Universidad de La Habana
Facultad de Matemática y Computación

Predicción de Mercado Utilizando Información de Noticias

Autor:
Alex Sánchez Saez, Carlos Manuel Gonzáles Peña, Jorge Alberto Aspiolea

Tutores:

Trabajo de Diploma
presentado en opción al título de
Licenciado en (Ciencia de la Computación)

Fecha

07-07-2024

Índice general

Introducción

Con este trabajo pretendemos contestar a la pregunta de : £Es posible predecir el mercado con eficacia. Este tama nos es de mucho interés y nos causa una gran curiosidad saber que requerimientos tendría en caso de que la respuesta a esta pregunta fuese afirmativa. Como es un marco demasiado amplio ya que el mercado es un componente complejo del sistema en que nos situamos y es dependiente de diversos factores de naturaleza distinta, como pueden ser las catástrofes, la influencia de personalidades o redes sociales, eventos políticos, pandemias globales etc. Es por esto que quisimos simplificar más nuestra pregunta, y la redujimos a : £Es posible predecir el comportamiento del mercado dada la información de noticias extraídas del mundo real?. Debido a la aparente aleatoriedad tanto del comportamiento del mercado como de las noticias, así como la dificultad de diseñar un algoritmo que se encargue de esta tarea, decidimos recurrir al aprendizaje automático (Machine Learning). Esta tecnología permite que una computadora aprenda posibles patrones y pueda predecir el comportamiento del mercado de manera aproximada.

Nuestra investigación abarcó varios temas del ámbito del aprendizaje automático, desde el estudio y análisis de varios modelos, hasta la recolección, tratamiento y preparación de los datos. Fue necesaria una profunda revisión de trabajos relacionados con esta temática, observando que muy pocos abordaban la perspectiva de utilizar noticias para dicha tarea.

Este estudio se enmarca en un ámbito mayor que es el análisis y predicción de series temporales. Las series temporales son datos secuenciales recogidos a intervalos regulares de tiempo y son fundamentales para el análisis predictivo en diversos campos, incluyendo la economía y las finanzas. La relación entre la serie de datos del mercado y las noticias es crucial, ya que las noticias pueden influir en las decisiones de los inversores y, por ende, en el comportamiento del mercado. Utilizando técnicas de "Forecasting with Covariance", podemos integrar estas variables adicionales (las noticias) en los modelos predictivos para mejorar la precisión de las predicciones.

Capítulo 1

Estado del Arte

En esta sección se revisa la literatura existente relacionada con la predicción del comportamiento del mercado financiero utilizando técnicas de aprendizaje automático. Esta revisión proporcionará el contexto necesario y destacará las contribuciones previas en este campo. Así como el análisis de resultados de cada una de las propuestas encontradas

1.1. Métodos Tradicionales de Predicción de Mercados

Se identificaron dos enfoques principales para el análisis y la predicción del comportamiento del mercado: el análisis técnico y el análisis fundamental. Estos métodos se han consolidado como pilares fundamentales en la disciplina financiera debido a sus enfoques distintos pero complementarios para interpretar y prever las tendencias del mercado.

1. **Análisis Técnico**

   El análisis técnico se centra en la identificación de patrones en los datos históricos del mercado, tales como precios, volúmenes de transacciones y otras variables derivadas. Este enfoque se basa en la premisa de que los movimientos de los precios no son aleatorios, sino que siguen tendencias y patrones que pueden ser identificados y utilizados para predecir movimientos futuros. Los analistas técnicos emplean una variedad de herramientas y técnicas, incluyendo gráficos, indicadores técnicos y modelos matemáticos, para detectar señales de compra y venta en el mercado. Este enfoque es particularmente útil en el corto plazo, donde los patrones históricos pueden repetirse con mayor frecuencia.

2. **Análisis Fundamental**

Por otro lado, el análisis fundamental se basa en la evaluación del estado financiero de una empresa y su entorno económico para determinar el valor intrínseco de sus acciones. Este enfoque examina factores como los estados financieros, la gestión de la empresa, las condiciones del sector y las tendencias macroeconómicas. Los analistas fundamentales creen que el valor de una acción está determinado por el rendimiento económico y financiero de la empresa, y que el mercado eventualmente reflejará este valor en el precio de la acción. Este tipo de análisis es esencial para inversiones a largo plazo, ya que proporciona una visión más profunda de los factores subyacentes que afectan el rendimiento de una empresa.

1.1.1. SVM (Support vector Machine)

Para regresión en un enfoque multi etapa.

1.1.2. Computational Efficient Functional Link Artificial Neural Network (CELANN)

Un modelo de red neuronal con una única capa oculta[1] que permite obtener un mejor performance .

1.1.3. Long Short Term Memory (LSTM)

Un tipo de red neuronal recurrente (RNN) Optimizada para corregir el problema de desvanecimiento de gradientes en estas (RNN) lo que permite utilizar modelos mas grandes .

1.1.4. Modelo de Medias Móviles (Simple Moving Average) (SMA)

Para suavizar las fluctuaciones de precios a corto plazo y revelar tendencias a largo plazo. Calcula el promedio de los precios en un periodo de tiempo determinado, eliminando el ruido"de las variaciones diarias.

1.1.5. Media Móvil Integrada Auto-regresiva (Auto-Regresive Integrated Moving Average)(ARIMA)

Combina tres componentes: autoregresión (AR), que usa la relación entre los valores pasados de la serie; diferenciación (I), que hace la serie estacionaria eliminando tendencias; y media móvil (MA), que captura dependencias residuales entre errores.

ARIMA[2] es útil para predecir futuros valores basándose en patrones históricos, como tendencias y ciclos, comúnmente en precios de acciones, tasas de cambio y otros datos financieros

1.1.6. Método Holt-Winters

Analiza series temporales que presentan patrones de tendencia y estacionalidad. Este modelo permite hacer pronósticos a corto y mediano plazo ajustando las fluctuaciones en los datos mediante tres componentes: nivel, tendencia y estacionalidad

1.1.7. LSTM Convolucionales

Para capturar tanto patrones espaciales como temporales en los datos financieros. Combina la capacidad de las redes convolucionales (CNN) para extraer características locales y la habilidad de las LSTM[3] para manejar dependencias a largo plazo en series temporales

1.2. Técnicas de ensembling observadas

1.2.1. XBoosting, usando Desission Tree y Random Forest

Funciona mediante el ensamble de múltiples árboles de decisión en secuencia, donde cada árbol nuevo corrige los errores del anterior.

1.2.2. Gradient Boosting

Combina múltiples árboles de decisión débiles en un modelo más robusto. Funciona ajustando errores de predicciones anteriores en pasos sucesivos.

1.2.3. ADA Boosting

En la predicción de valores de mercado, se utiliza para combinar varios modelos débiles (predicciones que tienen un rendimiento apenas mejor que el azar) en un modelo fuerte. Esto se logra al ajustar los pesos de los datos mal predichos, enfocándose en mejorar las predicciones para esos casos, lo que ayuda a reducir errores y mejorar la precisión global en la estimación de valores de mercado.

1.3. Forecasting with Covariance

El "Forecasting with Covariance"[4][5] implica la incorporación de múltiples series temporales relacionadas, permitiendo que las variaciones en una serie (como las noticias) informen las predicciones en otra serie (como los precios del mercado). Este enfoque reconoce que los mercados no operan de manera aislada, sino que están influenciados por una amplia gama de factores externos. Al incluir estas variables adicionales, nuestros modelos pueden captar mejor las dinámicas complejas que afectan los movimientos del mercado.

1.4. Aplicaciones de Aprendizaje Automático en Finanzas

A pesar de ser capaces de capturar no linealidades y comportamientos más complejos en los datos de series temporales financieras los algoritmos de Aprendizaje de Máquina no consiguen capturar toda la complejidad del comportamiento del mercado, debido a la gran cantidad de elementos de la realidad que interfieren en el mercado y la esencia casi aleatoria de dichos datos. Tienen en su mayoría el defecto de ser computacionalmente intensos y de requerir grandes volúmenes de datos, lo cuál no representa un problema, debido a que existen muchos datos financieros con disponibilidad. Modelos más potentes como las redes neuronales carecen de explicabilidad, por lo que no es posible extraer los patrones que detectan dichos modelos . Otros trabajos basados en modelos como random forest y árboles de decision , tienen mayor explicabiliad , pero presentan un rendimiento menor incluso que los modelos estadísticos clásicos. Modelos como las medias móviles , ARIMA y sus variantes son los más utilizados en la actualidad ya que requieren de tareas computacionalmente menos demandantes y aportan resultados comparables a los ofrecidos por los modelos de aprendizaje.

1.5. Uso de Noticias y Datos No Estructurados

En general el uso de noticias para relacionar los movimientos del mercado con los acontecimientos del mundo real es escaso en la bibliografía consultada, pero hay enfoques similares utilizando datos sobre política y relacionando las probabilidades de subir o bajar valores de acciones en dependencia de eventos ocurridos en el mundo real

1.6. Comparación y Análisis Crítico

En general para poder predecir efectivamente la esencia dinámica del comportamiento del mercado y su relación a eventos externos como eventos políticos, eventos sociales como por ejemplo el aislamiento social provocado por la pandemia , catástrofes naturales, influencias sociales etc, serían necesaios modelos muy potentes que sepan capturar la esencia de estos eventos , lo cuál lleva a la necesidad de una cantidad de datos muy grande y variables, necesitando de diferentes fuentes y datos de naturaleza muy distinta, lo cuál dificulta el tratamiento y modelación de estos datos para su uso. Por otra parte modelos complejos capaces de establecer relaciones entre estos datos requirirían de una amplia cantidad de recursos para tener un correcto funcionamiento, esto supone un freno en el desarrollo de herramientas para llevar a cabo predicciones efectivas en el ámbito financiero. Por su parte los modelos estadísticos siguen dando resultados cuanto menos comparables con los modelos de aprendizaje de máquina aunque estos cuentan con la limitación de no poder capturar en su mayoría no linealidades presentes en el histórico de los datos ni poder establecer una relación entre estos y datos de naturaleza diferente como las noticias u otros eventos.

1.7. Identificación de Huecos en la Literatura

Aunque se ha avanzado en el uso de aprendizaje automático para la predicción del mercado, la literatura muestra una falta de estudios que integren de manera efectiva noticias y datos no estructurados en los modelos predictivos.

1.8. Prophet

Una de las principales contribuciones de Prophet[6] es su capacidad para descomponer series temporales en componentes de tendencia, estacionalidad y eventos especiales (como días festivos) de manera intuitiva. A diferencia de modelos como ARIMA[2], que requieren que las series sean estacionarias, Prophet permite trabajar con datos no estacionarios, lo que amplía su aplicabilidad a una variedad de contextos del mundo real.

El modelo es aditivo, lo que significa que la suma de sus componentes (tendencia, estacionalidad y efectos) da como resultado la predicción final. Esta estructura permite ajustar fácilmente la flexibilidad del componente de tendencia mediante parámetros, lo que ayuda a capturar cambios abruptos en la serie temporal.

En resumen, Prophet ha revolucionado el campo del pronóstico de series temporales al ofrecer un enfoque accesible, flexible y eficaz que combina la simplicidad con la potencia analítica necesaria para abordar problemas complejos en diversos dominios.

Su capacidad para manejar datos no estacionarios y proporcionar interpretaciones claras lo posiciona como una herramienta valiosa en el arsenal del análisis predictivo.

