Documentación

Big Data I TerraWatt

Alma Gutierrez
Rafael Borge
Deniz Alcobendas
Íñigo Pérez

Índice

Descripción del Proyecto

El aumento constante de los precios de la electricidad, junto con los esfuerzos de la Unión Europea por promover un consumo energético más eficiente, pone de manifiesto la necesidad de contar con herramientas que permitan gestionar mejor el uso de energía en los hogares. En este contexto, TerraWatt surge como una solución innovadora diseñada para proporcionar a los usuarios una previsión precisa de sus facturas de electricidad y promover un uso más consciente y optimizado de la energía.

Los precios de la electricidad son sumamente variables y están influenciados por múltiples factores, como las condiciones económicas, los cambios sociales y el clima. Además, el consumo energético en los hogares depende de aspectos diversos, como el tipo de vivienda, la potencia contratada, el número de habitantes y las condiciones meteorológicas locales.

Por estas razones, el proyecto se enfoca en desarrollar dos modelos predictivos con el objetivo de:

1. **Anticipar los cambios en los precios de la electricidad.**
2. **Prever el consumo energético en los hogares para optimizar el uso de los recursos.**

La meta final es dotar a los usuarios de herramientas prácticas que les permitan:

- Planificar con antelación su consumo energético.
- Identificar momentos de alto costo y consumo.
- Fomentar un uso eficiente y sostenible de la energía.

Estos modelos predictivos se apoyan en la integración de datos históricos de consumo, precios y variables meteorológicas, enriquecidos con información adicional, como los días festivos y los patrones de actividad semanal. Al consolidar esta información, se generan datasets completos y estructurados que facilitan el análisis y la construcción de predicciones precisas de la factura eléctrica.

Objetivos

Objetivos Generales del Proyecto y de TerraWatt

El proyecto TerraWatt tiene como propósito general desarrollar una plataforma predictiva que permita anticipar el consumo energético y los precios de la electricidad, proporcionando herramientas de planificación y optimización de facturas para los hogares. TerraWatt tiene los siguientes objetivos generales:

1. Facilitar la comprensión del consumo energético a través de información detallada y predicciones basadas en datos históricos y contextuales.
2. Optimizar los gastos de las familias mediante predicciones precisas de los precios de la electricidad.
3. Proveer un entorno confiable de visualización y simulación de consumos y precios futuros.
4. Contribuir al fomento de un uso más eficiente y sostenible de los recursos energéticos.

Objetivos Específicos de Cada Modelo

Modelo de Predicción del Consumo Energético

1. Analizar los factores internos y externos que influyen en el consumo energético, como las características de la vivienda, los factores demográficos y las condiciones meteorológicas.
2. Identificar los factores más relevantes que afectan el consumo energético para optimizar las predicciones.
3. Desarrollar y comparar distintos modelos predictivos, como Random Forest, Gradient Boosting y redes neuronales, para seleccionar el modelo con mejor desempeño en la predicción del consumo.
4. Implementar un proceso de limpieza, integración y unificación de los datos históricos relacionados con el consumo, asegurando su calidad y consistencia.

5. Validar la precisión del modelo mediante técnicas avanzadas de evaluación y seleccionar el enfoque más robusto para las predicciones.

Modelo de Predicción del Precio de la Electricidad

1. Analizar los factores que influyen en la variación de los precios de la electricidad, considerando eventos predecibles, factores socioeconómicos y climatológicos.
2. Identificar los factores más relevantes en las fluctuaciones de precios a partir de datos históricos y condiciones externas.
3. Implementar un proceso de limpieza y organización de los datos históricos de precios, garantizando su uso eficiente en los modelos predictivos.
4. Desarrollar y comparar modelos predictivos de series temporales, como redes neuronales LSTM, para capturar patrones históricos y prever fluctuaciones.
5. Validar la precisión de los modelos mediante análisis de correlación cruzada y seleccionar aquel con menor margen de error en las predicciones.

Datos

Para construir el modelo de predicción del precio de la electricidad, se utilizaron diversas fuentes de datos que permiten captar los factores que influyen en las fluctuaciones de los precios. Estos datos fueron procesados y unificados en un único archivo CSV que consolidó toda la información necesaria para alimentar el modelo predictivo.

Entre las fuentes principales de datos se encuentran los registros meteorológicos proporcionados por la **AEMET** (Agencia Estatal de Meteorología), información histórica extraída mediante la **API de la Red Eléctrica Española** sobre los precios diarios de la electricidad y un archivo generado con los **días festivos en España**. La combinación de estas fuentes permitió un análisis integral que considera tanto aspectos climáticos como patrones de demanda influenciados por fechas festivas.por variables meteorológicas, especialmente en periodos de temperaturas extremas.

La descarga de los datos se realizó en un archivo comprimido que contenía los registros de aproximadamente 947 estaciones meteorológicas distribuidas por todo el territorio nacional. Cada archivo, en formato CSV, representa una estación específica e incluye datos diarios de las condiciones meteorológicas en un amplio rango temporal (en algunos casos, desde

1920 hasta la última fecha de actualización). La nomenclatura de los archivos permite identificar rápidamente el periodo cubierto por cada registro, lo que facilita su organización y análisis.

Variables climáticas más relevantes:

- **Temperatura:** media, mínima y máxima diaria.
- **Precipitaciones:** cantidad diaria en milímetros.
- **Viento:** velocidad media y rachas máximas.
- **Presión atmosférica:** valores de presión mínima y máxima.
- **Horas de luz:** número de horas de sol al día

Precios Electricidad (Archivo CSV Red Eléctrica Española)

La Red Eléctrica Española proporciona acceso a datos históricos sobre los precios de la electricidad, los cuales se extrajeron y consolidaron en un archivo CSV descargado. Este archivo contiene los precios diarios de los últimos 10 años y está estructurado para facilitar el análisis predictivo de los precios en distintos momentos del tiempo.

Características del Archivo de Precios de Electricidad:

- **Datos horarios:** los precios están organizados por horas, lo que permite estudiar las fluctuaciones a lo largo del día y observar los momentos de mayor y menor coste energético.
- **Promedio diario:** el archivo incluye columnas con los precios medios diarios, fundamentales para realizar comparaciones y detectar tendencias generales.
- **Tipo de tarifas:** incluye diferentes modalidades tarifarias, lo que permite analizar cómo varían los precios en función del tipo de contrato energético.
- **Formato de los datos:** el CSV contiene columnas como la fecha, la hora, el precio en euros/MWh y la categoría tarifaria, permitiendo filtrar y analizar según distintos criterios de interés.

Relevancia de los Datos de Precios de Electricidad:

Los datos extraídos permiten identificar patrones estacionales, como los incrementos de precio durante los meses de invierno y verano debido al aumento de la demanda energética. Además, facilitan la detección de horas pico y horas valle, proporcionando información valiosa para diseñar estrategias de consumo eficiente. Asimismo, estos datos permiten evaluar el impacto de eventos externos, como olas de calor o de frío, que provocan fluctuaciones significativas en los precios debido al aumento de la demanda. Gracias a la periodicidad diaria y la granularidad horaria, el modelo predictivo puede actualizarse de manera continua, reflejando las tendencias actuales del mercado eléctrico y mejorando así la precisión de sus resultados.

