

**MÁSTER DE FORMACIÓN PERMANENTE EN INTELIGENCIA
ARTIFICIAL**

**Aplicación de inteligencia artificial a un sistema de
trading en el par de divisas EURUSD**

Presentado por:

RONALD JAVIER VEZGA RUEDA

Dirigido por:

VICENTE CASTILLO FAULI

CURSO ACADÉMICO 2024-2025

Resumen

El presente estudio analiza el diseño, implementación y evaluación de un sistema de trading algorítmico aplicado al par de divisas EUR/USD, con el objetivo de mejorar la toma de decisiones mediante el uso de técnicas de inteligencia artificial. En un contexto financiero cada vez más dinámico y dominado por grandes volúmenes de datos, el uso de modelos de aprendizaje automático y procesamiento de lenguaje natural permite optimizar las estrategias de inversión, reducir la exposición al riesgo y aumentar la eficiencia operativa.

El sistema desarrollado se apoya en una arquitectura modular basada en eventos e integra dos bloques de inteligencia artificial: un modelo de detección de anomalías (Isolation Forest), que permite filtrar situaciones de mercado no convencionales, y un modelo de lenguaje natural (transformer) que analiza noticias económicas para contextualizar las decisiones de trading. El trabajo incluye un análisis comparativo con datos históricos (backtesting) comparativo entre una estrategia clásica y otra enriquecida con IA, demostrando mejoras significativas en términos de rentabilidad ajustada al riesgo.

Palabras clave: Machine learning, trading algorítmico, mercado de divisas, estrategias de inversión, detección de anomalías, análisis de sentimiento.

Abstract

This study examines the design, implementation, and evaluation of an algorithmic trading system applied to the EUR/USD currency pair, aiming to enhance decision-making through the use of artificial intelligence techniques. In an increasingly dynamic financial environment dominated by large volumes of data, the application of machine learning models and natural language processing helps optimize investment strategies, reduce risk exposure, and increase operational efficiency.

The developed system is based on a modular, event-driven architecture and integrates two artificial intelligence components: an anomaly detection model (Isolation Forest) to filter out unconventional market situations, and a natural language processing model (transformer) that analyzes economic news to provide context for trading decisions. The study includes a comparative analysis using historical data (backtesting) between a traditional strategy and an AI-enhanced strategy, demonstrating significant improvements in risk-adjusted returns.

Keywords: Machine learning, algorithmic trading, foreign exchange market, investment strategies, anomaly detection, sentiment analysis.

Índice de contenido

1. Introducción

- 1.1. Contextualización del mercado financiero actual
- 1.2. Planteamiento del problema
- 1.3. Motivación personal y relevancia del proyecto
- 1.4. Preguntas de investigación e hipótesis
- 1.5. Objetivos generales y específicos
- 1.6. Alcance y limitaciones del trabajo

2. Marco teórico

- 2.1. El mercado de divisas (FOREX)
 - 2.1.1. Características del mercado EURUSD
 - 2.1.2. Volatilidad e impacto macroeconómico
- 2.2. Tipos de análisis aplicados en trading
 - 2.2.1. Análisis técnico
 - 2.2.2. Análisis fundamental
 - 2.2.3. Análisis cuantitativo y algorítmico
- 2.3. Backtesting
- 2.4. Sistemas de trading automatizados
 - 2.4.1. Arquitectura modular de un sistema de trading

2.4.2. Componentes del sistema:

- a. Fuente de datos (Data Source)
- b. Generador de señales (Signal Generator)
- c. Dimensionador de posición (Position Sizer)
- d. Gestor de riesgos (Risk Manager)
- e. Ejecutor de órdenes (Order Executor)
- f. Notificador (Notifier)
- g. Analizador de sentimiento (Sentiment Analyzer IA)
- h. Detector de anomalías (Anomaly Detector IA)

2.5. Fundamentos de inteligencia artificial

2.5.1. Introducción a Machine Learning

2.5.2. Aprendizaje supervisado vs no supervisado

2.5.3. Algoritmos aplicados a trading

2.5.4. Detección de anomalías en finanzas

2.6. Revisión del estado del arte

2.6.1. Proyectos y plataformas similares

2.6.2. Principales avances y limitaciones actuales

3. Metodología

3.1. Diseño metodológico (CRISP-DM aplicado al TFM)

3.2. Estrategia de investigación y enfoque computacional

3.3. Herramientas y lenguajes de desarrollo (Python, MT5 API, scikit-learn)

3.4. Validación de resultados y métricas utilizadas

4. Diseño e implementación del sistema

4.1. Requerimientos del sistema

4.2. Diseño modular de componentes

4.3. Flujo de datos y eventos

4.4. Interacción con MetaTrader 5

4.5. Documentación del código

5. Procesamiento y análisis de datos

5.1. Recolección de datos de mercado

5.2. Limpieza y preprocesamiento de series temporales

5.3. Extracción de características e ingeniería de variables

6. Integración de modelos de IA

6.1. Implementación del detector de anomalías

6.1.1. Algoritmo Isolation Forest

6.1.2. Umbrales y evaluación del modelo

a. Análisis de la distribución de las puntuaciones de anomalía

b. Validación cualitativa con eventos históricos

c. Impacto en la estrategia de trading

d. Ajuste del umbral

e. Visualización

6.2. Implementación del analizador de sentimiento

6.2.1. Análisis de titulares financieros o métricas del mercado

6.2.2. Efecto sobre la toma de decisiones

6.3. Integración con el generador de señales

6.3.1. Matriz de confusión y informe de clasificación

7. Evaluación del sistema y análisis de resultados

7.1. Condiciones de la prueba y métricas utilizadas

7.2. Comparación de estrategias: sin IA vs con IA

7.3. Análisis cualitativo de resultados

7.4. Limitaciones detectadas y robustez del sistema

8. Conclusiones

8.1. Principales aportaciones del proyecto

8.2. Evaluación de objetivos cumplidos

9. Líneas de trabajo futuras

9.1. Ampliación a otros activos o marcos temporales

9.2. Inclusión de redes neuronales (LSTM, transformers)

9.3. Mejora del backtesting con datos sintéticos o simulados

10. Referencias bibliográficas

11. Anexos

11.1. Código en github

11.2. Glosario de acrónimos y términos técnicos

1. Introducción

1.1. Contextualización del mercado financiero actual

En la última década, los mercados financieros han experimentado una transformación radical impulsada por la digitalización, el acceso masivo a datos en tiempo real y la automatización de procesos de inversión. El auge del trading algorítmico (compraventa automatizada de activos financieros) y el desarrollo de infraestructuras tecnológicas más eficientes han permitido que tanto instituciones como inversores individuales diseñen estrategias de operación automatizadas capaces de procesar información a velocidades y volúmenes antes impensables.

Dentro de este nuevo paradigma, el mercado de divisas (FOREX, Foreign Exchange) se posiciona como uno de los entornos más dinámicos y líquidos, donde se intercambian diariamente billones de dólares entre pares de monedas. La elevada volatilidad y disponibilidad continua de datos lo convierte en un campo fértil para la aplicación de técnicas avanzadas de análisis, particularmente aquellas derivadas de la inteligencia artificial (IA).¹

1.2. Planteamiento del problema

Tradicionalmente, las estrategias de trading han dependido de modelos deterministas basados en reglas fijas o indicadores técnicos. Sin embargo, estos enfoques presentan limitaciones en contextos de alta incertidumbre, eventos inesperados o cambios estructurales en los mercados. Uno de los principales desafíos consiste en la incapacidad de estos sistemas para adaptarse a condiciones no convencionales, lo que puede derivar en pérdidas significativas.

¹ El Fondo Monetario Internacional declara que la inteligencia artificial puede mejorar la eficiencia de los mercados, y avivar su volatilidad, FMI Blogs, 15 de octubre de 2024.

En este contexto, surge la necesidad de incorporar mecanismos de aprendizaje automático y procesamiento de lenguaje natural que permitan identificar de forma automática patrones anómalos o interpretar eventos macroeconómicos desde una perspectiva contextual. La inclusión de estos elementos no solo aumenta la robustez del sistema, sino que también proporciona una ventaja competitiva frente a modelos tradicionales.

1.3. Motivación personal y relevancia del proyecto

Este trabajo de fin de máster nace del interés por explorar la intersección entre inteligencia artificial y mercados financieros. Como profesional con formación en ingeniería de computación y una fuerte orientación al análisis cuantitativo, he identificado el trading algorítmico como un campo de aplicación idóneo para poner en práctica conocimientos avanzados de IA.

Además de representar un desafío técnico y académico, este proyecto tiene una clara relevancia práctica, ya que propone una solución que puede ser aplicada y escalada en entornos reales de inversión. La combinación de técnicas como la detección de anomalías y el análisis de sentimiento aporta un valor añadido significativo a la toma de decisiones financieras automatizadas.

1.4. Preguntas de investigación e hipótesis

Este trabajo busca responder a las siguientes preguntas de investigación:

- ¿Puede un sistema de trading enriquecido con inteligencia artificial superar en rentabilidad y consistencia a una estrategia de trading tradicional?
- ¿Qué impacto tiene la detección de anomalías en la reducción del riesgo operativo?
- ¿Cómo influye el análisis de sentimiento de las comunicaciones del sector en la mejora de las decisiones de entrada y salida en el mercado?

A partir de estas preguntas, se plantea la hipótesis central:

La incorporación de módulos de inteligencia artificial en un sistema de trading algorítmico sobre el par EURUSD mejora significativamente su rendimiento ajustado al riesgo respecto a estrategias convencionales.

1.5. Objetivos generales y específicos

Objetivo general

- Diseñar, implementar y evaluar un sistema de trading algorítmico para el par EURUSD, que integre componentes de inteligencia artificial para optimizar la rentabilidad, reducir la exposición al riesgo y mejorar la contextualización de las decisiones.