1.9. Revisión Bibliográfica

Paper	**Año**	**Modelo**	**Resultados**	**Dataset**	**Basado en noticias**
Optimizing LSTM for time series prediction in Indian stock market [7]	2020	LSTM	Mejor Seleccion de Hiperparametros	Indian Stock Market	No
A Review of ARIMA vs. Machine Learning Approaches for Time Series Forecasting in Data Driven Networks [2]	2023	LSTM (Redes Neuronales Recurrentes) ARIMA(Predictor Estadistico)	Se descubrio que para predicciones de series de tiempo con relaciones no lineales , arima se comporta mejor , mientras que LSTM capta mejor las no linealidades		No

Stock price index movement classification using a CEFLANN with extreme learning machine [1]	2015	CEFLAN(Red neuronal de una sola Capa)	Modelo muy eficiente que logro predecir movimientos de mercado en alza o baja		No
Redes Neuronales Transformers aplicada a la predicción de activos financieros (Tesis de maestria de la facultad de Ingeniería de la Universidad Austral , no encontramos el paper pero si la exposicion en youtube)	2021	Transformers	Relacionan precios de mercado con capacidad de generar ganancias para las acciones obteniendo un volumen de falsos positivo de 0.73	Datos historicos de acciones en EEUU	No

Time Series Data Analysis for Stock Market Prediction [8]	2020	ARIMA HOTL-Winters, SMA	En largoplazo los que obtuveron errores menores fueron SMA y Hotl-Winters , mientras que ARIMA tuvo un mejor performance en corto plazo	10 años de Bombay Stock Exchange data	No
Deep Learning for Stock Market Prediction [9]	2020	Random Forest (Con ensembles , Boosting, Gradient Boosting) LSTM	Los valores de error generalmente aumentan cuando se crean modelos de predicción para un mayor número de días por adelantado	10 Años de mercado de acciones iraní	No

Explainable stock prices prediction from financial news articles using sentiment analysis [3]	2021	LSTM Convolucionales	Una arquitectura basada en en analisis de sentimientos de las noticias para cuantificar cuanto afectan al precio de los activos del mercado	Yahoo Market	Si
Geometric Case Based Reasoning for Stock Market Prediction [10]	2020	KVN	Propone la utilización de un método de medición de similitud de forma para determinar los vecinos más cercanos		No

Covariance matrix forecasting using support vector regression [4]	2021	SVM	Permite predecir la matriz de covarianza utilizando Matrices de covarianzas de segmentos o rangos de tiempo en las variables		No
Bitcoin, gold and the dollar A GARCH volatility analysis [11]	2016	Dynamic Conditional Correlation (DCC) GARCH	Hace una comparativa entre el bitcoin el oro y el dolar utilizando tecnicas de análisis de covarianza y predicción de series de tiempo con covarianza	Blockchain de bitcoin	No

EFSR: Ensemble Forecast Sensitivity to Observation Error Covariance [5]	2017	Lorenz 96	Aplica un modelo matemático Lorenz 96 que es utilizado para predecir entornos caótico y diversos como el clima		No
A convolutional neural network based approach to financial time series prediction [12]	2022	ARIMA Cart Random Forest	Mezcla modelos análiticos y random forest para obtener resultados más explicables		No
Forecasting at scale [6]	2017	Prophet	Es capaz de generar predicciones precisas y útiles, aunque su rendimiento puede variar según la calidad de los datos.		No, pero incluye un componente para modelar eventos especiales

Capítulo 2

Propuestas de solución

El objetivo central de nuestra investigación es encontrar la manera de relacionar los precios de las acciones en el mercado con factores externos, específicamente a través del análisis de noticias. Nuestra premisa inicial se centra en hallar una representación adecuada para las noticias que nos permita extraer su contenido y determinar su relación con la variación en el precio del activo en cuestión.

En este contexto, nos enfocamos en acciones de empresas tecnológicas, particularmente en Apple. El propósito es desarrollar un modelo que pueda identificar y cuantificar el impacto de las noticias sobre las fluctuaciones del precio de las acciones de Apple. Para lograr esto, empleamos técnicas avanzadas de procesamiento del lenguaje natural (NLP) y análisis de sentimientos, con el fin de extraer información relevante de las noticias y correlacionarla con los movimientos en el mercado bursátil.

2.1. Red Adversarial (GAN) para predecir movimientos del mercado

El proceso comenzaría con la recopilación y organización cronológica de las noticias, asociándolas con las fechas correspondientes y etiquetándolas según su impacto en el precio de la acción en ese momento del tiempo, es decir, si contribuyeron a una subida o bajada del precio del activo.

Con esta información, procederíamos a entrenar un predictor. Este predictor sería capaz de tomar como entrada los embeddings de las noticias y predecir el efecto que estas tendrán sobre el precio de las acciones.

Posteriormente, utilizando datos del comportamiento histórico del precio de las acciones, entrenaríamos un discriminador. Este discriminador tendría la tarea de evaluar si el comportamiento descrito por las noticias es congruente con los patrones observados en el comportamiento histórico del precio de las acciones. En esencia,

el discriminador verificaría la validez y precisión del predictor, asegurando que las predicciones se alineen con las tendencias históricas observadas.

Una vez entrenado el modelo este serpía capaz de relacionar efectivamente el contenido de las noticias con los patrones de comportamiento del precio de un activo en el mercado , es decir el movimiento de este.

No llevamos a cabo esta alternativa debido a la dificultad para encontrar un dataset de noticias donde cada una tuviera su efecto en el mercado y por cuestiones del tiempo disponible para el desarrollo de esta investigación

2.2. Clusterización del espacio de las noticias

Otra alternativa que tuvimos en cuenta fue realizar una clusterización del espacio de las noticias (conjunto de noticias en nuestro dataset) para así agruparlas por características comunes, luego de esto clasificar los centroides encontrados por su aporte, positivo, negativo o neutro respecto al activo que queremos predecir, así para clasificar una noticia solo tendríamos que ver a que centroide está mas cercana (a que cluster pertenece) y clasificar según dicho centroide; es decir, cada cluster tendría una clasificación de acción positiva, neutra o negativa y asumiríamos que todas las noticias que están presentes en ese cluster también, la representación de las noticias podría ser tanto en embeddings de las mismas como modelos más sencillos como bag of words o tf-idf si se requiriera de un análisis del corpus de noticias. Una vez hecho esto pudiéramos construir un dataset sintético donde colocáramos el movimiento del mercado en cada fecha y la acción de su noticia correspondiente a la fecha.Con esto entrenaríamos un regresor usando alguno de los modelos disponibles como una regresión lineal o una red neuronal. Esta idea fue descartada ya que introduce el cesgo de que todas las noticias de un cluster tienen el mismo efecto en el activo, aunque sean noticias semánticamente iguales, no quiere decir que tengan el mismo efecto en el mercado.

2.3. Red Neuronal Convolucional

Otra de las ideas que tuvimos presente fue la de utilizar una red neuronal convolucional que captara estructuras más complejas de la serie de tiempo representada por las noticias y el comportamiento del activo a predecir. Esto inspirado en la literatura citada requeriría de construir un dataset que involucrara tanto al estado de la moneda en cada fecha, como a la noticia presente en esa fecha, permitiendo al modelo capturar patrones y estructuras subyacentes en ventanas o márgenes de tiempo. Para llevar a cabo esta solución chocamos con la problemática de encontrar una forma de modelar este escenario ya que por ejemplo las noticias pueden estar afectando el

valor del activo un determinado tiempo, el cuál es muy difícil de predecir. Además de la dificultad que implica determinar el tamano de las ventanas de tiempo y como estructurar el dataset; por esto, esta idea fue descartada.

2.4. Árboles de Desición y Random Forest

Otra de las ideas latentes en la literatura consultada que nos llamó bastante la atención fue la idea de utilizar random forest a traves de ensembles como Gradient Boosting y Baggin que fueron analizados en la bibliografía encontrada. Estos enfoques son potentes pues los árboles de desición son capaces de capturar carácterísticas específicas de cada conjunto de datos y al combinar el conocimiento obtenido por varios de estos modelos se pueden capturar comportamientos más complejos en la estructuración de la relación Noticia-Mercado. Esto aportaría una mayor explicabilidad del modelo propuesto y aportaría en el caso de Bagging la capacidad de entrenar estos modelos por separado y después utilizar el conocimiento de estos en conjunto. Cada uno de los random forest se entrenaría con un fragmento del dataset donde capturaría diferentes características tanto de las noticias como del comportamiento del mercado, ayudando así a relacionar estas variables y permitiendo emerger patrones y características más complejas. Este enfoque tiene un problema con la tendencia al sobreajuste que haría que el modelo aprendiera muy bien de los datos de entrenamiento pero no sea capaz de generalizar.

2.5. Uso de Redes neuronales Recurrentes

Otro acercamiento a la resolución del problema planteado es el uso de redes neuronales recurrentes, las cuál permiten procesar series de tiempo con mayor eficacia ya que le permite al modelo no solo aprender de los datos de entrada sino que puede aprender de datos anteriormente aprendidos en estados anteriores. Este enfoque tiene le problema del desvanecimiento de gradiente, lo cuál limita la capacidad de estos modelos para aprender de series temporales de largo plazo. Para abordar este problema, se han desarrollado arquitecturas más avanzadas como las redes LSTM (Long Short-Term Memory) y GRU (Gated Recurrent Unit), que están diseñadas específicamente para manejar mejor las dependencias a largo plazo, estas agregan una puerta de olvido que permite a la red neuronal utilizar solo la información necesaria y olvidar la innecesaria

2.6. Redes neuronales con Arquitectura transformer

Los transformers, aunque efectivos en muchas tareas de procesamiento de secuencias, no son ideales para la predicción de series de tiempo financieras debido a varias limitaciones. En primer lugar, su complejidad dificulta la interpretabilidad, esencial en el ámbito financiero. Además, tienen una alta propensión al sobreajuste debido a su gran cantidad de parámetros, lo que puede resultar en un rendimiento deficiente con datos no vistos. Requieren grandes volúmenes de datos, lo cual es problemático en el contexto financiero donde los datos pueden ser limitados y obsoletos rápidamente. Su alta demanda computacional es otro inconveniente, especialmente para aplicaciones en tiempo real. Aunque buenos para capturar dependencias a largo plazo, los transformers pueden tener dificultades con los patrones cíclicos y estacionales típicos de las series financieras, y son sensibles al ruido característico de estos datos. Por último, la incorporación de conocimiento específico del dominio financiero es más compleja en transformers que en modelos tradicionales de series temporales.

2.7. Análisis de Sentimientos de las Noticias, teniendo en cuenta su relevancia

Haciendo un Análisis de sentimiento por las noticias (con un modelo pre entrenado) podríamos determinar si esta habla bien o mal de la moneda en cuestión, esto en principio debería verse reflejado en un alza o decaimiento del valor de la moneda. Este coeficiente de afectación del contenido de la noticia respecto a la moneda se vería limitado o potenciado por la relevancia de la noticia, siguiendo alguna métrica como la cantidad de lectores que tenga la cadena que la notifica , métricas de audiencia y alacance de las mismas, etc. Con esto podriamos introducir este nivel de afectación de la noticia al valor de la moneda en la serie de tiempo del movimiento de precios de la moneda o activo del mercado a analizar. Entonces con un modelo LSTM que tenga como entrada todos estos datos podríamos predecir efectivamente el valor de la moneda. Este enfoque es muy convincente y se encontró en la litratura trabajos que abordan el problema de forma muy similar. Pero haciendo esto agregamos el cesgo de afirmar que una noticia que hable bien sobre una moneda o activo haga que este aumente o decremente su valor de mercado. También se incluye la problemática de encontrar las métricas reales de alcance de estas noticias, podría utilizarse una simulación para obtener estos parámetros de forma aproximada, pero esto agregaría el error cometido por la simulación al no tratarse de datos reales, lo cuál lo hace poco fiable. En trabajos futuros se podría utilizar nuestro anterior trabajo simulador de redes sociales, para ser utilizado en la generación de dichas métricas.

Capítulo 3

Detalles de Implementación y Experimentos

3.1. Análisis de Dataset de valores del mercado

3.1.1. Obtención y análsis de los datos

Para obtener datos financieros con información relevante como el precio de apertura (el precio de las primeras operaciones realizadas en el día), precio de cierre (precio de las últimas operaciones realizadas en el día) y los precios más altos y más bajos que parecen a lo largo del día, conseguimos el dataset que se muestra en la tabla 3.1. El cual lo obtuvimos de Kaggle Yahoo Stock Prediction. Haciendo un análisis de estos datos encontramos que tiene 7 columnas (Open, High, Low, Close, Volume, Dividends, Stock Splits) y 6165 filas

Tabla 3.1: Data Set escogido(primeros 10 días)

Date	**Open**	**High**	**Low**	**Close**	**Volume**	**Div.**	**S. Splits**
2000-01-03	0.791669	0.849227	0.767607	0.844981	535796800	0.000	0.000
2000-01-04	0.817145	0.835073	0.763833	0.773740	512377600	0.000	0.000
2000-01-05	0.783176	0.834601	0.777515	0.785063	778321600	0.000	0.000
2000-01-06	0.801105	0.807709	0.717125	0.717125	767972800	0.000	0.000
2000-01-07	0.728448	0.762417	0.720900	0.751094	460734400	0.000	0.000
2000-01-10	0.769966	0.771853	0.715238	0.737884	505064000	0.000	0.000
2000-01-11	0.724202	0.750151	0.683156	0.700141	441548800	0.000	0.000
2000-01-12	0.717125	0.720900	0.652961	0.658151	976068800	0.000	0.000
2000-01-13	0.713232	0.745432	0.698253	0.730335	1032684800	0.000	0.000
2000-01-14	0.754868	0.771853	0.750151	0.758171	390376000	0.000	0.000

3.1.2. Análisis de los valores del dataset

Haciendo un análisis sobre los datos presentes en el dataset llegamos a que no hay datos cuyo valor sea nulo. También encontramos que todos los datos son numéricos como se observa en la tabla 3.2. Para evitar insertar ruido en los datos eliminamos los features Dividends y Stock Splits, pues estos no aportan información relevante para la predicción y solo agregan ruido a los datos.