Días Festivos de España

Para incorporar los efectos de los días festivos en el análisis del precio de la electricidad, se utilizó un archivo CSV que recopila las fechas de festivos nacionales y autonómicos en España. Este archivo es crucial para considerar cómo las festividades influyen en la demanda energética y, por lo tanto, en el precio de la electricidad.

Características del Archivo de Días Festivos:

- **Cobertura nacional y autonómica:** el archivo incluye tanto festivos oficiales nacionales como festivos locales específicos de cada comunidad autónoma.
- **Formato del archivo:** contiene columnas con la fecha del festivo, el nombre de la festividad, el ámbito geográfico (nacional o autonómico) y la región correspondiente.
- **Cobertura temporal:** cubre varios años de datos, abarcando los festivos relevantes durante el periodo analizado en el modelo.

Relevancia de los Días Festivos:

Los días festivos suelen generar cambios significativos en los patrones de consumo de electricidad, con una disminución del consumo energético industrial y un aumento en el consumo doméstico, lo que impacta de manera notable en los precios del mercado. Además, los festivos locales afectan de manera diferenciada a las regiones, provocando fluctuaciones regionales en los precios medios diarios. Incluir esta información en el modelo permite ajustar las predicciones para capturar correctamente las alteraciones de la demanda durante festividades importantes, contribuyendo así a estimaciones más precisas y representativas de los comportamientos reales del mercado.

Este archivo contribuye a una interpretación más precisa de los patrones de precios al modelar correctamente las desviaciones de comportamiento en fechas clave, como festivos nacionales y autonómicos.

Limpieza y transformación de los datos

Durante la revisión de los archivos de datos meteorológicos, se identificó un archivo con un error específico:

- **Archivo defectuoso:** `1111X-20120301-20241103.csv`
- **Problema:** Contiene una línea con un formato incorrecto que genera errores al intentar leer el archivo completo.

Se eliminó la línea defectuosa utilizando un script que filtra los datos válidos y crea un nuevo archivo limpio. Esto asegura que el archivo pueda ser procesado sin interrupciones, garantizando la integridad de los datos restantes.

Eliminación de Columnas Irrelevantes:

Los archivos originales contienen información detallada de las estaciones meteorológicas, pero algunas columnas no son necesarias para el análisis por provincias. Se eliminaron las siguientes columnas:

- **`INDICATIVO`**: Código identificador de la estación meteorológica.
- **`NOMBRE`**: Nombre específico de la estación meteorológica.

Motivo de eliminación:

La información a nivel de estación no es relevante para un análisis agregado por provincia, por lo que estas columnas se consideran redundantes y ocupan espacio innecesario.

Normalización de Valores en la Columna "PROVINCIA":

En algunos casos, las provincias aparecen con nombres duplicados o con variaciones de escritura. Para asegurar la consistencia de los datos, se realizaron las siguientes correcciones:

- **`ARABA/ALAVA`** → **`ALAVA`**

- `STA. CRUZ DE TENERIFE` → `SANTA CRUZ DE TENERIFE`
- `BALEARES` → `ILLES BALEARS`
- `GERONA` → `GIRONA`
- `ORENSE` → `OURENSE`
- `VIZCAYA` → `BIZKAIA`

Esta normalización permite una mejor agrupación y evita problemas de duplicación al calcular estadísticas por provincia.

Filtrado Temporal:

Los datos meteorológicos abarcan un amplio rango de fechas, pero el análisis debe restringirse a un período específico:

- **Fecha de corte:** 1 de abril de 2014.
- **Motivo:** Los datos de precios de electricidad solo están disponibles a partir de esa fecha, por lo que los datos anteriores no son relevantes para el análisis conjunto.

Proceso de filtrado:

Se eliminan todas las filas correspondientes a fechas anteriores al 1 de abril de 2014. Esto reduce el volumen de datos y mejora la eficiencia del procesamiento.

Agrupación por Provincia y Fecha:

Para obtener datos diarios por provincia, se realiza una agrupación de los datos de las estaciones meteorológicas:

- **Variables agrupadas:** Fecha y provincia.
- **Operación realizada:** Cálculo del promedio de todas las mediciones para cada combinación de provincia y día.

Este paso permite consolidar los datos en una estructura más simple y representativa, generando valores diarios promedio de las condiciones meteorológicas de cada provincia.

Consolidación y Exportación:

Una vez agrupados los datos por provincia, se realiza la exportación de los resultados:

1. **Archivos generados:** Se crea un archivo CSV independiente para cada provincia con el nombre de la misma (`ej.: MADRID.csv`, `TARRAGONA.csv`).

2. **Reemplazo de archivos:** Los archivos originales se eliminan y se reemplazan por los archivos consolidados.

Resultado final

Se obtiene un conjunto de archivos limpios y organizados, uno por cada provincia, con datos meteorológicos promediados por día. Este formato facilita el análisis posterior y la integración con otros conjuntos de datos, como los precios de la electricidad.

Limpieza de Datos Precio Energía Faltantes

Se realizó una limpieza exhaustiva para asegurar la integridad de los datos:

- **Eliminación de filas completamente vacías:**
    - Se descartaron registros donde todas las columnas presentaban valores nulos.
- **Relleno de valores faltantes en columnas numéricas:**
    - Los campos numéricos vacíos se completaron con la **media de la columna** correspondiente.

Este proceso es crucial para evitar errores en sistemas de almacenamiento y análisis como ElasticSearch, que requieren datos completos.

Revisión y Normalización de Nombres de Provincias:

Antes de realizar la combinación de datos, se revisaron los nombres de las provincias para evitar errores de consistencia:

- **Objetivo:** Garantizar que los nombres de las provincias coincidan en todos los archivos para prevenir errores al realizar la combinación de datos.
- **Proceso:**

    - Se detectaron nombres duplicados, variaciones de escritura y errores tipográficos.
    - Se empleó un diccionario de mapeo en Python para aplicar correcciones automáticamente al cargar los datos.

- `GERONA` → `GIRONA`
- `ORENSE` → `OURENSE`
- `VIZCAYA` → `BIZKAIA`

Esta normalización asegura que no haya conflictos al realizar la unión de los datasets de consumo energético y meteorología.

Unión de Datos de Precios y Datos Meteorológicos:

Para generar un modelo integral, se realizó la unión de los siguientes datasets:

- **precios_energía**: Contiene datos de la variación de precios eléctricos.
- **Archivos meteorológicos**: Contienen mediciones diarias promedio de las condiciones meteorológicas por provincia.

Detalle del proceso de combinación:

1. **Lectura de archivos:**
    - Se cargaron todos los archivos meteorológicos de la carpeta `Datos_limpios_meteorologicos`.
    - El archivo de consumo se leyó y se ajustó al rango temporal de interés.
2. **Preparación de los datos:**
    - Se añadió la columna `Provincia` al conjunto meteorológico para facilitar la combinación.
    - Se convirtieron las columnas de fecha al formato estándar `YYYY-MM-DD`.
3. **Unión de los datasets:**

- Se realizó una **Union Left** para preservar los datos de consumo y agregar la información meteorológica.
- Se validó la coincidencia de las provincias y se ajustaron los nombres en caso de discrepancias.

Adición de Columnas "Festivo" y "Entre Semana":

Para enriquecer los datos y evaluar el comportamiento en contextos específicos, se añadieron las columnas "Festivo" y "Entre Semana". La columna "Festivo" indica si una fecha corresponde a un día festivo y se construyó comparando las fechas de cada registro con las fechas de festivos oficiales por provincia, obtenidas del archivo `Festivos.csv`. A cada registro se le asignó "SI" si correspondía a un festivo y "NO" en caso contrario. La columna "Entre Semana" clasifica los días como laborales o no laborales, asignando "SI" para los días de lunes a viernes y "NO" para sábados y domingos.

