

Diseño de un modelo de Deep Learning basado en la arquitectura Temporal Fusion Transformer para la predicción horaria a corto plazo de la demanda eléctrica nacional

Silvio Aguilar¹, Santiago Vargas¹, Dr. Diego H. Stalder², Ing. Felix Morales², MSc. Carlos Sauer²

¹Alumno, Facultad de Ingeniería. Universidad Nacional de Asunción. San Lorenzo, Paraguay

²Orientador, Facultad de Ingeniería. Universidad Nacional de Asunción. San Lorenzo, Paraguay

Palabras clave:

- Predicción de demanda eléctrica
- Temporal Fusion Transformer (TFT)

Resumen

La predicción de la demanda eléctrica es una herramienta de importancia para la toma de decisiones en el suministro eléctrico, del cual es responsable la Administración Nacional de Energía (ANDE) dentro del territorio nacional. Este trabajo propone la implementación de un modelo basado en la arquitectura Temporal Fusion Transformer (TFT), el cual es entrenado con datos más actuales que trabajos anteriores y, además, cuenta con una nueva covariable para la predicción basada en los días feriados. Se utilizó la librería de optimización denominada Optuna para la determinación del mejor modelo ajustando los hiperparámetros. Los resultados y errores son analizados de forma trimestral, así como por días de la semana.

Keywords

- Electric demand prediction
- Temporal Fusion Transformer (TFT)

Abstract

Electric demand prediction is an important tool for decision-making in the electricity supply, which is the responsibility of the National Energy Administration (ANDE) within the national territory. This work proposes the implementation of a model based on the Temporal Fusion Transformer (TFT) architecture, which is trained with more recent data than previous works and, in addition, has a new covariate for prediction based on holidays. Frameworks such as Optuna were used in the determination of the best model. The results and errors are analyzed quarterly, as well as by days of the week.

1. Introducción

La Administración Nacional de Energía (ANDE) tiene como una de sus prioridades principales la de satisfacer de forma adecuada las necesidades de energía eléctrica del país, teniendo en cuenta la previsión de la demanda de consumo eléctrico a nivel nacional, factor primordial para poder tomar las mejores y más eficientes decisiones en ámbitos técnicos, administrativos y operativos [1]. Podemos destacar dos principales problemas que se pueden presentar si no se realiza de forma correcta este proceso: por un lado, la adquisición excesiva e innecesaria de energía, que puede desencadenar en pérdidas económicas para la entidad, o por otro lado, no contar con la cantidad necesaria para satisfacer la demanda mínima, repercutiendo directamente en los gastos del consumidor ya que se deberá recurrir a generación de contingencia, e incluso en un caso peor, generar apagones gene-

rales a niveles relativamente grandes. Cabe destacar además que toda la energía eléctrica que no se consume a nivel nacional es cedida o vendida a países vecinos, teniendo en cuenta que estos acuerdos no son necesariamente bien aprovechados en beneficio de nuestro país.

La planificación que realiza la ANDE, en distintas escalas de tiempo (diario, mensual y anual), y para distintos escenarios cuenta con cierta incertidumbre, considerando distintos factores que afectan directa e indirectamente a la demanda que se quiere predecir.

Existen modelos de previsión de carga basados en la estadística, considerados tradicionales, como la regresión lineal, así como modelos semi-analíticos compuestos a partir del perfil de consumo de clientes, e incluso, modelos basados en redes neuronales profundas [2]. Estos modelos fueron utilizados previamente para la predicción de demanda eléctrica, siendo considerados impulsores del estudio en el área.

En el mismo se establece que no existe un consenso que determine el uso particular de un modelo o método por encima de otro, generalmente suele ser resultado de las métricas de desempeño para seleccionar el mejor modelo, por lo cual la práctica común es determinar qué modelo se ajusta mejor para una determinada situación.

Como complemento, el estudio y aplicación de redes neuronales se ha convertido en una atracción para los investigadores del área, ya que han sido utilizadas para predicción de tareas que incluyen el pronóstico de carga a corto plazo (STLF: Short Term Load Forecasting) en microrredes [3]. En 2017 se introduce la arquitectura Transformer por parte de un grupo de investigadores de Google [4], creando una revolución en el área de procesamiento natural del lenguaje, implementando mecanismos de atención que son utilizados para determinar la importancia o peso de una palabra durante el proceso de traducción a otro idioma. La misma tiene una variante que se puede aplicar a la predicción de series temporales, denominada arquitectura Temporal Fusion Transformer (TFT) [5]. Esta última será tomada como línea de base para la realización de este Trabajo Final de Grado.

2. Objetivos

2.1. Objetivo General

Diseñar un modelo inteligente TFT (Temporal Fusion Transformer) para la predicción horaria a corto plazo de la demanda eléctrica nacional.

2.2. Objetivos Específicos

- Actualizar datos históricos de suministro de energía proveída por la ANDE.
- Validar datos de variables meteorológicas a nivel país e imputar datos faltantes.
- Implementar un modelo de predicción de demanda de energía basado en la arquitectura TFT.
- Realizar un ajuste fino de los hiper-parámetros.
- Comparar diferentes modelos basados en la Deep Learning para forecasting (previsión), incluyendo trabajos finales de grado de compañeros de FIUNA relacionados al área.

3. Marco Teórico

3.1. Sector Eléctrico del Paraguay

En el Paraguay, el sector eléctrico está marcado por la Administración Nacional de Electricidad (ANDE), un organismo descentralizado con el monopolio de generación, transmisión, distribución y venta de energía eléctrica. Tres centrales hidroeléctricas clave, Acaray, Itaipú (compartida con Brasil) y Yacyretá

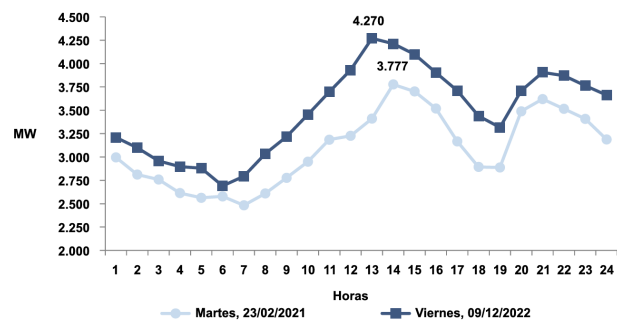


Figura 1: Curvas de Carga del Día de la Demanda Máxima. **Fuente:** [6]

(compartida con Argentina), cubren toda la demanda nacional. Proyectos a corto plazo incluyen nuevas líneas de transmisión para mejorar la cobertura y estabilidad. El mapa eléctrico interno facilita un control operativo efectivo, dividido en sistemas regionales estratégicos [1].

3.2. Demanda Eléctrica en Paraguay

La demanda eléctrica experimenta un crecimiento continuo a nivel nacional, impulsado por factores climáticos y necesidades energéticas. Diferentes escenarios de crecimiento se presentan, con el escenario Medio de Crecimiento proyectando tasas promedio del 6,6 % para la demanda máxima y del 6,8 % para la energía requerida en el Sistema Interconectado Nacional (SIN).