Objetivos específicos

- Diseñar una arquitectura modular que permite integrar múltiples componentes de análisis y ejecución.
- Implementar un módulo de detección de anomalías utilizando el algoritmo Isolation Forest.
- Desarrollar un analizador de sentimiento basado en modelos de lenguaje tipo transformer.
- Realizar pruebas comparativas entre una estrategia base y otra enriquecida con IA mediante backtesting.
- Analizar los resultados obtenidos para validar la eficacia del sistema propuesto.

1.6. Alcance y limitaciones del trabajo

El presente trabajo se centra exclusivamente en el par de divisas EUR/USD, operando en intervalos de tiempo intradía, con datos históricos comprendidos durante el año 2023. El sistema se ejecuta en modo simulado, utilizando herramientas de backtesting y APIs públicas para la obtención de noticias.

Limitaciones:

- No se contempla ejecución en tiempo real con dinero real.
- Las fuentes de noticias pueden no cubrir todos los eventos relevantes del mercado.
- La evaluación de IA se restringe a técnicas no supervisadas (para anomalías) y a modelos pre entrenados (en NLP).

2. Marco teórico

2.1. El mercado de divisas (FOREX)

El mercado de divisas, conocido como FOREX (Foreign Exchange), es el mayor mercado financiero del mundo en términos de volumen negociado, con más de 6 billones de dólares transaccionados diariamente, según datos del Banco de Pagos Internacionales (BIS, 2022).

A diferencia de otros mercados financieros, el FOREX no posee una ubicación física ni opera en una bolsa centralizada, sino que funciona de forma descentralizada a través de una red global de bancos, brokers, instituciones financieras y operadores individuales.²

El atractivo del mercado FOREX radica en su alta liquidez, operativa continua las 24 horas durante los días hábiles, y apalancamiento flexible. Estas características lo convierten en un

² Bank for International Settlements. (2022). Triennial Central Bank Survey: Foreign exchange turnover in April 2022. https://www.bis.org/statistics/rpfx22_fx.htm

entorno ideal para el desarrollo de estrategias algorítmicas, en las que la rapidez y precisión en la ejecución de órdenes son factores clave de éxito.

2.1.1. Características del mercado EURUSD

El par EUR/USD es el instrumento más negociado en el mercado FOREX. Representa la relación entre el euro (moneda base) y el dólar estadounidense (moneda cotizada). Su volumen de negociación, liquidez y profundidad del mercado lo convierten en un activo particularmente atractivo para estrategias sistemáticas y backtesting fiable. Además, el comportamiento del EUR/USD refleja la interacción económica entre la Eurozona y los Estados Unidos, ofreciendo patrones técnicamente estables y predecibles.

2.1.2. Volatilidad e impacto macroeconómico

La volatilidad del EUR/USD está influenciada por una variedad de factores macroeconómicos: decisiones de política monetaria del BCE y la FED, datos de inflación, empleo, PIB y eventos geopolíticos. Esta sensibilidad a eventos exógenos hace que sea crucial incorporar mecanismos de detección de anomalías y análisis contextual para mejorar la fiabilidad de las estrategias de trading.³

2.2. Tipos de análisis aplicados en trading

El trading profesional se apoya en tres pilares principales de análisis, cada uno con sus fortalezas y limitaciones. Estos enfoques no son excluyentes y, de hecho, suelen combinarse para obtener una visión más completa del mercado.⁴

³ Investopedia, *Economic factors that affect the forex market* 2024.

⁴ eToro. (2024). Understanding technical, fundamental & quantitative analysis.

2.2.1. Análisis técnico

Se basa en el estudio de los movimientos históricos del precio y del volumen a través de gráficos e indicadores. Los traders técnicos buscan patrones como soportes, resistencias, líneas de tendencia o figuras de continuación. Entre los indicadores más utilizados están el RSI, MACD, medias móviles y bandas de Bollinger. El principio fundamental es que “el precio lo descuenta todo”, es decir, toda la información relevante ya está reflejada en el gráfico.

2.2.2. Análisis fundamental

Estudia los factores económicos, políticos y financieros que afectan a la oferta y demanda de una moneda. En el caso del EUR/USD, se consideran variables como tasas de interés, crecimiento económico, deuda pública, inflación o decisiones de los bancos centrales. El análisis fundamental busca entender el “por qué” detrás de los movimientos del mercado.

2.2.3. Análisis cuantitativo y algorítmico

Implica el uso de modelos matemáticos, estadísticos o computacionales para identificar patrones en los datos y desarrollar estrategias automatizadas. Incluye técnicas de regresión, optimización de portafolios, simulaciones de Monte Carlo, redes neuronales, aprendizaje automático y más. Este enfoque es el núcleo de la presente investigación, ya que permite automatizar decisiones de trading y evaluar su eficacia mediante backtesting.

2.3. Backtesting

El proceso de backtesting consiste en simular la ejecución del sistema de trading sobre datos históricos para evaluar su comportamiento en condiciones pasadas. Esta fase es esencial para validar la eficacia de la estrategia antes de considerar su uso en producción o entorno real.

2.4. Sistemas de trading automatizados

Un sistema de trading automatizado es un conjunto de reglas programadas que permiten ejecutar operaciones sin intervención humana, basado en criterios objetivos. Su principal ventaja radica en la eliminación del sesgo emocional, la posibilidad de operar múltiples activos simultáneamente, y la ejecución sistemática de señales.⁵

2.4.1. Arquitectura modular de un sistema de trading

El diseño modular permite dividir el sistema en bloques funcionales que se comunican entre sí mediante eventos. Esta estructura favorece el mantenimiento, la reutilización de código, y la integración de nuevas tecnologías como IA o big data.⁶

2.4.2. Componentes del sistema:

a. Fuente de datos (Data Source)

Obtiene y estructura los datos necesarios para alimentar el sistema: precios OHLC, volumen, indicadores técnicos, y en algunos casos, información textual. En este proyecto, la fuente es MetaTrader 5 mediante su API de Python.

b. Generador de señales (Signal Generator)

Aplica una estrategia técnica base (como cruce de medias móviles) para generar señales de entrada y salida. Puede contener múltiples estrategias que se activan bajo distintas condiciones del mercado.

⁵ Luchian y Strat, “The trustworthiness of AI algorithms and the simulator bias in trading,” 2024

⁶ Meso Software. (2025). *Considerations in Trading Systems Architecture*. Recuperado de

c. Dimensionador de posición (Position Sizer)

Calcula el tamaño de la operación según criterios como capital disponible, nivel de riesgo máximo aceptado o apalancamiento permitido. Implementa técnicas como “fixed size”, “risk percent” o “minimum size”.

d. Gestor de riesgos (Risk Manager)

Evalúa si una operación cumple con los criterios de riesgo definidos, como exposición máxima por operación, drawdown acumulado (pérdida máxima desde un pico de capital), o factores de apalancamiento. Filtra las señales no viables desde el punto de vista de la seguridad operativa.

e. Ejecutor de órdenes (Order Executor)

Encargado de enviar las órdenes al mercado a través de la plataforma MT5. Admite órdenes de mercado, limitadas o condicionadas, y gestiona la confirmación y el monitoreo del estado de las operaciones.

f. Notificador (Notifier)

Informa al usuario de eventos relevantes como la ejecución de órdenes, detección de anomalías, señales rechazadas o cambios en la exposición. Utiliza canales como Telegram o notificaciones por email.

g. Analizador de sentimiento (Sentiment Analyzer IA)

Procesa textos relacionados con noticias financieras para determinar si el sentimiento general del mercado es positivo, negativo o neutral. Esto permite ajustar la agresividad de la estrategia en función del contexto emocional predominante.

h. Detector de anomalías (Anomaly Detector IA)

Módulo de inteligencia artificial que detecta patrones de comportamiento inusuales en los datos. En este TFM, se implementa mediante un algoritmo de aprendizaje no supervisado (Isolation Forest), que evita operar en condiciones no previstas.

2.5. Fundamentos de inteligencia artificial

La inteligencia artificial (IA) busca dotar a los sistemas informáticos de capacidades que simulan la inteligencia humana, como el aprendizaje, la toma de decisiones o el razonamiento. En particular, su aplicación al análisis financiero ha ganado tracción debido a la gran cantidad de datos generados por los mercados y la necesidad de detectar patrones no evidentes.

2.5.1. Introducción a Machine Learning

Machine Learning (ML) es una subdisciplina de la IA que permite a los sistemas aprender patrones a partir de los datos. Se clasifica en tres grandes categorías:

- Aprendizaje supervisado: el modelo aprende a partir de datos etiquetados (por ejemplo, precios futuros conocidos).
- Aprendizaje no supervisado: detecta patrones o agrupaciones sin etiquetas (clustering, detección de anomalías).
- Aprendizaje por refuerzo: el agente aprende por ensayo y error optimizando una recompensa.

2.5.2. Aprendizaje supervisado vs no supervisado

En el contexto financiero, el aprendizaje supervisado se aplica para predecir tendencias, identificar activos ganadores o construir modelos de scoring. El aprendizaje no

supervisado, en cambio, es útil para detectar cambios de régimen, condiciones atípicas del mercado o relaciones ocultas entre activos.

Este TFM utiliza aprendizaje no supervisado en el módulo de detección de anomalías, ya que no se cuenta con etiquetas explícitas sobre cuándo el mercado se comporta de forma anormal.

2.5.3. Algoritmos aplicados a trading

Algunos algoritmos relevantes aplicados a estrategias algorítmicas incluyen:

- Árboles de decisión y Random Forest
- Support Vector Machines (SVM)
- k-Means y DBSCAN para clustering
- Autoencoders y LSTM para series temporales
- Isolation Forest para detección de outliers

2.5.4. Detección de anomalías en finanzas

Detectar anomalías en datos financieros permite evitar operaciones en momentos de alta incertidumbre o ruido del mercado. Las anomalías pueden ser eventos de alta volatilidad, errores en los datos, movimientos inesperados por noticias o incluso manipulaciones del mercado. Isolation Forest es un algoritmo eficiente y escalable para esta tarea, y ha sido utilizado con éxito en diversas aplicaciones financieras (Liu et al., 2008).⁷

⁷ Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). *Isolation Forest*. Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, 413–422.