Para garantizar la precisión de esta integración, se realizaron varios pasos detallados. Primero, se convirtió el formato de las fechas en ambos datasets (consumo y festivos) al mismo formato, asegurando así una combinación correcta. Posteriormente, se generó una clave única basada en la combinación de "Fecha" y "Provincia" para identificar de manera precisa los días festivos. Finalmente, se añadió la columna "Entre Semana" utilizando la función `dt.dayofweek`, que clasifica los días según su posición en la semana (0 para lunes y 6 para domingo), permitiendo así identificar automáticamente los días laborales y los fines de semana.

Generación del Archivo Final "Modelo_Precios_Met_Fest.csv":

Este archivo contiene la información consolidada para el análisis de precios de la electricidad.

Resultado final

- Datos meteorológicos consolidados por provincia y día.
- Información sobre precios diarios de electricidad.
- Variables adicionales para indicar si el día es festivo o entre semana.

Pasos realizados:

1. **Lectura y modificación del archivo de precios:**
    - Se corrigió el delimitador para garantizar la correcta lectura del archivo `precios_energia.csv`.
2. **Unión de los datos meteorológicos y de precios:**
    - Se realizó una unión por `FECHA` para agregar la información de precios al dataset meteorológico.
3. **Limpieza de datos:**
    - Se eliminaron columnas irrelevantes, como impuestos específicos, para mantener únicamente los precios relevantes.
4. **Exportación:**
    - El archivo final `Modelo_Precios_Met_Fest.csv` se guardó en formato CSV

Desarrollo del proyecto

1. Modelo de Predicción del Consumo Energético por Vivienda

Este modelo busca estimar el consumo diario de electricidad en función de las características específicas de cada hogar. Para ello, se analizarán:

- **Aspectos de la vivienda:** tipo de construcción, tamaño, potencia contratada.
- **Factores demográficos:** número de habitantes y sus hábitos de uso energético.
- **Condiciones meteorológicas:** temperatura, humedad, precipitaciones y horas de luz.
- **Patrones de uso:** horarios y frecuencia de consumo a lo largo de la semana.

El objetivo es generar predicciones precisas sobre el consumo y ofrecer recomendaciones personalizadas que permitan reducir gastos y fomentar un uso más eficiente de la energía.

1.1. Primera fase: Evaluación inicial de los modelos

En la fase inicial del proyecto, probamos diferentes modelos de predicción utilizando los datos completos, con el objetivo de estimar tanto el consumo energético de los hogares como la evolución de los precios de la electricidad. Sin embargo, los resultados obtenidos fueron poco precisos y presentaban altos errores de predicción. Para entender mejor las razones de esta baja precisión, realizamos un análisis más profundo de las predicciones y sus desviaciones respecto a los valores reales.

Uno de los hallazgos más relevantes fue que los patrones de consumo y los precios de la electricidad varían significativamente en función de la **provincia**. En otras palabras, no existía un único modelo que pudiera generalizarse para todas las regiones de España con un nivel de precisión aceptable. Diferencias en el clima, en la estructura de la red eléctrica, en los hábitos de consumo y en la disponibilidad de energías renovables hacían que la predicción basada en un modelo único no fuera viable.

Este descubrimiento nos llevó a replantearnos la estrategia y considerar una **segmentación por provincias**, creando modelos específicos para cada una de ellas en lugar de un único modelo nacional.

1.2. Implementación de modelos por provincia

Dado que cada provincia presenta características propias en términos de consumo eléctrico y precios de la energía, decidimos **entrenar modelos específicos para cada provincia** en lugar de uno solo para todo el país. Esto nos permitió mejorar la precisión de las predicciones y capturar mejor las variaciones locales en los datos.

Para implementar esta solución, estructuramos el código de tal manera que los modelos se entrenan de forma independiente para cada provincia. Esto implicó una **optimización del procesamiento de datos** y un enfoque más modular que permitiera la comparación entre distintos modelos de machine learning en cada una de las provincias.

1.3. Selección de modelos de machine learning

Para entrenar los modelos, evaluamos varios algoritmos de aprendizaje automático, seleccionando aquellos que han demostrado ser eficaces en la predicción de series temporales y problemas de regresión. Los modelos seleccionados fueron:

Regresión Lineal

La regresión lineal es el modelo más simple que probamos, ya que busca ajustar una línea recta a los datos minimizando el error cuadrático medio (MSE). Se considera un buen punto de partida para la predicción del consumo y del precio, ya que puede capturar relaciones lineales entre las variables predictoras. Sin embargo, observamos que su desempeño era muy deficiente en provincias con alta variabilidad en el consumo o en los precios, lo que sugiere que las relaciones no son estrictamente lineales.

Random Forest

El modelo de Random Forest se basa en la construcción de múltiples árboles de decisión y la combinación de sus resultados para obtener una predicción más estable y precisa. Probamos este modelo debido a su capacidad para capturar relaciones no lineales entre las variables y su robustez frente a datos ruidosos. Además, este algoritmo es menos sensible a la multicolinealidad y puede manejar un gran número de variables sin problemas. Funcionó especialmente bien en provincias con tendencias irregulares y fluctuaciones marcadas.

XGBoost (Extreme Gradient Boosting)

XGBoost es un modelo basado en árboles de decisión con una optimización del algoritmo de Gradient Boosting. Se eligió debido a su alta capacidad predictiva, su eficiencia en el manejo de grandes volúmenes de datos y su resistencia al sobreajuste. Se observó que este modelo proporcionaba los mejores resultados en provincias donde el consumo y los precios seguían patrones estacionales claros.

Gradient Boosting

Gradient Boosting es una técnica de aprendizaje supervisado que construye modelos de predicción en secuencia, de manera que cada nuevo modelo trata de corregir los errores del anterior. Se probó debido a su capacidad para capturar relaciones complejas en los datos. En algunas provincias, este modelo superó a XGBoost, sobre todo en aquellas donde las variaciones en el consumo eran más dependientes de las condiciones climáticas.

1.4. Comparación y selección del mejor modelo por provincia

Tras probar estos cuatro modelos en todas las provincias, nos dimos cuenta de que **no había un modelo único que funcionara mejor en todas las regiones**. En su lugar, el rendimiento variaba dependiendo de la provincia, lo que reafirmó la importancia de realizar una evaluación individualizada para seleccionar el modelo más adecuado en cada caso.

Para determinar qué modelo ofrecía las mejores predicciones para cada provincia, implementamos una **evaluación automática basada en métricas de desempeño**. Se calcularon las siguientes métricas:

- **Error Cuadrático Medio (MSE - Mean Squared Error):** Esta métrica penaliza los errores grandes en la predicción y es útil para evaluar la precisión del modelo en términos absolutos. Se utilizó como referencia principal para minimizar las desviaciones de las predicciones.

- **Coeficiente de Determinación (R^2 - R Squared):** Mide qué porcentaje de la variabilidad de los datos es explicada por el modelo. Un valor de R^2 cercano a 1 indica un ajuste muy bueno, mientras que valores cercanos a 0 indican que el modelo no logra explicar la variabilidad en los datos.

