Universitat Oberta de Catalunya (UOC)

Máster Universitario en Ciencia de Datos (*Data Science*)

TRABAJO FINAL DE MÁSTER

Área: Data Analytics in Industrial and Business Environments

Predicción de tendencias en los precios de lonja del bovino usando técnicas de aprendizaje automático

---

Autor: Víctor Jarreta Espligares

Tutor: Rafael Luque Ocaña

Profesor: Susana Acedo Nadal

Zaragoza, 29 de diciembre de 2024

Créditos/Copyright

FICHA DEL TRABAJO FINAL

Título del trabajo:	Predicción de tendencias en los precios de lonja del bovino usando técnicas de aprendizaje automático
Nombre del autor:	Víctor Jarreta Espligares
Nombre del colaborador/a docente:	Rafael Luque Ocaña
Nombre del PRA:	Susana Acedo Nadal
Fecha de entrega (mm/aaaa):	01/2025
Titulación o programa:	Máster en Ciencia de Datos
Área del Trabajo Final:	Data Analytics in Industrial and Business Environments
Idioma del trabajo:	Español
Palabras clave	Machine Learning, Natural Language Processing, Smart Farming

Dedicatoria/Cita

A mi familia, por su apoyo incondicional, y a mis amigos, por su ánimo y su capacidad de distracción.

Agradecimientos

A mi tutor, Rafael Luque Ocaña, por sus consejos y recomendaciones a lo largo de este proyecto. Al Grupo de Sistemas de Información Avanzados de la Universidad de Zaragoza, en el que durante los últimos años he tenido la oportunidad de trabajar y aprender de grandes profesionales. A mis compañeros de máster, por su apoyo y su compañía durante estos meses. A la Lonja Agropecuaria de Binéfar, por su colaboración y su disposición en este proyecto.

Resumen

In all kinds of markets and sectors, there are moments of price fluctuations that can be caused by different factors, such as changes in demand, supply, the economy, politics, society, etc. These changes can be predictable, either because they are cyclical or because they have occurred in the past, or they may be unpredictable, such as, for example, those caused by a pandemic.

Neural network models based solely on time series cannot acknowledge these latter changes since they do not consider the context in which they occur at certain prices.

Therefore, in this project, the integration of natural language processing is proposed in price prediction models to improve efficiency, or at a minimum, minimize errors in price predictions at abnormal times.

En todo tipo de mercados y sectores existen momentos de fluctuaciones de precios que pueden ser provocados por diferentes factores, como cambios en la demanda, en la oferta, en la economía, en la política, en la sociedad, etc. Estos cambios pueden ser predecibles, ya sea porque son cíclicos o porque se han producido en el pasado, o pueden ser impredecibles, como por ejemplo, los provocados por una pandemia.

Los modelos de redes neuronales basados únicamente en series temporales son incapaces de detectar estos últimos cambios, ya que no tienen en cuenta el contexto en el que se producen ciertos precios.

Por ello, en este proyecto se propone la integración de procesamiento de lenguaje natural en modelos de predicción de precios con el objetivo de mejorar la eficiencia, o como mínimo, minimizar los errores en las predicciones de precios en momentos anómalos.

Palabras clave: Machine Learning, Natural Language Processing, Smart Farming.

Índice general

Índice de figuras

Índice de cuadros

1. Introducción

Los sectores agropecuarios son fundamentales para la economía de muchos países, ya que proporcionan alimentos, materias primas y empleo a una gran parte de la población. En estos sectores, los precios tienen un componente volátil y cambiante, cuyo grado de variabilidad depende de múltiples factores e impacta de forma directa en los consumidores, los productores, los intermediarios y los gobiernos. Esta volatilidad en los precios puede ser causada por factores internos, como la oferta, la demanda o por la temporada, o por factores externos, como eventos geopolíticos, cambios en la política económica o desastres naturales.

Para los consumidores, controlar la volatilidad de los precios es fundamental para garantizar la seguridad alimentaria y el acceso a alimentos asequibles y de calidad. Para los productores, controlar la volatilidad significa mejorar la rentabilidad de sus explotaciones y la sostenibilidad de sus negocios. Para los intermediarios, significa reducir riesgos y aumentar beneficios. Y para los gobiernos, significa garantizar la estabilidad económica y social del país.

En muchos mercados de estos sectores, los procesos de fijación de precios se llevan a cabo por consenso entre los agentes del mercado, que se reúnen periódicamente en lonjas o mercados para acordar los precios de los productos. Estos agentes proponen incrementos o decrementos en los precios en función de su interpretación del estado del mercado, información pública e información privada que poseen. Este proceso de fijación es injusto para los agentes que no tienen tanta información y los pone en una posición de desventaja.

En este contexto, la predicción de precios de productos agropecuarios es una herramienta fundamental para la toma de decisiones en los sectores agropecuarios, ya que permite a los consumidores, los productores, los intermediarios y los gobiernos anticipar posibles fluctuaciones

en los precios y tomar medidas preventivas para mitigar los riesgos asociados a estos cambios.

En este proyecto se propone la creación de un sistema de predicción de precios de productos ganaderos, en concreto de vacuno, basado en redes neuronales. Este sistema utilizará datos históricos públicos de precios de productos ganaderos y comentarios de contexto socio-económico asociados a los precios privados, ambos recopilados de la Lonja de Binéfar, una lonja de productos agropecuarios de la provincia de Huesca.

Este sistema de predicción será comparado posteriormente con otro modelo de predicción de precios que no cuenta con los comentarios de contexto socio-económico, que se está utilizando actualmente en la lonja, para evaluar si la integración de técnicas de procesamiento de lenguaje natural en los modelos de predicción de precios mejora la eficiencia de los modelos en la detección de cambios de tendencias en los precios.

1.1. Contexto y motivación

En el sector agropecuario, la predicción de precios es una tarea compleja debido a la naturaleza volátil y no lineal de los precios, que pueden ser influenciados por múltiples factores internos y externos. En este contexto, la integración de técnicas de procesamiento de lenguaje natural en los modelos de predicción de precios puede mejorar la eficiencia de los modelos en la detección de cambios de tendencias en los precios, ya que permite capturar factores externos que influyen en los precios, como noticias, informes, redes sociales y otros datos no estructurados.

En la figura 1 se muestra la evolución del precio de un producto ganadero desde 2015 hasta 2024, donde se puede observar un incremento significativo en el precio a finales de 2021, provocado por la guerra en Ucrania que ha sido uno de los mayores productores agrícolas del mundo. Esta situación no puede ser capturada por los modelos tradicionales de predicción de precios, ya que al basarse únicamente en los históricos de precios, no son capaces de detectar cambios puntuales en los precios causados por eventos inesperados.

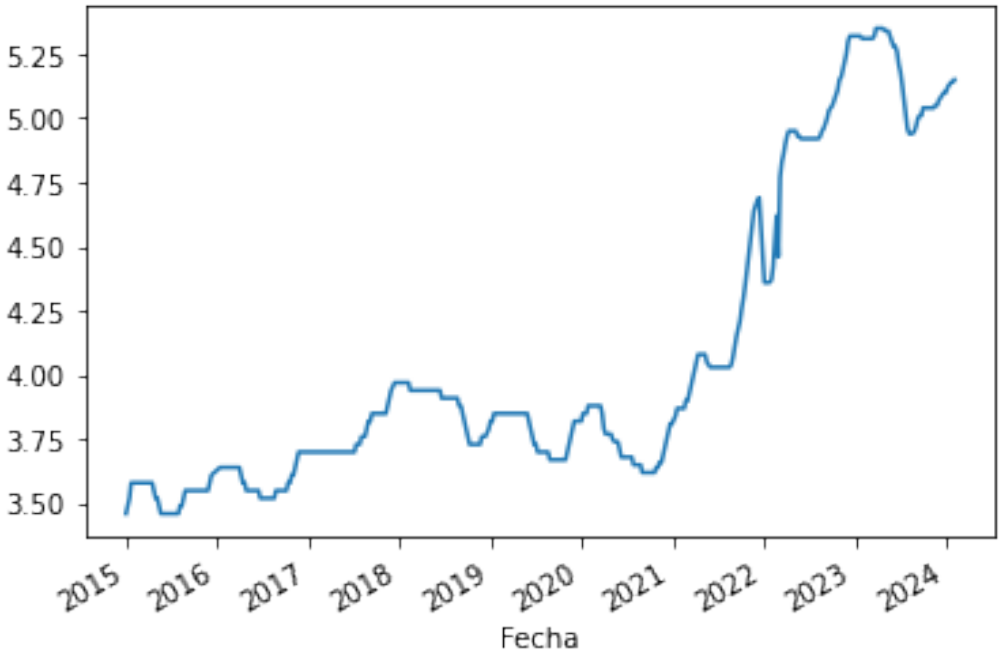

Figura 1: Evolución del precio durante los años.

Esta situación pone en manifiesto la necesidad imperativa de contar con modelos de predicción que no se basen únicamente en los históricos de precios, sino que también es necesario contar con información adicional que aporte un sentido a los precios, que pueda entender el sentimiento de los agentes del mercado sobre una situación concreta.

Este sentimiento de los agentes suele estar reflejado en sus opiniones a la hora de llegar a un consenso en la fijación de precios, donde una situación de incertidumbre o de tensión suele provocar un estancamiento en los precios, ya que el desconocimiento de la dirección que tomará el mercado provoca que los agentes sean más conservadores a la hora de proponer cambios en los precios.