La demanda mensual refleja el patrón climático típico del país, con picos durante los meses más cálidos debido al uso intensivo de sistemas de enfriamiento. La demanda diaria muestra un consumo más alto de lunes a viernes, relacionado con la actividad laboral y escolar, mientras que los fines de semana experimentan reducciones. Se destaca la importancia de considerar estaciones y clima para la planificación eléctrica.

El análisis horario que se puede observar en la Figura 1 revela variaciones marcadas a lo largo del día, con picos durante las horas de elevadas temperaturas y actividades cotidianas. El Sistema Interconectado Nacional (SIN) es esencial para el estudio, representando el sistema eléctrico interno del país. Dos subsistemas, el Sistema Eléctrico Brasileño (SEB) y el Sistema Argentino de Interconexión (SADI), alimentados por diversas hidroeléctricas, son componentes clave del SIN. El parque de generación existente es 100 % hidroeléctrico, con centrales como Acaray que desempeñan un papel fundamental en el suministro eléctrico del país.

3.3. Predicción de la demanda eléctrica

La predicción precisa de la demanda eléctrica emerge como un componente crítico para garantizar

la fiabilidad y eficiencia del suministro eléctrico, además de tener implicaciones significativas en la eficiencia energética, la sostenibilidad y las decisiones gubernamentales. Este trabajo evalúa la eficacia de la arquitectura TFT en esta tarea. La importancia de esta evaluación radica en mejorar predicciones y comprender variables relevantes.

3.4. Modelos Basados en Atención

3.4.1. Atención en Machine Learning

El mecanismo de atención es fundamental en arquitecturas basadas en Transformers, permite focalizarse en partes específicas de datos, asignando diferentes niveles de importancia a elementos individuales del input. Este principio, inspirado en la atención humana, utiliza funciones de compatibilidad y proyecciones para asignar pesos a elementos de entrada [4].

3.4.2. Atención por Producto de Punto Escalado:

Esta función de atención eficaz mide la similitud entre consulta y clave, normalizando las puntuaciones mediante softmax(). La salida ponderada de valores constituye el resultado de la atención.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \quad (1)$$

donde Q (*Query*) como un vector que representa la consulta, K (*Key*) como un conjunto de vectores que representan las claves, V (*Value*) como un conjunto de vectores que representan los valores, y d_k como la dimensión de los vectores clave y valor. Además, se utiliza la función $\text{softmax}()$ como una función de activación que normaliza las puntuaciones de atención para que sumen 1.

3.4.3. Fundamentos de la arquitectura Transformer

La arquitectura Transformer es una arquitectura de redes neuronales que se basa en un mecanismo de atención para aprender relaciones entre los datos, principalmente utilizada para tareas relacionadas al procesamiento natural del lenguaje [4]. La arquitectura Transformer se compone de dos componentes principales como se puede observar en la Figura 2:

- **Encoder:** El encoder es responsable de codificar la entrada en una representación que pueda ser utilizada por el decoder.
- **Decoder:** El decoder es responsable de generar la salida a partir de la representación codificada.

El encoder y el decoder están formados por una serie de capas de atención, que permiten a la red aprender a centrarse en las partes más relevantes de la entrada para realizar una tarea determinada [4]. Entre las capas internas caben destacar las siguientes:

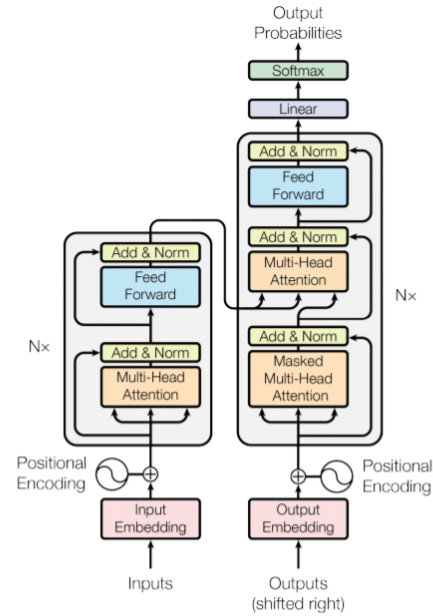


Figura 2: Arquitectura Transformer.

Fuente:[4]

- **Autoatención multicabezal:** La autoatención multicabezal es un mecanismo de atención que permite a una red neuronal aprender relaciones a largo plazo entre los tokens de una secuencia.
- **Conexión residual:** Una conexión residual es una conexión que permite a una red neuronal aprender funciones más complejas saltando capas anteriores de la red.
- **Normalización de capas:** La normalización de capas es una técnica que ayuda a estabilizar el entrenamiento de redes neuronales profundas.
- **Enmascaramiento:** El enmascaramiento es una técnica que se utiliza para evitar que un modelo de aprendizaje automático aprenda relaciones relevantes entre los datos [4].

4. Modelos De Pronóstico de Múltiples Horizontes

Para que el modelo predictivo sea aplicable al uso real es necesario tener estimaciones predictivas en múltiples puntos futuros, permitiendo visualizar tendencias a lo largo de un horizonte futuro y optimizar acciones. Desde un punto de vista estadístico, el pronóstico de múltiples horizontes se considera una modificación del problema de predicción de un paso adelante, representado por la ecuación:

$$y'_{t+\tau} = f(y_{t-k:\tau}, x_{t-k:\tau}, u_{t-k:\tau}, s, \tau), \quad (2)$$

donde $y'_{t+\tau}$ es el valor predicho de la variable objeti-

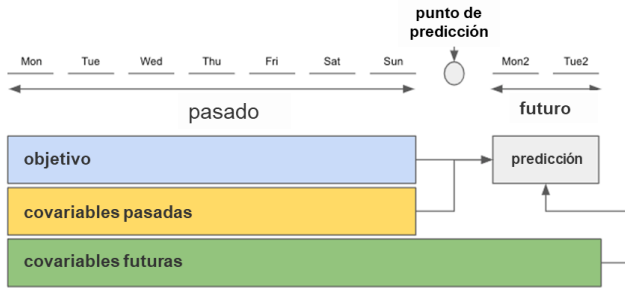


Figura 3: Ilustración de una predicción a múltiples horizontes.

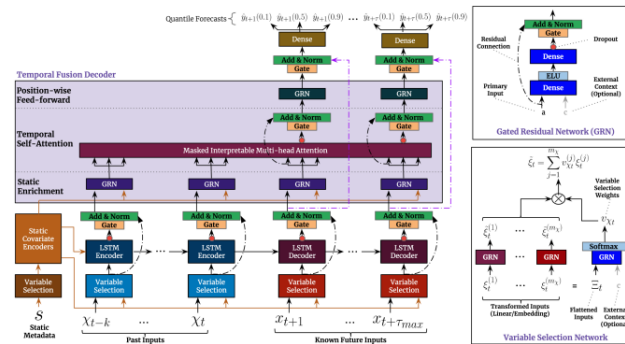


Figura 4: Arquitectura TFT.

vo en el tiempo $t + \tau$, τ es el horizonte de pronóstico discreto, u_t son entradas futuras conocidas, x_t son entradas que solo se pueden observar históricamente, s representa covariables adicionales o información relevante, y f es la función de predicción [7], lo que se busca es predecir una curva en vez de un solo punto como en otros métodos como se puede observar en la Figura 3.