El código que desarrollamos para entrenar y evaluar los modelos siguió los siguientes pasos:

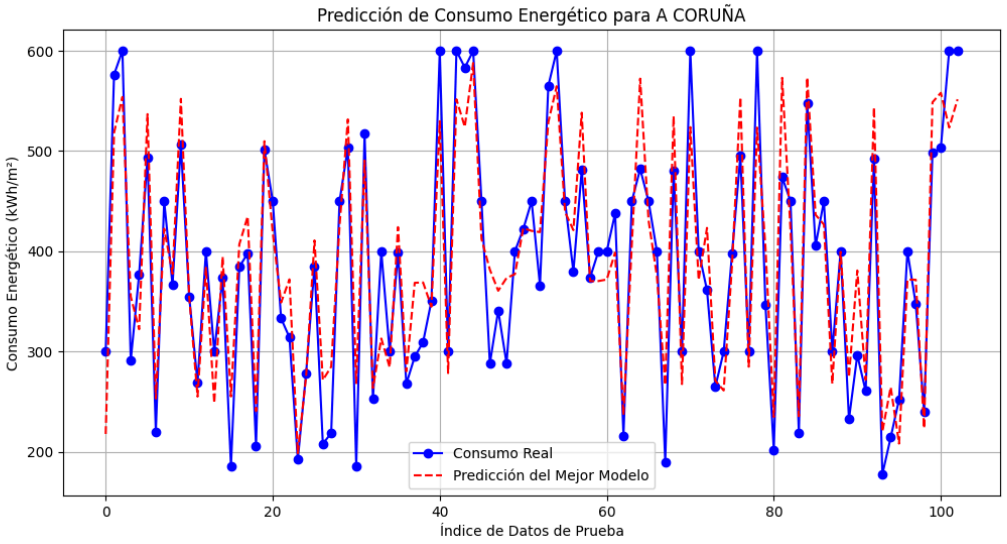
1. **Entrenamiento de los modelos:** Se ejecutaban los cuatro modelos de aprendizaje automático en los datos de cada provincia.
2. **Evaluación de las métricas:** Se calculaban el MSE y el R^2 para cada modelo.
3. **Selección automática:** Se elegía el modelo con el menor MSE y el mayor R^2 para cada provincia.
4. **Almacenamiento de resultados:** Se registraban los modelos seleccionados y sus métricas para futuras comparaciones.

1.5. Resultados y conclusiones

Gracias a esta estrategia, logramos mejorar significativamente la precisión de las predicciones. Algunos de los hallazgos clave fueron:

- En provincias con **patrones de consumo estables**, como Madrid o Barcelona, **Random Forest y XGBoost** fueron los modelos que mejor funcionaron.
- En provincias con **altas variaciones estacionales**, como Asturias o Galicia, **Gradient Boosting** ofreció mejores predicciones, ya que pudo capturar las fluctuaciones en el consumo energético relacionadas con la climatología.

- En provincias con **precios más impredecibles**, como algunas de las islas, **XGBoost** mostró mejores resultados debido a su capacidad para manejar variaciones bruscas en las series temporales.



En general, el uso de **un modelo por provincia, en lugar de un único modelo nacional**, permitió mejorar la precisión de las predicciones y ofrecer recomendaciones más ajustadas a la realidad de cada región.

1.6. Conclusión final

La implementación de modelos predictivos específicos para cada provincia representó un **gran avance en la capacidad de anticipar la factura de la luz**. Esta estrategia no solo mejoró la precisión de las estimaciones, sino que también permitió capturar las particularidades de cada región en términos de consumo y precios de la electricidad.

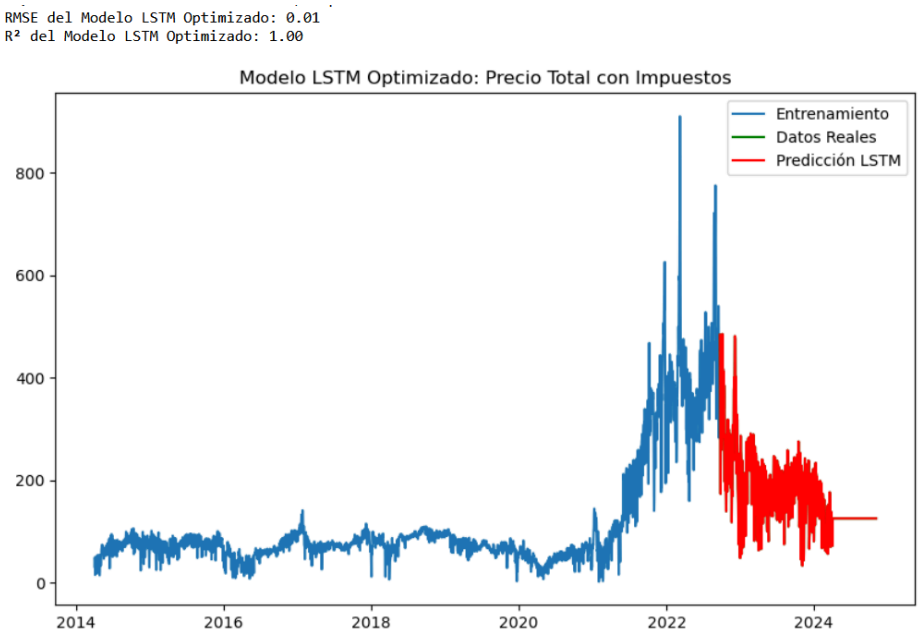
El desarrollo de este sistema no solo permite ofrecer predicciones precisas a los usuarios, sino que también abre la puerta a futuras mejoras, como la integración de más factores en los modelos (políticas energéticas, impacto de energías renovables, etc.). En definitiva, este enfoque representa un paso importante hacia la **optimización del consumo energético y la planificación eficiente de los gastos eléctricos en los hogares.**

2. Modelo de Predicción del Precio de la Electricidad

Este modelo pretende anticipar las fluctuaciones en el precio de la electricidad utilizando datos históricos y factores externos como:

- Cambios en las condiciones meteorológicas.
- Eventos socioeconómicos importantes.
- Días festivos o períodos de alta demanda.
- Situaciones globales, como conflictos que afectan los mercados energéticos.

Para este modelo se emplearán técnicas avanzadas de series temporales y redes neuronales tipo **LSTM**, con el fin de identificar patrones de comportamiento y prever los precios con alta precisión.