1.2. Objetivos

En este proyecto se propone la creación de un sistema de predicción de precios de productos ganaderos, en concreto de vacuno, basado en redes neuronales. El modelo contará con una parte de procesamiento de lenguaje natural para analizar documentos de texto y una parte de redes neuronales *Long Short-Term Memory* (LSTM) [1] para analizar series temporales de precios. El análisis de los documentos de texto se efectuará gracias a *Transformers* [2], una técnica de procesamiento de lenguaje natural que ha demostrado ser muy eficiente en dichas tareas, y el análisis de los precios se llevará a cabo con redes neuronales LSTM, que son muy eficientes con series temporales. El sistema se evaluará con datos reales y se comparará con otros modelos de

predicción de precios que no cuentan con el análisis de los documentos de texto.

El objetivo principal del proyecto es demostrar que al añadir información que aporte un contexto socio-económico a ciertos datos de entrada, se puede mejorar el rendimiento de los modelos de predicción de precios en situaciones poco comunes, es decir, comprobar si se puede reducir el error máximo en las predicciones.

Se ha dividido el objetivo principal en los siguientes objetivos parciales:

- Recopilar históricos de precios de los productos ganaderos.
- Recopilar comentarios de contexto socio-económico asociados a los precios.
- Desarrollar un modelo que integre los históricos de precios y los comentarios.
- Entrenar y validar el modelo con los datos recopilados.
- Evaluar el rendimiento del modelo en comparación con otros modelos de predicción de precios.

1.3. Sostenibilidad, diversidad y desafíos ético/sociales

No existe un impacto directo en la sostenibilidad, ya que no se trata de un proyecto que afecte a las materias primas o a la energía. Sin embargo, el proyecto puede tener un impacto indirecto en la sostenibilidad si se demuestra que el modelo propuesto es más eficiente en la predicción de precios y, por lo tanto, permite a la lonja tomar decisiones más informadas sobre la compra y venta de productos ganaderos.

El proyecto no tiene un impacto directo en aspectos éticos ni en términos de género, diversidad o derechos humanos, ya que se trata de un proyecto técnico cuyos datos de entrada son precios y documentos, y no implica ningún aspecto legal, social, ético, de género o de derechos humanos.

Respecto al impacto social, el proyecto contribuye a tomar decisiones más informadas en el sector ganadero, reduciendo la volatilidad en la planificación económica, mejorando la seguridad alimentaria y contribuyendo a los Objetivos de Desarrollo Sostenible, especialmente al ODS 2 (Hambre Cero), al ODS 8 (Trabajo Decente y Crecimiento Económico), ODS 9 (Industria, Innovación e Infraestructura) y al ODS 12 (Producción y Consumo Responsables).

1.4. Enfoque y metodología

Las estrategias potenciales para este proyecto respecto a las metodologías de desarrollo de software son las siguientes:

- Metodología en cascada: esta metodología es la más tradicional y se basa en una secuencia de fases en las que cada fase depende de la anterior. En este caso, la primera fase sería la recopilación de datos, seguida del desarrollo del modelo, la evaluación del modelo y la comparación con otros modelos. Esta metodología es adecuada para proyectos en los que los requisitos son claros y no se espera que cambien a lo largo del proyecto.

- Metodología ágil: esta metodología se basa en el desarrollo iterativo e incremental, en el que se van desarrollando pequeñas partes del proyecto y se van añadiendo funcionalidades a medida que se van completando. En este caso, se podrían desarrollar pequeñas partes del modelo y se podrían ir añadiendo funcionalidades a medida que se van completando. Esta metodología es adecuada para proyectos en los que los requisitos no están claros o pueden cambiar a lo largo del proyecto.

- CRIPS-DM: esta metodología se basa en un ciclo de vida de desarrollo de proyectos de minería de datos que consta de seis fases: comprensión del negocio, comprensión de los datos, preparación de los datos, modelado, evaluación y despliegue.

La metodología seleccionada para este proyecto es la metodología CRISP-DM, ya que es una metodología específica para proyectos de minería de datos cuya mayor desventaja es la falta de flexibilidad, la cual no es un problema en este caso ya que los objetivos del proyecto son claros y no se espera que cambien a lo largo del proyecto.

A continuación se define cada una de las fases de la metodología CRISP-DM:

- Comprensión del negocio: se entienden los objetivos y requisitos del proyecto desde un punto de vista empresarial.

- Comprensión de los datos: se recopilan los datos necesarios para el proyecto y se entiende su estructura y contenido.

- Preparación de los datos: se preparan los datos para el análisis, lo cual incluye la limpieza de los datos, la selección de los datos relevantes y la transformación de los datos.

- Modelado: se desarrolla el modelo de predicción de precios.

- Evaluación: se evalúa el rendimiento del modelo con los datos recopilados.

- Despliegue: se despliega el modelo en un entorno de producción.

La última fase se obvia en este proyecto ya que no se va a desplegar el modelo en un entorno de producción.

1.5. Planificación

La planificación del proyecto se dividide en diversas tareas y entregables que podemos ver en la Figura 2. La planificación se ha dividido en varias fases: definición del trabajo, estado del arte, implementación del proyecto, redacción de la documentación y defensa del proyecto.

A continuación se describen las tareas y sus sub-tareas, si las hubiera:

- Definición del trabajo: en esta fase se proporciona un contexto general del proyecto y se definen los objetivos, la metodología y la planificación.
- Estado del arte: en esta fase se revisa la literatura existente sobre el tema del proyecto y se identifican los modelos de predicción de precios existentes.
- Implementación del proyecto: esta fase se divide en varias sub-tareas: recopilación de datos, desarrollo del modelo, evaluación del modelo y comparación con otros modelos.
- Redacción de la documentación: en esta fase se redacta la documentación preliminar y final del proyecto, además de una presentación audiovisual.
- Defensa del proyecto: en esta fase se defiende el proyecto ante un tribunal, exponiendo el proyecto en sí, los resultados obtenidos y las conclusiones.

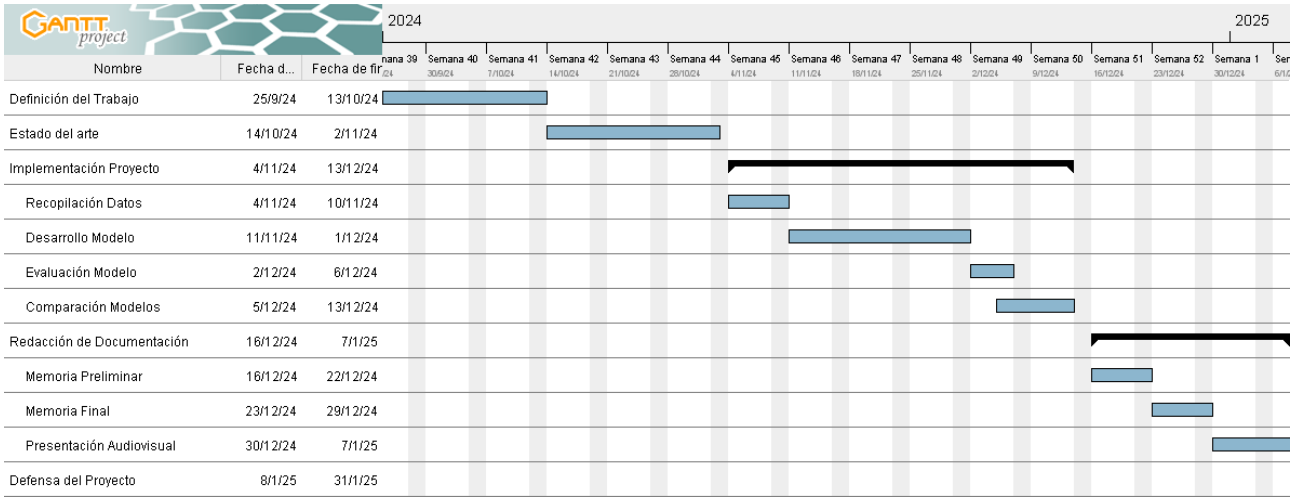

Figura 2: Planificación del proyecto.

1.6. Resumen de los productos del proyecto

Los productos obtenidos en este proyecto son:

- Un modelo de predicción de precios de productos de vacuno basado en redes neuronales.
- Un análisis de los resultados obtenidos con el modelo.
- Un informe final del proyecto (este documento).

El código fuente para la creación del modelo de predicción de precios, así como el análisis de los resultados obtenidos y el informe final del proyecto, se encuentran disponibles en el repositorio de GitHub del autor: https://github.com/VJarreta/uoc_tfm_data_science.

1.7. Breve descripción de los demás capítulos del informe

En el apartado de *Estado del Arte* se revisarán las metodologías más avanzadas en la predicción de precios, incluyendo los modelos de redes neuronales, las redes LSTM, los modelos híbridos y los modelos de transformadores, y se evaluarán las oportunidades y retos actuales y futuros para identificar las prácticas más prometedoras.

En el apartado de *Métodos y recursos* se describen los aspectos más relevantes del diseño y desarrollo del proyecto, la metodología utilizada y los productos creados.

En el apartado de *Resultados* se describen los resultados obtenidos con tres modelos de predicción de precios diferentes, incluyendo un modelo basado en redes neuronales con LSTM, un modelo basado en redes neuronales con una parte de procesamiento de lenguaje natural y el modelo anterior pero con optimización de hiperparámetros.

En el apartado de *Conclusiones y trabajo futuro* se describen las conclusiones del trabajo, una evaluación del grado de logro de los objetivos iniciales, una evaluación de la planificación y metodología utilizadas en el proyecto y una discusión de temas para futuros trabajos.