2.6. Revisión del estado del arte

2.6.1. Proyectos y plataformas similares

Entre los sistemas existentes destacan QuantConnect, Backtrader, MetaTrader con robots programados en MQL5, y plataformas institucionales como MetaStock o TradeStation. Estas herramientas proporcionan entornos robustos de backtesting, conexión con datos en tiempo real y ejecución automatizada. Sin embargo, presentan ciertas limitaciones:

- Falta de integración nativa con IA: La mayoría de estas plataformas no está diseñada para incorporar de forma sencilla modelos avanzados de machine learning o procesamiento de lenguaje natural.
- Restricciones de personalización: Algunas requieren lenguajes propietarios o entornos cerrados que dificultan la implementación de arquitecturas modulares basadas en eventos como la propuesta en este TFM.
- Orientación al mercado institucional: Aunque son potentes, su coste y complejidad las hacen inaccesibles para desarrolladores independientes o proyectos académicos.

Por el contrario, frameworks como Backtrader o zipline permiten una mayor libertad en la definición de estrategias y la integración de librerías como scikit-learn o transformers, lo que los hace más adecuados para investigaciones con IA aplicada al trading.

2.6.2. Principales avances y limitaciones actuales

En los últimos años se han producido avances significativos en la aplicación de inteligencia artificial al trading financiero:

- Deep learning para series temporales: Modelos como LSTM y transformers temporales se han utilizado para predecir precios futuros o detectar señales de

reversión. Estas redes capturan dependencias no lineales y patrones de largo plazo en datos financieros.

- Procesamiento de lenguaje natural (NLP): Se han entrenado modelos específicos como FinBERT para extraer sentimiento desde noticias o publicaciones económicas, integrando el contexto informativo en las decisiones de trading.
- Aprendizaje por refuerzo: Algoritmos como Deep Q-Learning o Proximal Policy Optimization (PPO) se han explorado para construir agentes que ajustan dinámicamente sus estrategias según el feedback del mercado.

No obstante, persisten limitaciones importantes:

- Generalización y adaptabilidad: Muchos modelos entrenados con datos históricos presentan un rendimiento inferior en condiciones no vistas o cambios de régimen. La sobredependencia de patrones pasados puede generar fallos en mercados altamente dinámicos.
- Interpretabilidad: Las decisiones de modelos complejos, especialmente redes neuronales profundas, son difíciles de explicar. Esto reduce la confianza de los inversores y dificulta el control del riesgo.
- Coste computacional y de datos: El entrenamiento de modelos profundos requiere grandes cantidades de datos etiquetados y recursos computacionales significativos, lo cual no siempre es viable en entornos reales o académicos.

Por ello, en este TFM se ha optado por una IA ligera, explicable y eficiente, integrando modelos como Isolation Forest (para detección de anomalías) y FinBERT (para análisis de sentimiento), que permiten mejorar la toma de decisiones sin añadir complejidad excesiva al sistema.

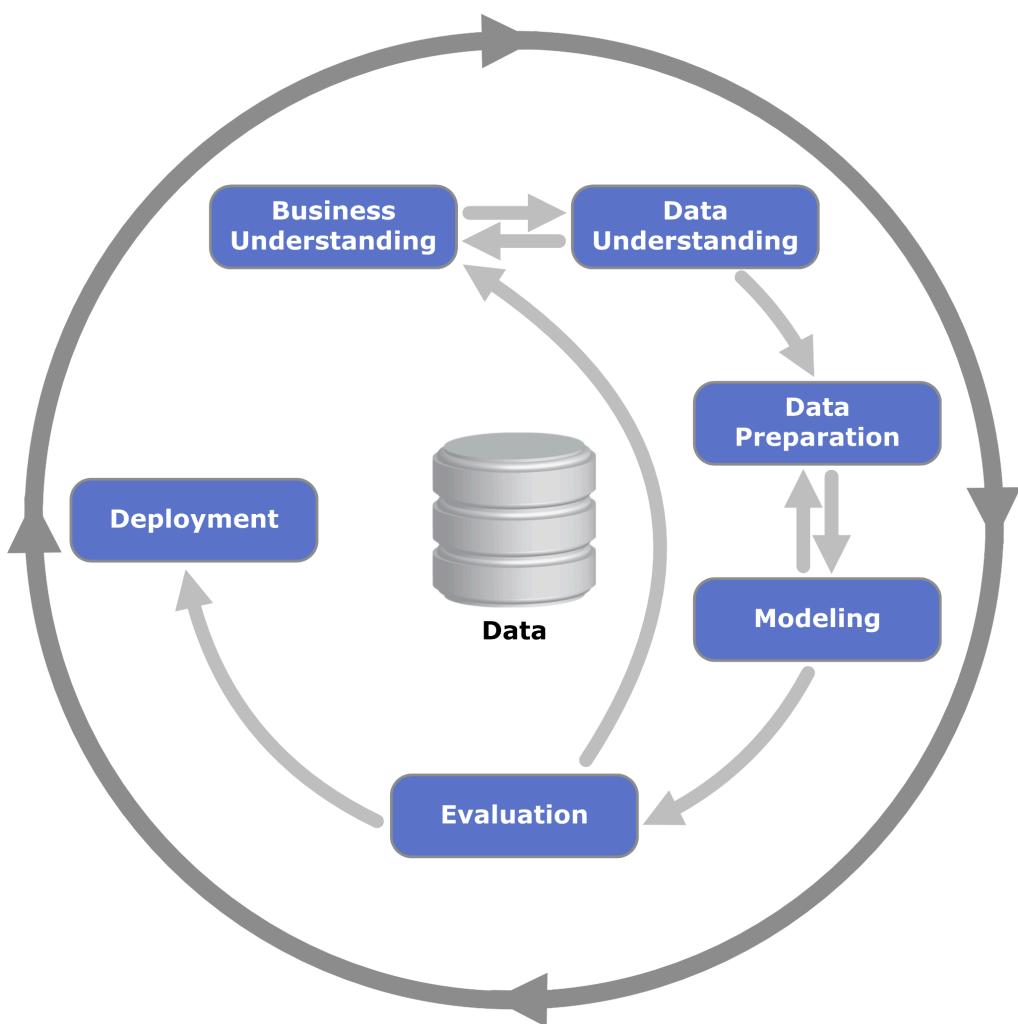
3. Metodología

3.1. Diseño metodológico (CRISP-DM aplicado al TFM)

Para estructurar el desarrollo de este trabajo, se adopta el modelo metodológico CRISP-DM (Cross Industry Standard Process for Data Mining), una de las metodologías más utilizadas en proyectos de ciencia de datos e inteligencia artificial. Este modelo se adapta perfectamente al contexto del TFM, ya que permite un flujo iterativo y flexible entre etapas.

Figura 1

Fases de la Metodología CRISP-DM



Nota. Las seis fases son: comprensión del negocio, comprensión de los datos, preparación de los datos, modelado, evaluación y despliegue.

Las fases aplicadas son:

- Comprensión del negocio: En esta etapa se analiza el objetivo principal del sistema de trading, que es operar el par EUR/USD de forma automatizada, optimizando la rentabilidad ajustada al riesgo mediante el uso de inteligencia artificial.
- Comprensión de los datos: Se realiza un estudio exploratorio de los datos históricos de precios (velas OHLC), volumen, indicadores técnicos y noticias económicas. Esta fase incluye la identificación de variables relevantes para el sistema y el análisis de su comportamiento.
- Preparación de los datos: Se pre procesan los datos para limpiar valores faltantes o atípicos, normalizar series, generar features derivadas y convertir formatos textuales en métricas de sentimiento mediante procesamiento de lenguaje natural (NLP).
- Modelado: Se implementa un algoritmo de detección de anomalías basado en Isolation Forest, que se entrena con datos históricos estables para identificar contextos operativos inusuales. También se integran modelos de sentimiento para modular la toma de decisiones.
- Evaluación: Se realiza una comparación de rendimiento entre dos versiones del sistema: una estrategia técnica base, y otra enriquecida con IA. La evaluación incluye métricas cuantitativas de rentabilidad y riesgo, así como visualización de decisiones.
- Despliegue (simulado): Aunque el sistema no se implementa en producción con dinero real, se ejecuta en un entorno simulado utilizando backtesting sobre datos históricos representativos del mercado EUR/USD.

Este enfoque garantiza una trazabilidad completa del desarrollo, desde la definición del problema hasta la validación de resultados.

3.2. Estrategia de investigación y enfoque computacional

El presente TFM se enmarca en una investigación aplicada con enfoque cuantitativo y experimental. Se parte de un problema práctico real (la mejora del desempeño de un sistema de trading) y se propone una solución técnica basada en el diseño, implementación y validación de un sistema informático.

El enfoque computacional adoptado combina elementos de:

- Análisis algorítmico: desarrollo de reglas técnicas de entrada y salida.
- Aprendizaje automático no supervisado: aplicación de modelos de detección de anomalías.
- Procesamiento de lenguaje natural (NLP): extracción de sentimiento desde fuentes de texto económico.
- Simulación y evaluación: uso de técnicas de backtesting para probar hipótesis de forma controlada.

Este enfoque mixto busca no solo construir un sistema funcional, sino también evaluar científicamente su rendimiento frente a versiones sin inteligencia artificial.

3.3. Herramientas y lenguajes de desarrollo (Python, MT5 API, scikit-learn)

El sistema ha sido desarrollado en Python, debido a su versatilidad, amplio ecosistema científico y compatibilidad con librerías de trading e inteligencia artificial. A continuación, se detallan las herramientas utilizadas:

- Lenguaje: Python 3.10+

- Entorno de ejecución: Jupyter Notebook y scripts .py modulares
- Plataforma de conexión: MetaTrader 5 (MT5) mediante la librería MetaTrader5
- Visualización: Matplotlib, Seaborn, Plotly
- Machine Learning: Scikit-learn (Isolation Forest), Pandas, Numpy
- Procesamiento de texto (NLP): Transformers, Hugging Face
- Notificaciones: Python-telegram-bot
- Control de versiones: Git y GitHub para trazabilidad del desarrollo

Esta combinación de herramientas permite cubrir todo el ciclo de vida del sistema: desde la adquisición de datos hasta el análisis de resultados, pasando por la ejecución de órdenes simuladas.