Análisis exploratorio de los datos

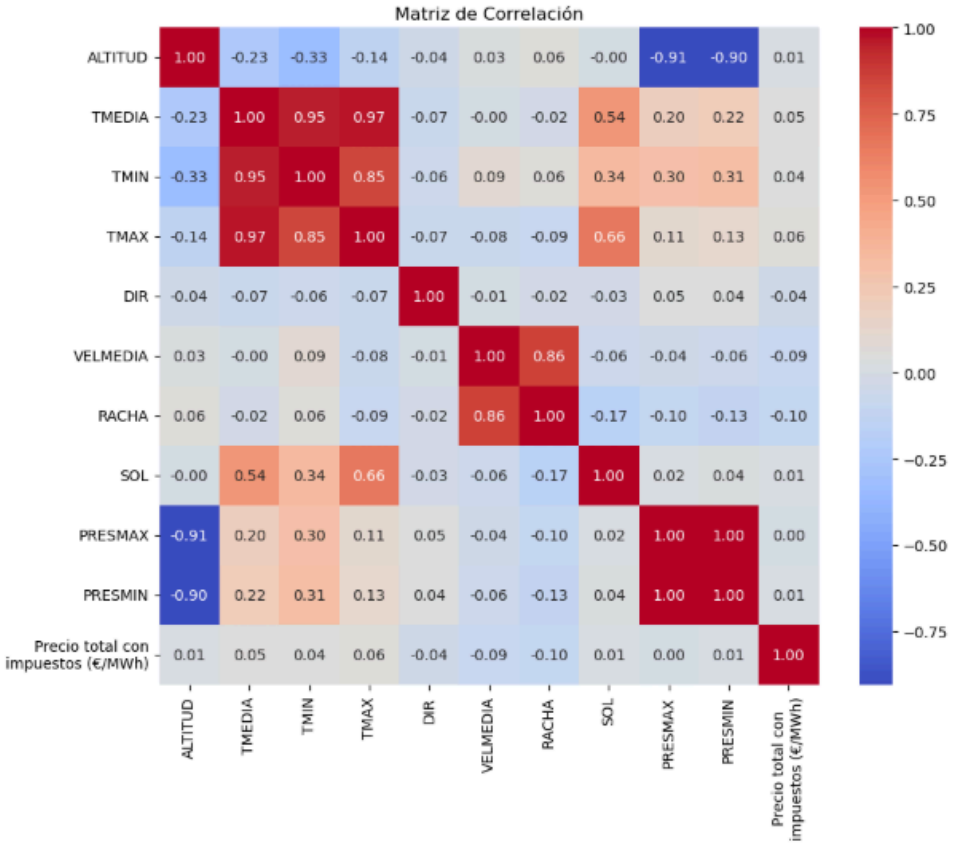
Nuestros datos, almacenados en un archivo CSV con 193,500 filas y 15 columnas, incluyen un conjunto completo de variables sin valores nulos. Entre las variables numéricas se encuentran la altitud, las temperaturas (media, máxima y mínima), las horas de sol y el precio total con impuestos (€/MWh). Las variables categóricas incluyen las provincias, días festivos y si corresponde a entre semana. La amplitud y variedad de los datos proporcionan una base sólida para analizar tendencias, relaciones y patrones.

La altitud promedio es de 561.6 metros, mientras que las temperaturas tienen una media de 15.3°C, con valores que oscilan entre -13.8°C y 41.7°C. Las horas de sol promedian 7.4 horas diarias, y la velocidad media del viento es de 2.7 m/s, aunque se registran ráfagas de hasta 31.6 m/s. En cuanto a los precios energéticos, presentan una media de 125.6 €/MWh,

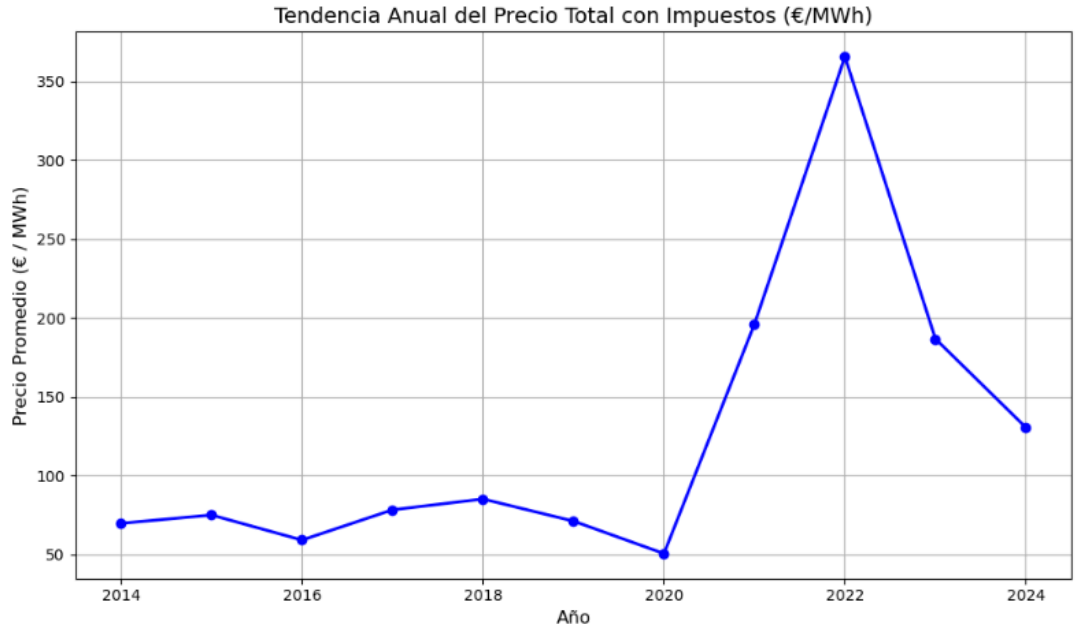
pero con una alta dispersión, desde 2.2 €/MWh hasta 989.8 €/MWh, reflejando la alta variabilidad climática y de mercado en los datos.

La columna "Provincia" incluye 50 valores únicos, representando regiones tanto peninsulares como insulares. Esto permite realizar análisis específicos por ubicación. En el caso de las temperaturas, el análisis de boxplots muestra distribuciones razonables, aunque con valores atípicos en los extremos inferiores de la temperatura mínima y superiores en la máxima, que podrían reflejar eventos climáticos extremos.

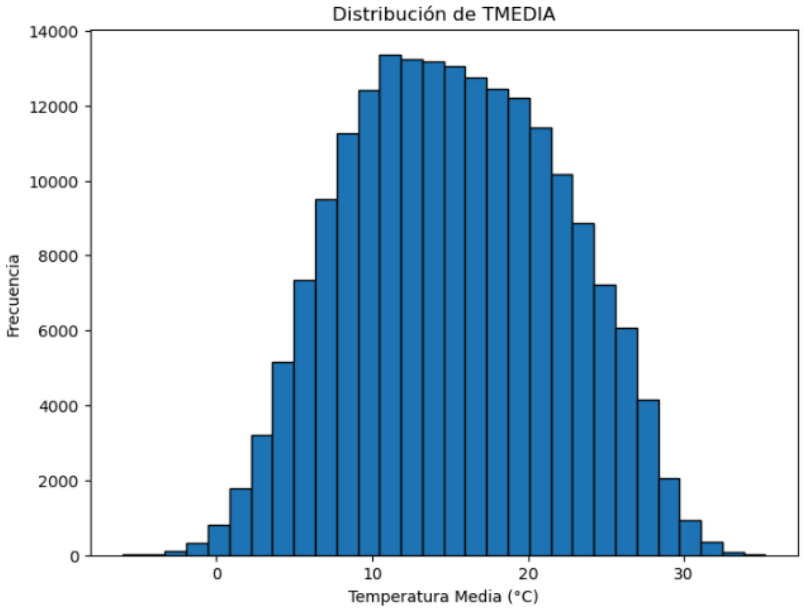
La matriz de correlación destaca relaciones fuertes entre variables climáticas. Las temperaturas están correlacionadas entre sí, y las horas de sol tienen una relación moderada con ellas. La presión atmosférica muestra una relación inversa con la altitud (-0.91), mientras que las ráfagas están relacionadas con la velocidad media del viento (0.86). Sin embargo, no hay correlaciones significativas entre las variables climáticas o geográficas y los precios de la energía, sugiriendo que estos últimos están influenciados por factores externos al clima.