2. Estado del arte

Este Estado del Arte se centra en explorar las metodologías más avanzadas en la predicción de precios, revisando los modelos de redes neuronales, en particular las redes LSTM, en series temporales, las limitaciones de estos modelos y el impacto de integrar técnicas de NLP para capturar factores externos que influyen en los precios. También se abordan los enfoques híbridos que combinan modelos de series temporales y NLP. Finalmente, se analizarán las fortalezas y debilidades de cada enfoque y se evaluarán las oportunidades y retos actuales y futuros para identificar las prácticas más prometedoras.

2.1. Introducción a la predicción de precios

La predicción de precios es un campo de estudio amplio y complejo que abarca una extensa gama de aplicaciones, desde la predicción de precios de acciones y criptomonedas hasta la predicción de precios de productos agrícolas y ganaderos. La predicción de precios es un problema desafiante debido a la naturaleza volátil y no lineal de los precios, que pueden ser influenciados por múltiples factores externos, como eventos geopolíticos, cambios en la oferta y la demanda o factores macroeconómicos.

2.2. Técnicas tradicionales para la predicción de precios

Las técnicas tradicionales para la predicción de precios incluyen modelos econométricos, como el modelo ARIMA (AutoRegressive Integrated Moving Average), que se basa en la descomposición de una serie temporal en componentes estacionales, cíclicos y residuales para predecir los precios futuros. Los modelos ARIMA son eficaces en la predicción de precios a corto plazo, pero tienen limitaciones en predicciones a largo plazo.

Este tipo de modelos se han utilizado para la predicción en el mercado de valores [3] en los que se ha demostrado que obtienen un buen rendimiento, llegando a conseguir un 95.85 % de "accuracy" en algunos casos.

2.3. Redes neuronales para la predicción de precios

Las redes neuronales son un enfoque más avanzado para la predicción de precios que se basa en la simulación de las conexiones neuronales en el cerebro humano para aprender patrones complejos en los datos. Las redes neuronales recurrentes (RNN) son un tipo de red neuronal que es utilizada en la predicción de series temporales, aunque los datos más recientes tienen más peso a la hora de predecir, restando importancia a los datos más antiguos. Además, existe el problema de la desaparición del gradiente, que cuando se calcula el gradiente de la función de pérdida, este se va desvaneciendo a medida que se propaga hacia atrás en el tiempo, lo que hace que las RNN no sean lo suficientemente eficientes en la predicción de precios.

Para solucionar estos problemas, se diseñaron las redes LSTM, que son un tipo de red neuronal recurrente que supera las limitaciones de las RNN [4–6] al introducir una estructura de memoria a largo plazo que permite a la red recordar información relevante de los datos pasados y utilizarla en la predicción de precios. Las redes LSTM 3 cuentan con una celda de memoria que permite recordar u olvidar información en función de los datos de entrada, con tres puertas que controlan el flujo de información:

- Puerta de olvido: controla cuánta información de la celda de memoria se debe olvidar.

- Puerta de entrada: controla cuánta información nueva se debe añadir a la celda de memoria.

- Puerta de salida: controla cuánta información de la celda de memoria se debe utilizar en la predicción de precios.

Dichas puertas ajustan de manera dinámica la memoria de la celda para retener los datos más relevantes y descartar los menos relevantes a lo largo del tiempo, algo necesario a la hora de trabajar con series temporales complejas y no lineales.

Este modelo ya se ha utilizado en un contexto similar al de este proyecto, en el que se comparaban los resultados de diferentes modelos de predicción, incluido el LSTM, para predecir los precios de productos porcinos con datos nacionales españoles y europeos [7].

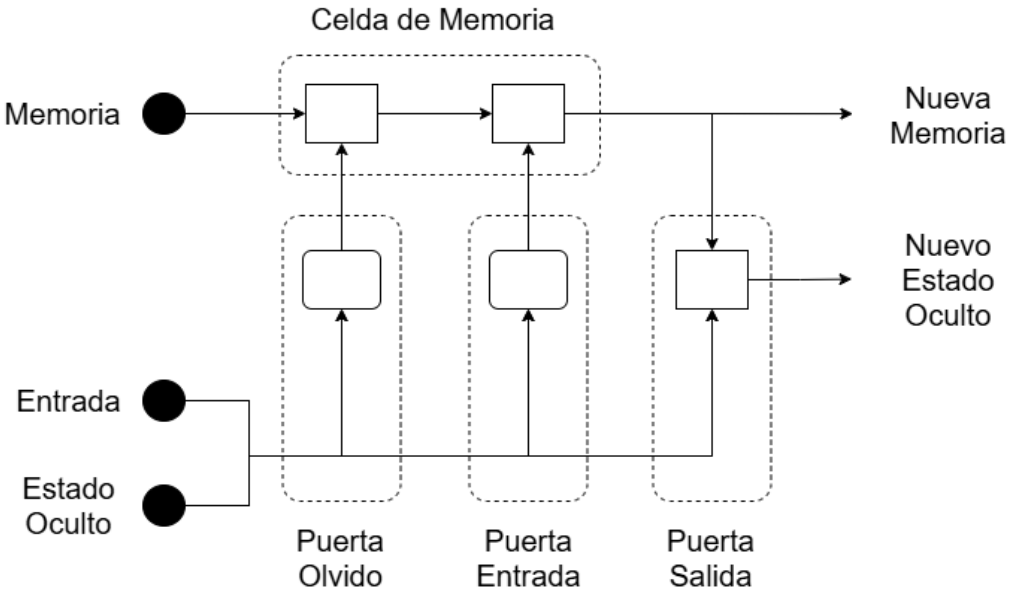

Figura 3: Arquitectura de una red LSTM.

2.4. Limitaciones de las redes neuronales en la predicción de precios

A pesar de su eficacia, las redes neuronales tienen algunas limitaciones en la predicción de precios. Una de las limitaciones más importantes es su incapacidad para capturar factores externos que influyen en los precios, por lo que no pueden detectar cambios puntuales en los precios causados por eventos inesperados, ya que no se trata de un evento cíclico o que se repita en el tiempo.

2.5. Procesamiento de lenguaje natural

El procesamiento de lenguaje natural (NLP) es un campo de la inteligencia artificial que se centra en la interacción entre las computadoras y el lenguaje humano. El *Natural Language Processing* (NLP) es una técnica que permite entender, interpretar y generar lenguaje humano de forma natural, y utiliza técnicas de aprendizaje automático y procesamiento de texto para analizar y extraer información de los datos de texto.

El NLP se ha utilizado en una amplia variedad de aplicaciones, como la traducción automática, la generación de texto, el análisis de sentimientos, la clasificación de texto, la extracción de información, la detección de spam y *fake news* [8], entre otros.

2.6. Procesamiento de lenguaje natural para la predicción de precios

El procesamiento de lenguaje natural se ha utilizado en la predicción de precios para capturar factores externos que influyen en los precios, como noticias, informes, redes sociales y otros datos no estructurados. Esto se consigue con la técnica *Sentiment Analysis*, que analiza el tono y la emoción de los textos para determinar si son positivos, negativos o neutrales [9, 10] , o con *Embeddings*, que convierten las palabras en vectores numéricos para que puedan ser procesados por los modelos de aprendizaje automático [11].

2.7. Modelos híbridos para la predicción de precios

Los modelos híbridos son aquellos que combinan técnicas de series temporales y procesamiento de lenguaje natural para mejorar la predicción de precios. Estos modelos utilizan redes neuronales LSTM para analizar las series temporales de precios y técnicas de NLP para analizar los datos de texto que aportan un contexto socio-económico a los precios.

Estos modelos híbridos han demostrado, en algunos casos, ser más eficientes en la predicción de precios que los modelos tradicionales y las redes neuronales por separado[12–16], gracias a la capacidad de conocer el contexto actual de los datos que se están analizando. Estos modelos se están utilizando actualmente de forma expiremental en el sector financiero, más concretamente en la predicción de precios del mercado de valores [17], donde se analizan noticias y/o redes sociales junto con los precios históricos para predecir los precios futuros.

2.8. Modelos de Transformadores

Los modelos de *Transformers* son una técnica de procesamiento de lenguaje natural diseñados inicialmente para tareas de traducción automática, generación de texto y análisis de sentimientos [18]. Los modelos más conocidos son BERT, GPT, T5, y Llama 4 que son capaces

de procesar largas secuencias de datos y captar relaciones complejas en los textos. La "auto-atención" es una de las características más importantes de los transformadores, que permite a los modelos conocer las partes más relevantes de la entrada con independencia de la posición de las palabras.

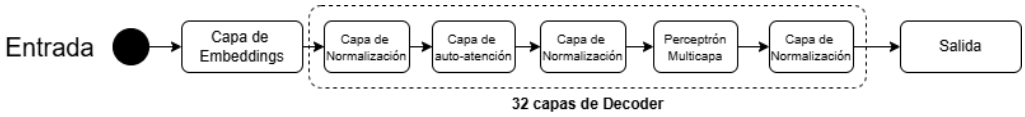

Figura 4: Arquitectura de Llama.

En los últimos años se han desarrollado variaciones de los modelos diseñados específicamente para series temporales, como *Temporal Fusion Transformers* (TFT), que combinan las capacidades de los transformadores con las de los modelos de series temporales y el *Informer*, diseñado para optimizar la eficiencia en series temporales extensas. Un uso común de estos modelos es la predicción de series temporales, como por ejemplo la predicción de la demanda energética [19].

