Edrington, C. (2016). Real-time sensor fault detection and isolation in power system with hardware implementation. 2016 North American Power Symposium (NAPS). doi:10.1109/naps.2016.7747936

[7] Mohammadi, F., Nazri, G.-A., Saif, M. (2019). A Fast Fault Detection and Identification Approach in Power Distribution Systems. 2019 International Conference on Power Generation Systems and Renewable Energy Technologies (PGSRET). doi:10.1109/pgsret.2019.8882676

[8] Jia Qingquan, Yang Qixun, Yang Wei, Yang Yihan, Song Jiahua. (n.d.). Multi-criteria relaying strategy for single phase to ground fault in MV power systems. Proceedings. International Conference on Power System Technology. doi:10.1109/icpst.2002.1047485

[9] L. Zhang, J. Lin, B. Liu, Z. Zhang, X. Yan and M. Wei, .A Review on Deep Learning Applications in Prognostics and Health Management,ïn IEEE Access, vol. 7, pp. 162415-162438, 2019, doi: 10.1109/ACCESS.2019.2950985.

[10] A.Daneels and W.Salter, "What is Scada?", International Conference on Accelerator and Large Experimental Physics Control System, Trieste, Italy, 1999, pp.1

[11] G. Yue, G. Ping and L. Lanxin, "An End-to-End model based on CNN-LSTM for Industrial Fault Diagnosis and Prognosis," 2018 International Conference on Network Infrastructure and Digital Content (IC-NIDC), 2018, pp. 274-278, doi: 10.1109/ICNIDC.2018.8525759.

[12] H. M. Rai, K. Chatterjee and C. Mukherjee, "Hybrid CNN-LSTM model for automatic prediction of cardiac arrhythmias from ECG big data,"2020 IEEE 7th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON), 2020, pp. 1-6, doi: 10.1109/UPCON50219.2020.9376450.

[13] P. -J. Lin et al., ÇNN-Based Prognosis of BCI Rehabilitation Using EEG From First Session BCI Training,ïn IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 29, pp. 1936-1943, 2021, doi: 10.1109/TNSRE.2021.3112167.

[14] A. Vijayalakshmi, M. Shahaana, N. C. D. Nivetha and K. Subramaniam, "Development of Prognosis Tool for Type-II Diabetics using Tongue Image Analysis,"2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020, pp. 617-619, doi: 10.1109/ICACCS48705.2020.9074437.

[15] N. Arya and S. Saha, "Multi-Modal Classification for Human Breast Cancer Prognosis Prediction: Proposal of Deep-Learning Based Stacked Ensemble Model,ïn IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 19, no. 2, pp. 1032-1041, 1 March-April 2022, doi: 10.1109/TCBB.2020.3018467.

[16] Barua, Arnab. (2020). Human Activity Recognition in Prognosis of Depression Using Long Short-Term Memory Approach. International Journal of Advanced Science and Technology. 29.

[17] C. Darab, R. Tarnovan, A. Turcu and C. Martineac, .Artificial Intelligence Techniques for Fault Location and Detection in Distributed Generation Power Systems,"2019 8th International Conference on Modern Power Systems (MPS), 2019, pp. 1-4, doi: 10.1109/MPS.2019.8759662.

[18] M. Alali, F. N. Shimim, Z. Shahooei and M. Bahramipanah, Ïntelligent Line Congestion Prognosis in Active Distribution System Using Artificial Neural Network,"2021 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT), 2021, pp. 1-5, doi: 10.1109/ISGT49243.2021.9372244.

[19] Y. Huang, P. Zhao and Y. Jiang, "Data-driven fault risk warning method for distribution system,"2021 China International Conference on Electricity Distribution (CICED), 2021, pp. 143-147, doi: 10.1109/CICED50259.2021.9556654.