En el análisis temporal, el precio promedio anual muestra un aumento drástico a partir de 2021, alcanzando un máximo en 2022, probablemente reflejando crisis de suministro o aumentos en la demanda. Aunque en 2024 se observa una ligera estabilización, los precios siguen siendo notablemente altos en comparación con años anteriores.



Finalmente, la distribución de la temperatura media sigue un patrón aproximadamente normal, con la mayoría de los valores entre 10°C y 20°C, reflejando las condiciones promedio en las regiones analizadas. Estos análisis iniciales subrayan la riqueza del conjunto de datos y la necesidad de explorar factores externos que puedan influir en los patrones observados.



Infraestructura del proyecto

Ingesta de Datos

Fuentes de Datos:

-API meteorológica para obtener temperatura, humedad y otros factores climáticos.

-Datos históricos de consumo energético de usuarios.

-Precios del mercado eléctrico en tiempo real.

Procesamiento Inicial:

-Obtención de datos mediante scripts en Python.

-Transformación y limpieza de datos antes del almacenamiento.

Almacenamiento de Datos

ElasticSearch:

-Base de datos NoSQL utilizada para indexar y almacenar datos históricos y predicciones.

-Permite búsquedas eficientes y análisis rápidos de tendencias de consumo y precio.

Volúmenes Persistentes en Docker:

-Para mantener la integridad de los datos almacenados en contenedores.

Procesamiento y Modelado

Modelos de Predicción en Python:

-Modelo de Consumo Energético:

Basado en series temporales (LSTM).

Toma en cuenta patrones históricos y factores climáticos.

-Modelo de Predicción de Precios:

Basado en distintos modelos según el más adecuado para los datos en concreto (Random forest, regresión lineal, GB y XGB).

Integra tendencias del mercado eléctrico, consumos, localización, tipo de vivienda, número de habitantes y potencia contratada .

Ejecutados dentro de contenedores Docker:

-Para garantizar portabilidad y replicabilidad del entorno.

API y Backend

FastAPI:

-Expone endpoints REST para interactuar con los modelos.

-Permite recibir solicitudes y devolver predicciones.

Gestión de Solicitudes:

-API recibe parámetros necesarios para los modelos.

-Consulta ElasticSearch para datos históricos.

-Llama a los modelos predictivos y devuelve resultados.

Interfaz Web

Interfaz y funcionamiento web:

-Html como estructura de la página

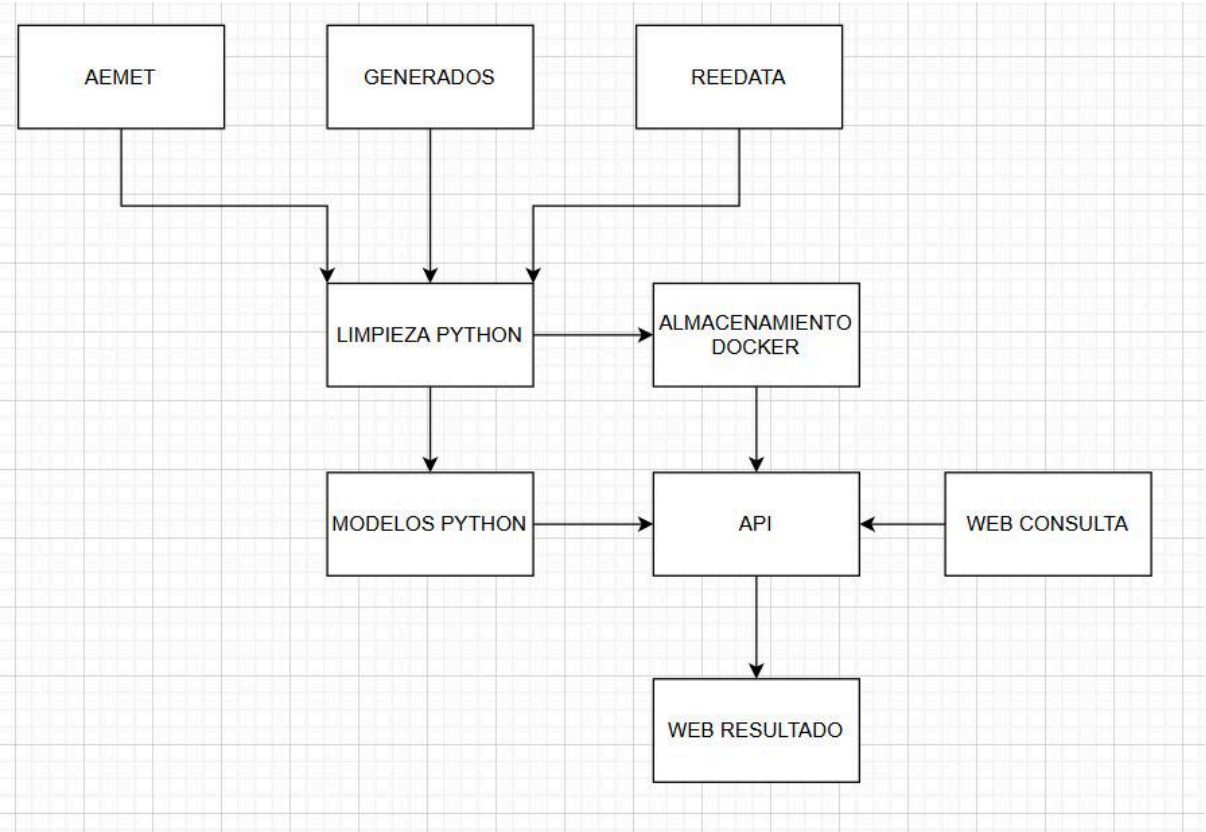
-Funcionalidad de la página aportada por JavaScript tales que panel de usuario que muestra predicciones y tendencias en gráficos o botones dentro de la web.

Comunicación con API:

-Consultas dinámicas al backend para obtener estimaciones de facturación.

-JavaScript utiliza un .py que se conecta con FastAPI para obtener datos de las predicciones y visualizarlos.

Diagrama de infraestructura del proyecto



Flujo de Datos

1. **Ingesta de Datos**
    - **Fuentes de datos**:
        - **API meteorológica**: Datos de temperatura, humedad, precipitación y viento de la AEMET.
        - **Red Eléctrica Española**: Datos de precios horarios de electricidad.
        - **Datos históricos de consumo**: Información generada artificialmente basada en patrones de consumo doméstico.
        - **Días festivos**: Archivo CSV con festivos nacionales y autonómicos en España.
    - **Procesamiento inicial**:
        - Descarga y almacenamiento de los datos en formato **CSV**.
        - Conversión de fechas al formato estándar `YYYY-MM-DD`.
        - Normalización de nombres de provincias.
2. **Preprocesamiento y Transformación**
    - **Limpieza de Datos**:

        - Eliminación de valores nulos.
        - Corrección de nombres de provincias.
        - Filtrado temporal (datos posteriores a 2014).
    - **Enriquecimiento de Datos**:
        - Unión de datos meteorológicos y precios de electricidad.
        - Adición de columnas como **"Festivo"** y **"Entre Semana"**.
    - **Exportación del Dataset Final**:
        - Generación del archivo **"Modelo_Precios_Met_Fest.csv"**, que contiene:
            - **Datos meteorológicos**
            - **Precios de electricidad**
            - **Días festivos**
            - **Día de la semana (laboral o fin de semana)**
3. **Modelado Predictivo**
    - **Modelo de Predicción del Consumo Energético**:
        - Variables: Características de la vivienda, factores demográficos, condiciones meteorológicas y patrones de uso.
        - Modelos: **Regresión Lineal, Random Forest, Gradient Boosting, XGBoost**.
        - Se elige el mejor modelo por provincia según **MSE (Error Cuadrático Medio)** y **R^2 (Coeficiente de Determinación)**.
    - **Modelo de Predicción del Precio de la Electricidad**:
        - Modelos utilizados: **LSTM** (Redes Neuronales Recurrentes).
        - Se ajustan los modelos con datos históricos y factores externos (clima, demanda, festivos).
4. **Almacenamiento de Datos**
    - Se utiliza **ElasticSearch** para almacenar los datos y resultados de las predicciones.
    - Uso de **Docker** para mantener la integridad y portabilidad de los contenedores con modelos entrenados.
5. **API y Backend**
    - **FastAPI** gestiona la comunicación entre la interfaz web y los modelos predictivos.
    - La API recibe solicitudes con parámetros de consumo y devuelve:
        - Predicción de precios de electricidad.
        - Predicción del consumo energético del hogar.
        - Estimaciones de factura mensual.

6. **Interfaz Web**
    - **HTML y JavaScript** para la estructura y funcionalidad de la página.
    - **Gráficos dinámicos** para mostrar tendencias de precios y consumo.
    - Conexión con **FastAPI** para obtener predicciones.

Conclusiones y trabajo futuro

En este apartado se destacan los logros principales del proyecto TerraWatt y las conclusiones clave obtenidas a partir del trabajo realizado. Se analizan los avances alcanzados en la gestión energética doméstica mediante modelos predictivos y cómo estos pueden ayudar a optimizar el consumo y reducir costos. Además, se identifican las áreas de mejora y las oportunidades de seguir perfeccionando el proyecto, planteando propuestas concretas para su expansión y aplicación futura.

Resultados obtenidos

Nuestro proyecto logró resultados importantes que muestran cómo los modelos predictivos pueden ser una herramienta práctica para mejorar la gestión energética en los hogares. Entre los principales logros están:

- Modelos predictivos eficientes: Se desarrollaron dos modelos, uno para anticipar los precios de la electricidad y otro para predecir el consumo energético en los hogares. Ambos funcionan bien al analizar datos históricos, clima y patrones de uso semanal, ofreciendo predicciones bastante precisas.

- Uso de datos variados: Se integraron diferentes fuentes de información, como precios históricos, consumo, clima y hasta días festivos, lo que permitió crear datasets muy completos que mejoraron la calidad de las predicciones.

- Mejora en la planificación: Los usuarios de TerraWatt obtuvieron herramientas útiles para planificar su consumo, detectar periodos de alto costo y ajustar sus hábitos de manera más eficiente.

- Consumo más sostenible: Además, las herramientas fomentan un uso más consciente de la energía, ayudando a las familias a ahorrar y alineándose con los esfuerzos por promover la sostenibilidad energética en Europa.

Conclusiones del trabajo

El proyecto demostró que los modelos predictivos pueden ser una solución práctica y efectiva para enfrentar los desafíos de la gestión energética en los hogares. Utilizando datos históricos y contextuales, se logró anticipar tanto los precios de la electricidad como el consumo energético de manera precisa, facilitando que los usuarios puedan planificar mejor su consumo y optimizar sus gastos.

Un aspecto clave fue la incorporación de datos adicionales como el clima, los días festivos y los patrones de actividad semanal, lo que mejoró significativamente la calidad de las predicciones. Esto resalta la importancia de usar un enfoque integral al analizar datos energéticos, ya que permite generar resultados más útiles y relevantes.

Los modelos desarrollados demostraron ser escalables y adaptables, lo que significa que pueden ajustarse para funcionar en diferentes regiones y contextos. Esto abre la puerta a que TerraWatt se implemente en una variedad de entornos, ampliando su alcance y beneficios.

En resumen, TerraWatt es una herramienta innovadora y útil para la gestión energética en los hogares. Sin embargo, aún hay áreas por mejorar y muchas oportunidades para seguir desarrollando el proyecto en el futuro.

Trabajo futuro

Si bien el proyecto logró cumplir con los objetivos principales, existen diversas áreas que podrían explorarse para mejorar y ampliar sus capacidades. Vamos a presentar algunas propuestas y direcciones futuras que podrían fortalecer la precisión de los modelos predictivos según las horas del día, mejorar la accesibilidad a la web y aumentar el valor que ofrecemos a nuestros usuarios.

- **Optimización del consumo por horas.**

Actualmente, los datos utilizados en nuestros modelos no incluyen información específica sobre las horas del día, lo que limita nuestra capacidad para optimizar el consumo en función de las variaciones horarias de los precios. Por esta razón, uno de nuestros próximos objetivos será desarrollar modelos predictivos específicos para cada franja horaria, permitiendo a los usuarios ajustar su consumo de manera más precisa y eficiente según el momento del día.

Para lograr esto, diseñaremos modelos avanzados que analicen el consumo energético en diferentes regímenes horarios, basados en datos históricos y actuales. Estos modelos no solo identificarán patrones de consumo, sino que también proporcionarán recomendaciones personalizadas para cada usuario, teniendo en cuenta las siguientes franjas horarias:

- **Horas punta:** Estas son las horas de mayor demanda y costos más altos. Los modelos buscarán estrategias para minimizar el consumo durante estos periodos.
- **Horas llano:** Durante este horario los precios son más moderados y ofreceremos consejos para equilibrar el uso de energía.
- **Horas valle:** Estas representan los momentos de menor costo de la electricidad. A través de los modelos, fomentaremos un mayor uso de energía en estas franjas.

Con esta optimización, no solo mejoraremos la precisión y relevancia de las predicciones, sino que también ayudaremos a los usuarios a adoptar hábitos energéticos más conscientes, aprovechando al máximo las variaciones de precio a lo largo del día y contribuyendo a una gestión energética más sostenible.

- **Adaptar página web para uso móvil.**

Actualmente, la página web de TerraWatt está diseñada para su uso en ordenadores, lo que limita su accesibilidad y comodidad para usuarios que prefieren dispositivos móviles. Por ello, uno de nuestros objetivos principales es transformar la plataforma en una versión completamente adaptable a diferentes tamaños de pantalla, asegurando una experiencia fluida y eficiente en teléfonos y tablets.

Para lograr esta transición, desarrollaremos una interfaz intuitiva que mantenga el diseño funcional y visual de la versión original, pero que se ajuste dinámicamente a cualquier dispositivo. Esto garantizará que los usuarios puedan acceder a las funcionalidades de TerraWatt en cualquier lugar y momento, sin restricciones.

Además, realizaremos pruebas exhaustivas en una amplia variedad de dispositivos móviles para identificar posibles errores o problemas de compatibilidad. Tomaremos en cuenta los comentarios de los usuarios durante estas pruebas para realizar ajustes y mejoras continuas, asegurando que la plataforma cumpla con sus necesidades.

Como parte de esta actualización, también implementaremos un sistema de notificaciones, que mantendrá a los usuarios informados sobre cambios en los precios de la electricidad, consejos de ahorro energético, recordatorios de consumo eficiente, etc. De este modo,

TerraWatt no solo será más accesible, sino también más proactivo en ayudar a los usuarios a gestionar su energía de manera consciente y efectiva.