2.9. Limitaciones de los modelos híbridos

A pesar de su eficacia, los modelos híbridos tienen algunas limitaciones en la predicción de precios. Una de las limitaciones más importantes es su dependencia de los datos de texto, ya que si los datos de texto no aportan información relevante, el modelo no será capaz de detectar los cambios en los precios causados por eventos inesperados. Otro problema es el ruido y el sobreajuste, ya que los datos de texto pueden contener información irrelevante o sesgada que puede afectar a la predicción de precios. Por último, el coste computacional de los modelos híbridos es mayor que el de los modelos tradicionales, ya que requieren más recursos y tiempo de entrenamiento.

En el caso de este proyecto, los documentos de texto se obtendrán de una lonja de productos agropedecuarios, escritos semanalmente por trabajadores de la misma, que contienen información relevante y no exclusivamente de los precios del vacuno, sino también de otro productos ganaderos y agrícolas. Esto último es importante, ya que los precios de los productos agropecuarios suelen estar influenciados entre sí y los precios de uno pueden afectar a los precios de otros, ya que suelen ser productos sustitutivos o complementarios.

2.10. Conclusiones del Estado del Arte

En este Estado del Arte se han revisado las metodologías tradicionales y avanzadas en la predicción de precios, incluyendo los modelos econométricos como ARIMA, las regresiones

lineales, los árboles de decisión, las redes neuronales recurrentes, las redes LSTM, los modelos híbridos y los modelos de transformadores. Se ha que la integración de técnicas de NLP en los modelos de predicción de precios puede mejorar la eficiencia de los modelos en la detección de cambios de tendencias en los precios a cambio de la dependencia de los datos de texto de calidad, cuyo problema se considera resuelto por el origen de los datos de texto utilizados en este proyecto y un mayor coste computacional. Se han identificado que los modelos híbridos, que combinan técnicas de series temporales y procesamiento de lenguaje natural, podrían ser los más adecuados para el proyecto propuesto.

3. Métodos y recursos

3.1. Diseño y desarrollo del proyecto

En esta sección se describen los aspectos más relevantes del diseño y desarrollo del proyecto.

3.1.1. Recopilación de datos

Para la recopilación de datos se han utilizado dos fuentes de información procedentes de la lonja de productos agropecuarios de Binéfar. La primera fuente de información son los históricos de precios de los productos ganaderos, en concreto de vacuno, que se han recopilado semanalmente desde 2015. La segunda fuente de información son los comentarios de contexto socio-económico asociados a los precios, que se han recopilado semanalmente desde 2020.

Se ha decidido utilizar la técnica de *Transfer Learning* para el procesamiento de los textos, utilizando el modelo de lenguaje Llama3, una herramienta de procesamiento de lenguaje natural que permite analizar los textos y extraer información relevante de ellos. Gracias a este procesamiento se han obtenido los datos vectoriales que representan el texto de los comentarios.

Es importante mencionar que aunque los históricos de precios son públicos, los comentarios de contexto socio-económico son privados e internos de la lonja, por lo que no se pueden compartir. Por este motivo, en el proyecto se proporciona un fichero que contiene los precios unidos con los vectores numéricos de los comentarios, pero no se proporcionan los comentarios originales.

3.1.2. Preparación de los datos

Los datos recopilados se han preparado para el análisis, lo cual incluye la unión de los distintos conjuntos de datos, los cuales se encuentran en tres archivos diferentes, el primero con los precios de los productos ganaderos desde 2015 hasta 2023 incluido, el segundo con los

precios restantes de 2024 y el tercero contiene los vectores numéricos de los comentarios de contexto socio-económico con su correspondiente fecha.

Esto sugiere que los datos disponibles para este proyecto son los de 2020 hasta la actualidad, descartando los datos anteriores debido a la carencia de comentarios que expliquen el contexto de esos precios. Esto supone una limitación en el modelo, ya que estamos perdiendo un 57.26 % de los datos, lo cual puede afectar a la eficiencia de los modelos a desarrollar.

Durante el análisis de los datos se ha detectado la presencia de valores extremos para un producto en concreto, cuyos valores eran de 5 céntimos, pero los valores vecinos eran de 5 euros. Tras una consulta con los trabajadores de la lonja, se ha determinado que se trata de un error en la recopilación de los datos, y que el valor correcto es de 5 euros. Por lo tanto, se ha procedido a corregir el error en los datos.

3.1.3. Desarrollo de los modelos

Los modelos de predicción de precios se han desarrollado en Python utilizando la biblioteca de aprendizaje automático TensorFlow, que nos proporciona una amplia gama de herramientas y funciones para el desarrollo de modelos de redes neuronales.

Se han desarrollado tres modelos de predicción de precios:

- Modelo de series temporales: este modelo se basa en una red LSTM que analiza los históricos de precios de los productos ganaderos para predecir los precios futuros. La arquitectura del modelo sería la siguiente:

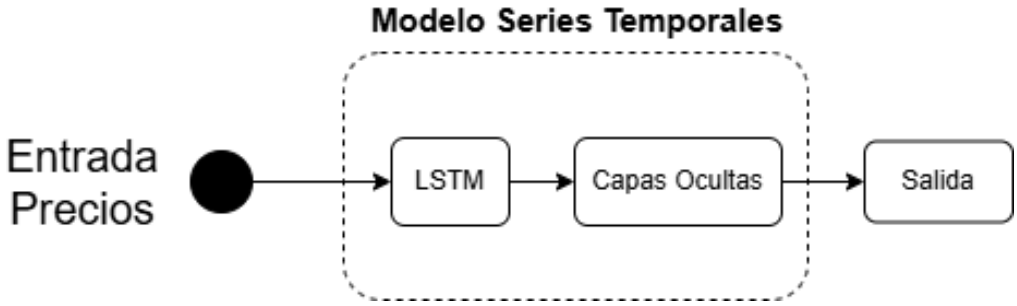

Figura 5: Arquitectura del modelo de series temporales.

- Modelo de series temporales con NLP: este modelo se basa en una capa LSTM para analizar los históricos de precios y una capa LSTM para analizar la salida de la primera capa LSTM y los vectores de texto de los comentarios. La arquitectura del modelo sería la siguiente:

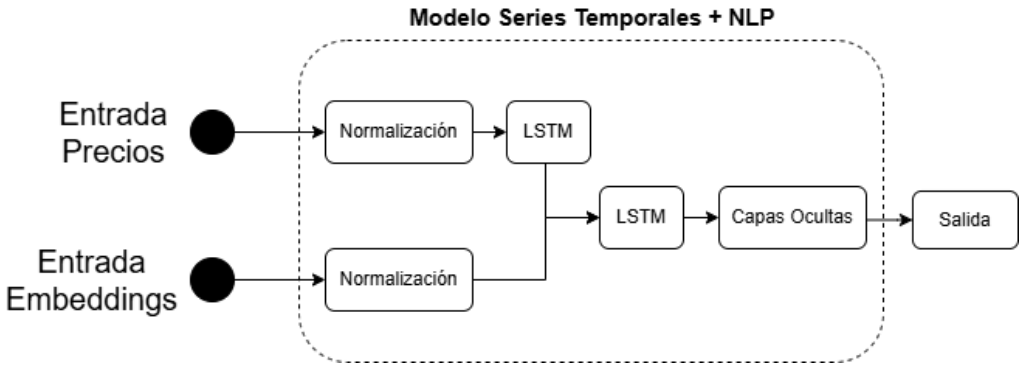

Figura 6: Arquitectura del modelo de series temporales con NLP.

- Modelo de series temporales con NLP optimizado: este modelo está basado en el modelo anterior, pero se ha utilizado Optuna, una herramienta de optimización de hiperparámetros, para encontrar los hiperparámetros y número de capas óptimos para el modelo. La arquitectura del modelo es la misma que la del modelo anterior, pero se diferencian en el número de capas (ocultas) y en los hiperparámetros, lo cual no influye en la arquitectura del modelo.

En el modelo de series temporales no existe una capa de normalización, ya que los precios de los productos ganaderos son muy poco oscilantes, y el valor mínimo y máximo de los precios es muy similar (aproximadamente una diferencia de 3€ entre el mínimo y el máximo entre diferentes productos). Sin embargo, en los modelos de series temporales con procesamiento de lenguaje natural sí que se ha añadido una capa de normalización, con el objetivo de que los vectores numéricos tengan la misma escala que los precios, ya que estos van desde -1 a 1.

3.1.4. Evaluación de los modelos

Los modelos se han evaluado con una serie de métricas de rendimiento, como el coeficiente de determinación R^2, el error absoluto medio (MAE), el error cuadrático medio (MSE), la raíz del error cuadrático medio (RMSE), la precisión y el error máximo.

Dichas métricas se han calculado tras una simulación del modelo en un entorno de producción, es decir, con los datos de testing se ha simulado la predicción de precios de forma semanal, donde se predice el precio de la semana siguiente con los datos accesibles en la semana actual, y posteriormente se reentrena el modelo.

3.2. Metodología

En esta sección se describe la metodología utilizada en el proceso de desarrollo del proyecto, describiendo las alternativas posibles, las decisiones que se han tomado y los criterios utilizados

para tomar estas decisiones.

3.2.1. Metodología de desarrollo

La metodología de desarrollo utilizada en este proyecto es la metodología CRISP-DM, que se basa en un ciclo de vida de desarrollo de proyectos de minería de datos. La metodología CRISP-DM consta de seis fases: comprensión del negocio, comprensión de los datos, preparación de los datos, modelado, evaluación y despliegue.

3.3. Metodología de los modelos

La metodología de los modelos se basa en el desarrollo de tres modelos de predicción basados en redes neuronales: un modelo de series temporales, un modelo de series temporales con NLP y un modelo de series temporales con NLP optimizado.

El modelo de series temporales, actualmente usado en producción, se ha creado con el objetivo de poder comparar los resultados obtenidos con los modelos de series temporales con NLP.