3.4. Validación de resultados y métricas utilizadas

La evaluación del sistema se realiza mediante backtesting, simulando operaciones con datos históricos del EURUSD en diferentes contextos de mercado (tendencias alcistas, bajistas y laterales). Para cuantificar el rendimiento se utilizan métricas ampliamente aceptadas en el ámbito financiero y de IA:

Métricas de rendimiento del sistema de trading:

- Retorno total (%)
- Retorno anualizado
- Máximo drawdown (%)
- Número de operaciones ganadoras/perdedoras
- Ratio de operaciones ganadoras

Comparación de resultados con y sin IA: se evalúa la diferencia de rendimiento al activar o desactivar los módulos de IA, especialmente en condiciones volátiles otras noticias económicas.

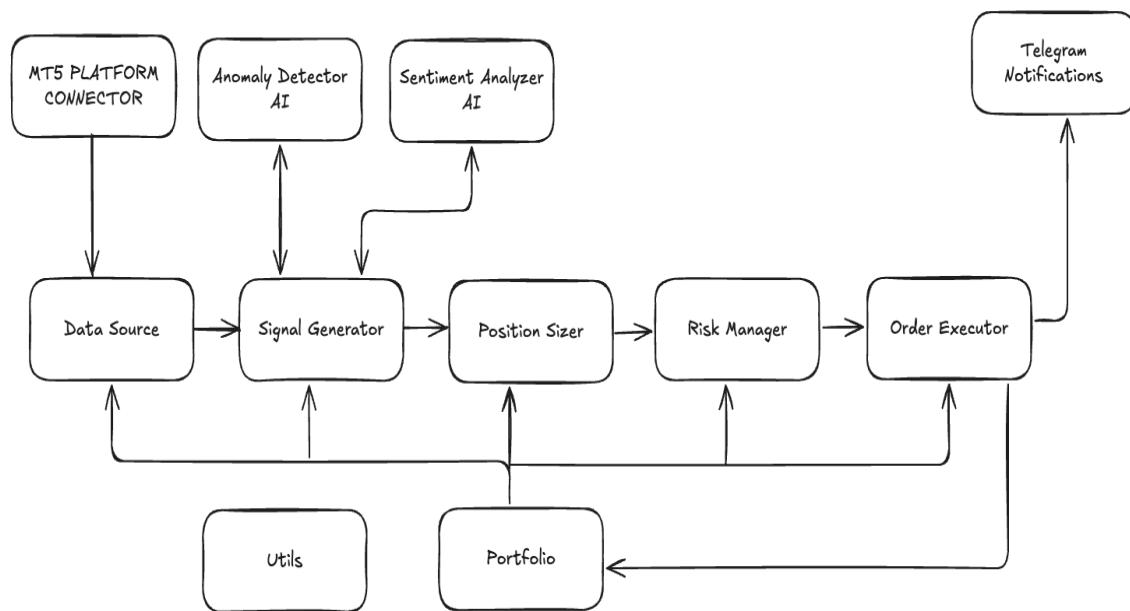
Además, se incluyen visualizaciones de equity curves (curvas de capital), histogramas de resultados y gráficos de dispersión para interpretar la evolución temporal y la consistencia de la estrategia.

4. Diseño e implementación del sistema

El sistema desarrollado se basa en una arquitectura modular orientada a objetos, lo que facilita su mantenibilidad, escalabilidad y reutilización. Se trata de una solución programada íntegramente en Python, conectada en tiempo real con la plataforma MetaTrader 5 (MT5), y con componentes de inteligencia artificial integrados para la toma de decisiones informadas y seguras.

Figura 2

Diagrama de arquitectura del sistema de trading



Nota. El diagrama representa toda la arquitectura usada en el sistema de trading en vivo y en backtesting.

4.1. Requerimientos del sistema

Para el correcto diseño e implementación del sistema de trading automatizado, es necesario establecer un conjunto claro de requisitos funcionales y no funcionales, que guíen tanto la arquitectura como la lógica del desarrollo.

Los requisitos funcionales definen qué debe hacer el sistema, es decir, las funcionalidades específicas que debe implementar para cumplir sus objetivos. En este caso, se trata de tareas como obtener datos de mercado, generar señales de trading, evaluar el riesgo, ejecutar órdenes, y aplicar modelos de inteligencia artificial.

Los requisitos no funcionales, por otro lado, especifican cómo debe comportarse el sistema en términos de calidad, rendimiento, estructura y mantenibilidad. Estos requisitos no se refieren directamente a funciones, sino a atributos como la modularidad, eficiencia en tiempo de ejecución, legibilidad del código o capacidad para realizar pruebas sobre datos históricos.

A continuación, se detallan ambos tipos de requisitos según las necesidades del sistema propuesto.

Requisitos funcionales:

- Obtener datos del par EUR/USD en tiempo real desde MetaTrader 5.
- Calcular indicadores técnicos y generar señales de compra o venta.
- Determinar el tamaño de las posiciones y ejecutar órdenes de manera automatizada.
- Evaluar el riesgo y filtrar operaciones según criterios predefinidos.

- Detectar condiciones anómalas del mercado mediante un modelo de IA.
- Integrar un módulo de análisis de sentimiento que influya en la toma de decisiones.
- Notificar al usuario sobre las operaciones y alertas a través de Telegram.

Requisitos no funcionales:

- Modularidad en el código (cada componente aislado y reemplazable).
- Tiempos de ejecución eficientes para análisis en tiempo real.
- Alta legibilidad del código, con comentarios y documentación.
- Capacidad de ser probado en backtesting sobre datos históricos.
- Trazabilidad de decisiones y registros (logs de operaciones y eventos).

4.2. Diseño modular de componentes

La arquitectura del sistema se estructura en módulos independientes, cada uno con una responsabilidad clara. Esta separación por capas permite escalar el sistema sin interferir con el resto de componentes.

Módulos principales:

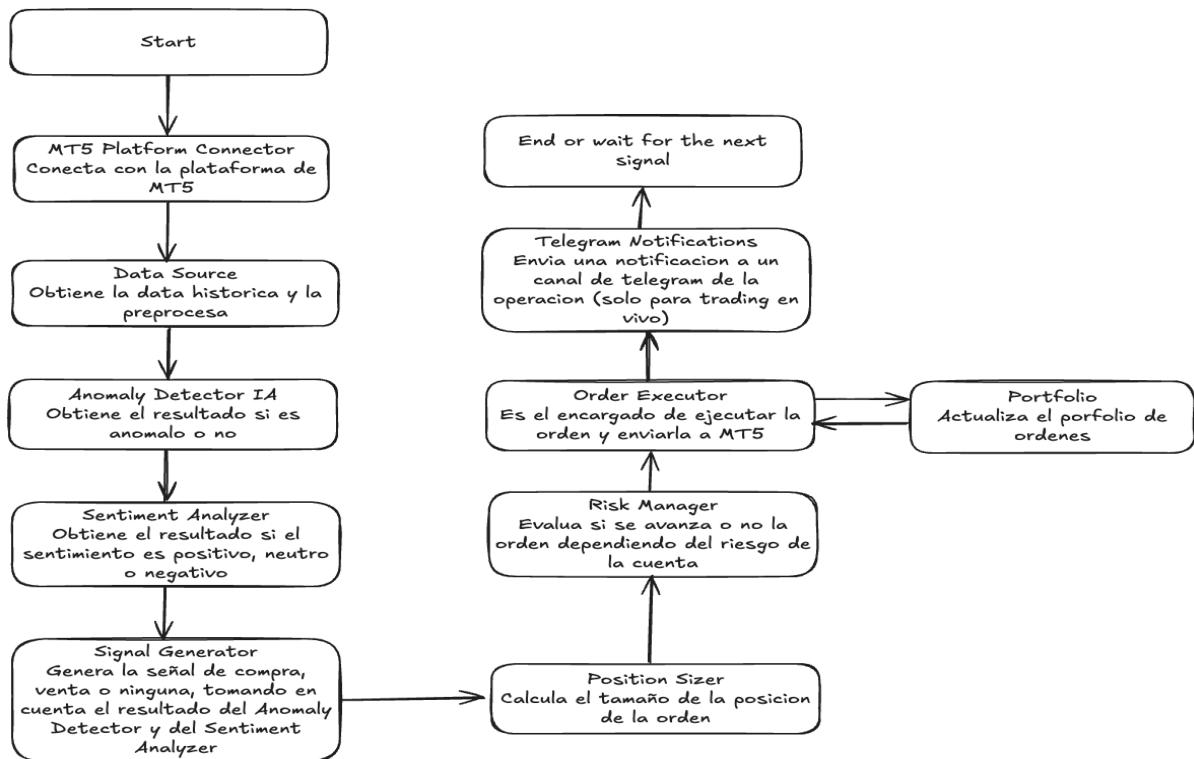
- Data Source: gestiona la conexión a MetaTrader 5 y la obtención de datos de mercado (precios OHLC, volumen, etc.).
- Strategy Manager: contiene la lógica algorítmica para generar señales basadas en indicadores técnicos (por ejemplo, cruce de medias móviles).
- Position Sizer: determina el tamaño de cada posición en función del riesgo permitido.
- Risk Manager: filtra señales que superen los límites de exposición o no cumplan condiciones de seguridad.
- Order Executor: ejecuta operaciones a través de la API de MT5 y monitoriza su estado.

- Anomaly Detector: modelo de machine learning que evalúa si el contexto de mercado es anómalo.
- Sentiment Analyzer: procesa texto económico para obtener un indicador de sentimiento.
- Notifier: gestiona alertas al usuario mediante mensajería automática (Telegram).
- Portfolio Manager: monitoriza las operaciones abiertas, el capital disponible y las estadísticas de rentabilidad.