[20] C. M. Furse, M. Kafal, R. Razzaghi and Y. -J. Shin, "Fault Diagnosis for Electrical Systems and Power Networks: A Review,ïn IEEE Sensors Journal, vol. 21, no. 2, pp. 888-906, 15 Jan.15, 2021, doi: 10.1109/JSEN.2020.2987321.

[21] T. Wu, G. Tu, Z. Q. Bo and A. Klimek, "Fuzzy Set Theory and Fault Tree Analysis based Method Suitable for Fault Diagnosis of Power Transformer,"2007 International Conference on Intelligent Systems Applications to Power Systems, 2007, pp. 1-5, doi: 10.1109/ISAP.2007.4441664.

[22] L. Zamboni, I. Nunes da Silva, L. Nascimento Soares and R. A. Souza Fernandes, "Fault Detection in Power Distribution Systems Using Automated Integration of Computational Intelligence Tools,ïn IEEE Latin America Transactions, vol. 9, no. 4, pp. 522-527, July 2011, doi: 10.1109/TLA.2011.5993738.

[23] Shabanzadeh, Morteza & Moghaddam, Mohsen. (2013). What is the Smart Grid? Definitions, Perspectives, and Ultimate Goals. Power System Conference. 10.13140/2.1.2826.7525.

[24] Mar, Adriana & Pereira, Pedro & Martins, João. (2019). A Survey on Power Grid Faults and Their Origins: A Contribution to Improving Power Grid Resilience. Energies. 12. 4667. 10.3390/en12244667.

[25] Martinez. H (9 de octubre de 2020) IArtificial.net https://www.iartificial.net/precision-recall-f1-accuracy-en-clasificacion/more-1845

[26] F. J. Ariza-Lopez, J. Rodriguez-Avi and M. V. Alba-Fernandez, Çomplete Control of an Observed Confusion Matrix,ÏGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium, 2018, pp. 1222-1225, doi: 10.1109/IGARSS.2018.8517540.

[27] Y. Xiong, "Building text hierarchical structure by using confusion matrix,"2012 5th International Conference on BioMedical Engineering and Informatics, 2012, pp. 1250-1254, doi: 10.1109/BMEI.2012.6513202.

[28] Zhang Jialu,Qi Shiqian,Yu ge.Assessment methods of speech synthesis systems for Chinese,Acta Acustica,Vol.23,No.1,1998: 19-30

[29] S. L. Andresen, "John McCarthy: father of AI,ïn IEEE Intelligent Systems, vol. 17, no. 5, pp. 84-85, Sept.-Oct. 2002, doi: 10.1109/MIS.2002.1039837.

[30] Russell, Stuart J. (Stuart Jonathan). (2010). Artificial intelligence : a modern approach. Upper Saddle River, N.J. :Prentice Hall,

[31] Brown, S. (2021, Apr 21). "Machine Learning explained".MIT.https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained"

[32] L.White, Richard,(16 aug 1996) "Steps in Developing a Classifier".

[33] IBM Cloud Education (14, sep 2020). Recurrent Neural Networks".IBM.https://www.ibm.com/cloud/learn/recurrent-neural-networks

[34] Fang, Y., Ma, Z., Zhang, Z., Zhang, X. Y., Bai, X. (2017, August). Dynamic Multi-Task Learning with Convolutional Neural Network. In IJCAI (pp. 1668-1674).

[35] Sadouk, L. (2018). CNN Approaches for Time Series Classification. In (Ed.), Time Series Analysis - Data, Methods, and Applications. IntechOpen. https://doi.org/10.5772/intechopen.81170

[36] L. Chen and G. Lin, .Extending Sliding-Window Semantics over Data Streams,"2008 International Symposium on Computer Science and Computational Technology, 2008, pp. 110-113, doi: 10.1109/ISCSCT.2008.187.

[37] A. Arasu, B. Babcock, S. Babu, J. Cieslewicz, M. Datar, K.Ito, et al,“STREAM: The Stanford Stream Data Manager”, IEEE Data Engineering Bulletin. vol. 26, no. 1, 2003, pp. 19-26.