El modelo de series temporales con NLP se ha creado con el objetivo de mejorar la eficiencia de los modelos en la predicción de precios, o por lo menos minimizar los errores máximos durante la predicción de precios.

Una vez elegida la arquitectura del modelo que utiliza los textos, se ha utilizado la librería Optuna para optimizar los hiperparámetros y número de capas del modelo con el objetivo de mejorar la eficiencia de los modelos en la predicción de precios.

Cabe destacar que no se han utilizado capas de atención en los modelos, ya que se ha utilizado Llama3 para procesar los textos, el cual ya utiliza capas de atención en su arquitectura, lo que hace innecesario su uso en los modelos. Sin embargo, si se decidiese prescindir de Llama3, y por lo tanto procesar los textos directamente con los modelos, se debería añadir una capa de atención a los modelos para poder capturar las relaciones entre las palabras de los textos.

3.3.1. Metodología de evaluación

La metodología de evaluación utilizada se trata de una simulación de los modelos en un entorno de producción, donde se simula el rendimiento de las predicciones de los modelos en un entorno real. Para ello, se han utilizado los datos de testing para simular la predicción de precios de forma semanal, y tras finalizar la simulación se obtienen las métricas resultantes.

3.4. Productos creados

Los productos creados en este proyecto son:

- Un modelo de predicción de precios de vacuno basado en redes neuronales que utiliza series temporales.
- Un modelo de predicción de precios de vacuno basado en redes neuronales que utiliza series temporales y NLP.
- Un modelo de predicción de precios de vacuno basado en redes neuronales que utiliza series temporales, NLP y optimización de hiperparámetros y número de capas.
- Un análisis de los resultados y comparación de los mismos entre los distintos modelos.
- Un informe final del proyecto.

4. Resultados

4.1. Resultados del modelo de series temporales

Mostramos los resultados obtenidos con el modelo de series temporales en la Tabla 1.

Métrica	Valor
R^2	0.99355
MAE	0.0073767
MSE	0.0001376
RMSE	0.0117285
Accuracy	99.8663 %
Error Máx.	0.05 €

Cuadro 1: Resultados del modelo de series temporales.

En la figura 7 se muestra la evolución de los precios reales y predichos por el modelo de series temporales en los datos de testing.

4.2. Resultados del modelo de series temporales con NLP

Mostramos los resultados obtenidos con el modelo de series temporales con NLP en la Tabla 2.

En la figura 8 se muestra la evolución de los precios reales y predichos por el modelo de series temporales con NLP en los datos de testing.

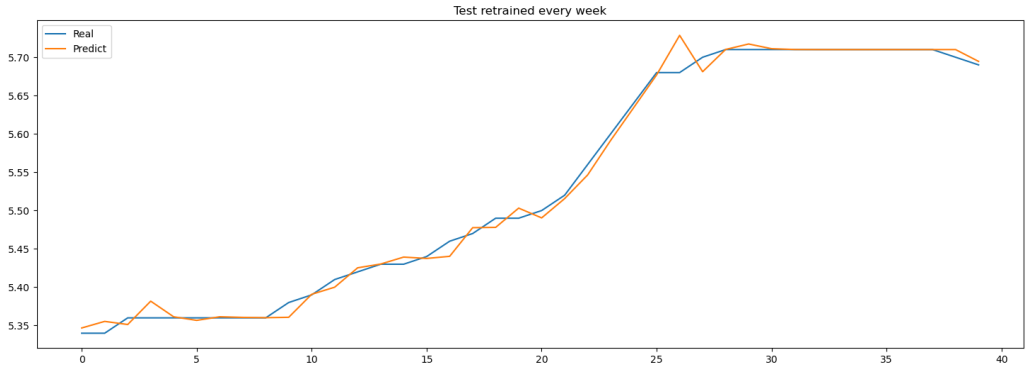

Figura 7: Resultados del modelo de series temporales.

Métrica	Valor
R^2	0.98947
MAE	0.010852
MSE	0.0002246
RMSE	0.0149882
Accuracy	99.8042 %
Error Máx.	0.04 €

Cuadro 2: Resultados del modelo de series temporales con NLP.

4.3. Resultados del modelo de series temporales con NLP optimizado

Mostramos los resultados obtenidos con el modelo de series temporales con NLP optimizado en la Tabla 3.

Métrica	Valor
R^2	0.99195
MAE	0.00985
MSE	0.0001717
RMSE	0.0131053
Accuracy	99.8205 %
Error Máx.	0.03 €

Cuadro 3: Resultados del modelo de series temporales con NLP optimizado.

En la figura 9 se muestra la evolución de los precios reales y predichos por el modelo de series temporales con NLP optimizado en los datos de testing.

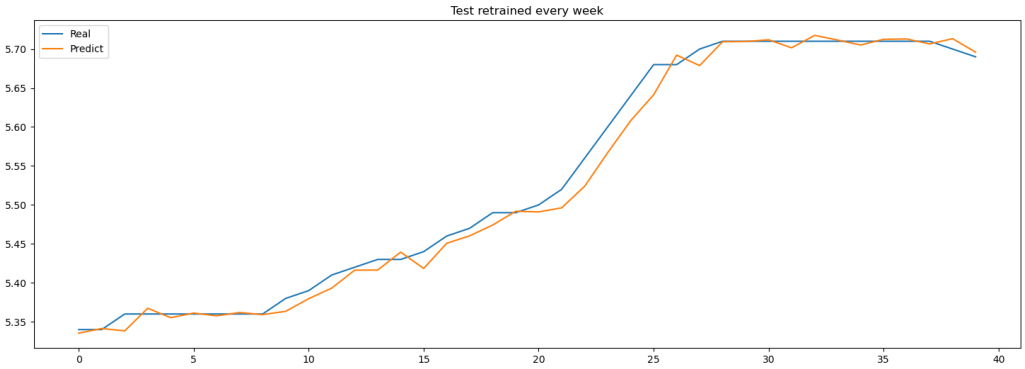

Figura 8: Resultados del modelo de series temporales con NLP.

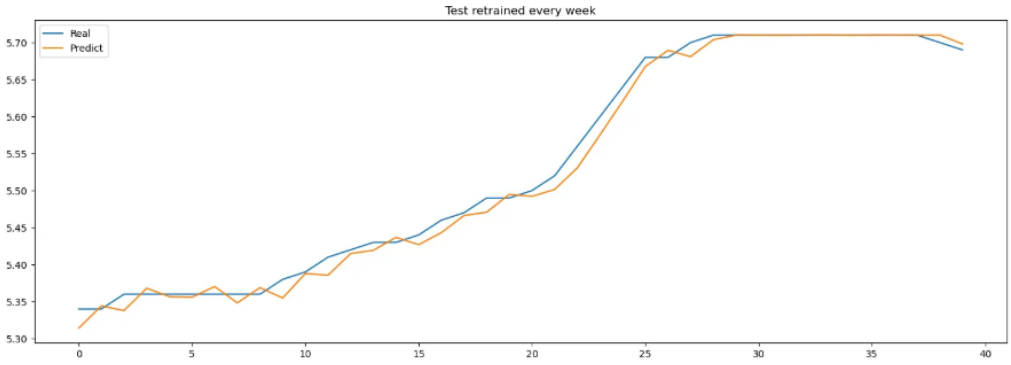

Figura 9: Resultados del modelo de series temporales con NLP optimizado.

4.4. Comparación de los modelos

En la Tabla 4 se muestra una comparación de los resultados obtenidos con los tres modelos desarrollados.

Modelo	R^2	MAE	MSE	RMSE	Accuracy	Error Máx.
Series Temporales	**0.99355**	**0.0073767**	**0.0001376**	**0.0117285**	**99.8663 %**	0.05 €
Series + NLP	0.98947	0.010852	0.0002246	0.0149882	99.8042 %	0.04 €
Series + NLP Opt	0.99195	0.00985	0.0001717	0.0131053	99.8205 %	**0.03 €**

Cuadro 4: Comparación de los resultados de los modelos.

Podemos ver que el modelo que utiliza únicamente series temporales obtiene unos resultados muy buenos, teniendo en cuenta el contexto, aunque el error máximo es relativamente elevado.

El modelo que utiliza series temporales con NLP obtiene unos resultados ligeramente peores que el modelo de series temporales, aunque el error máximo es menor. Dado que el rendimiento del modelo es pobre en comparación al anterior, se espera que el modelo optimizado mejore los resultados.

El modelo optimizado, como se esperaba, obtiene los mejores resultados de los tres modelos, con un error máximo aún más bajo que los otros dos modelos, y unas métricas de rendimiento incluso superiores a las del modelo de series temporales. Esto indica que la optimización de los hiperparámetros y número de capas del modelo ha funcionado correctamente y ha mejorado la eficiencia del modelo.

Como último apunte de los resultados, se puede observar que ambos modelos que utilizan los vectores de los documentos de texto tienen un error máximo menor que el modelo previo, lo cual podría ser un indicador de que los documentos de texto aportan información relevante para la predicción de los precios en situaciones extremas, donde el conocimiento de los precios anteriores no es suficiente para predecir el precio futuro.

5. Conclusiones y trabajo futuro

5.1. Conclusiones

En este proyecto se han desarrollado tres modelos de predicción de precios de vacuno basados en redes neuronales: un modelo de series temporales, un modelo de series temporales con NLP y un modelo de series temporales con NLP optimizado. Los resultados obtenidos con los modelos muestran que el modelo de series temporales con NLP optimizado obtiene un error máximo menor, pero unas métricas de rendimiento levemente inferiores a las del modelo de series temporales.