Tabla 3.2: Análisis de la nulidad de los datos

	Column	**Non-Null Count**	**Dtype**
0	Open	6165 non-null	float64
1	High	6165 non-null	float64
2	Low	6165 non-null	float64
3	Close	6165 non-null	float64
4	Volume	6165 non-null	int64
5	Dividends	6165 non-null	float64
6	Stock Splits	6165 non-null	float64

3.1.3. Análisis de la distribución de los datos

Hallando la asimetría o sesgo de los datos (Skewness) para detectar anomalías en la distribución de los datos [3.1, 3.2, 3.3, 3.4, 3.5]

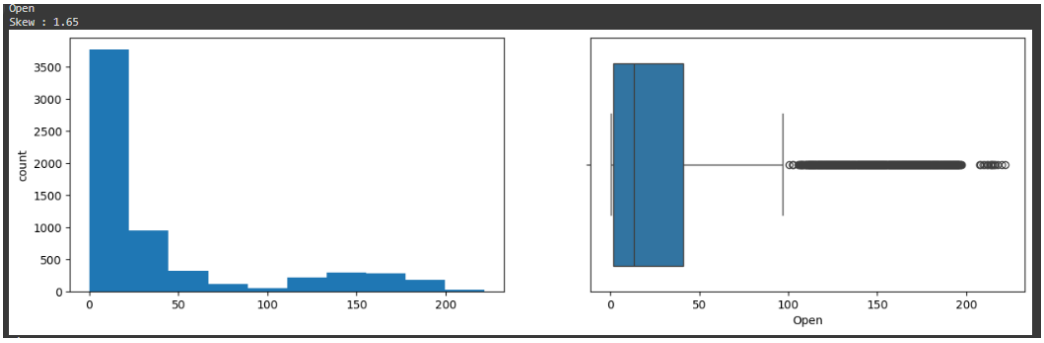

Figura 3.1: Distribución de la característica Open

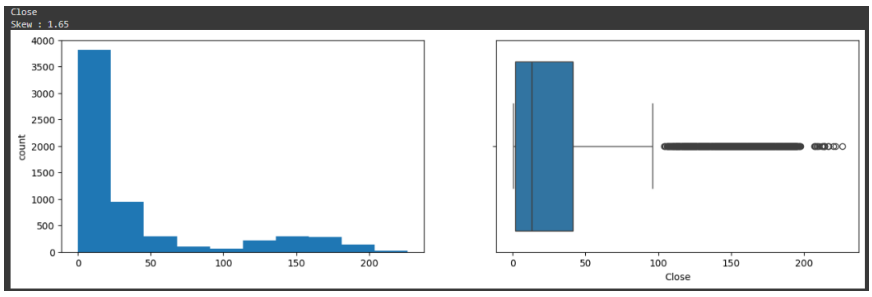

Figura 3.2: Distribución de la característica Closed

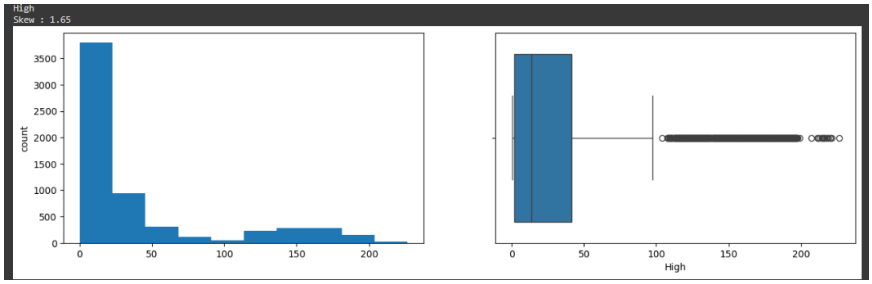

Figura 3.3: Distribución de la característica High

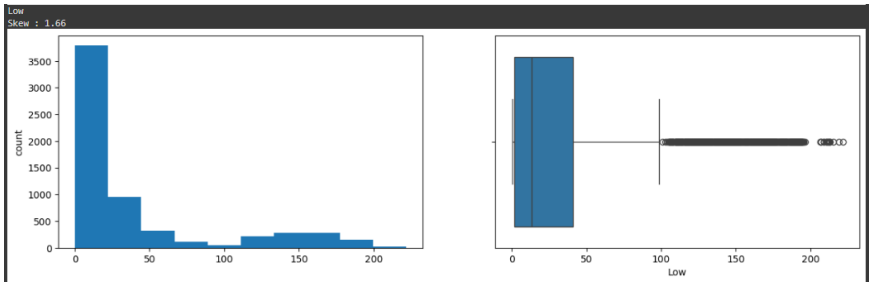

Figura 3.4: Distribución de la característica Low

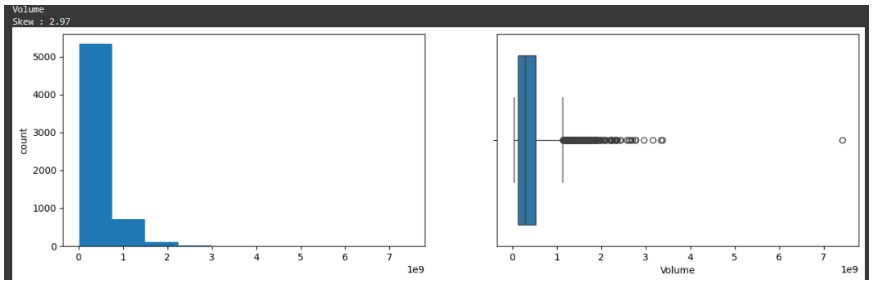

Figura 3.5: Distribución de la característica Volume

Podemos observar en los datos un cesgo positivo; los valores en su mayoría son bajos y hay una cola larga que se extiende a la derecha. Lo cuál indica que hay un número significativo de valores altos en la distribución, pero con menor frecuencia. El gráfico de caja y bigotes (boxplot) a la derecha muestra que la mediana está cerca del primer cuartil, con una cola larga extendiéndose hacia valores más altos. En la mayoría de casos el cesgo fluctúa alrededor de 1.65 lo cuál indica una asimetría hacia valores altos. Esto sugiere que aunque hay una gran cantidad de valores bajos, hay valores muy altos que están afectando la media y extendiendo la cola de distribución, por lo que es necesario normalizar los datos.

3.1.4. Análisis bivariable de los datos

Con este Análisis pretendemos encontrar relaciones entre las diferentes carácteristicas presentes en los datos para encontrar así relaciones, ruidos, etc.

Utilizando un grafico de pares para ver relaciones entre las diferentes variables

En general la relación entre las variables es lineal , notando no linealidades sobretodo en variables relacionadas con el volumen de las operaciones

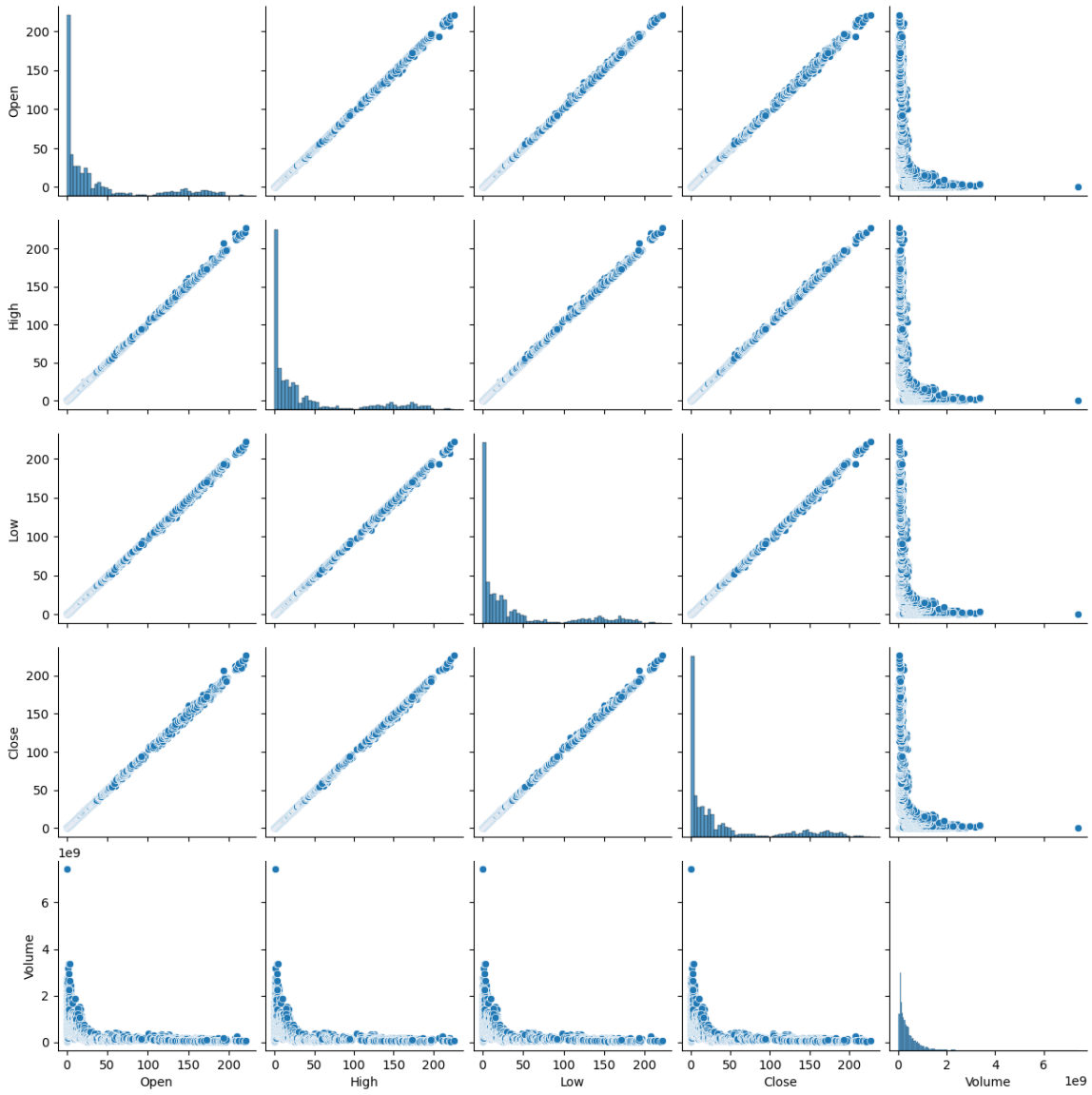

Figura 3.6: Gráfico de pares con los datos

Analizando la figura 3.6 de pares llegamos a la conclusión de la mayoría de datos tienen relaciones lineales entre sí, salvo por el volumen el cuál presenta relaciones no lineales respecto al resto de variables.