Cada uno de estos módulos se comunica mediante objetos y eventos definidos, lo que permite intercambiar implementaciones fácilmente (por ejemplo, sustituir una estrategia técnica por otra sin modificar el flujo general del sistema). La Figura 3 presenta el diagrama de flujo detallado de este sistema propuesto.

Figura 3

Diagrama de flujo del sistema de trading



Nota. El diagrama representa el flujo de ejecución del sistema de trading algorítmico propuesto. Desde la conexión inicial con MetaTrader 5 hasta la ejecución y notificación de operaciones, se muestra la interacción secuencial entre los módulos principales: adquisición y preprocesamiento de datos, detección de anomalías, análisis de sentimiento, generación de señales, evaluación del riesgo, dimensionamiento de posición, ejecución de órdenes y actualización del portafolio.

4.3. Flujo de datos y eventos

El flujo general del sistema sigue una secuencia ordenada que puede describirse en los siguientes pasos:

- Inicialización: El sistema carga los parámetros de configuración, conecta con MT5 y prepara el entorno de trabajo.
- Actualización de datos: El Data Source obtiene el último bloque de datos del EUR/USD, incluyendo precios e indicadores.
- Generación de señal: El Strategy Manager evalúa si existe una señal de compra o venta según la estrategia activa.
- El Anomaly Detector valida si el contexto de mercado es normal.
- El Sentiment Analyzer consulta y analiza noticias económicas relevantes.
- Gestión de riesgos: El Risk Manager evalúa si la operación es aceptable según el perfil de riesgo definido.
- Cálculo de tamaño: Si la señal es válida, el Position Sizer calcula el tamaño óptimo de la operación.
- Ejecución: El Order Executor envía la orden al mercado vía MT5.
- Notificación: El Notifier informa al usuario sobre la acción tomada por la api de telegram.

- Monitoreo: El Portfolio Manager actualiza el estado de las operaciones abiertas y calcula métricas de rendimiento.

Este flujo puede representarse mediante un diagrama de secuencia o de componentes (Como podemos ver en la Figura 2).

4.4. Interacción con MetaTrader 5

La interacción con el broker se realiza mediante la librería MetaTrader 5 para Python, que proporciona acceso directo al terminal de MT5 instalado localmente.

Funciones clave utilizadas:

- `mt5.initialize()` – Establece la conexión con el terminal.
- `mt5.symbol_info_tick()` – Obtiene el último precio del símbolo.
- `mt5.copy_rates_from_pos()` – Descarga series de tiempo históricas (OHLC).
- `mt5.order_send()` – Envía una orden de mercado o pendiente.
- `mt5.positions_get()` – Consulta las posiciones abiertas.
- `mt5.shutdown()` – Finaliza la conexión de manera segura.

El sistema ha sido diseñado para lanzar sesiones independientes de trading sin comprometer otras cuentas conectadas al mismo terminal.

4.5. Documentación del código y arquitectura

El proyecto ha sido documentado mediante comentarios dentro del código fuente, siguiendo las buenas prácticas de desarrollo en Python (PEP-8 y docstrings). Cada módulo contiene:

- Una descripción de su funcionalidad general.
- Interfaces públicas (métodos disponibles para otros módulos).

- Dependencias externas utilizadas (por ejemplo, sklearn, pandas, MetaTrader 5).
- Ejemplos de uso en formato notebook (cuando aplica).

Además, se han creado diagramas de arquitectura del sistema, incluyendo:

Diagrama de módulos: muestra los componentes y sus interacciones (Como podemos ver en la Figura 1).

Estos recursos están incluidos en el github con todo el código para facilitar su comprensión por parte del tribunal evaluador.

5. Procesamiento y análisis de datos

Uno de los pilares fundamentales de cualquier sistema de trading algorítmico basado en inteligencia artificial es la calidad y el tratamiento de los datos. Las decisiones que toma el sistema dependen en gran medida de cómo se procesan y presentan las variables de entrada. En este capítulo se describe el proceso completo de preparación de los datos del mercado para su uso por los modelos técnicos y de IA.

5.1. Recolección de datos de mercado

La fuente principal de datos utilizada en este proyecto es la plataforma MetaTrader 5, accedida mediante su API oficial para Python (MetaTrader5). A través de esta interfaz, se recopilan los datos históricos del par EUR/USD, incluyendo precios y volumen, con la siguiente estructura:

- Open: precio de apertura
- High: precio máximo
- Low: precio mínimo
- Close: precio de cierre

- Tick volume: número de transacciones dentro de la vela
- Time: timestamp de cada vela (datetime)
- Frecuencia de los datos: Intervalo temporal de 1 hora (H1), seleccionado por ser un marco temporal adecuado para estrategias de corto-medio plazo y lo suficientemente representativo para evaluar el comportamiento del modelo.
- Período: desde enero hasta diciembre 2023, para asegurar diversidad de escenarios macroeconómicos y evitar sobreajuste en condiciones particulares.
- Formato: Los datos se almacenan en estructuras de tipo DataFrame (pandas) con el timestamp como índice, permitiendo un fácil manejo y manipulación para análisis técnico, modelado y visualización.

5.2. Limpieza y preprocesamiento de series temporales

Dado que los datos financieros pueden contener errores, lagunas o irregularidades, se realiza un proceso de preprocesamiento para garantizar su integridad y consistencia antes de alimentar los modelos del sistema.

Etapas del preprocesamiento:

- Verificación de duplicados: se eliminan registros con timestamps repetidos o inconsistencias en la secuencia temporal.
- Interpolación de valores faltantes: en caso de datos faltantes por desconexión o días sin actividad (por ejemplo, fines de semana), se realiza un relleno inteligente (por ejemplo, forward fill), especialmente importante para indicadores como las medias móviles.
- Conversión de timestamp: se ajusta el índice temporal a formato datetime y se transforma a la zona horaria GMT+0 para mantener consistencia.

- Normalización y escalado: para modelos como el detector de anomalías, algunas variables (como volumen o desviación estándar) se normalizan usando MinMaxScaler o StandardScaler para evitar dominancia numérica.
- Filtrado de extremos: se identifican posibles outliers que puedan distorsionar la lógica del sistema (por ejemplo, spikes por errores del broker) mediante reglas estadísticas (z-score) o inspección visual.

Este tratamiento asegura que los datos de entrada estén limpios y estructurados para su uso tanto en estrategias deterministas como en modelos de IA.

5.3. Extracción de características e ingeniería de variables

La ingeniería de características consiste en transformar los datos brutos en variables informativas y representativas que mejoren el rendimiento del sistema de trading.

Variables generadas:

- Indicadores técnicos clásicos: SMA(10), SMA(20) medias móviles simples.

Variables de contexto para IA:

- Sentimiento textual agregado: valor en texto de positivo, neutral o negativo, derivado de análisis de noticias.
- Flag de anomalía (booleano): indicador binario generado por el modelo Isolation Forest sobre las features técnicas.

Estas variables enriquecen el conjunto de datos y permiten que tanto los módulos técnicos como los modelos de machine learning puedan identificar condiciones del mercado, evaluar riesgos y tomar decisiones fundamentadas.

6. Integración de modelos de IA

La incorporación de inteligencia artificial al sistema de trading tiene como objetivo mejorar su capacidad de adaptación al entorno, reducir riesgos en condiciones adversas y añadir una capa de contexto informativo. En este proyecto se han integrado dos modelos principales: un detector de anomalías basado en machine learning y un analizador de sentimiento derivado de procesamiento de lenguaje natural (NLP).

6.1. Implementación del detector de anomalías

Las condiciones del mercado pueden presentar patrones inusuales que escapan al alcance de los indicadores técnicos tradicionales. Estas anomalías pueden ser resultado de eventos inesperados, errores en los datos, intervenciones económicas o cambios abruptos en la volatilidad. Para evitar operaciones en estos contextos, se ha desarrollado un módulo basado en aprendizaje no supervisado.

6.1.1. Algoritmo Isolation Forest

Este apartado describe el modelo de detección de anomalías utilizado en el sistema de trading. El modelo seleccionado es Isolation Forest, un algoritmo de aprendizaje no supervisado especializado en detectar outliers o comportamientos atípicos en grandes volúmenes de datos.

¿Cómo funciona Isolation Forest?

- No calcula densidades ni distancias (como otros métodos), sino que aísla observaciones anómalas dividiendo aleatoriamente el espacio de datos mediante árboles binarios.
- Las observaciones que se aíslan más rápidamente (menos divisiones) son consideradas anómalas.

- Esto lo hace muy eficiente, incluso en conjuntos de datos grandes.

Ventajas del modelo:

- Escalable a grandes conjuntos de datos
- Rápido entrenamiento y predicción
- Capacidad para detectar múltiples tipos de anomalías simultáneamente
- No requiere datos etiquetados

Implementación técnica:

- Librería: sklearn.ensemble.IsolationForest
- Parámetros utilizados:
 - n_estimators=100
 - contamination=0.02 (2% de datos esperados como anomalías)
 - random_state=42
- Variables de entrada: combinación de features técnicas (open, high, low, close, tickvol, vol, spread)
- Salida: la puntuación de anomalía siendo ésta un valor entre -1 y 1, y una etiqueta binaria (1 = normal, -1 = anómalo)

Este modelo se entrena una vez con datos históricos estables y se utiliza en producción para evaluar, vela a vela, si el contexto actual justifica tomar decisiones de trading o abstenerse.