Dada la complejidad del problema, de los datos, de los modelos y de las técnicas utilizadas, es imperativo seguir investigando y mejorando los modelos para poder obtener resultados más precisos, dado que este proyecto es una primera aproximación al uso de datos históricos apoyados por documentos de texto para llevar a cabo predicciones de precios.

5.2. Evaluación crítica del grado de logro de los objetivos iniciales

Una vez comentada la complejidad del trabajo, se puede afirmar que los objetivos iniciales del proyecto se han cumplido, ya que se ha desarrollado un modelo de predicción de precios de vacuno basado en redes neuronales que utiliza series temporales y procesamiento de lenguaje natural con un error máximo menor que el modelo de series temporales.

Cabe destacar que es necesario seguir investigando, ya que es posible que la complejidad del modelo de series temporales con NLP optimizado haya mejorado las predicciones, pero no tenga en cuenta los documentos de texto de forma significativa.

5.3. Evaluación crítica de la planificación y metodología

Tras finalizar la planificación y ejecución del proyecto, se puede afirmar que la metodología utilizada ha sido satisfactoria para el desarrollo del proyecto, ya que ha permitido seguir un ciclo de vida de desarrollo de proyectos de minería de datos y ha facilitado la organización y ejecución de las tareas.

La planificación del proyecto ha sido adecuada, ya que se han cumplido los plazos establecidos y se han obtenido los resultados esperados. Sin embargo, se podría mejorar la planificación en futuros proyectos, ya que se han producido retrasos en algunas tareas debido a la complejidad del problema y de los modelos, como la optimización de los hiperparámetros y número de capas del modelo, que ha requerido más tiempo del previsto al tener un factor de aleatoriedad.

5.4. Desafíos de sostenibilidad, diversidad y ético-sociales

En este proyecto se han tenido en cuenta los desafíos de sostenibilidad, diversidad y ético-sociales vinculados al proyecto, ya que se han utilizado datos de la lonja de productos agropecuarios de Binéfar, que son datos públicos y accesibles para cualquier persona interesada en el tema. Además, este proyecto es de interés general para la comunidad científica y para el sector agropecuario, ya que puede ayudar a mejorar la eficiencia en la predicción de precios de productos ganaderos, lo que puede tener un impacto positivo en la economía y en la sostenibilidad del sector.

5.5. Temas para trabajo futuro

En este proyecto se han explorado las posibilidades de utilizar datos de series temporales y procesamiento de lenguaje natural para la predicción de precios de vacuno, pero existen otros enfoques y técnicas que podrían obtener mejores resultados, o que podrían complementar los modelos desarrollados en este proyecto.

Algunos temas para trabajo futuro podrían ser:

- Explorar la posibilidad de utilizar modelos de transformadores.
- Investigar la posibilidad de utilizar modelos híbridos que combinen técnicas de series temporales y procesamiento de lenguaje natural.
- Explorar la posibilidad de utilizar modelos de aprendizaje profundo más tradicionales, como ARIMA o regresiones lineales.
- Investigar sobre técnicas de imputación, interpolación y generación de datos, para intentar solucionar la pérdida de datos en los históricos de precios.

En resumen, este proyecto es una primera aproximación al uso de datos históricos apoyados por documentos de texto para llevar a cabo predicciones de precios, y aunque los resultados obtenidos son prometedores, es necesario seguir investigando y mejorando los modelos para poder obtener resultados más precisos.

6. Glosario

Definición de los términos y acrónimos más relevantes utilizados en este informe.

- *Transfer Learning*: técnica de aprendizaje automático que permite utilizar un modelo preentrenado de forma parcial o total para una tarea específica y adaptarlo a otra tarea.
- *Long Short-Term Memory* (LSTM): tipo de red neuronal recurrente que supera las limitaciones de las RNN al introducir una estructura de memoria a largo plazo.
- *Natural Language Processing* (NLP): técnica de procesamiento de lenguaje natural que permite entender, interpretar y generar lenguaje humano de forma natural.
- *Sentiment Analysis*: técnica de NLP que analiza el tono y la emoción de los textos para determinar si son positivos, negativos o neutrales.
- *Embeddings*: técnica de NLP que convierte las palabras en vectores numéricos para que puedan ser procesados por los modelos de aprendizaje automático.
- *Transformers*: técnica de procesamiento de lenguaje natural diseñada para tareas de traducción automática, generación de texto y análisis de sentimientos.
- *Temporal Fusion Transformers* (TFT): modelo de transformadores diseñado específicamente para series temporales.
- *Informer*: modelo de transformadores diseñado para optimizar la eficiencia en series temporales extensas.
- *Auto-atención*: característica de los transformadores que permite a los modelos conocer las partes más relevantes de la entrada con independencia de la posición de las palabras.

Bibliografía

[1] Ralf C. Staudemeyer and Eric Rothstein Morris. Understanding LSTM - a tutorial into long short-term memory recurrent neural networks. *CoRR*, abs/1909.09586, 2019. URL http://arxiv.org/abs/1909.09586.

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL http://arxiv.org/abs/1706.03762.

[3] Prapanna Mondal, Labani Shit, and Saptarsi Goswami. Study of effectiveness of time series modeling (arima) in forecasting stock prices. *International Journal of Computer Science, Engineering and Applications*, 4, 04 2014. doi: 10.5121/ijcsea.2014.4202.

[4] Sima Siami-Namini, Neda Tavakoli, and Akbar Siami Namin. A Comparison of ARIMA and LSTM in Forecasting Time Series, December 2018. URL https://ieeexplore.ieee.org/document/8614252/.

[5] Fang Liu, Shaobo Guo, Qianwen Xing, Xinye Sha, Ying Chen, Yuhui Jin, Qi Zheng, and Chang Yu. Application of an ann and lstm-based ensemble model for stock market prediction, 2024. URL https://arxiv.org/abs/2410.20253.

[6] Xintao Li, Sibei Liu, Dezhi Yu, Yang Zhang, and Xiaoyu Liu. Predicting 30-day hospital readmission in medicare patients: Insights from an lstm deep learning model, 2024. URL https://arxiv.org/abs/2410.17545.

[7] Mario E. Suaza-Medina, F. Javier Zarazaga-Soria, Jorge Pinilla-Lopez, Francisco J. Lopez-Pellicer, and Javier Lacasta. Effects of data time lag in a decision-making system using machine learning for pork price prediction. *Neural Computing and Applications*, 35(26): 19221–19233, June 2023. ISSN 1433-3058. doi: 10.1007/s00521-023-08730-7. URL http://dx.doi.org/10.1007/s00521-023-08730-7.

[8] Ray Oshikawa, Jing Qian, and William Yang Wang. A survey on natural language processing for fake news detection, 2020. URL https://arxiv.org/abs/1811.00770.

[9] Zenun Kastrati, Fisnik Dalipi, Ali Shariq Imran, Krenare Pireva Nuci, and Mudasir Ahmad Wani. Sentiment analysis of students' feedback with nlp and deep learning: A systematic mapping study. *Applied Sciences*, 11(9), 2021. ISSN 2076-3417. doi: 10.3390/app11093986. URL https://www.mdpi.com/2076-3417/11/9/3986.

[10] Nhan Cach Dang, María N. Moreno-García, and Fernando De la Prieta. Sentiment analysis based on deep learning: A comparative study. *Electronics*, 9(3), 2020. ISSN 2079-9292. doi: 10.3390/electronics9030483. URL https://www.mdpi.com/2079-9292/9/3/483.

[11] Jose Camacho-Collados and Mohammad Taher Pilehvar. Embeddings in natural language processing. In Lucia Specia and Daniel Beck, editors, *Proceedings of the 28th International Conference on Computational Linguistics: Tutorial Abstracts*, pages 10–15, Barcelona, Spain (Online), December 2020. International Committee for Computational Linguistics. doi: 10.18653/v1/2020.coling-tutorials.2. URL https://aclanthology.org/2020.coling-tutorials.2.

[12] Jaydip Sen and Sidra Mehtab. A robust predictive model for stock price prediction using deep learning and natural language processing. July 2021. doi: 10.36227/techrxiv.15023361.v1. URL http://dx.doi.org/10.36227/techrxiv.15023361.v1.

[13] Faisal Khalil and Gordon Pipa. Is Deep-Learning and Natural Language Processing Transcending the Financial Forecasting? Investigation Through Lens of News Analytic Process. *Computational Economics*, 60(1):147–171, June 2022. ISSN 1572-9974. doi: 10.1007/s10614-021-10145-2. URL https://doi.org/10.1007/s10614-021-10145-2.

[14] Om Mane and Saravanakumar kandasamy. Stock market prediction using natural language processing – a survey, 2022. URL https://arxiv.org/abs/2208.13564.

[15] Rahul Rao and Jyothi R. Integrating Time Series Forecasting, NLP, and Financial Analysis for Optimal Investment Strategy: A Case Study on Adani Ports, June 2024. URL https://ieeexplore.ieee.org/document/10605257/.

[16] Pratyush Muthukumar and Jie Zhong. A stochastic time series model for predicting financial trends using nlp, 2021. URL https://arxiv.org/abs/2102.01290.

[17] Kevin Taylor and Jerry Ng. Natural language processing and multimodal stock price prediction, 2024. URL https://arxiv.org/abs/2401.01487.

[18] Narendra Patwardhan, Stefano Marrone, and Carlo Sansone. Transformers in the real world: A survey on nlp applications. *Information*, 14(4), 2023. ISSN 2078-2489. doi: 10.3390/info14040242. URL https://www.mdpi.com/2078-2489/14/4/242.