Análisis de correlación entre los datos utilizando un mapa de calor

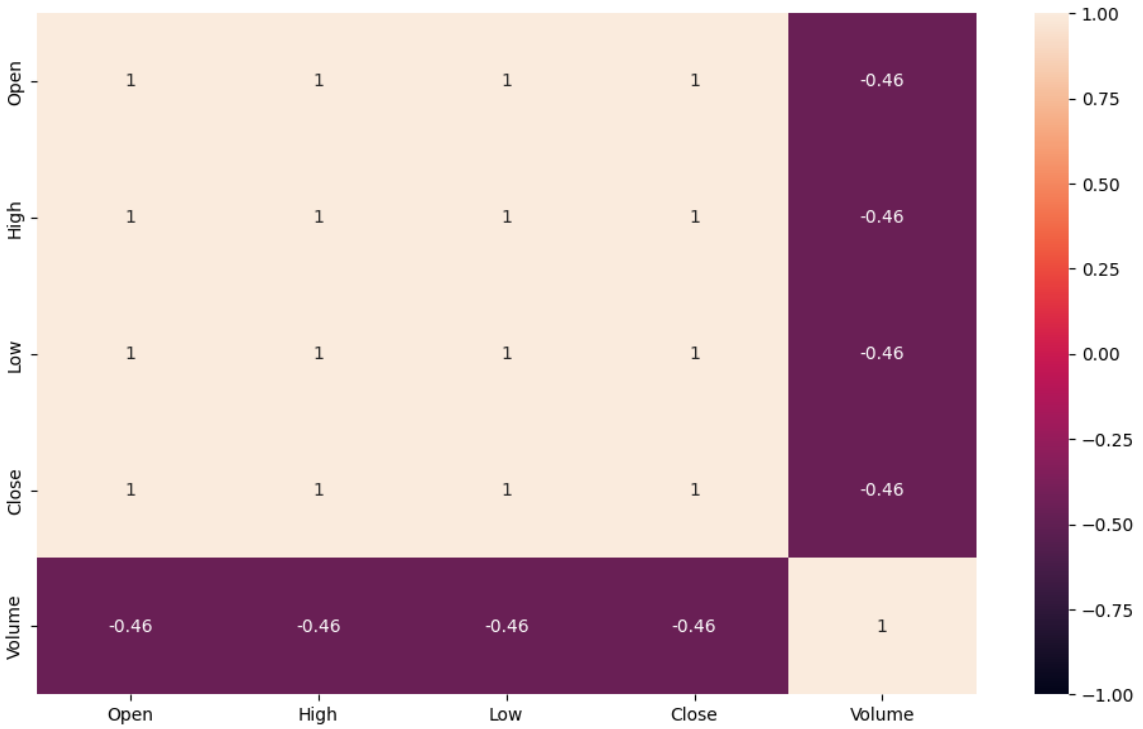

Figura 3.7: Mapa de calor de las variables del dataset

Es notable la correlación negativa del volumen al resto de características capturadas en el dataset, mientras que el resto mantienen una correlación lineal entre sí como se observa en la figura 3.7. Esto corrobora la información sacada de la gráfica de pares analizada anteriormente

3.1.5. Detección de Outlayers

Utilizando Rango Interquartílico (IQR)

Figura 3.8: Detección de Outliers en Open

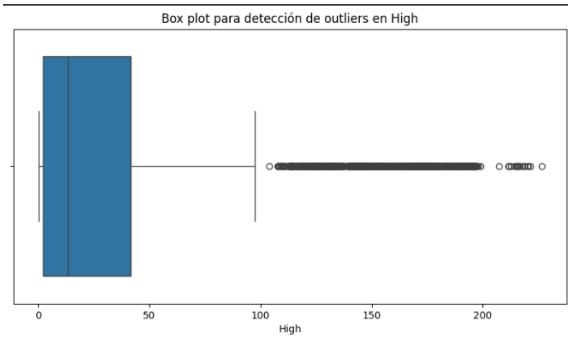

Figura 3.9: Detección de Outliers en High

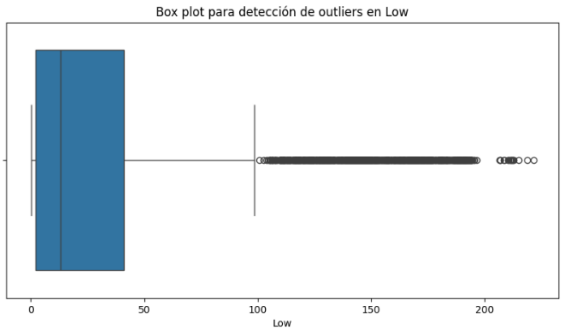

Figura 3.10: Detección de Outliers en Low

Figura 3.11: Detección de Outliers en Close

Analizando las gráficas [3.8, 3.9, 3.10, 3.11[] de bigotes notamos una gran cantidad de valores moviéndose a la derecha de cada dato, alejándose de la media, lo que indica una gran cantidad de valores desproporcionados al resto, mucho mayores que el resto , este comportamiento es común dada la naturaleza de estos datos financieros la cuál tiende a ser muy volátil y dinámica en el tiempo. Las variables Open, High, Low y Close parecen tener distribuciones similares, todas con un número significativo de outliers en el extremo superior. Esto sugiere que las variaciones extremas en los precios pueden estar correlacionadas entre estas diferentes medidas.

3.1.6. Limpieza y normalización de los datos

Para Mantener los datos normalizados exploramos varias alternativas de métodos de normalización.

MinMax Scaling

Escala los datos para mantenerlos en un rango específico, en este caso (0,1) como se observa en la tabla 3.3. Tiene el problema de ser suceptible a los outliers.

Tabla 3.3: Resultados de la normalización con MinMax

	Open	**High**	**Low**	**Close**	**Volume**
0	0.002689	0.002873	0.002599	0.002861	0.069178
1	0.002804	0.002811	0.002582	0.002546	0.066012
2	0.002651	0.002809	0.002644	0.002596	0.101962
3	0.002732	0.002690	0.002371	0.002295	0.100563
4	0.002404	0.002490	0.002388	0.002445	0.059031

Standard Scaler

Transforma los datos para que tengan una media de 0 y una desviación estándar de 1 como se observa en la tabla 3.4. Menos sensible a outliers que Min-Max Scaling

Tabla 3.4: Resultado de la normalización usando Standard Scaler

	Open	**High**	**Low**	**Close**	**Volume**
0	-0.673803	-0.672878	-0.673852	-0.672678	0.372172
1	-0.673336	-0.673134	-0.673922	-0.673982	0.311264
2	-0.673959	-0.673143	-0.673669	-0.673775	1.002914
3	-0.673630	-0.673630	-0.674786	-0.675018	0.976000
4	-0.674961	-0.674451	-0.674717	-0.674396	0.176954

Transformación logarítmica

La transformación logarítmica permite reducir los cesgos y hacer los datos más normales, lo que ayuda a manejar los outliers, esta normalizacón será la que utilizaremos en nuestra implementación

3.1.7. Análisis de los outliers después de normalizar con la transformación logarítmica

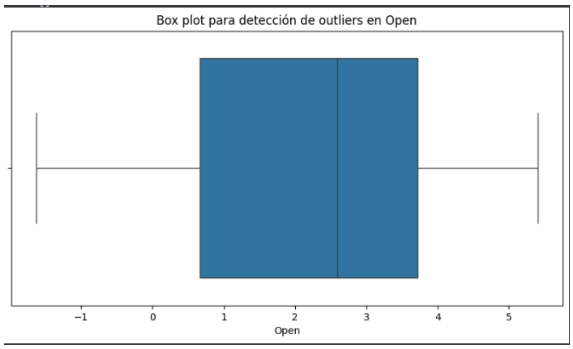

Figura 3.12

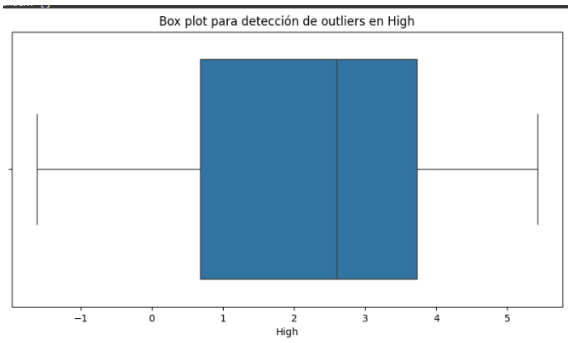

Figura 3.13

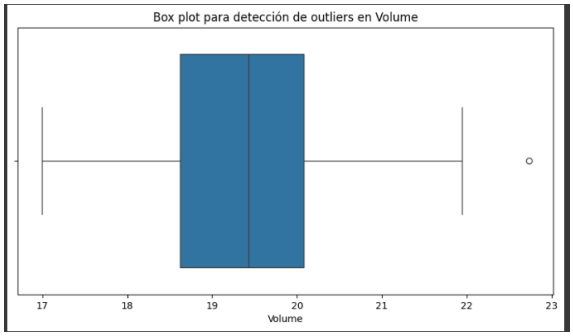

Figura 3.14

Se puede notar como se reduce la cantidad de outliers luego de la normalización y se obtiene una distribución más normal de los datos

Distribución de los datos luego de normalizar

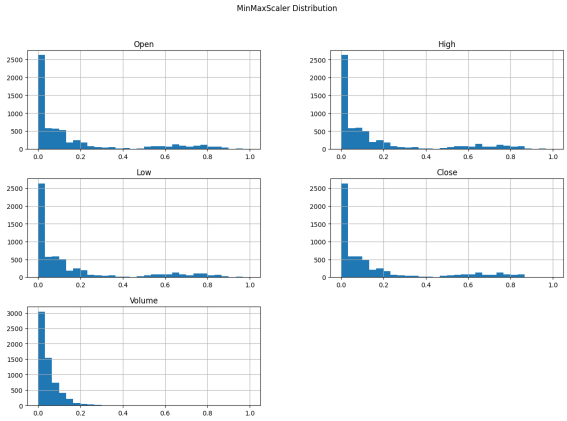

Figura 3.15: Distribución de los datos después de MinMax

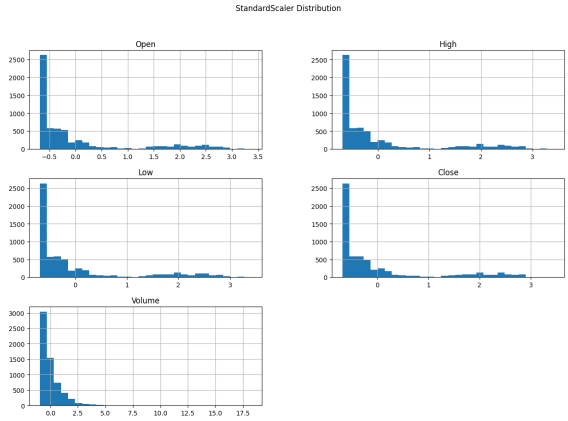

Figura 3.16: Distribución de los datos desués de Standard Scaler

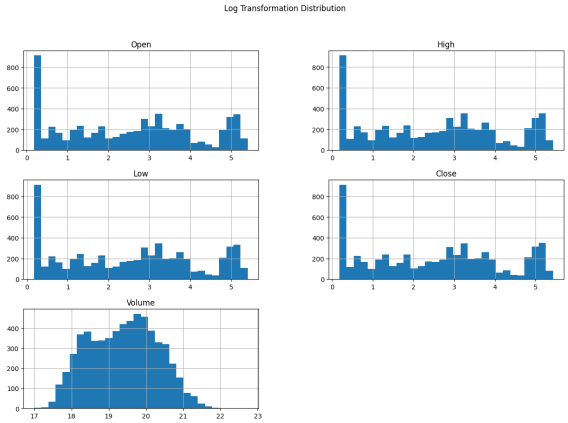

Figura 3.17: Distribución de los datos después de Transformación Logarítmica

La transformación logarítmica resuelve el problema del cesgo en los datos y los outliers como se observa en la figura 3.17. Lo cuál hace que los datos tengan escalas semejantes y elimina el ruido que agregan los datos con tamaños muy grandes que es el principal problema que nos habíamos encontrado.

3.2. Análisis del dataset de noticias

El dataset de noticias en nuestro caso, de igual manera fue extraído de Kaggle y contiene noticias de los últimos años de la empresa tecnológica Apple(tabla 3.5). El dataset contiene 12 columnas y 15975 entradas. Para nuestro trabajo sólo nos quedaremos con el contenido de la noticia que es el que queremos vincular al movimiento de los precios. El resto de columnas son valores que no tienen sentido para este trabajo como Label, Ticket, Category que no aportan información relevante. También hay datos que redundan como el precio, el volumen etc.

Tabla 3.5: Dataset de noticias(primeras 3 noticias)

Date		**Ticker**	**Category**	**Title**	**Content**	**Open**	**High**	**Low**	**Close**
2020-01-27	0	AAPL	opinion	Apple Set ...	Technology ...	77.514	77.942	76.22	77.237
2020-01-27	1	AAPL	opinion	Tech Daily ...	The top ...	77.514	77.942	76.22	77.23
2020-01-27	2	AAPL	opinion	7 Monster ...	S P 500 ...	77.514	77.942	76.22	77.23

Eliminación de datos innecesarios y redundantes

Notamos que todas las fechas estaban en tipo de dato Object y que a diferencia de las fechas en los datos de los precios estas no tenían la hora del día, por lo que fue necesario transformar estos datos para poder así vincular cada fecha en que el activo se investigó con su respectiva noticia vigente en ese día. También nos encontramos con muchas fechas sin noticias y muchas noticias repetidas a lo largo deldataset como se nota en la tabla 3.2, las cuáles eliminamos para evitar así redundancias en los datos.