5. Fundamentos teóricos de la Arquitectura TFT

La arquitectura Transformers for Time Series Forecasting (TFT) es una extensión de las redes neuronales basadas en Transformers, que originalmente se desarrollaron para tareas de procesamiento de lenguaje natural. Los Transformers son especialmente adecuados para modelar relaciones en secuencias de datos, lo que los hace prometedores en la predicción de series temporales, de sus componentes observados en la Figura 4 podemos destacar los siguientes componentes [5]:

- **Mecanismos de Control:** TFT utiliza mecanismos de control (gating mechanisms) que permiten omitir componentes no utilizados de la arquitectura, proporcionando adaptabilidad en cuanto a la profundidad y complejidad de la red, haciéndola adecuada para una amplia gama de conjuntos de datos y escenarios.

- **Redes de Selección de Variables:** TFT incorpora redes de selección de variables que eligen las variables de entrada relevantes en cada paso temporal, permitiendo al modelo determinar qué variables son importantes en cada momento, lo que es esencial para la adaptación a datos cambiantes.

- **Codificadores de Covariables Estáticas:** TFT integra codificadores que incorporan características estáticas en la red. Esto se logra mediante la codificación de vectores de contexto que condicionan la dinámica temporal. Las características estáticas, como la ubicación o el tipo de entidad, pueden influir en las tendencias a lo largo del tiempo.

- **Procesamiento Temporal:** TFT se encarga de aprender las relaciones temporales a largo y corto plazo a partir de las entradas variables en el tiempo (observadas y conocidas). Para el procesamiento a nivel local, se emplea una capa secuencia-a-secuencia. Además, TFT utiliza un innovador bloque de atención multi-cabezal interpretable para capturar las dependencias a largo plazo.

- **Intervalos de Predicción:** TFT proporciona intervalos de predicción que indican el rango de valores probables para el objetivo en cada horizonte de predicción. Esto es útil para comprender la incertidumbre en las predicciones y tomar decisiones basadas en riesgos.

6. Metodología Aplicada

El enfoque metodológico empleado para abordar la predicción de la demanda de energía a nivel nacional en este estudio. Se describe de manera exhaustiva el proceso de recolección, preparación y manipulación de los datos provenientes de la Administración Nacional de Electricidad (ANDE), así como la selección de variables y el ajuste de la arquitectura y la evaluación del sistema de predicción. Se presentan las fases clave del preprocesamiento de datos, el diseño experimental, la configuración de los modelos y las estrategias de evaluación utilizadas para validar y optimizar la precisión de las predicciones. La Figura 5 sintetiza los pasos aplicados en cada etapa. Este capítulo proporciona una visión detallada del marco metodológico que sustenta el desarrollo y la evaluación del modelo predictivo de demanda energética.

6.1. Obtención y Preparación de Datos

El conjunto de datos ya preprocesado en gran parte en [8] fue reutilizado y complementado con datos actuales y recientes de la Dirección de Planificación de la ANDE, manteniendo el intervalo horario e

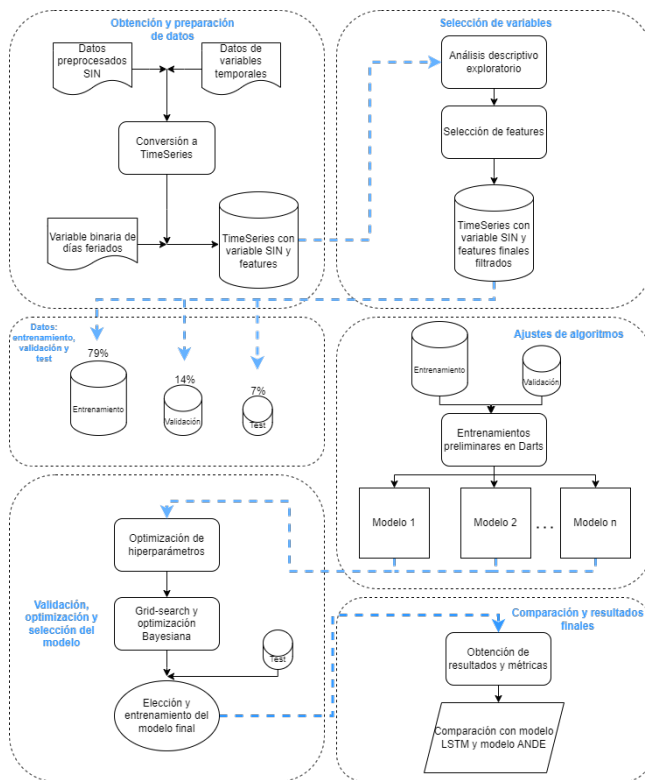


Figura 5: Esquema general de la metodología aplicada.

Fuente: Elaboración propia.

incluyendo los años 2020, 2021 y 2022. Se encuentran agrupadas todas las series relevantes en crudo para su análisis e inclusión en los distintos algoritmos de entrenamiento y validación. Además de la variable objetivo o target, que es el SIN (MW), variables que son consideradas exógenas, que pueden impactar en el desenvolvimiento de la variable objetivo y dar lugar a predicciones más precisas, las variables meteorológicas disponibles que son la temperatura seca del aire y la humedad relativa, de distintas ciudades y sistemas regionales del país y, por otro lado, generamos variables temporales que indican la continuidad del tiempo en distintos intervalos (anual, mensual, semanal).

Las variables exógenas temporales de calendario fueron generadas por código, categorizando las mismas en Año, Mes y Día de la Semana y así poder interpretar su impacto en el consumo eléctrico. La serie correspondiente al Año consta del valor numérico respectivo al año analizado (2009, 2010, 2011, etc). Para los casos del Mes y el Día de la Semana, de forma a obtener una continuación secuencial lógica, en términos cuantitativos, se obtienen sus formas senoidales y cosenoidales presentes en las Figuras 6 y 7.

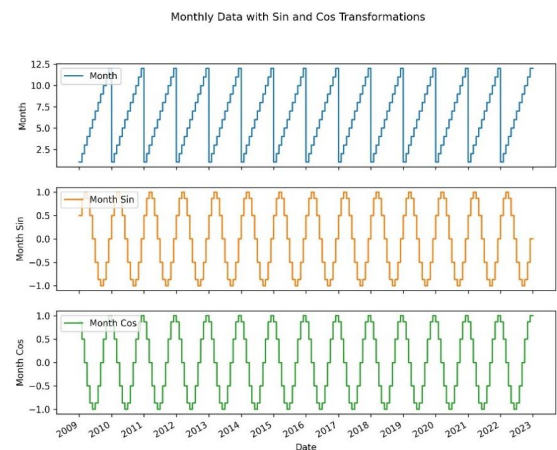


Figura 6: Representación gráfica de la variable Mes y sus transformaciones senoidales y cosenoidal.