6.1.2. Umbrales y evaluación del modelo

Aquí se explica cómo se evalúa y ajusta el modelo para que sea útil en el sistema de trading.

a. Análisis de la distribución de las puntuaciones de anomalía

- Aunque Isolation Forest ya marca valores anómalos usando el parámetro contamination, el autor analiza la distribución de puntuaciones manualmente.
- Esto permite ajustar el umbral de decisión de forma más precisa, visualizando los datos (por ejemplo, con histogramas).

b. Validación cualitativa con eventos históricos

Se verifica si el modelo marca como anómalos momentos conocidos de alta volatilidad, como:

- Brexit
- Pandemia COVID-19
- Intervenciones de bancos centrales

Esto ayuda a comprobar que el modelo tiene sentido en la práctica.

c. Impacto en la estrategia de trading

- El modelo actúa como un filtro: si detecta una anomalía, se evita operar.
- Esto reduce la frecuencia operativa durante condiciones adversas y mejora la seguridad.

d. Ajuste del umbral

En este caso, el umbral final elegido es 0.0595, lo que corresponde al 2% de las observaciones más extremas.

Es un compromiso entre:

- No filtrar demasiadas operaciones.
- Pero sí proteger contra riesgos extremos.

e. Visualización

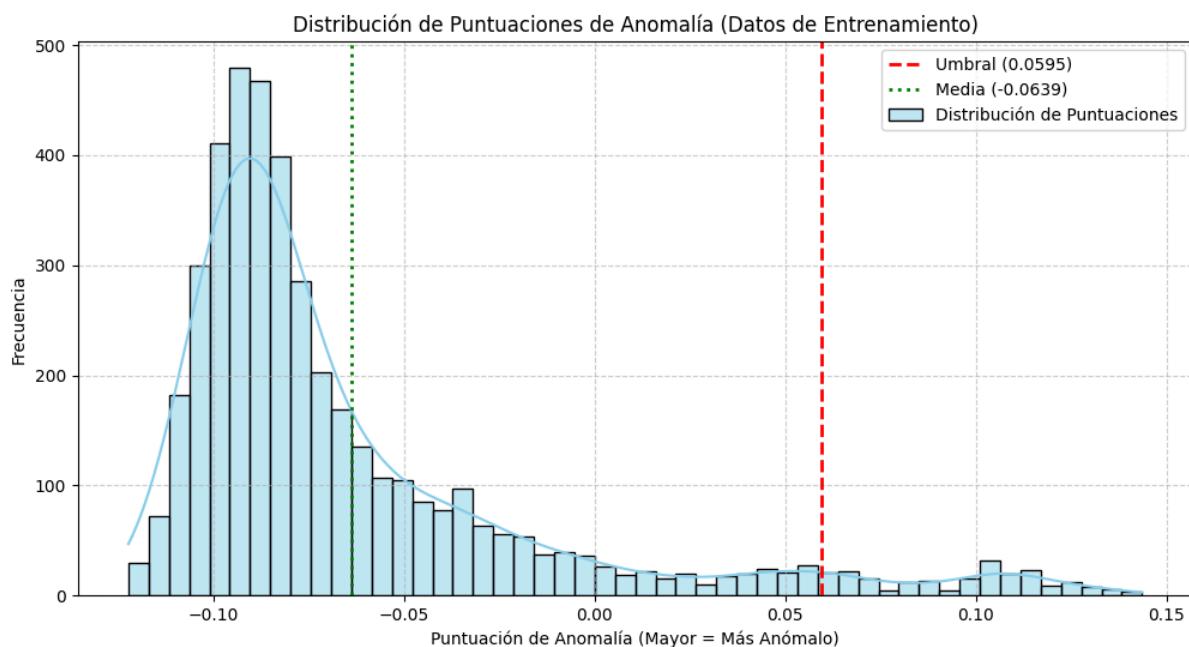
Se incluye un gráfico que muestra la distribución de las puntuaciones de anomalía y el corte visual del umbral.

Esto permite interpretar el comportamiento del modelo y justificar su inclusión.

Tomando en cuenta los valores obtenidos después de entrenar el modelo, podemos observar en la Figura 4 la distribución de puntuaciones de anomalía y el valor del umbral que determina si es anómalo o no.

Figura 4

Distribución de puntuaciones de anomalía en el entrenamiento



Nota. En este gráfico se puede observar tanto la media como el umbral de detección de datos anómalos.

6.2. Implementación del analizador de sentimiento

Los mercados no se mueven únicamente por razones técnicas o fundamentales cuantificables. El sentimiento del mercado derivado de noticias, declaraciones y expectativas tiene un impacto directo en los precios. Incorporar esta dimensión permite tomar decisiones más informadas y adaptativas.

6.2.1. Análisis de titulares financieros o métricas del mercado

El sistema obtiene titulares financieros de fuentes como:

- Noticias económicas (por ejemplo, newsapi para el trading en vivo) y para el backtesting se utiliza un conjunto de datos anotado de noticias financieras de Forex, diseñado específicamente para el análisis de sentimiento. Este dataset incluye texto de noticias y su correspondiente polaridad de sentimiento (positivo, negativo o neutral), lo que lo hace idóneo para la tarea de clasificación. El recurso fue desarrollado y puesto a disposición por Fatouros (2023).
- Calendarios económicos con resúmenes de eventos
- Base de datos precompilada de titulares históricos etiquetados (para entrenamiento y backtesting)

Estos textos son procesados usando un modelo NLP pre entrenado:

Prosus AI. (n.d.). finbert [modelo pre entrenado]. Hugging Face.

<https://huggingface.co/ProsusAI/finbert>

El modelo ProsusAI/finbert es un transformador pre-entrenado específicamente para el análisis de sentimiento en el dominio financiero. A diferencia de modelos de lenguaje generales como BERT, RoBERTa o incluso GPT-3/4, FinBERT ha sido ajustado

(fine-tuned) en un corpus masivo de texto financiero, lo que le otorga una comprensión matizada del lenguaje y la terminología específicos de este sector.

Para este estudio se utilizó el análisis de todas las noticias ocurridas 1 semana antes del día actual de la toma de decisión, ya que es un tiempo promedio en que las noticias hacen efecto en los mercados.

Proceso de inferencia:

- Tokenización del texto
- Evaluación de polaridad: positiva, neutral o negativa
- Agregación de sentimiento para generar un índice de mercado [-1, 1]

Este índice se actualiza cada semana y se añade como variable adicional en la toma de decisiones.

6.2.2. Efecto sobre la toma de decisiones

El índice de sentimiento afecta el sistema en dos formas:

Modulación de señales técnicas: si el sentimiento es negativo y la señal es de compra, se ignora la señal; si es coherente (sentimiento positivo + señal de compra), se refuerza la decisión.

Prevención de entrada en contexto adverso: se define un umbral de sentimiento (por ejemplo, < -0.5) bajo el cual se evita operar, incluso si la señal técnica es positiva.

Esto permite que el sistema adopte un enfoque más cauteloso en momentos de pánico informativo o euforia irracional, dos fuentes típicas de volatilidad extrema en el FOREX.

6.3. Integración con el generador de señales

Ambos módulos se integran en el flujo de decisión antes de ejecutar cualquier operación.

La lógica condicional puede resumirse como:

Tabla 1

Lógica de acción del signal generator

Detector de Anomalías	Análisis de Sentimiento	Acción del Signal Generator
Normal	Positivo	Se ejecuta la señal si es de compra. Se ignora la señal si es de venta
Normal	Negativo	Se ignora la señal si es de compra. Se ejecuta la señal si es de venta.
Normal	Neutral	Se ejecuta la señal solo si tiene un fuerte respaldo técnico.
Anómalo	Cualquiera	Se bloquea la señal. No se ejecuta ninguna operación.
No disponible	No disponible	Se aplica lógica técnica pura sin filtros IA.

Nota. La tabla muestra las decisiones tomadas por el generador de señales en función de la detección de anomalías en el mercado y del análisis de sentimiento extraído de noticias financieras. Solo cuando ambos módulos indican condiciones favorables, la señal técnica

se ejecuta. En caso contrario, la operación es descartada o bloqueada para reducir el riesgo operativo.

Descripción de la lógica

- Detector de anomalías (Isolated Forest Anomaly Detector)
 - Si detecta una anomalía (anomaly score > threshold), se considera que el contexto de mercado es inusual.
 - En este caso, el Strategy Manager ignora toda señal generada por la estrategia, incluso si es técnica y válida.
- Analizador de sentimiento (FinBERT)
 - Devuelve una polaridad de mercado: positiva, negativa o neutral.
 - Si la señal técnica y el sentimiento están alineados (por ejemplo, señal de compra + sentimiento positivo o señal de venta + sentimiento negativo), se refuerza la ejecución.
 - Si están desalineados (por ejemplo, señal de compra + sentimiento negativo), se descarta la operación.
 - En caso de sentimiento neutral, se evalúa con menor prioridad, exigiendo una señal técnica fuerte o un contexto no anómalo.
- Lógica compuesta
 - La señal solo se ejecuta si no hay anomalía detectada, el sentimiento va acorde al tipo de señal y la señal técnica es válida.