Cantidad de noticias	15975
Cantidad de fechas con noticias	1654
Noticias sobrantes	14321

Finalmente nos quedamos con un dataset de 1654 entradas como en la tabla 3.6 que solo contiene la fecha y la noticia en sí

Tabla 3.6: Dataset Resultante

Date	**Content**
2012-07-23	Summer Heat Scorches Europe ...
2012-07-24	Market Bait And Switch that is ...
2012-07-27	Will AAPL Fall From The Tree Apple ...
2012-07-30	Bulls Snatch Victory From Jaws ...
2012-07-31	What's Driving China Real Estate ...

Procesamiento del contenodo de las noticias con técnicas de NLP

Para procesar las noticias empleamos técnicas de NLP como la eliminación de stopwords, signos de puntuación, tokenización de los contenidos de las noticias y entrenamos un Modelo Word2Vect para así obtener ua representación de embedding de el contenido de dichas noticias obteniendo la tabla 3.7. Esto nos permite una representación numérica y cómoda para entrenar nuestro modelo y una forma de capturar relaciones más complejas entre los precios y el contenido de las noticias.

Tabla 3.7: Resultado de aplicación de técnica nlp al dataset de noticias

Date	Content	Processed	Embedding
2012-07-23	Summer Heat Scorches...	[summer, heat, scorches...	[0.05905736, 0.78830993...
2012-07-24	Market Bait And...	[market, bait, switch...	[0.37770188, 0.25995106...
2012-07-27	Will AAPL Fall...	[aapl, fall, tree...	[-0.34234846, 0.27494824...
2012-07-30	Bulls Snatch Victory...	[bulls, snatch, victory...	[0.110929586, 0.7759576...
2012-07-31	What's Driving China...	[driving, china, real,...	[0.18154113, 0.86173123...

3.2.1. Clusterización de las noticias

Con el objetivo de observar el agrupamiento de las noticias dado su contenido y cuán distantes o agrupadas están unas de otras decidimos crear clusters con el objetivo de observar esto. Utilizamos el algoritmo K-Means con diferentes k. Para determinar cuál sería el valor óptimo del hiperparámetro K hicimos un análisis del índice de silueta.

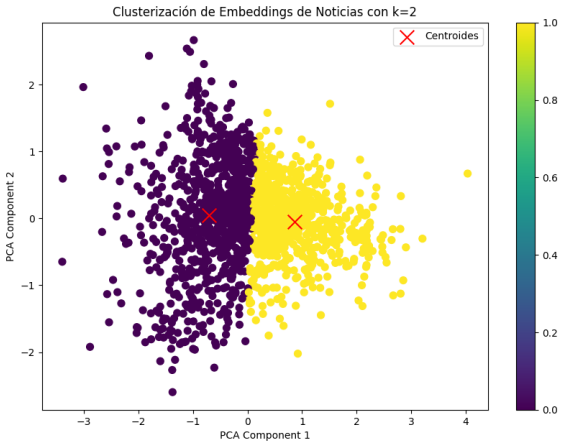

Figura 3.18: Clusters para K=2

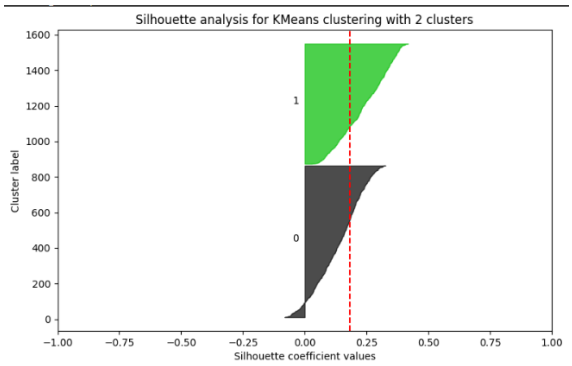

Figura 3.19: Análsis de silueta para K=2

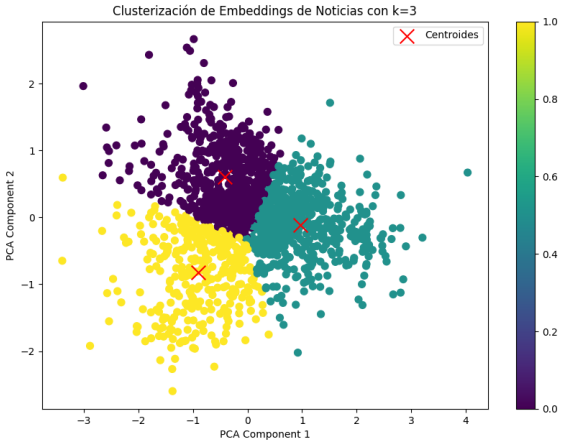

Figura 3.20: Clusters para K=3

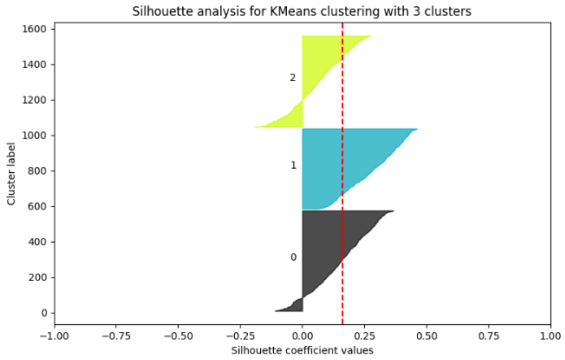

Figura 3.21: Análisis de silueta para K=3

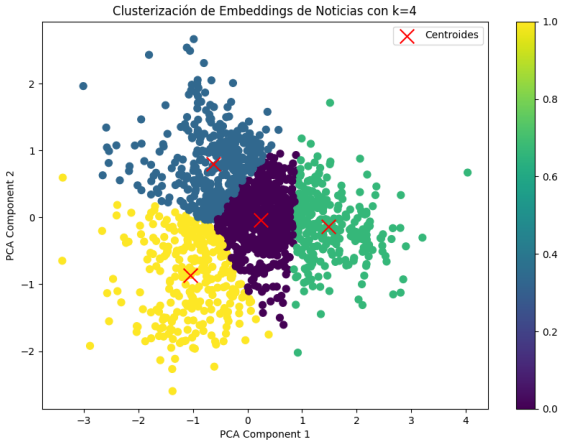

Figura 3.22: Clusters para K=4

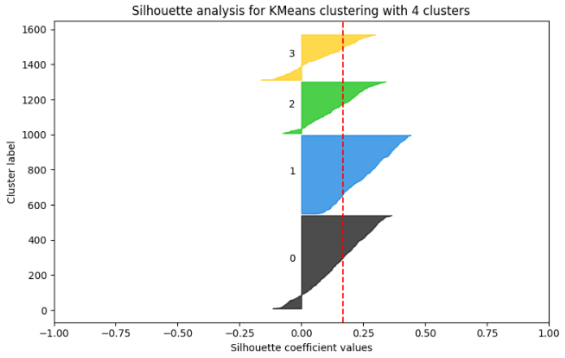

Figura 3.23: Análisis de silueta para K=4

Después de este análisis llegamos a la conclusión de que la mejor forma de clusterizar el conjunto de noticias es con 2 clusters como en las figuras 3.18 y 3.19, pues mantiene consistente la distancia media de los elementos del cluster. En un análisis posterior de los centroides pretendíamos analizar el contenido de estos para determinar si eran noticias que afectaban positiva o negativamente a los precios de las noticias y así clasificar las noticias en base a esto, por cuestiones ya mencionadas esta idea fue descartada en el presente trabajo.

Por cuestiones de tiempo y complejidad de estos algoritmos descartamos la idea de hacer un análisis utilizando modelos como DBSCAN o HDBSCAN que serían más

adecuados dada la naturaleza no lineal de los datos textuales como el contenido de las noticias y su capacidad de encontrar grupos basándose en la densidad y proximidad que sería un enfoque beneficioso ya que las noticias que tratan temas semejantes entre sí tienden a estar agrupadas pues usan regularmente palabras muy similares, cosa que los embeddings son capaces de captar en los vectores resultantes. También temas muy presentes en los datos, tienden a estar más concentrados, creando puntos de alta densidad, mientras que temas menos tratados tienden a estar más dispersos.

3.3. Distribución del dataset para las fases de entrenamiento y test

Para abordar la limitada disponibilidad inicial de datos, se utilizó la estrategia de validación cruzada específicamente para series temporales. Inicialmente, se optó por una división de datos en proporciones de 80 % para entrenamiento, 10 % para validación y 10 % para pruebas. Diversas configuraciones alternativas fueron probadas manteniendo coherencia en la distribución del conjunto de datos, incluyendo divisiones del tipo 70-20-10 y 30-40-20. Aunque no se observaron diferencias sustanciales en el rendimiento del modelo en la mayoría de los casos, se destacó una disminución significativa en el desempeño cuando se asignaba una cantidad reducida de datos para el entrenamiento. Es relevante señalar que durante el proceso de entrenamiento y validación del modelo se empleó el método de validación cruzada, considerado el más adecuado para abordar las particularidades inherentes a las series temporales.

3.4. Entrenamiento y diseño del Modelo

El modelo escogido es una red neuronal recurrente usando la arquitectura LSTM que permite un mejor análisis de series temporales. Hicimos varias pruebas entrenando al modelo con diferentes conjuntos del dataset, agregando datos extras como los indicadores EMA.

Inicialmente tenemos un modelo LSTM cuyos datos de entrada sólo pertenecen a los registros históricos de movimiento del activo que queremos predecir, este modelo tiene dos capas, una capa oculta con 128 neuronas y una capa densa para el output. El entrenamiento se relizó con 160 epochs y un learning rate de $5 * 10^{-5}$. A continuación algunos resultados:

3.4.1. Primera Iteración : Entrenando solo con los datos de Close

Tabla 3.8: Resultados del entrenamiento

Performance	
Train RMSE	0.049
Validation RMSE	0.117
Test RMSE	0.272

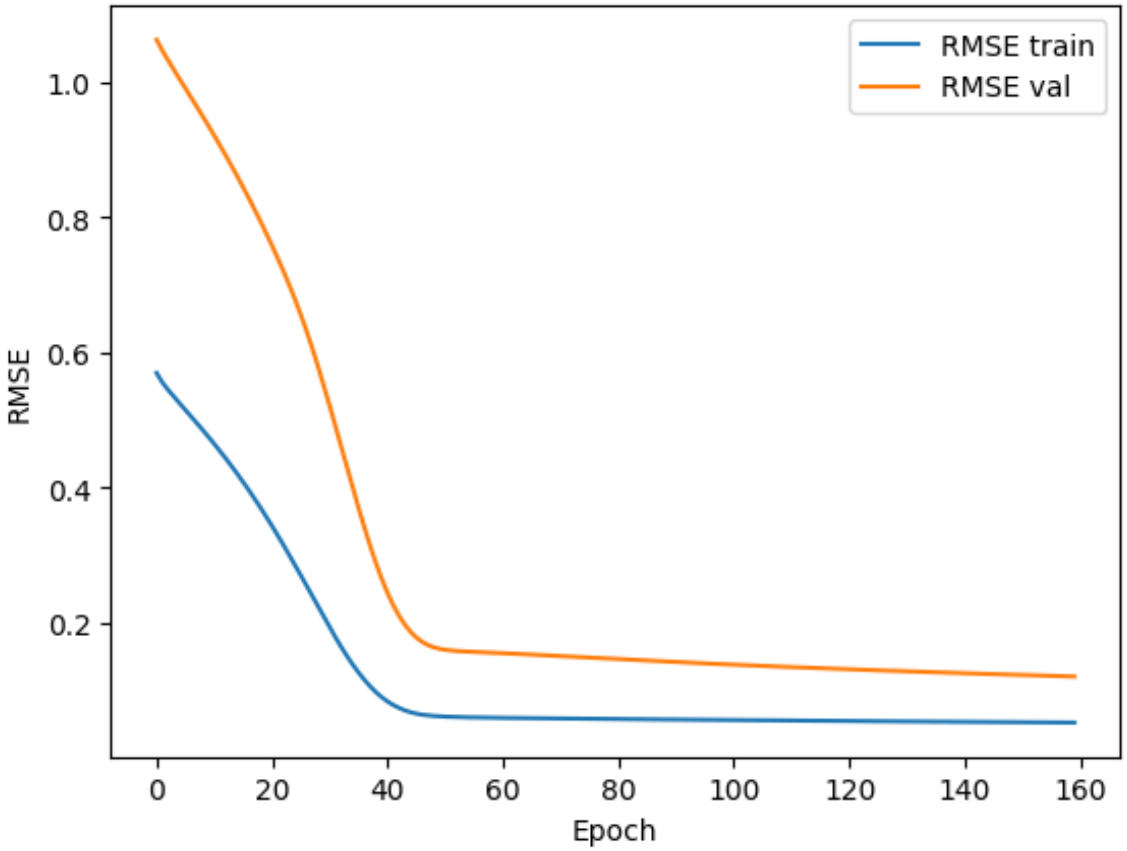

Figura 3.24: Resultados del entrenamiento

Tiene una buena consistencia entre los datos de entrenamiento, aunque aún no vincula la información relacionada con las noticias.