Fuente: Elaboración propia.

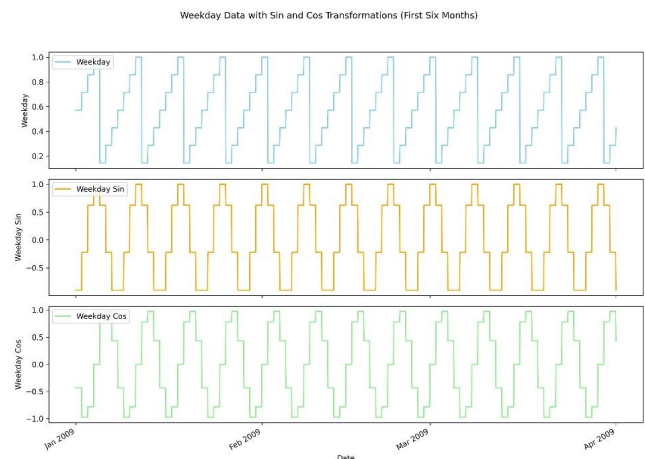


Figura 7: Representación gráfica de la variable Día de la Semana y sus transformaciones senoidales y cosenoidales.

Fuente: Elaboración propia.

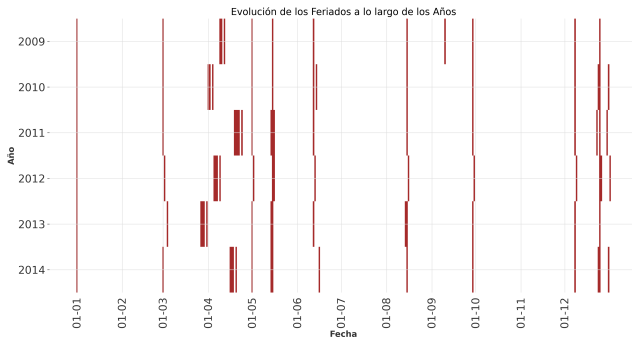


Figura 8: Distribución de los días feriados en los primeros cinco años del conjunto de datos. **Fuente:** Elaboración propia.

Tabla 1: Descripción de hiperparámetros en la arquitectura TFT.

Hiperparámetros	Descripción
input_chunk_length	Número de pasos, en horas, que se mira para atrás
output_chunk_length	Número de pasos, en horas, predichos a partir del punto de predicción
hidden_size	El hiperparámetro principal y común a lo largo de la arquitectura TFT
lstm_layers	Número de capas para los codificadores y decodificadores LSTM
num_attention_heads	Número de cabezas de atención de la arquitectura
dropout	Fracción de neuronas afectadas por el dropout
batch_size	Número de pasos (input y output) utilizado en cada iteración de entrenamiento

Fuente: Elaboración propia.

6.2. Preprocesamiento de Datos

Las tareas necesarias para lograr un preprocesamiento típico a la hora de construir un análisis de datos pueden ser resumidas en cinco principales [9]: (i) Limpieza; (ii) Reducción y Selección; (iii) Escalamiento y normalización; (iv) Transformación; (v) Partición.

La primera tarea ya fue realizada en el trabajo realizado en [8], así como gran parte de la reducción y selección de datos, adoptando intervalos de tiempo idénticos para lograr una comparación más precisa. Sin embargo se agregaron datos actualizados de 2020 al 2022 para extender el conjunto de datos.

6.2.1. Escalamiento y normalización

La importancia de esta etapa radica en verificar y asegurarse que las escalas de las variables de entrada se encuentren en la misma escala, optando por normalizarlas en una escala [0, 1] y así mejorar el rendimiento y estabilidad del entrenamiento del modelo. La normalización de los datos como tal se realiza con métodos proveídos por la librería Darts que cuenta con su escalador y transformador nativo y se realiza en todo el conjunto de datos que abarca desde el 2009 hasta el 2022.

6.2.2. Transformación de Datos

Se realiza una conversión de formato para garantizar la compatibilidad con la librería utilizada en Python. Se pasa de DataFrames a TimeSeries, la clase principal utilizada para el tratamiento de las series temporales en el entrenamiento de los modelos. Además, se incorpora la variable binaria temporal de feriados, que indica la presencia de fechas festivas y su posible impacto en la demanda eléctrica, la consideración de esta variable tiene importancia en este trabajo ya que la demanda eléctrica generalmente se encuentra influenciada por el comportamiento de las personas en estas fechas [10].

6.2.3. Partición de Datos

Adoptamos técnicas tradicionales de ML y dividimos el conjunto de datos en tres partes: entrenamiento, validación y prueba.

El conjunto de entrenamiento se utiliza para entrenar el modelo. El conjunto de validación es utilizado para optimizar la precisión del modelo ajustando los hiperparámetros del modelo. Con el conjunto de prueba realizamos la comparación final de datos predichos con los observados para determinar las métricas correspondientes.

6.3. Modelo TFT implementado con Darts

El desarrollo de un modelo TFT en Darts se basa en la implementación realizada por otra librería conocida: PyTorch. El modelo aplica consultas de atención multi-cabeza a las entradas futuras a partir del conjunto de datos ingresado de forma obligatoria como *future_covariates*. Si se especifican codificadores futuros con el parámetro *add_encoders*, se pueden generar automáticamente variables futuras, lo que permite omitir la entrada del parámetro *future_covariates* al aplicar el entrenamiento (fit) y la predicción (predict).

Por defecto, el modelo usa la probabilidad de *QuantileRegression* (Regresión Cuantil), teniendo la capacidad de realizar predicciones probabilísticas. La arquitectura del modelo cuenta con ciertos hiperparámetros que deben ser elegidos en base a la cantidad de datos que se va a manejar, la potencia computacional y el tiempo disponible, partiendo siempre desde un conjunto de valores iniciales, que se deberán ir ajustando con la realización de múltiples experiencias que puede tomar varios días e incluso semanas. En la Tabla 1 se describen los principales hiperparámetros que se tienen en cuenta para el entrenamiento del modelo, con el nombre utilizado dentro de las funciones de Darts.

Tabla 2: Definición de las covariables.

Covariables	Tipo de Covariable
Temperatura	Covariable futura
Humedad	Covariable futura
Año	Covariable futura
Mes (cosenoidal)	Covariable futura
Mes (senoidal)	Covariable futura
Día de la Semana (cosenoidal)	Covariable futura
Día de la Semana (senoidal)	Covariable futura

Fuente: Elaboración propia.

6.4. Definición de las covariables

La definición de covariables que acompañan a la variable objetivo SIN (MW) en el conjunto de datos. Se identifican un total de siete covariables, las cuales se consideran esenciales para mejorar la precisión de la predicción. Según la clasificación presentada en [11], estas covariables se definen mayormente como "covariables futuras". Este enfoque se basa en la regla que establece que si los valores son conocidos con antelación, entonces se consideran covariables futuras (o también pueden utilizarse como covariables pasadas). En este contexto, al conocer previamente los valores de las covariables para el año 2022, que se utiliza como conjunto de prueba, se clasifican como covariables futuras. La Tabla 2 correspondiente a esta definición se presenta en [11]. Este enfoque se adapta a nuestro horizonte de predicción, que se extiende hasta el año 2022.