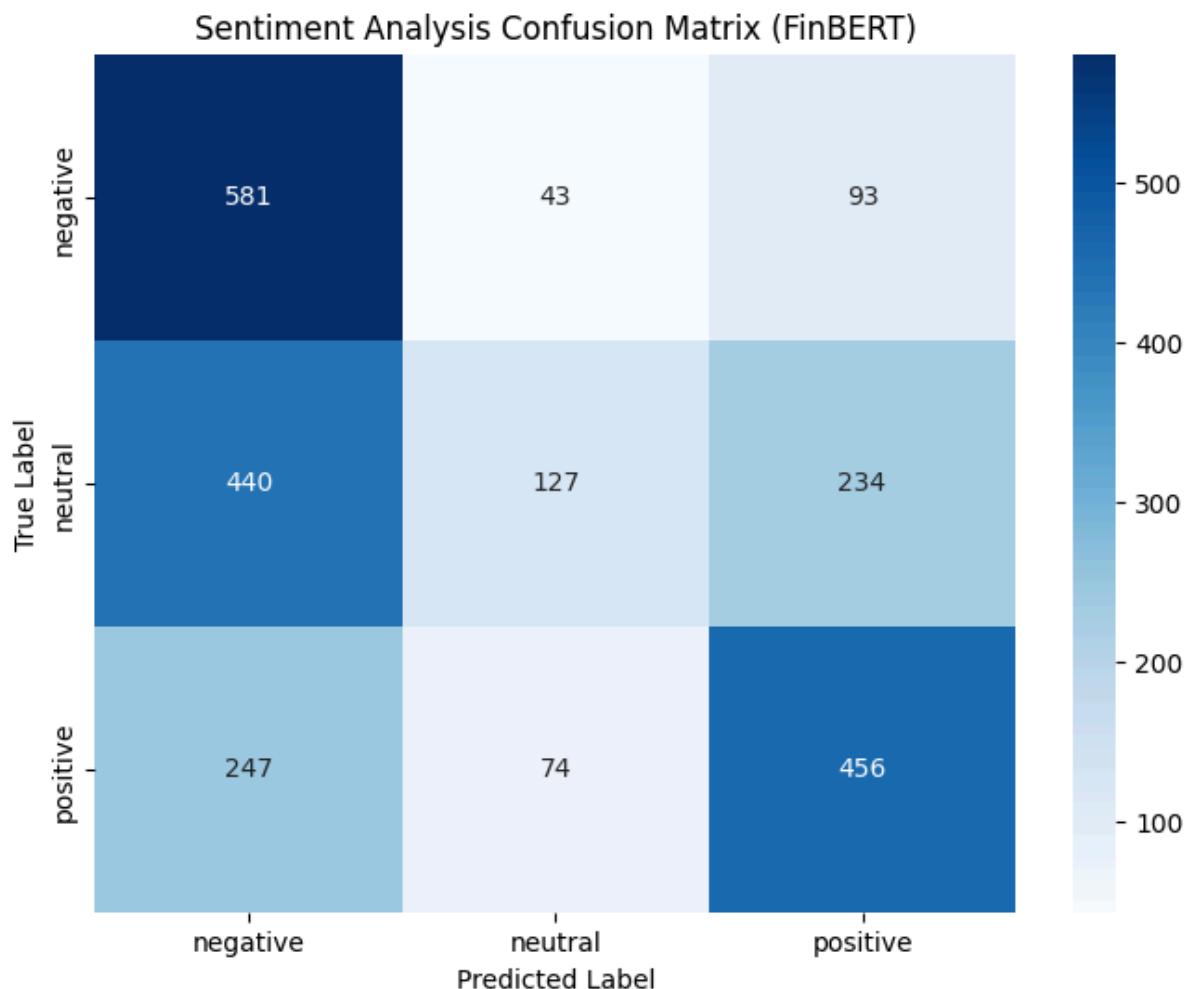
6.3.1. Matriz de confusión y informe de clasificación

Las filas de la figura 5 representan las etiquetas verdaderas (True Label) y las columnas representan las etiquetas predichas (Predicted Label).

- Fila 1: Sentimiento "negative" (Real)
 - 581: El modelo predijo correctamente 581 casos como "negative" cuando realmente eran "negative" (Verdaderos Negativos - TN).
 - 43: El modelo predijo 43 casos como "neutral" cuando realmente eran "negative" (Falsos Neutrales).
 - 93: El modelo predijo 93 casos como "positive" cuando realmente eran "negative" (Falsos Positivos para la clase "positive", pero Falsos Negativos para la clase "negative" - Tipo II error).
- Fila 2: Sentimiento "neutral" (Real)
 - 440: El modelo predijo 440 casos como "negative" cuando realmente eran "neutral" (Falsos Negativos para la clase "neutral").
 - 127: El modelo predijo correctamente 127 casos como "neutral" cuando realmente eran "neutral" (Verdaderos Neutrales).
 - 234: El modelo predijo 234 casos como "positive" cuando realmente eran "neutral" (Falsos Positivos para la clase "positive", Falsos Negativos para la clase "neutral").
- Fila 3: Sentimiento "positive" (Real)
 - 247: El modelo predijo 247 casos como "negative" cuando realmente eran "positive" (Falsos Negativos para la clase "positive").
 - 74: El modelo predijo 74 casos como "neutral" cuando realmente eran "positive" (Falsos Neutrales).
 - 456: El modelo predijo correctamente 456 casos como "positive" cuando realmente eran "positive" (Verdaderos Positivos - TP).

Figura 5

Matriz de confusión del análisis de sentimiento



Nota. La figura representa la matriz de confusión del modelo FinBERT aplicado al análisis de sentimiento financiero. En el eje vertical se indican las etiquetas reales y en el eje horizontal las predicciones del modelo.

Observaciones de la Matriz de Confusión

- Clase "negative": El modelo es bastante bueno identificando sentimientos negativos (581 correctos). Sin embargo, confunde un número significativo de casos negativos con neutrales (43) y, especialmente, con positivos (93). Esto podría indicar que

algunos textos negativos son difíciles de distinguir de los neutrales o incluso de los que el modelo considera positivos.

- Clase "neutral": Esta es la clase donde el modelo tiene más dificultades. De los 801 casos neutrales reales ($440 + 127 + 234$), solo 127 fueron clasificados correctamente. El modelo tiende a clasificar erróneamente los neutrales como negativos (440) o positivos (234) en una proporción mucho mayor que la predicción correcta. Esto es un problema común en el análisis de sentimiento, ya que la "neutralidad" puede ser ambigua o estar sub representada en el conjunto de entrenamiento.
- Clase "positive": El modelo también tiene un rendimiento decente para la clase positiva (456 correctos), pero también confunde un número considerable de positivos con negativos (247) y, en menor medida, con neutrales (74).

Total de ítems evaluados: $581+43+93 + 440+127+234 + 247+74+456 = 2295$.

El informe de clasificación proporciona métricas clave para cada clase (precisión, recall, f1-score) y promedios generales.

Tabla 2

Reporte de clasificación del análisis de sentimiento

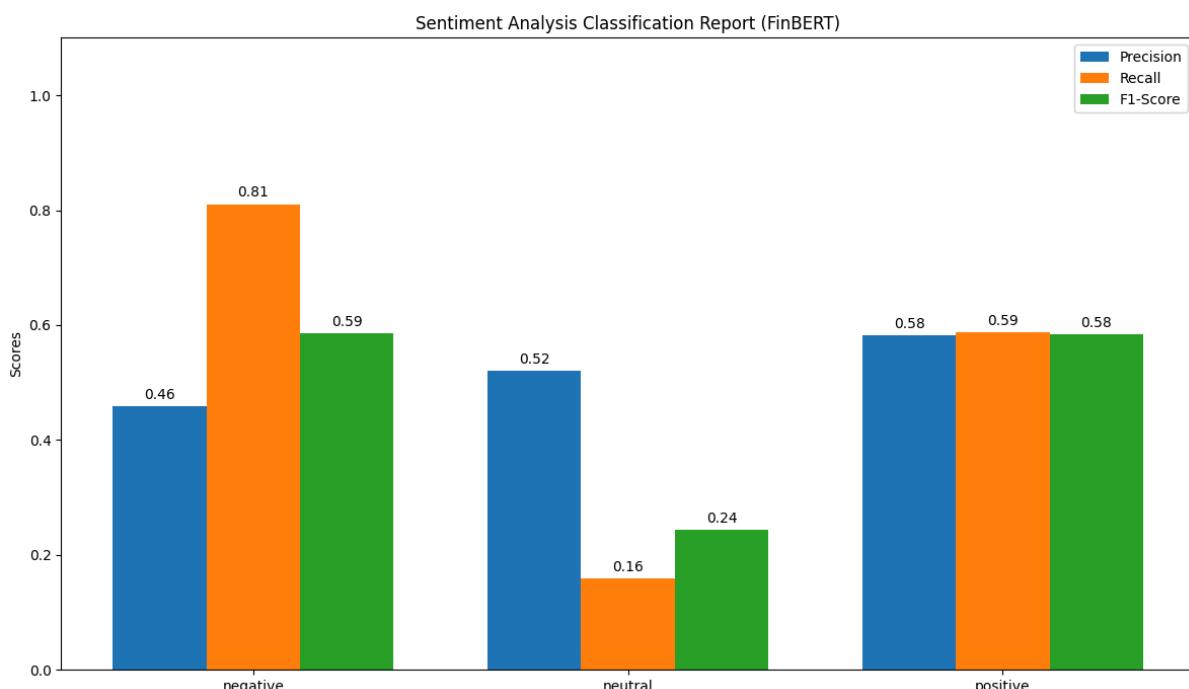
	precisión	recall	f1-score	support
negative	0.458202	0.810321	0.585390	717
neutral	0.520492	0.158552	0.243062	801
positive	0.582375	0.586873	0.584615	777
accuracy	0.507190	0.507190	0.507190	0.50719

Nota. La tabla presenta las métricas de rendimiento del modelo FinBERT en el análisis de sentimiento. La precisión indica la proporción de verdaderos positivos entre las

predicciones positivas. El recall refleja la capacidad del modelo para identificar correctamente las instancias reales de cada clase. El F1-score representa la media armónica entre precisión y recall. La clase “neutral” obtuvo los resultados más bajos, lo que sugiere dificultad del modelo para distinguir textos sin polaridad clara.

Figura 6

Gráfica del reporte de clasificación del análisis de sentimiento



Nota. La figura ilustra las métricas de evaluación del modelo FinBERT para cada clase de sentimiento (negativo, neutral, positivo), incluyendo precisión, recall y F1-score. Se observa un rendimiento significativamente más bajo en la clase “neutral”, especialmente en la métrica de recall, lo que indica dificultades del modelo para identificar correctamente textos sin polaridad marcada. Las clases “negativa” y “positiva” muestran resultados más equilibrados.