3.4.2. Segunda Iteración: Entrenando Agregando un Indicador EMA-50

Performance	
Train RMSE	0.009
Validation RMSE	0.052
Test RMSE	0.193

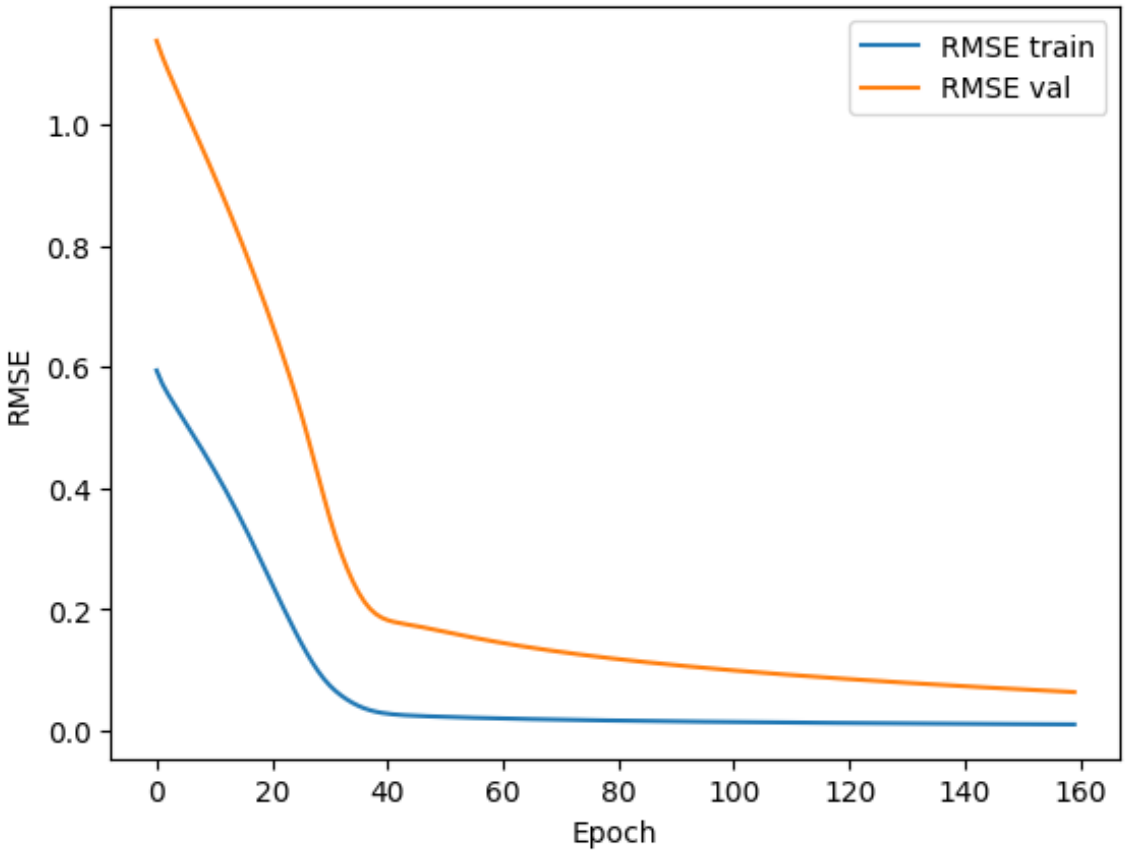

Figura 3.25: Resultados del entrenamiento

Se puede notar que al añadir la EMA50 al modelo la predicción de los datos mejoró notablemente, bajando del 27 al 19 porciento.

3.4.3. Tercera Iteración: utilizando todas las columnas del dataset (Close, Open, High, Low, Volume)

Performance	
Train RMSE	0.134
Validation RMSE	0.073
Test RMSE	0.066

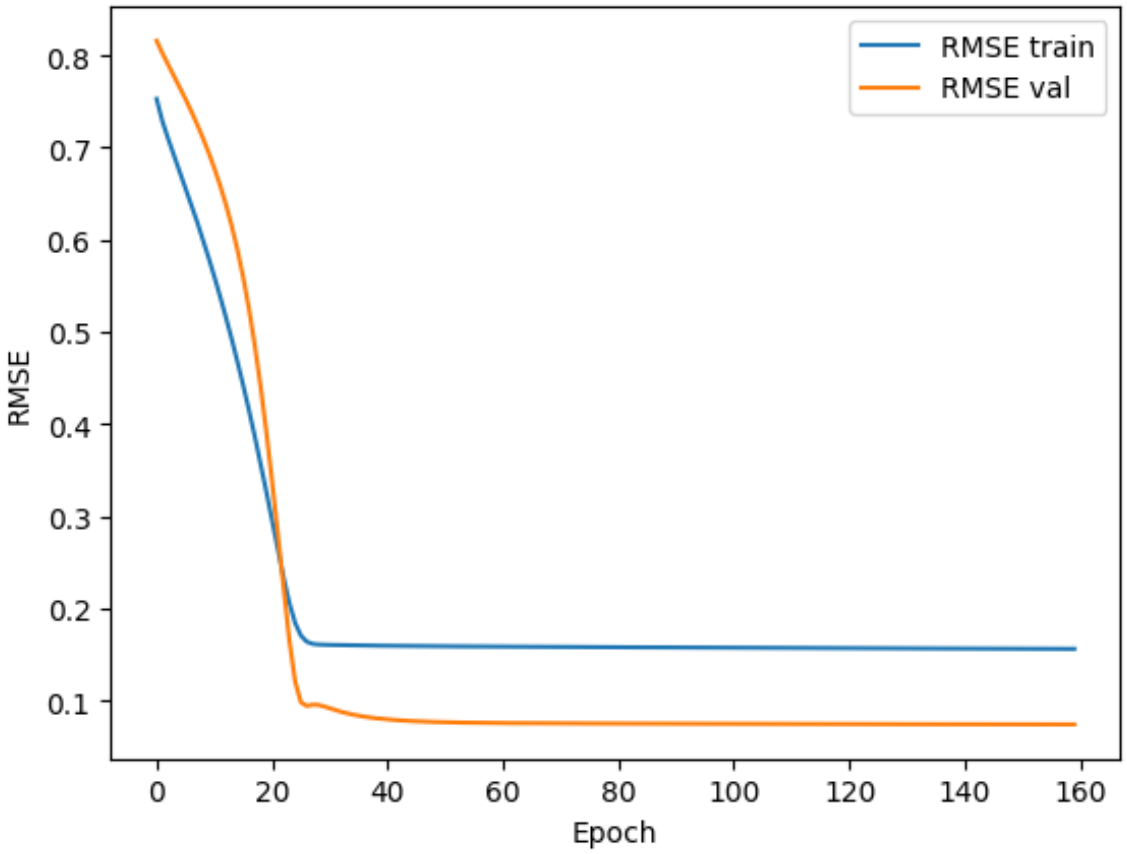

Figura 3.26: Resultados del entrenamiento

3.4.4. Iteración 4 (Añadiendo noticias)

Para tratar de encontrar la relación semántica entre la evolución del precio y las palabras o contenido de las noticias, utilizamos **WordToVect** para hallar el embedding de cada noticia, tomamos cada componente del embedding como un feature junto

al resto de covariables. Nuestro modelo es ahora una red Neuronal con arquitectura LSTM con un 105 covaraibles nuevas, una por cada componente del embedding, 256 neuronas en las capas ocultas y un learning rate de 5×10^{-5}. Hicimos una separación 80, 10, 10 ,donde 80 % es entrenamiento 10 % es validación y el otro 10 % es de test. Obtuvimos los resultados siguientes:

Performance	
Train RMSE	0.210
Validation RMSE	0.160
Test RMSE	0.195

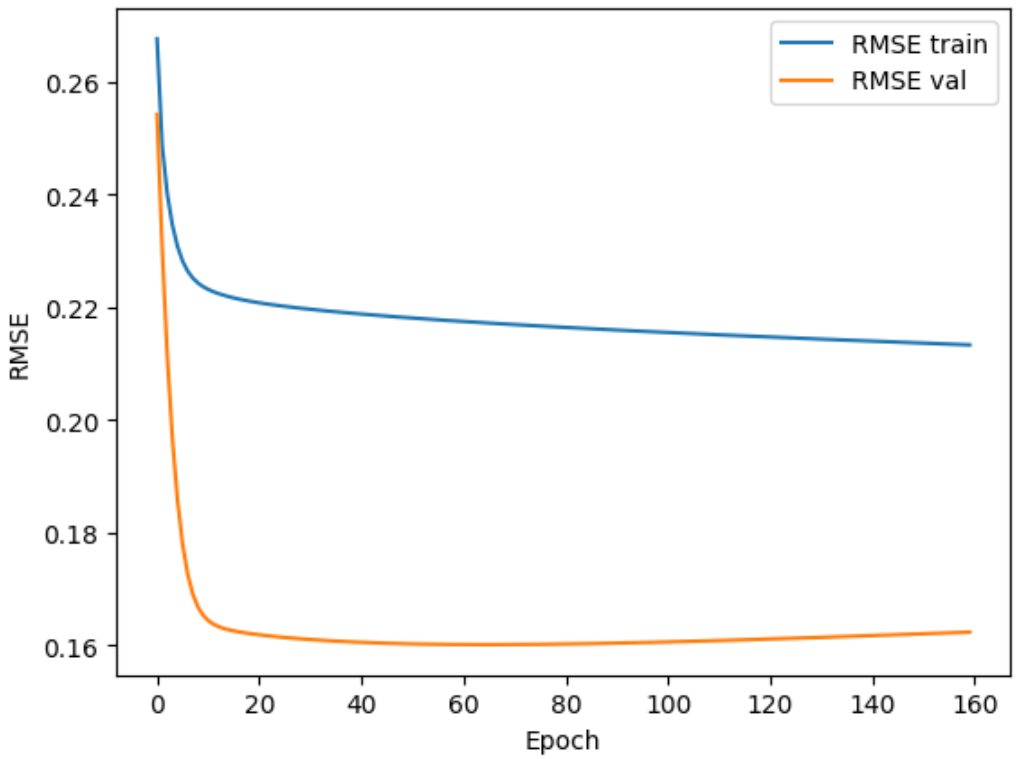

Figura 3.27: Resultados Obtenidos después del entrenamiento

3.4.5. Iteración 5 Ampliando el entrenamiento

Después de Probar con varias configuraciones y parámetros decidimos ampliar el modelo agregando más neuronas en las capas ocultas hasta 200 y aumentando

el número de iteraciones del entrenamiento a 340. Haciendo esto el modelo mejoró mínimamente los resultados anteriores, sinó que el rango de error se mantivo muy similar a la iteración anterior, o sea, que añadiendo mas neuronas e interaciones al modelo este no mejora su desempeño.