6.5. Optimización de velocidad con EarlyStopper

Debido a la arquitectura compleja del modelo TFT, el tiempo de entrenamiento y la posterior optimización de hiperparámetros puede resultar bastante largo, incluso contando con GPUs consideradas potentes como en nuestro caso, al poseer una NVIDIA RTX 3070 Ti (Laptop). Es por eso que la librería Darts acepta como parámetro de entrada un EarlyStopper, una función heredada de PyTorch, que permite controlar la mejoría y convergencia del entrenamiento en cada iteración y parar el mismo en caso de no presentar mejoras, según la función de evaluación definida y la paciencia establecida. Esta función puede ser el valor de pérdida en la validación (`val_loss`), el valor de pérdida en el entrenamiento (`train_loss`), que no es recomendado, o bien, alguna función de métricas importada de PyTorch como el MAPE.

7. Analisis e Interpretacion de los Resultados

Para el primer entrenamiento seleccionamos como función de validación al MAPE (Mean Absolute Percentage Error), a ser comparado cada quince iteraciones de entrenamiento, con un delta mínimo de 0,05, logrando un modelo inicial luego de 5 horas de entrenamiento. Comparando la totalidad de los resul-

Tabla 3: Métricas generales del primer modelo

R2	0,63163
MSE	106746,695
RMSE	326,721 (MW)
MAXERROR	2096,142 (MW)

Fuente: Elaboración propia.

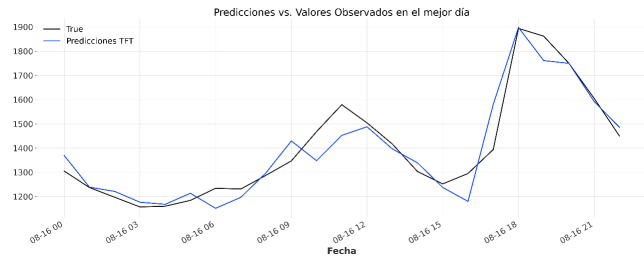


Figura 9: Comparación de valores observados con predichos del mejor día 16-08-2020 para el modelo inicial entrenado.

Fuente: Elaboración propia.

tados se realiza un análisis más profundo para poder encontrar el día con mejor desempeño en el intervalo de validación, dando como resultado la fecha 16-08-2020, según podemos observar en la Figura 13, obteniendo un valor RMSE de 50.2723 (MW). A partir de los primeros resultados, evaluamos la posibilidad de cambiar la función de pérdida para el EarlyStopper y así verificar si hay mejoría y, además, mayor o menor velocidad de convergencia, incluyendo entrenamientos con `val_loss` y `train_loss`. Se observa que con estas dos funciones el modelo empeoró según las métricas, especialmente para el caso de `train_loss`, como se muestra en la Tabla 4 y se observa en la Figura 10, por lo tanto, fijamos al MAPE y re-evaluamos los valores de iteraciones a controlar y los deltas mínimos.

Tabla 4: Comparación de métricas con funciones distintas de validación.

Métrica	Valor con <code>val_loss</code>	Valor con <code>train_loss</code>
R2	0,58931	0,54169
MSE	119013,6	132810
RMSE (MW)	344,983	364,431
MAXERROR (MW)	2096,142	1745,375

Fuente: Elaboración propia.

El modelo tiende a converger más rápido y devolver mejores resultados a medida que se ingresan valores más estrictos para el control según se observa en la Tabla 5, finalizando esta etapa preliminar con un valor de paciencia de 2 y un delta mínimo de 0,0001. Es importante resaltar que estos modelos iniciales fueron entrenados con un valor máximo de 50

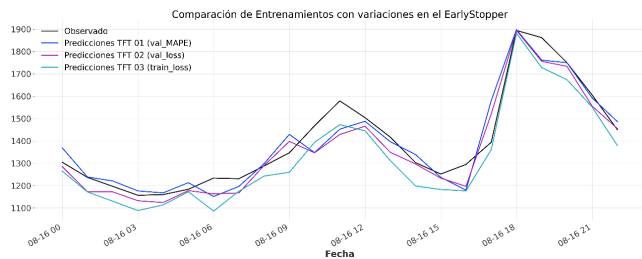


Figura 10: Comparación del mismo modelo con funciones de validación distintas para el EarlyStopper

Fuente: Elaboración propia.

Tabla 5: Comparación de métricas para distintos valores de paciencia y Δ usando el MAPE como función de validación.

Función	Paciencia	Δ	R2	RMSE	MAXERROR
MAPE	20	0.1	0.67	309	1617
	15	0.15	0.68	306	1805
	2	0.0001	0.74	282	1851

Fuente: Elaboración propia.

iteraciones o epochs y en ningún caso se llegó hasta el final debido al EarlyStopper.

7.1. Impacto de la covariable Feriados

Los primeros análisis y entrenamientos de los modelos fueron realizados con las covariables mencionadas en la Tabla 2 sin considerar los días feriados. Una vez determinada la mejor función de convergencia podemos proceder a concatenar esta variable y establecer si el patrón de comportamiento de la misma incide de forma positiva o negativa en las métricas del modelo.

Primeramente, podemos inferir la relación entre la variable Feriados y la variable objetivo SIN utilizando gráficos de cajas mostrado en la Figura 11, que nos permite comparar la tendencia central y la propagación del consumo de energía en días feriados o no feriados. La Figura 12, que nos da una perspectiva acerca de la densidad y estructura de la distribución de los datos del SIN en los mencionados días.

Según lo observado se puede afirmar que en días feriados el consumo a nivel nacional es levemente inferior a los días normales, pero a un nivel que se puede considerar significativo como para ser incluido como covariable dentro del modelo.

Los resultados arrojados se encuentran bastante cercanos al modelo sin Feriados según podemos observar en la Tabla 6. Comparando las métricas analizadas por sobre todo el conjunto de datos de validación existe una diferencia leve pero con valores mayores indicando un rendimiento mínimamente inferior (de aproximadamente unos 7 MW).

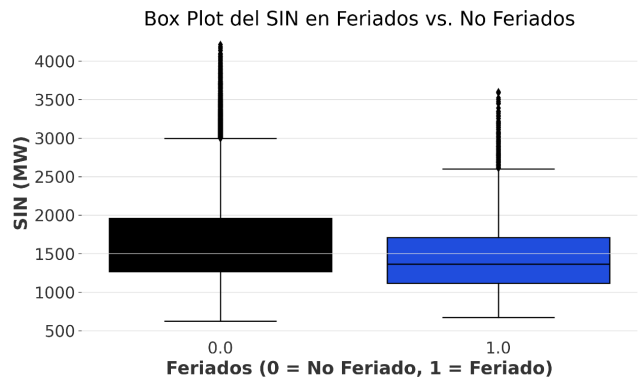


Figura 11: Distribución del SIN en forma de Caja para Feriados y No Feriados

Fuente: Elaboración propia.