- **Mayor interacción con el cliente.**

Para mejorar la interacción con nuestros usuarios, uno de los objetivos principales será ofrecerles recomendaciones personalizadas para optimizar su consumo energético. Estas sugerencias están basadas en un análisis detallado de sus patrones de consumo y datos contextuales, como horarios de mayor demanda y tarifas energéticas. De esta manera, los usuarios podrán gestionar su consumo de manera más eficiente, reduciendo costos y fomentando un uso más consciente de la energía.

Las recomendaciones incluirán consejos prácticos, como estrategias para reducir el uso de energía durante las horas pico, maximizar el aprovechamiento de tarifas bajas en horarios valle, o incluso sugerencias sobre electrodomésticos más eficientes que podrían sustituir a los actuales para lograr un ahorro significativo a largo plazo. Estas sugerencias estarán diseñadas para ser fáciles de aplicar y adaptadas a las necesidades específicas de cada usuario.

Para centralizar esta funcionalidad, implementaremos una nueva sección en la plataforma dedicada exclusivamente a las recomendaciones personalizadas. En esta área, los usuarios podrán acceder a sus consejos energéticos de forma clara y sencilla, con información que les ayude a tomar decisiones más inteligentes sobre su consumo diario.

Con estas mejoras, buscamos no solo ofrecer un servicio más completo, sino también empoderar a los usuarios con herramientas útiles que los motiven a adoptar hábitos de consumo más sostenibles y a optimizar su experiencia energética con TerraWatt.

Cómo ejecutar el proyecto

Para implementar el proyecto debemos de tener en cuenta varios aspectos, entre otros, que debemos de tener instalado python (última versión), así como la última versión de las librerías utilizadas en el trabajo, es importante a tener en cuenta que si se cuenta con otra versión tanto de python como de las librerías el trabajo no se ejecutará.

Por otro lado debemos de tener en cuenta que el proyecto tiene dos partes fundamentales, la conexión a Docker, en la cuál se suben y ejecutan todos los archivos, guardándose en el contenedor correspondiente y con su conexión a elasticsearch. Luego en la segunda parte

vemos la página web con su propia API y endpoint, en la cuál se conectan nuestros datos de manera que nos permite implementar el modelo.

Por lo que primeramente vamos a explicar como conectar nuestros archivos a Docker y elasticsearch, para ello debemos de guardar el trabajo en nuestro ordenador y abrir el cmd para poner los siguientes comandos.

Primero deberemos de buscar nuestra carpeta:

```
cd {Ruta_de_la_carpeta}
```

```
cd Desktop/TerraWatt/Limpieza_datos
```

Una vez hecho esto primeramente deberemos de crear nuestra network y el contenedor de elasticsearch, es recomendable primeramente visualizar los puertos disponibles para no tener ningún solapamiento.

```
docker network create elasticsearch-network-pbd1
```

```
docker run -d --name elasticsearch-pdb1 --net elasticsearch-network-pdb1 -p 9202:9200 -p 9203:9300 -e "discovery.type=single-node" -e "xpack.security.enabled=true" -e "ELASTIC_PASSWORD=changeme" elasticsearch:8.10.2
```

Una vez hecho esto debemos de crear la imagen de nuestra image:

```
docker build -t terrawatt_image .
```

A continuación, crearemos el contenedor sin ejecutarlo ya que vamos a proceder a forzar la conexión con la network, ya que ha sido uno de los problemas que hemos tenido y que más nos han retrasado.

```
docker create --name Terrawatt_container terrawatt_image
```

Como hemos comentado procederemos a la conexión forzada a la network

```
docker network connect elasticsearch-network-pdb1 Terrawatt_container
```

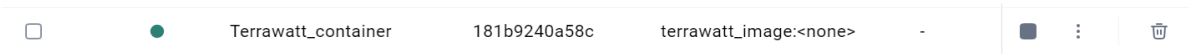
Por lo que una vez hecho esto ya podremos inicializar el contenedor, se nos mostrarán los resultados dentro de la app de docker, dentro del contenedor.

```
docker start Terrawatt_container
```

Una vez ejecutado esto, que tardará unos 30 minutos aproximadamente, se recomienda descargar los datos limpios y los modelos para poder visualizar todos los cambios así como para entender mejor los datos y el modelo, luego los subiremos a elasticsearch a partir de ahora realizar las consultas desde nuestra nueva base de datos. Para ellos en la terminal de Visual Studio (o del programa que se haya usado) y deberemos de ejecutar los siguiente comandos:

```
docker cp {id_container}:/app/{archivos_a_descargar} ./{archivos_a_descargar}
```

El id_container lo encontraréis dentro de la aplicación de docker en la misma sección de Containers en la segunda columna. Como se puede ver aquí, en mi caso el id sería 181b9240a58c



Por otro lado los archivos también podemos obtener estos archivos dentro de la propia app de Docker, dentro de nuestro contenedor, si nos vamos al apartado de “Files”, debemos de entrar en la subcarpeta “App”, dentro de esta carpeta encontraremos todos nuestros datos.

Una vez hecho esto ya tendremos nuestros archivos ejecutados ya se habrán cargado los datos es nuestro docker.

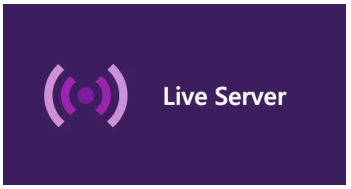
Por otro lado vamos a explicar primeramente que archivos tenemos inicialmente y cuáles debemos de obtener después de ejecutar todos nuestros .py para luego poder guardarlos y ejecutar nuestra página web.

Este es el proceso simplificado de le ejecución del proyecto, primeramente deberemos de ejecutar todos los archivos que se encuentran indicados en el apartado de Limpieza_datos, para obtener el archivo final de “Modelo_Precios_Met_Fest.csv”, el cuál usa diferentes archivos generados de manera intermedia para obtener la información.

Y por último deberemos de ejecutar los archivos de creación de los modelos, el cúal creará y ejecutará una carpeta con los modelos guardados, que deberá de estar guardada para poder implementar la página web.

En nuestro caso el método que hemos utilizado para poder ejecutarla es una extensión en Visual Studio (aunque esto también se podría hacer mediante el cmd simplemente ejecutando el archivo en python), nosotros hemos decidido esta opción ya que nos permitía, poder visualizar todos los cambios que realizamos en tiempo real.

La extensión utilizada es



x

Antes de poder visualizar la página web debemos de realizar varios comandos en nuestro cmd para poder establecer la conexión con nuestra API.

```
cd {Ruta_de_la_carpeta}
```

```
cd Desktop/TerraWatt/API_conexion
```

```
uvicorn main:app --reload
```

Con esto ya tendremos establecida nuestra conexión con la API, por lo que ya está todo internamente conectado, por lo que ya en nuestra página web podremos hacer peticiones.

Una vez realizado esto, deberemos de seleccionar nuestro archivo que se encuentra dentro de la carpeta *Web_Terrawatt*, podremos elegir cualquiera de los archivos .html, pero recomendamos realizarlo en la index.html, ya que este nos llevará a la página principal de la web.

Una vez estemos en nuestro archivo deberemos de hacer click derecho y elegir la opción de "*Open with Live Server*" y automáticamente se nos abrirá en nuestro navegador la página web.