Performance	
Train RMSE	0.203
Validation RMSE	0.164
Test RMSE	0.185

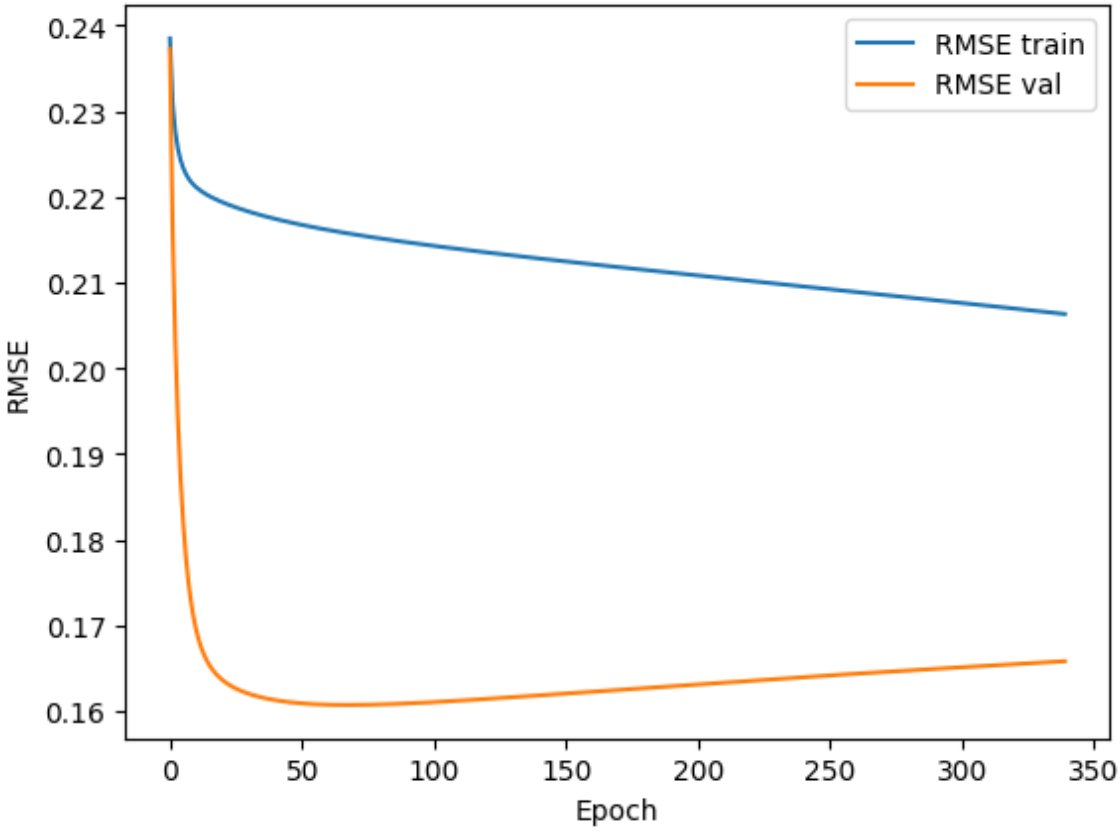

Figura 3.28: Resultados de extender el modelo

3.4.6. Iteración 6: Modificación de los hiperparámetros del modelo

Aqui colocamos 150 neuronas en las capas ocultas y redujimos el EPOCH en 200

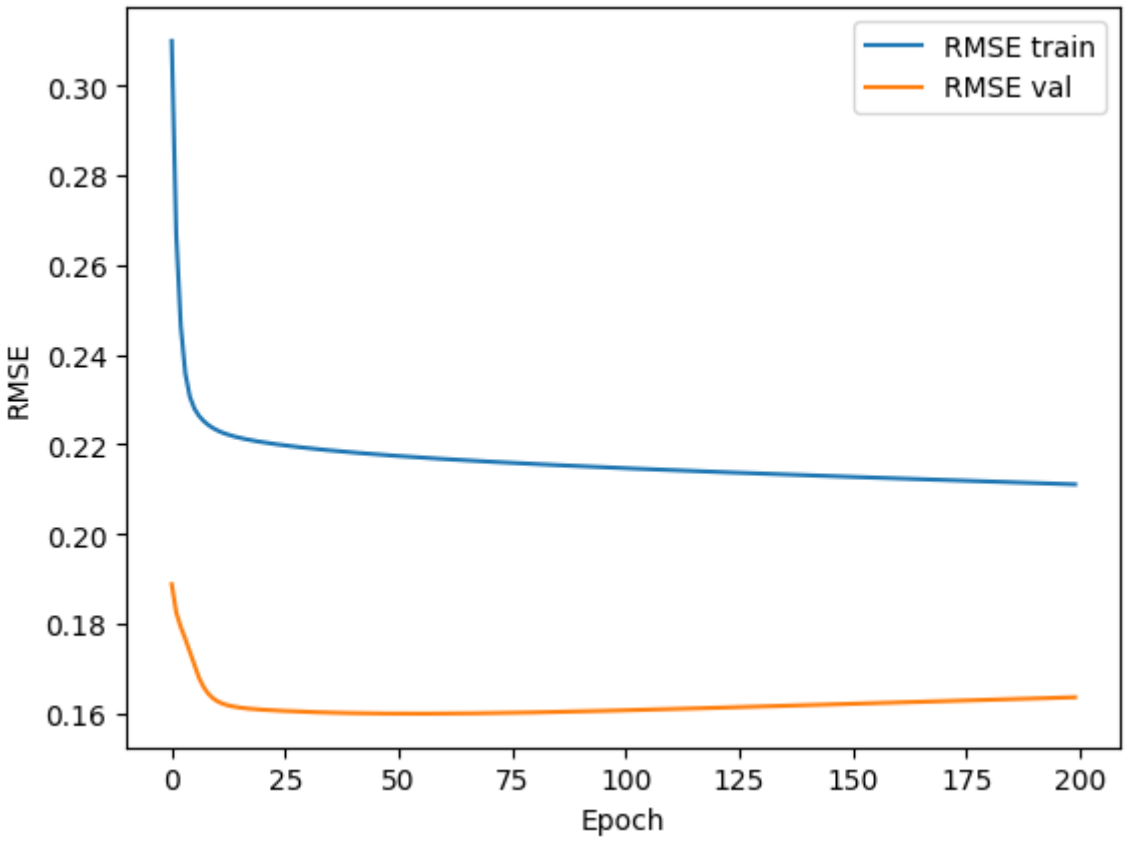

Figura 3.29: Resultados

Performance	
Train RMSE	0.208
Validation RMSE	0.162
Test RMSE	0.201

3.4.7. Iteración 7: Rectificando errores en los datos

Después de varias iteraciones y muchas prubas cambiando los hiperparámetros del modelo notamos que existían en los datos un faltante de noticias. El dataset de noticias está desde julio del 2012 hasta febrero del 2020, mientras que el de los precios estaba desde el 2000 hasta el 2024. Por esto cambiamos y acortamos el dataset para hacer coincidir estos rangos de fecha, dejándonos los siguientes resultados.

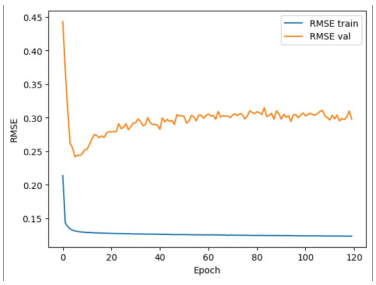

Figura 3.30: Prueba 1

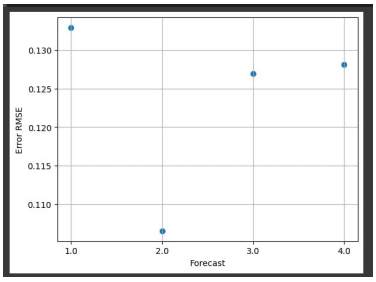

Figura 3.31: Errores en la predicción de varios parámetros

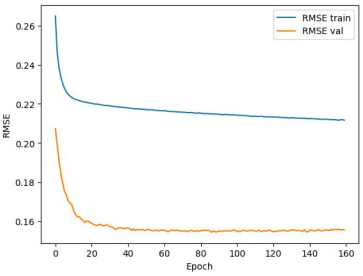

Figura 3.32: Prueba 2

Se hace notar la baja capacidad de generalización del modelo, así como la baja calidad de los resultados obtenidos.

3.5. Iteración 8: Redefiniendo la forma de abordar el problema

Dado el mal desempeño que estabamos observando en nuestra idea original, decidimos cambiar de perpectiva y abordar el problema de una manera diferente. Nuestro nuevo enfoque sería clasificar las noticias en positivas o negativas lo cuál favorece o penaliza el movimiento del activo financiero. Esto lo llevamos a cabo utilizando el modelo DistilBERT que encontramos en Hugging Face que nos permite no solo clasificar las noticias en positivas o negativas, sino que también nos permite tener un valor de pertenencia a cada categoría. Volvimos a hacer el análisis exploratorio y el tratamiento de datos anteriormente descrito, con la diferencia que en este caso no eliminamos las noticias que estaban en días repetidos, sino que clasificamos todas las noticias y las que estaban en un mismo día las promediamos. Esto para intentar capturar la esencia del comportamiento medio de las noticias que hablan sobre el activo en el momento. Luego agregamos las features de incidencia positiva e incidencia negativa al dataset donde teníamos los precios normalizamos los valores y entrenamos a una red LSTM para predecir las series temporales de los parámetros de movimiento de un activo financiero (Forecasting with Covariance).

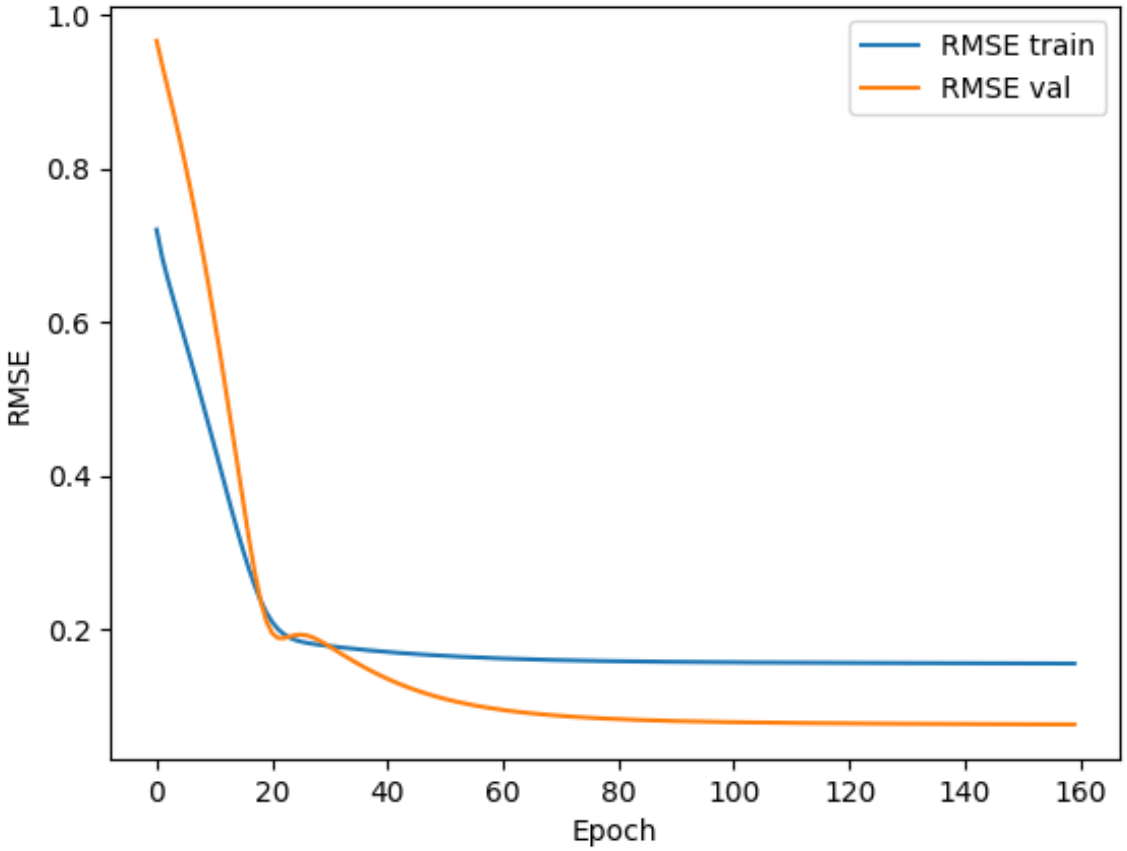

Figura 3.33: Resultados

Performance	
Train RMSE	0.133
Validation RMSE	0.074
Test RMSE	0.083

Luego de este cambio de enfoque obtuvimos mejores resultados evidenciando una mejor capacidad de generalización por parte del modelo y una reducción del error medio en los distintos predictores.

3.6. Iteración 9: Añadir relación entre noticias y cambio de precio

Además de comprobar si la noticia es positiva o negativa en sí misma, también comprobamos cuánto afecta esta noticia al precio en sí, para esto restamos por cada día el precio de cierre con el precio de apertura y lo dividimos entre que tan positiva y negativa es la noticia de ese día, esto nos da una relación entre crecimiento del precio y clasificación de la noticia.

Para este caso los resultados fueron peores que los obtenidos solo con la clasificación de las noticias en sí, con un error de hasta el 32%.