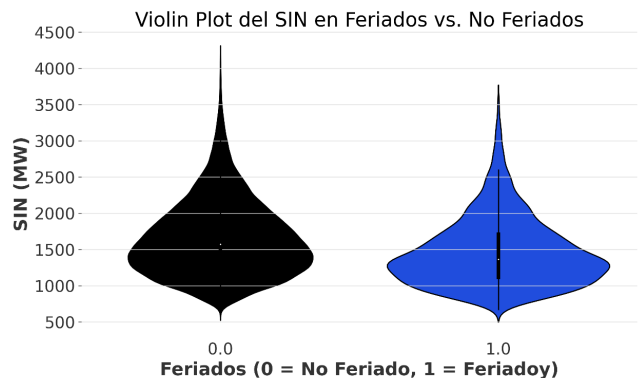


Figura 12: Distribución del SIN en forma de Violin para Feriados y No Feriados

Fuente: Elaboración propia.

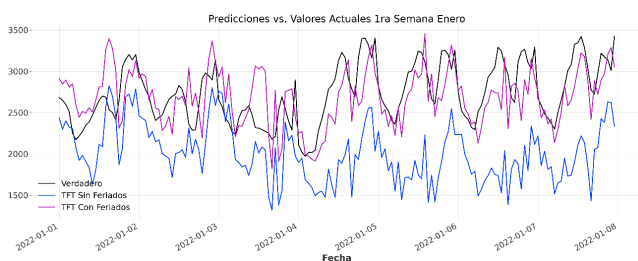


Figura 13: Comparación de resultados para modelo con y sin covariable Feriado en la primera semana de Enero.

Fuente: Elaboración propia.

Tabla 6: Comparación de modelos sin covariable Feriados vs modelo con covariable Feriados.

Métrica	Valor sin Feriados	Valor con Feriados
R2	0,74	0,711
MSE	79900,2	83712,97
RMSE (MW)	282,666	289,34
MAXERROR (MW)	1851,334	1980,201

Fuente: Elaboración propia.

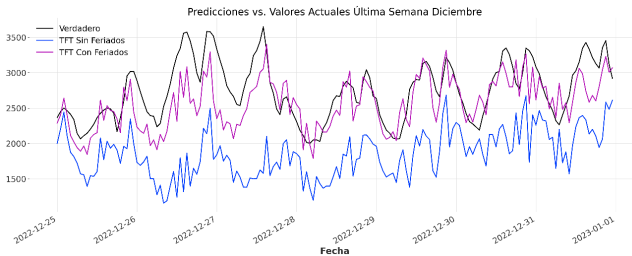


Figura 14: Comparación de resultados para modelo con y sin covariable Feriado en la última semana de Diciembre.

No obstante, el objetivo de incluir esta covariable es por su comportamiento en períodos específicos de concentración de días festivos, siendo los meses más relevantes Enero y Diciembre. Es por ello que se realiza un enfoque en los resultados predichos en semanas puntuales en estos intervalos mencionados, tal y como se observa en las Figuras 13 y 14.

Además, aprovechando las herramientas proveídas por Darts, podemos obtener gráficos de “explicabilidad” que, según lo mencionado en la metodología, nos permiten visualizar la importancia dada por nuestro modelo a la covariable Feriados según la Figura 15. Se observa que para el codificador de entrada la covariable Feriado (holidays) ocupa el cuarto lugar en importancia y atención, a la par de las covariables de Mes tanto senoidal como coseinodal, así como el tercer lugar para el decodificador, solamente por detrás de la Temperatura y la variable temporal de calendario ascendente Año, ambas de suma importancia en el entrenamiento y predicción de cualquier modelo de predicción con series temporales.

Considerando toda esta información y teniendo en cuenta que la caída de precisión es muy pequeña considerando que los feriados pueden tener mayor impacto si aumentan en gran cantidad, decidimos dejar esta covariable como fija para los siguientes entrenamientos en donde se busca la optimización del modelo.

7.2. Optimización de hiperparámetros con Optuna

Existen múltiples maneras de realizarlo y para este trabajo se utilizó la librería Optuna [12], debido a su versatilidad y eficiencia en la optimización de mode-

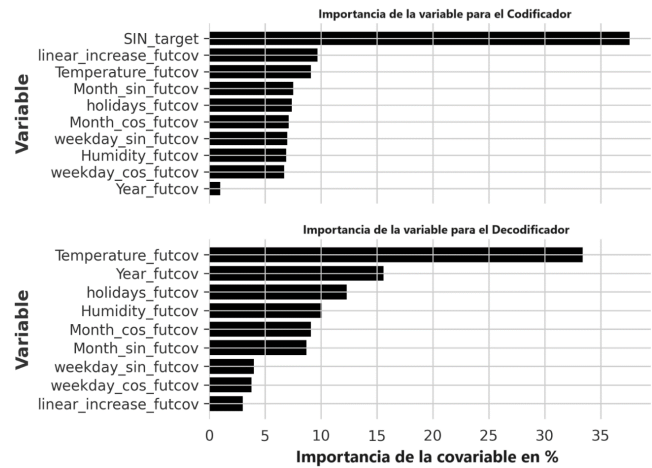


Figura 15: Gráficos de explicabilidad de las variables de entrenamiento del modelo

Fuente: Elaboración propia.

los de lenguaje de máquinas y aprendizaje profundo. El intervalo de valores que pueden adoptar los hiperparámetros se observa en la Tabla 7, mientras que la función utilizada para comparar cada modelo devuelto en el proceso de optimización es el SMAPE (Error simétrico medio porcentual absoluto).

Tabla 7: Intervalo de valores utilizados en la optimización de hiperparámetros.

Hiperparámetro	Valor mínimo	Valor máximo
input_chunk_length	14 días (336 horas)	30 días (720 horas)
output_chunk_length	1 día (24 horas)	1 día (24 horas)
hidden_size	32	64
lstm_layers	1	2
num_attention_heads	2	4
dropout	0,0	0,4
batch_size	16	32

Fuente: Elaboración propia.

7.2.1. Optimización con tiempo de parada

Se realiza primeramente un estudio inicial de optimización utilizando la funcionalidad de tiempo de parada, con un valor de 7200 segundos, debido a la gran demanda computacional que genera el entrenamiento del modelo, que corta el proceso de optimización una vez finalizado el tiempo asignado.

7.2.2. Optimización con límite máximo de iteraciones

A pesar de obtener resultados relativamente aceptables, se procedió a realizar una optimización completa en base a un número fijo de 50 iteraciones, con la posibilidad de obtener un mejor utilizando un tiempo aproximado de 4 días completos. Según la Figura 16, se determina que el mejor modelo obtenido

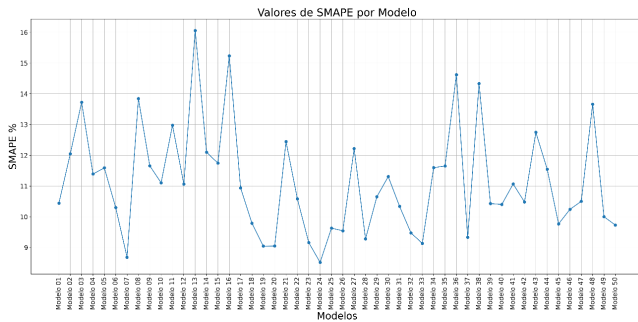


Figura 16: SMAPE para cada modelo en la optimización por Optuna.