[19] Pham Canh Huy, Nguyen Quoc Minh, Nguyen Dang Tien, and Tao Thi Quynh Anh. Short-term electricity load forecasting based on temporal fusion transformer model, 2022.

7. Anexos

7.1. Preparación de los datos

Los datos numéricos que contienen los históricos de precios de la Lonja de Binéfar están divididos en dos archivos con formato CSV. Esta división se debe a un cambio en la forma de recopilar los datos. En el primer archivo, "datos1" tiene los datos desde 2015 hasta enero de 2024, mientras que el segundo archivo, "datos2" tiene los datos desde febrero de 2024 hasta diciembre de 2024. Por lo tanto, es necesario juntarlos en un solo archivo para poder trabajar con ellos. Además, los nombres de los productos son diferentes, aunque representan lo mismo, y por lo tanto el nombre de las columnas también es diferente.

Junto con el código fuente, se adjunta un archivo CSV con los datos de los precios de los productos ganaderos unidos con los datos de los comentarios de contexto socio-económico, ya que aunque los datos de los precios son públicos, los comentarios son internos de la lonja y no pueden ser compartidos. Por este motivo, algunas celdas del código fuente están configuradas para no ejecutarse, ya que producirían un error al no encontrar los archivos necesarios.

Tras unir los datos en un solo archivo, procedemos a mostrar las distribuciones de los precios de los productos ganaderos, para poder observar si existen valores extremos o errores en los datos. En la siguiente figura se muestra la distribución de los precios de los productos ganaderos.

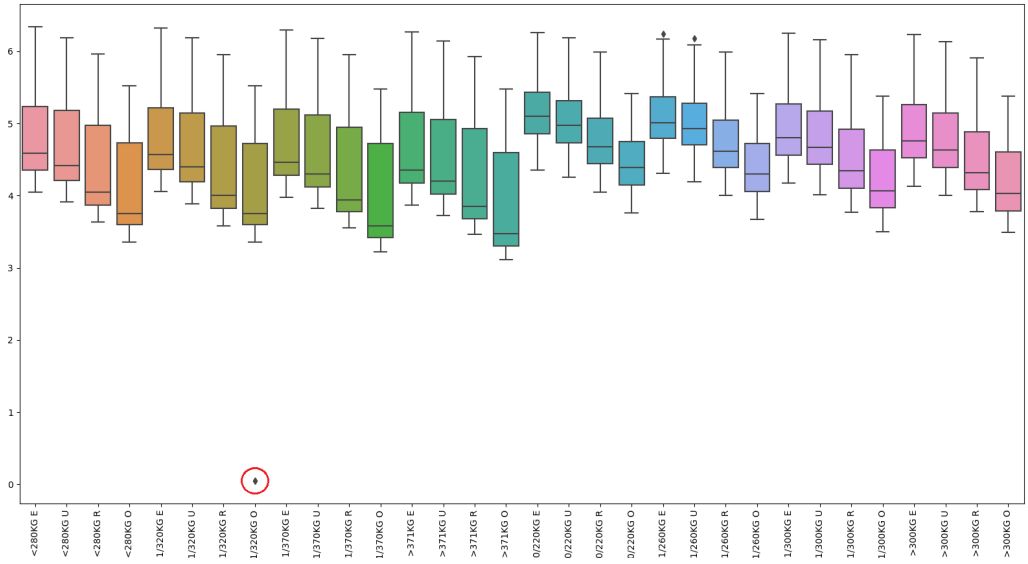

Figura 10: Distribución de los precios de los productos ganaderos.

En la figura anterior se puede observar la presencia de valores extremos en uno de los productos, redondeando con un círculo rojo, que se corresponden con valores de 5 céntimos. Estos valores extremos son un error en la recopilación de los datos, ya que los valores vecinos son de 5 euros y tras una confirmación con los trabajadores de la lonja, se ha determinado que el valor correcto es de 5 euros.

Una vez corregido el error, mostramos las correlaciones entre los preciios de los distintos productos, para observar si existe alguna relación entre ellos. En la figura 11 se muestra la matriz de correlaciones de los precios de los productos ganaderos.

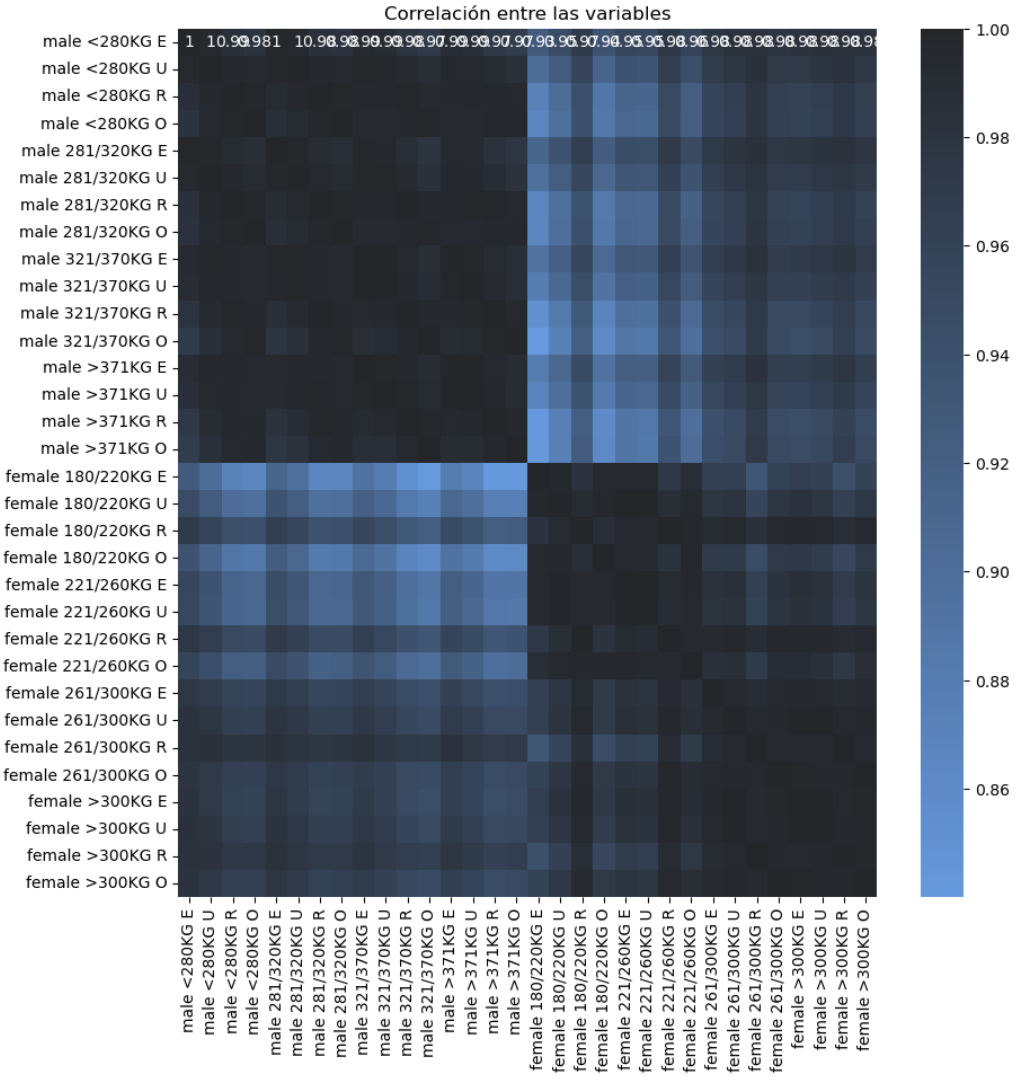

Figura 11: Matriz de correlaciones de los precios de los productos ganaderos.

Los productos tienen una correlación positiva altísima entre ellos, lo cual es de esperar ya que los precios de los productos ganaderos suelen estar influenciados entre sí, dado que son productos sustitutivos o complementarios. Esta correlación es incluso mayor entre los productos del mismo sexo, es decir, entre los precios de machos y hembras.

7.2. Optuna

Optuna es una herramienta de optimización de hiperparámetros que permite encontrar los mejores hiperparámetros y número de capas para un modelo de aprendizaje automático. Esta técnica se basa en definir un espacio de búsqueda y ejecutar múltiples pruebas para encontrar los parámetros que maximizan o minimizan una métrica de rendimiento determinada.

En este proyecto se ha utilizado Optuna para optimizar los hiperparámetros y número de capas del modelo de series temporales con NLP, con el objetivo de mejorar la eficiencia del modelo en la predicción de precios. La métrica de rendimiento utilizada para la optimización ha sido el error durante la simulación de los modelos en un entorno de producción. Esta técnica nos proporciona un número determinado de modelos, con sus respectivos hiperparámetros y número de capas, y nos permite seleccionar el mejor modelo para nuestro problema.

Además, se ha descubierto un problema en el uso de Optuno para la optimización de modelos de series temporales. En estos modelos se suele proporcionar la variable a predecir en un momento temporal previo al momento actual, es decir, la variable a predecir se encuentra en el momento temporal $t + 1$ y los datos de entrada en el momento temporal t.

Esto es muy útil para predecir el futuro, pero puede llegar a provocar sobreajuste en el modelo, ya que el modelo puede comenzar a repetir los datos de entrada en lugar de predecir el futuro. Por ejemplo, si el modelo recibe los datos de entrada en el momento temporal t para predecir $t + 1$, el modelo predeciría el valor t continuamente. Esto produce unas predicciones que parecen desplazadas hacia la derecha, ya que el modelo no es capaz de predecir el futuro, sino que repite el pasado. Aunque este modelo es técnicamente incorrecto, recibe unas métricas buenas, porque los datos son muy poco oscilantes y parecidos entre sí. Por lo tanto, el "mejor" modelo devuelto por Optuna es un modelo que está sobreajustado, en siguiente figura podemos ver las predicciones del modelo sobreajustado.