Performance	
Train RMSE	0.159
Validation RMSE	0.344
Test RMSE	0.325

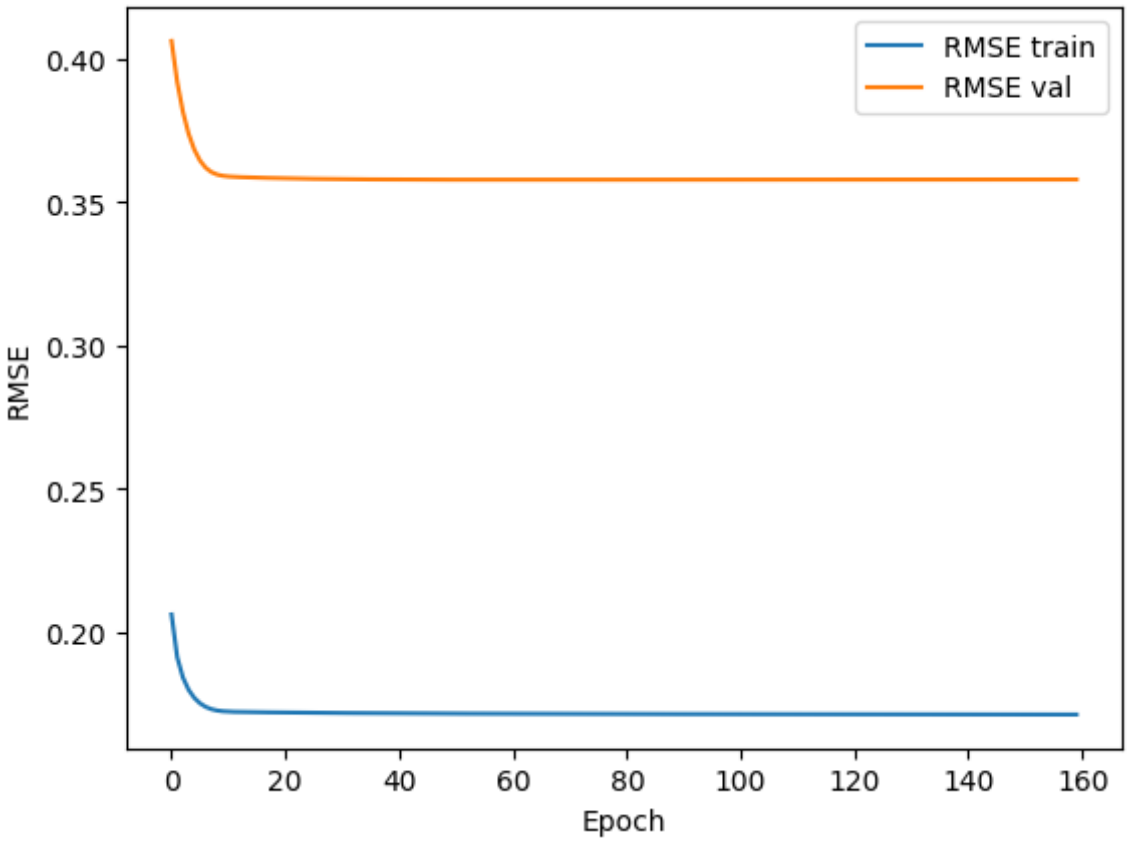

Figura 3.34: Resultados

Capítulo 4

Análisis de los resultados

4.1. Comparación de los modelos utilizados

A lo largo del proyecto hemos creado tres modelos para predecir el precio de los activos, uno utilizando el precio de los mismos, el segundo usando un clasificador de noticias, y el tercero utilizando el embeding de noticias. En la tabla 4.1 se reflejan los resultados obtenidos.

Tabla 4.1: Tabla comparativa de todas las iteraciones

Performance					
	epochs	learning rate	Train RMSE	Test RMSE	Val RMSE
Iter 1	160	5×10^{-5}	0.049	0.117	0.272
Iter 2	160	5×10^{-5}	0.009	0.052	0.193
Iter 3	160	5×10^{-5}	0.134	0.073	0.066
Iter 4	160	5×10^{-5}	0.210	0.160	0.195
Iter 5	340	5×10^{-5}	0.203	0.164	0.185
Iter 6	200	5×10^{-5}	0.208	0.162	0.201
Iter 8	160	5×10^{-5}	0.133	0.074	0.083
Iter 9	160	5×10^{-5}	0.159	0.344	0.325

Podemos ver que, además de la propia predicción de la serie de tiempo usando solo los valores por defecto que vienen en la serie (Iter 3) fue con un error de apenas el 6,6 % y, al usar las noticias, la mejor forma de predecir la serie es clasificando las noticias en positivas y negativas, y usando estos como dos covariables nuevas. Con este enfoque logramos una predicción con apenas el 8,3 % de error, la segunda mejor entre todas las iteraciones, y la mejor al usar las noticias en la serie de tiempo.

4.2. Resultados

Después de varias iteraciones de experimentación e intentos de optimizar nuestro modelo para resolver la tarea de predicción del mercado utilizando noticias como elemento de soporte para este análisis, observamos que la forma en que relacionamos las noticias con los precios solo introdujo ruido en el proceso. Por tanto, es necesario encontrar una representación más adecuada para llevar a cabo esta tarea. Este cambio de enfoque se presentó en forma de análisis de las noticias utilizando otro modelo en lugar de encargar todo el trabajo a un único modelo, esto evidencia una vez más la efectividad que tiene utilizar y mezclar varias opciones en pos de conseguir resolver los problemas y la necesidad de cambiar de perspectiva ante la no resolución satisfactoria de uno. Al explorar la bibliografía consultada, encontramos diversos enfoques para resolver este problema, cada uno con sus pros y sus contras. Entre todos los artículos revisados, hubo una idea que nos llamó particularmente la atención: la de proporcionar explicabilidad a los resultados, lo cual se aleja del enfoque que tomamos, pero es un factor crucial para comprender si estos modelos son realmente capaces de captar la complejidad del sistema financiero y relacionarlo efectivamente con datos del mundo real.

4.3. Implicaciones éticas referentes a la resolución de nuestro problema

La solución efectiva de el problema que tratamos por parte de nuestra solución o alguna otra tendría consecuencias graves para el comercio ya que este tipo de herramientas son muy probables de caer en manos de personas con poder y dinero esto traería consigo un desbalance de poder y más injusticia social. Es tema de debate en caso de que la solución al problema se hiciera pública que ocurriría y que consecuencias tendría, pues todos aquellos con la información y el conocimiento suficiente podrían utilizar las herramientas concebidas de estas soluciones y generar de igual manera un desbalance económico a gran o media escala. También está la posibilidad de una represión por parte de quienes quieren perpetuar su poder sobre los demás y este casi siempre está en evidencia por el dinero. También este tipo de problema se pudiera llevar a otros campos donde podría ser de utilidad y benedficio para la humanidad desarrollando así mejores herramientas que nos faciliten la vida. Por parte de nuestra solución esta solo nos sirve para evidenciar como una noticia (conjunto de palabras) puede ser capaz de incidir sobre cosas como el precio y movimiento del mercado de tal forma que llegan a cambiar y moldear la forma en que este fluctúa. Esto se puede llevar a cualquier escala de la sociedad donde las noticias y la influencia pueden hacer cambiar desde como se comporta el valor de nuestro dinero y pertenencias hasta como

nos comportamos nosotros.

Conclusiones

La realización de esta investigación nos confrontó con la compleja y errática realidad de intentar predecir entornos tan volátiles como el mercado financiero. Este desafío representa un reto significativo para los modelos de aprendizaje actuales. Al analizar los resultados de varios estudios y aplicar nuestras propias metodologías, constatamos la dificultad inherente y la imperiosa necesidad de desarrollar herramientas que integren variables externas, como las noticias, en los sistemas de predicción.

Nuestra investigación nos sirvió como base para comprender mejor la utilización y los problemas presentes en las soluciones basadas en *machine learning*. Revela que, aunque existe una abundancia de datos y una considerable inversión en este campo, los resultados no son tan sobresalientes como en otras áreas que también emplean técnicas de aprendizaje automático.

En resumen, si predecir el mercado fuese una tarea sencilla, todos seríamos ricos. Sin embargo, esto mismo cambiaría el mercado una vez más, creando nuevos desafíos que resolver.

Recomendaciones

Consideramos que futuros trabajos deberían enfocarse en la integración de técnicas de aprendizaje profundo con métodos de interpretación de modelos. Esto no solo aumentaría la precisión de las predicciones, sino que también facilitaría la comprensión de cómo las noticias y otros factores externos influyen en el mercado. La incorporación de análisis de sentimientos y el uso de modelos híbridos que combinan datos estructurados y no estructurados podrían ofrecer una perspectiva más holística y precisa. De esta manera, no solo se mejoraría la exactitud de los modelos predictivos, sino que también se incrementaría la confianza en su aplicabilidad práctica en entornos financieros reales.

Bibliografía

[1] Rajashree Dash y P. K. Dash. «Stock price index movement classification using a CEFLANN with extreme learning machine». En: *2015 IEEE Power, Communication and Information Technology Conference (PCITC)*. 2015, págs. 22-28. DOI: 10.1109/PCITC.2015.7438176.

[2] Vaia I. Kontopoulou et al. «A Review of ARIMA vs. Machine Learning Approaches for Time Series Forecasting in Data Driven Networks». En: *Future Internet* 15.8 (2023). ISSN: 1999-5903. DOI: https://doi.org/10.3390/fi15080255. URL: https://www.mdpi.com/1999-5903/15/8/255.

[3] Shilpa Gite et al. «Explainable stock prices prediction from financial news articles using sentiment analysis». En: *PeerJ Computer Science* 7 (2021). DOI: 10.7717/peerj-cs.340. URL: https://api.semanticscholar.org/CorpusID:231827830.

[4] Piotr Fiszeder y Witold Orzeszko. «Covariance matrix forecasting using support vector regression». En: *Applied Intelligence* 51 (2021), págs. 7029-7042. DOI: 10.1007/s10489-021-02217-5. URL: https://link.springer.com/article/10.1007/s10489-021-02217-5.

[5] Daisuke Hotta et al. «EFSR: Ensemble Forecast Sensitivity to Observation Error Covariance». En: *Monthly Weather Review* 145 (2017), págs. 5015-5031. DOI: 10.1175/MWR-D-17-0122.1. URL: https://journals.ametsoc.org/view/journals/mwre/145/12/mwr-d-17-0122.1.xml.

[6] Sean J Taylor y Benjamin Letham. «Forecasting at scale(Prophet)». En: *PeerJ Preprints* 5 (2017). DOI: 10.7287/peerj.preprints.3190v2. URL: https://peerj.com/preprints/3190v2/.

[7] Anita Yadav, C K Jha y Aditi Sharan. «Optimizing LSTM for time series prediction in Indian stock market». En: *Procedia Computer Science* 167 (2020). International Conference on Computational Intelligence and Data Science, págs. 2091-2100. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2020.03.257. URL: https://www.sciencedirect.com/science/article/pii/S1877050920307237.

[8] Mugdha Kulkarni, Anil Jadha y Deepika Dhingra. «Time Series Data Analysis for Stock Market Prediction». En: *Proceedings of the International Conference on Innovative Computing Communications (ICICC)* (2020). DOI: 10.2139/ssrn.3563111. URL: https://ssrn.com/abstract=3563111.

[9] M. Nabipour et al. «Deep Learning for Stock Market Prediction». En: *Entropy* 22.8 (2020). ISSN: 1099-4300. DOI: 10.3390/e22080840. URL: https://www.mdpi.com/1099-4300/22/8/840.

[10] Se-Hak Chun y Young-Woong Ko. «Geometric Case Based Reasoning for Stock Market Prediction». En: *Sustainability* 12.17 (2020). ISSN: 2071-1050. DOI: 10.3390/su12177124. URL: https://www.mdpi.com/2071-1050/12/17/7124.

[11] Anne Haubo Dyhrberg. «Bitcoin, gold and the dollar - A GARCH volatility analysis». En: *Finance Research Letters* 16 (2016), págs. 85-92. DOI: 10.1016/j.frl.2015.10.008. URL: https://www.sciencedirect.com/science/article/abs/pii/S1544612315001038.

[12] Dr. M. Durairaj y B. H. Krishna Mohan. «A convolutional neural network based approach to financial time series prediction». En: *Neural Comput & Applic* 34 (2022), págs. 13319-13337. DOI: 10.1007/s00521-022-07143-2. URL: https://link.springer.com/article/10.1007/s10489-021-02217-5.