Fuente: Elaboración propia.

Tabla 8: Mejores conjuntos de hiperparámetros.

Modelo	ICL	OCL	HS	LSTM	NAtt	D	BS
1	456	24	46	1	3	0.3505	24
2	480	24	50	1	4	0.2681	21
3	408	24	48	1	4	0.2693	17
4	408	24	46	1	4	0.3319	22
5	432	24	50	1	4	0.2872	18

Fuente: Elaboración propia.

es el número 24, cuyo análisis exhaustivo se realiza posteriormente. Aún así, se toman los 5 mejores modelos junto con sus respectivos conjuntos de valores para realizar las comparaciones pertinentes de los mismos.

7.3. Análisis de los modelos TFT

El resumen de los mejores cinco conjuntos de hiperparámetros extraídos de la optimización se observa en la Tabla 8. A partir de estos valores se realiza un re-entrenamiento de estos modelos para obtener las métricas correspondientes. Es importante aclarar que se realiza la desnormalización de los resultados para obtener una mejor visualización y panorama en las comparaciones finales.

El resultado de las métricas aplicadas en el conjunto de datos de prueba se puede observar en la Tabla 9.

Tabla 9: Métricas aplicadas a todo el conjunto de prueba.

Modelo	MSE	RMSE	R2	MAXERROR
1	76757.93	277.05	0.735	1728.80
2	92928.09	304.84	0.679	1700.03
3	68073.19	260.91	0.765	1761.95
4	62221.84	249.44	0.785	1599.63
5	87992.68	296.64	0.696	1923.10

Fuente: Elaboración propia.

Consideramos importante también realizar el

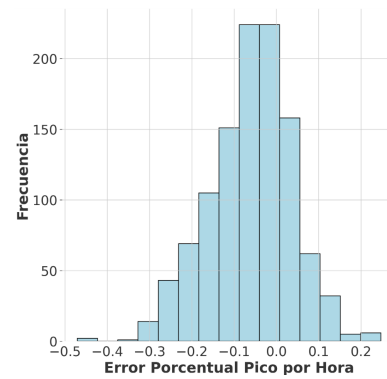


Figura 17: Histograma del error pico porcentual por hora.

Fuente: Elaboración propia.

análisis de los errores en distintas formas que se presentan en los valores predichos. Se puede observar en la Figura 17 la distribución del porcentaje del error pico, en intervalos horarios, a lo largo del conjunto de datos. El error máximo se calcula encontrando el error absoluto máximo para cada período de 24 horas, y luego se normaliza al dividir por el valor máximo de entre los valores reales para ese día en particular. Los valores menores a 0 indican que los resultados de la predicción fueron menores en comparación a los valores reales, mientras que los mayores a 0 indican justamente lo contrario.

La distribución del porcentaje de error se centra alrededor de cero sugiere que en una gran mayoría de los días el error pico fue muy pequeño porcentualmente entre los valores predichos y los reales, además de que el rango en el cual caen los errores es relativamente pequeño, indicando un rendimiento constante del modelo a lo largo de los días en el conjunto de datos. El histograma de la Figura 18 indica una distribución de los errores máximos que se alcanzan durante los días, es decir, el peor caso en cuanto a error para cada día evaluado dentro del conjunto de datos. Se observa que está sesgado a la derecha, indicando que hay muchos más días con errores bajos porcentuales, cuyo rango más poblado es entre 0,1 y 0,2.

8. Comparativa de resultados

Es posible observar el comportamiento del modelo en los considerados mejores días según la métrica del MSE según la Figura 19. Vemos como el modelo sigue la tendencia según los picos y valles para cada hora en todos los días mostrados, buscando adaptarse a cada cambio brusco que se presenta.

Además de observar el comportamiento en cada hora por día, podemos observar el desempeño en la mejor y peor semana del modelo según las métricas en las Figuras 20 y 21 respectivamente.

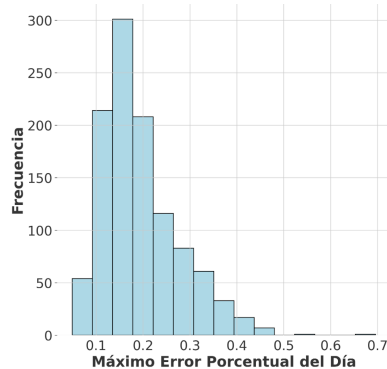


Figura 18: Histograma del máximo error porcentual del día

Fuente: Elaboración propia.

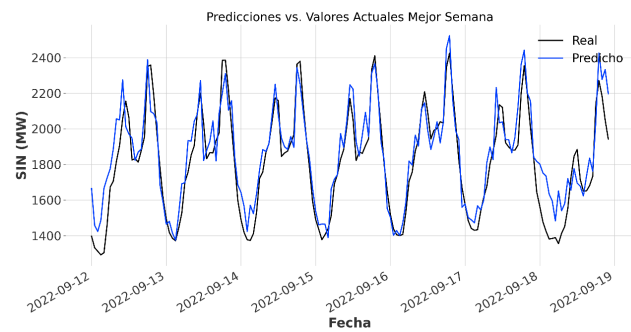


Figura 20: Comparativa en la mejor semana de los valores del mejor modelo con los valores reales.

Fuente: Elaboración propia.

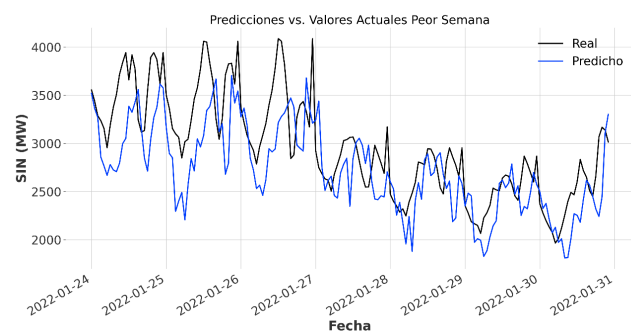


Figura 21: Comparativa en la peor semana de los valores del mejor modelo con los valores reales.

Fuente: Elaboración propia.