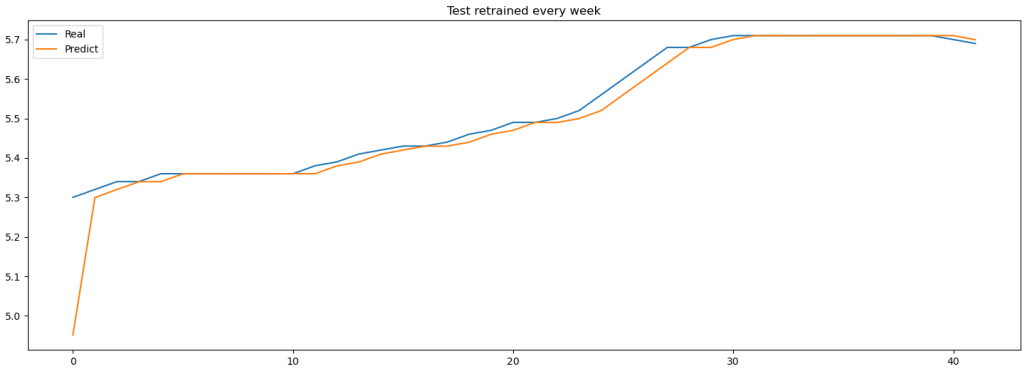

Figura 12: Resultados del mejor modelo de Optuna.

En la figura anterior se puede ver el concepto de desplazamiento hacia la derecha de las predicciones del modelo sobreajustado, pero para confirmarlo completamente se procede a graficar los datos de testing, y las predicciones de este modelo hacia la izquierda, y si coinciden en la gran mayoría de los casos, se puede confirmar que el modelo está sobreajustado.

En la figura anterior se puede confirmar el sobreajuste del modelo, por lo que es necesario comprobar todos los modelos devueltos por Optuna, ya que no podemos confiar en que el mejor modelo, es decir el que menor error tenga, sea el mejor modelo para nuestro problema.

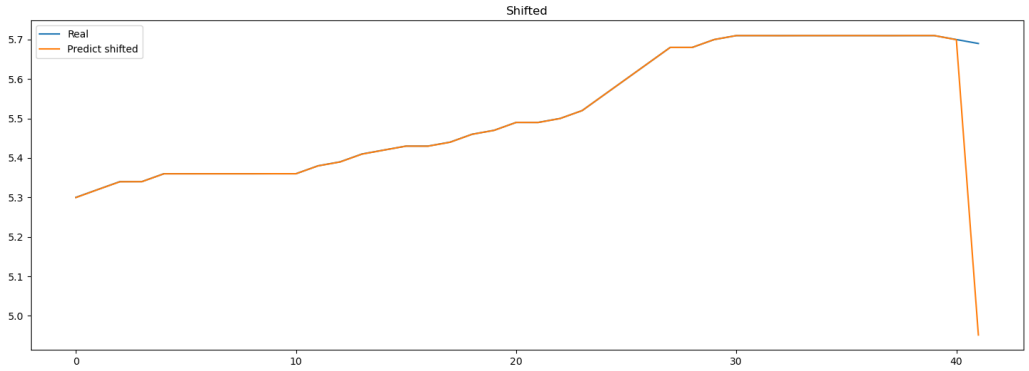

Figura 13: Resultados del mejor modelo de Optuna desplazado.

7.3. Obtención de los embeddings

Para obtener los embeddings de los comentarios de contexto socio-económico, se ha utilizado el modelo de lenguaje Llama3, el cual se ha obtenido a partir de la biblioteca de procesamiento de lenguaje natural Hugging Face. Este modelo ha sido entrenado con un corpus de textos en español y permite obtener los vectores numéricos de los textos de forma sencilla y eficiente.

En el siguiente código se muestra cómo obtener los embeddings de los comentarios en Python utilizando el modelo Llama3 de forma local:

```
from transformers import AutoTokenizer, AutoModel
import torch

class EmbeddingExtractorCPU:
    # Por defecto usamos el modelo 'pesos3.2' (Llama)
    def __init__(self, model_name: str = "pesos3.2"):
        # Cargamos el tokenizer y el modelo
        self.tokenizer = AutoTokenizer.from_pretrained(model_name)
        self.model = AutoModel.from_pretrained(model_name)

    def get_embeddings(self, text: str):
        # Tokenizamos y obtenemos los embeddings
        inputs = self.tokenizer(text, return_tensors="pt")
        with torch.no_grad():
            outputs = self.model(**inputs)

        # Obtenemos los embeddings
        embeddings = outputs.last_hidden_state.mean(dim=1)  # Mean pooling
```

```
        return embeddings

class EmbeddingExtractorGPU():
    # Por defecto usamos el modelo 'pesos3.2' (Llama)
    def __init__(self, model_name: str = "pesos3.2"):
        # Cargamos el tokenizer y el modelo
        self.tokenizer = AutoTokenizer.from_pretrained(model_name)
        self.model = AutoModel.from_pretrained(model_name).to('cuda')

    def get_embeddings(self, text: str):
        # Tokenizamos y obtenemos los embeddings
        inputs = self.tokenizer(text, return_tensors="pt").to('cuda')
        with torch.no_grad():
            outputs = self.model(**inputs)

        # Obtenemos los embeddings
        embeddings = outputs.last_hidden_state.mean(dim=1)  # Mean pooling
        return embeddings
```

En el código anterior, creamos dos clases, `EmbeddingExtractorCPU` que permite obtener los embeddings utilizando la CPU, y `EmbeddingExtractorGPU` que utiliza la GPU. Ambas clases tienen un método `get_embeddings` que recibe un texto y devuelve los embeddings del texto utilizando el modelo Llama3 por defecto, pero si se tiene otro modelo entrenado y localmente localizado, se puede cambiar el nombre del modelo en la inicialización de la clase.

Procedemos a cargar los ficheros y crear un extractor de embeddings para poder obtener los vectores numéricos, ya sea usando la CPU o la GPU, dicha elección debe hacerse de forma manual (cambiando el nombre de la clase o comentando y descomentando las líneas correspondientes).

```
from PyPDF2 import PdfReader
import os

# Directorio de los comentarios procesados
comentarios_procesados_dir = './datos/comentarios_procesados'

# Listar los archivos en el directorio de comentarios procesados
comentarios_procesados_files = os.listdir(
```

```
    comentarios_procesados_dir
    )

# Borramos el fichero de output de los embeddings si existe
if os.path.exists('./datos/embeddings.json'):
    os.remove('./datos/embeddings.json')

# Iterar sobre los archivos de comentarios procesados
embeddings_mean_list = []
# Pruebas locales con CPU
embedding_extractor = EmbeddingExtractorCPU()
# Pruebas en la máquina con GPU
# embedding_extractor = EmbeddingExtractorGPU()
```

Para cada uno de los comentarios cargados, obtenemos el texto dividido en secciones (las secciones están divididas por el conjunto de caracteres "*- "), para cada sección obtenemos los embeddings y calculamos la media de los mismos, con el fin de obtener un único vector numérico que represente el comentario completo. Tras la obtención de los embeddings, se guardan en un fichero JSON asociando la fecha del comentario con el vector numérico obtenido para posteriormente unirlo con los datos de los precios. Se ha decidido hacer la escritura en el fichero para cada comentario, en lugar de hacerlo al final del proceso, para evitar problemas en caso de que el proceso se interrumpa, ya que se podrían perder los datos obtenidos hasta el momento.

```
for file in comentarios_procesados_files:
    # Crear la ruta del archivo
    date = file.split('_')[1].split('.')[0].replace('-', '/')
    file_path = os.path.join(comentarios_procesados_dir, file)
    text = ''
    # Leer el contenido del PDF
    with open(file_path, 'rb') as file_output:
        pdf = PdfReader(file_output)
        for page in pdf.pages:
            text += page.extract_text()

    # Dividir el texto en temas
    paragraphs = text.split('*- ')
    # Eliminamos el primer párrafo, que es introductorio
```

```
    paragraphs = paragraphs[1:]
    embeddings_list = []
    # Obtener los embeddings de cada párrafo
    for paragraph in paragraphs:
        embeddings = embedding_extractor.get_embeddings(paragraph)
        embeddings_list.append(embeddings)

    # Calculamos la media de los embeddings de los párrafos
    mean_embeddings = torch.stack(embeddings_list).mean(dim=0)
    # Creamos el diccionario con la fecha y los embeddings
    embeddings_mean_list.append(
        {
            "date": date,
            "embeddings": mean_embeddings
            })

    with open(f'./datos/embeddings.json', 'a') as embeddings_file:
        # Si el fichero está vacío, añadimos un corchete de apertura
        if os.stat('./datos/embeddings.json').st_size == 0:
            embeddings_file.write('[')
            embeddings_file.write(f'{
                    {
                    "date": "{date}",
                    "embeddings": {mean_embeddings.tolist()}
                    }
                }')
        else:
            embeddings_file.write(f',\n{
                {
                    "date": "{date}",
                    "embeddings": {mean_embeddings.tolist()}
                    }
                }')
# Añadimos un corchete de cierre al final del fichero
with open(f'./datos/embeddings.json', 'a') as embeddings_file:
    embeddings_file.write(']')
```

Como resultado de este proceso, se obtiene un fichero JSON con las fechas de los comentarios y los vectores numéricos asociados a cada comentario, que se utilizarán para entrenar los modelos de predicción de precios.