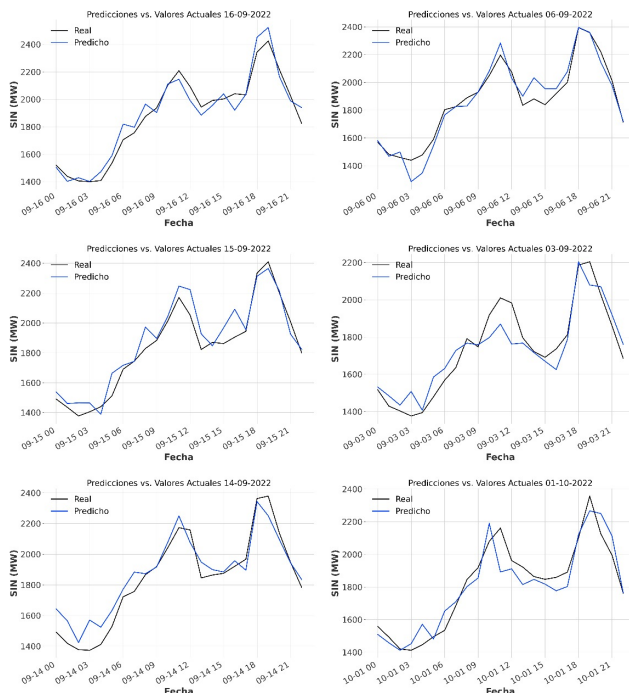


Figura 19: Comparativa en los mejores días de los valores del mejor modelo con los valores reales.

Fuente: Elaboración propia.

En la mejor semana (12-09-22 al 19-09-22) la curva de los valores predichos sigue bastante cerca a la de los valores reales en el conjunto de datos de prueba, indicando una gran precisión en este período con desviaciones muy pequeñas.

En el caso de la peor semana (24-01-2022 al 31-01-2022) existen discrepancias mucho más notables debido a los cambios bruscos en los valores reales, que no siguen un patrón muy claro ni constante, destacando que esta semana corresponde al mes de Enero, donde es más probable que se presente este tipo de comportamientos

9. Conclusión

En las distintas etapas atravesadas para el pre-procesamiento de datos en este Trabajo Final de Grado se ha cumplido en cada una de ellas su respectiva meta, pudiendo escalar y normalizar correctamente todo el conjunto de datos que fueron proveídos, para poder consecuentemente transformarlos y particionarlos de modo a ser utilizados según los requisitos para el modelo y la arquitectura elegida.

Los datos históricos pertenecientes a la AN-

DE fueron actualizados por los más recientes según los materiales y estadísticas proveídas en sus canales oficiales públicos, que incluyen todos sus planes maestros de transmisión, distribución y generación para el período 2021-2030.

Se logró implementar de manera satisfactoria un modelo basado en la arquitectura TFT, considerado innovador debido al conjunto de conceptos y otras arquitecturas que aplica por dentro, para la predicción de la demanda de energía a nivel nacional, con la utilización de novedosas librerías para trabajos enfocados al entrenamiento de modelos de series temporales como, por ejemplo, Darts.

A partir de los entrenamientos preliminares del modelo, con distintas configuraciones y conjuntos de valores, fue posible realizar la optimización de los parámetros considerados relevantes dentro de la arquitectura TFT que condujeron a la producción de modelos más precisos para obtener mejores resultados.

Para finalizar, las comparaciones realizadas con los valores reales del SIN arrojaron resultados bastante precisos en períodos complejos del año debido a los picos y variaciones constantes en esas fechas, específicamente durante el verano. Es justamente este aspecto el que resalta también en las comparaciones con el trabajo que sirvió de base a este, obteniendo un modelo que se adapta mejor durante el conjunto de datos de prueba.

10. Trabajos futuros

Futuras investigaciones deben centrarse en evaluar nuevos modelos derivados del presentado en este trabajo mediante técnicas de transferencia de aprendizaje y ajuste fino [13]. Estos modelos deben ser adaptados para los nuevos datos de demanda eléctrica y condiciones climáticas que puedan surgir y que no estén contemplados en el conjunto de datos utilizado en este estudio.

El progreso hacia la obtención de predicciones más precisas y una comprensión más profunda de cómo las variables utilizadas para el entrenamiento afectan dichos modelos es crucial. Por lo tanto, investigaciones futuras deberían considerar la implementación de nuevas arquitecturas, como Crossformer [14] y N-Beats [15].

Este trabajo proporcionó una solución para la predicción de la demanda a nivel nacional. Sin embargo, estudios futuros podrían abordar este problema de manera más detallada, reutilizando la implementación existente pero centrándose en la predicción a nivel de subsistema.

Referencias

- [1] Administración Nacional de Electricidad, *Plan Maestro de Transmisión*, Febrero 2021, período: 2021 – 2030.
- [2] C. Kuster, Y. Rezgui, and M. Mourshed, "Electrical load forecasting models: A critical systematic review," *Sustainable Cities and Society*, 2017. [Online]. Available: <https://doi.org/10.1016/j.scs.2017.10.017>
- [3] O. A. Barboza, "Automatización de previsión de demanda horaria de potencia eléctrica en el sistema interconectado nacional," *Revista Científica de la UCSA*, pp. <https://www.mdpi.com/1996-1073/11/7/1636>, 2014.
- [4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, and L. Kaiser, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017. [Online]. Available: <https://arxiv.org/abs/1706.03762>
- [5] B. Lima, S. O. Arik, N. Loeff, and T. Pfister, "Temporal fusion transformers for interpretable multi-horizon time series forecasting," *Elsevier*, p. arXiv:1912.09363, 2019.
- [6] Administración Nacional de Electricidad (ANDE), "Memoria anual 2022," 2023.
- [7] B. Lim and S. Zohren, "Time series forecasting with deep learning: A survey," *Department of Engineering Science, University of Oxford, Oxford, UK*, p. arXiv:2004.13408, 2020.
- [8] M. Verdún and F. Morales, "Diseño de un modelo inteligente para la predicción horaria a corto plazo de la demanda eléctrica nacional basado en aprendizaje profundo y propuesta de implementación en el despacho de carga de la ande," 2020.
- [9] C. Fan, M. Chen, X. Wang, J. Wang, and B. Huang, "A review on data preprocessing techniques toward efficient and reliable knowledge discovery from building operational data," *Frontiers in Energy Research*, vol. 9, p. 652801, 2021, this article is part of the Research Topic "Future Energy Systems for Building Clusters and Districts". [Online]. Available: <https://doi.org/10.3389/fenrg.2021.652801>
- [10] F. Ziel, "Modeling public holidays in load forecasting: a german case study," *Journal of Modern Power Systems and Clean Energy*, vol. 6, pp. 191–207, 2018, special Section on Forecasting in Modern Power Systems, Open access.
- [11] J. Herzen, S. G. Piazzetta, T. Neuer, G. , and T. Van Pottelbergh, "Darts: User-friendly modern machine learning for time series," *Journal of Machine Learning Research*, 2022.

- [12] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," *arXiv preprint arXiv:1907.10902*, 2019.
- [13] Z. Al-Ali. (2022) Production forecasting using transfer learning of pre-trained deep model. [Online]. Available: <https://suetrid.stanford.edu/production-forecasting-using-transfer-learning-pretrained-deep-model>
- [14] Y. Zhang and J. Yan, "Crossformer: Transformer utilizing crossdimension dependency for multivariate time series forecasting," 2023, published as a conference paper at ICLR 2023.
- [15] B. N. Oreshkin, D. Carpow, N. Chapados, and Y. Bengio, "N-beats: Neural basis expansion analysis for interpretable time series forecasting," in *International Conference on Learning Representations (ICLR)*. Element AI and Mila, 2020, published as a conference paper at ICLR 2020.