- **Support (Soporte):** El número real de ocurrencias de cada clase en el conjunto de datos de prueba.
 - Negative: 717
 - Neutral: 801
 - Positive: 777
- **Precisión (Precisión):** De todas las instancias que el modelo predijo como una clase, ¿cuántas fueron realmente de esa clase?
 - **Negative:** **0.46** (De todos los que el modelo dijo que eran negativos, solo el 46% realmente lo eran). Esto es bajo, lo que significa que el modelo tiene bastantes "falsos positivos" para la clase negativa (es decir, clasifica otras cosas como negativas erróneamente).
 - **Neutral:** **0.52** (De todos los que el modelo dijo que eran neutrales, el 52% realmente lo eran). Un poco mejor que el negativo, pero aún indica que hay bastantes predicciones neutrales incorrectas.
 - **Positive:** **0.58** (De todos los que el modelo dijo que eran positivos, el 58% realmente lo eran). Es la clase con mayor precisión, indicando que sus predicciones positivas son más fiables.
- **Recall (Exhaustividad/Sensibilidad):** De todas las instancias que realmente pertenecen a una clase, ¿cuántas fueron correctamente identificadas por el modelo?
 - **Negative:** **0.81** (El modelo identificó el 81% de todos los sentimientos negativos reales). Este es un valor alto y bueno, lo que significa que el modelo es muy bueno "encontrando" los verdaderos negativos.
 - **Neutral:** **0.16** (El modelo solo identificó el 16% de todos los sentimientos neutrales reales). Este es un recall extremadamente bajo, confirmando lo que

vimos en la matriz de confusión: el modelo tiene serios problemas para detectar textos neutrales.

- **Positive: 0.59** (El modelo identificó el 59% de todos los sentimientos positivos reales). Un recall moderado, indicando que pierde casi la mitad de los positivos reales.
- **F1-Score:** Es la media armónica de la precisión y el recall. Es una métrica útil cuando se busca un equilibrio entre precisión y recall, y es especialmente relevante en casos de clases desequilibradas.
 - **Negative: 0.59**
 - **Neutral: 0.24** (Muy bajo, debido al bajo recall).
 - **Positive: 0.58**
- **Accuracy (Exactitud): 0.507190**
 - El modelo clasificó correctamente aproximadamente el 50.7% de todos los ítems. Esto es un valor relativamente bajo para un modelo de clasificación, especialmente considerando que hay 3 clases. Una exactitud del 50% significa que está un poco mejor que una clasificación aleatoria en un problema con 3 clases balanceadas. Sin embargo, dado el desequilibrio y la dificultad de la clase neutral, este número debe interpretarse junto con las métricas por clase.

Conclusiones para el modelo FinBERT

- **Problema con la clase Neutral:** Claramente, el mayor punto débil del modelo FinBERT es la clasificación de textos neutrales. Su recall para esta clase es extremadamente bajo (0.16), lo que significa que la mayoría de los textos neutrales no son identificados como tales. En cambio, son clasificados erróneamente como negativos o positivos. Esto podría deberse a:

- **Ambigüedad intrínseca de los datos neutrales:** Los textos financieros neutrales pueden ser difíciles de distinguir de los que tienen un ligero sesgo positivo o negativo.
 - **Desequilibrio de clases:** Aunque el soporte es relativamente similar, es posible que los ejemplos "neutrales" sean menos distintivos o más difíciles de aprender para el modelo.
 - **Calidad del etiquetado:** Es posible que los datos de entrenamiento para la clase neutral sean inconsistentes o contengan ruido.
- **Trade-off entre Precision y Recall:**
 - Para la clase "**negative**": Alta recall (encuentra la mayoría de los negativos reales) pero baja precisión (muchas de sus predicciones negativas son incorrectas). El modelo tiende a ser "demasiado agresivo" al clasificar algo como negativo.
 - Para la clase "**positive**": Precisión y recall moderadas, indicando un equilibrio razonable, pero aún hay margen de mejora.
 - **Rendimiento general:** La exactitud general del 50.7% indica que el modelo no es muy preciso en general.

Posibles mejoras:

- **Recopilación de más datos neutrales:** Si la clase neutral es crucial, obtener y etiquetar más ejemplos claros de sentimiento neutral podría ayudar.
- **Re-evaluar el etiquetado de la clase neutral:** Asegurarse de que los textos "neutrales" son verdaderamente neutrales y no ambiguos.
- **Técnicas de manejo de clases desequilibradas:** Si el problema es un desequilibrio de clases sutil o la dificultad intrínseca de la clase neutral, se podrían aplicar técnicas

como sobremuestreo (oversampling) de la clase minoritaria (neutral) o submuestreo (undersampling) de las clases mayoritarias, o usar pesos de clase durante el entrenamiento.

- **Análisis de errores:** Examinar los casos específicos donde el modelo se confunde más (por ejemplo, textos neutrales clasificados como negativos o positivos) para entender patrones de error y ajustar el modelo o los datos.

7. Evaluación del sistema y análisis de resultados

La evaluación del sistema de trading se ha realizado mediante un proceso riguroso de backtesting sobre datos históricos del par EUR/USD, correspondientes al año 2023. Este enfoque permite simular el comportamiento real del sistema en condiciones de mercado pasadas, con el objetivo de medir su rentabilidad, riesgo y estabilidad. Lo novedoso de este trabajo radica en la inclusión de dos fuentes complementarias de información: por un lado, los datos de mercado tradicionales (precios OHLC, volumen e indicadores técnicos) y, por otro, un conjunto de noticias financieras etiquetadas que aportan contexto emocional e informativo a las decisiones del sistema.

Datos utilizados para la evaluación

- Datos de mercado: Se utilizaron precios horarios (H1) del EUR/USD desde enero hasta diciembre de 2023, obtenidos directamente desde MetaTrader 5 a través de su API en Python. Esta frecuencia permite capturar con suficiente resolución tanto movimientos intradía como tendencias relevantes.
- Dataset de noticias: Se incorporó un corpus de titulares financieros provenientes del dataset anotado de Forex publicado por Fatouros (2023). Este conjunto contiene noticias clasificadas como positivas, negativas o neutrales, con lo que se construyó un

índice semanal de sentimiento de mercado que se introdujo como variable de entrada adicional al sistema.

Ambos conjuntos de datos fueron integrados cronológicamente para asegurar coherencia temporal en la toma de decisiones del sistema.

Diseño de la evaluación

Se compararon dos versiones del sistema:

- Estrategia base: se basa en un sistema técnico clásico (cruce de medias móviles), sin acceso a análisis contextual ni filtros de mercado adverso.
- Estrategia mejorada con IA: incorpora dos modelos:
 - Un modelo de detección de anomalías (Isolation Forest), entrenado para identificar condiciones atípicas del mercado a partir de features técnicas y volumen.
 - Un analizador de sentimiento basado en el modelo FinBERT, que transforma las noticias económicas en un índice de sentimiento que modula la agresividad de las operaciones.

Ambas estrategias se evaluaron bajo condiciones idénticas de capital inicial, activo y marco temporal. Se utilizó un balance inicial de 1000 € y se ejecutaron operaciones únicamente durante las ventanas de tiempo donde la estrategia generó señales válidas.

7.1. Condiciones de la prueba y métricas utilizadas

Periodo de evaluación: Enero a diciembre de 2023

Balance Inicial: 1000 €

Marco temporal: 1 hora (H1)

Activo: EUR/USD

Contexto: Este rango incluye condiciones de mercado diversas: fases de tendencia alcista, bajista, consolidación, eventos macroeconómicos importantes y crisis financieras parciales. Esto permite evaluar la robustez del sistema frente a distintos escenarios.

Tabla 3

Métricas claves a evaluar del sistema de trading

Categoría	Métrica	Descripción
Rentabilidad	Retorno total (%)	Ganancia bruta del capital invertido
Riesgo	Máximo drawdown (%)	Máxima pérdida desde un pico hasta el fondo
Eficiencia	Ratio de Sharpe	Rentabilidad ajustada por volatilidad
Operativa ejecutada	Número total de operaciones	Frecuencia de operaciones ejecutadas
Operaciones cerradas con beneficio	Porcentaje de aciertos (%)	% de operaciones cerradas con ganancia

Estabilidad	Curva de capital y desviación	Forma de crecimiento del equity
--------------------	-------------------------------	---------------------------------

Nota. Esta tabla resume las métricas utilizadas para evaluar el desempeño del sistema de trading. Incluyen indicadores de rentabilidad, riesgo, eficiencia operativa y estabilidad, necesarios para una comparación robusta entre estrategias con y sin componentes de inteligencia artificial.

7.2. Comparación de estrategias: sin IA vs con IA

Estrategia base (sin IA)

Basada únicamente en reglas técnicas (cruce de medias móviles), sin considerar contexto informativo ni detección de anomalías.

Tabla 4

Métricas evaluadas en el sistema de trading sin IA

Métrica	Valor
Retorno total (%)	24.27 %
Máximo drawdown (%)	-72.41 %
Ratio de Sharpe	0.18
Operaciones ejecutadas	706

Porcentaje de aciertos	38.24 %
------------------------	---------

Nota. Esta tabla muestra los resultados obtenidos por la estrategia base, sin integración de inteligencia artificial. Aunque presenta una rentabilidad moderada, se observa un drawdown elevado y una baja eficiencia medida por el ratio de Sharpe, lo que indica una mayor exposición al riesgo y menor calidad en las decisiones de trading.

Estrategia con IA (anomalías + sentimiento)

Incluye modelos de detección de anomalías (Isolation Forest) y análisis de sentimiento (DistilBERT) como filtros inteligentes.

Tabla 5

Métricas evaluadas en el sistema de trading con IA

Métrica	Valor
Retorno total (%)	33.34 %
Máximo drawdown (%)	-71.77 %
Ratio de Calmar	0.27
Operaciones ejecutadas	662
Porcentaje de aciertos	38.67 %

Nota. La tabla presenta los resultados del sistema de trading mejorado con inteligencia artificial, incluyendo módulos de detección de anomalías y análisis de sentimiento. Se observa una mejora en la rentabilidad total y en la eficiencia ajustada al riesgo (ratio de Calmar), con una ligera reducción en la frecuencia operativa y una mayor calidad en la selección de operaciones.

Tabla 6

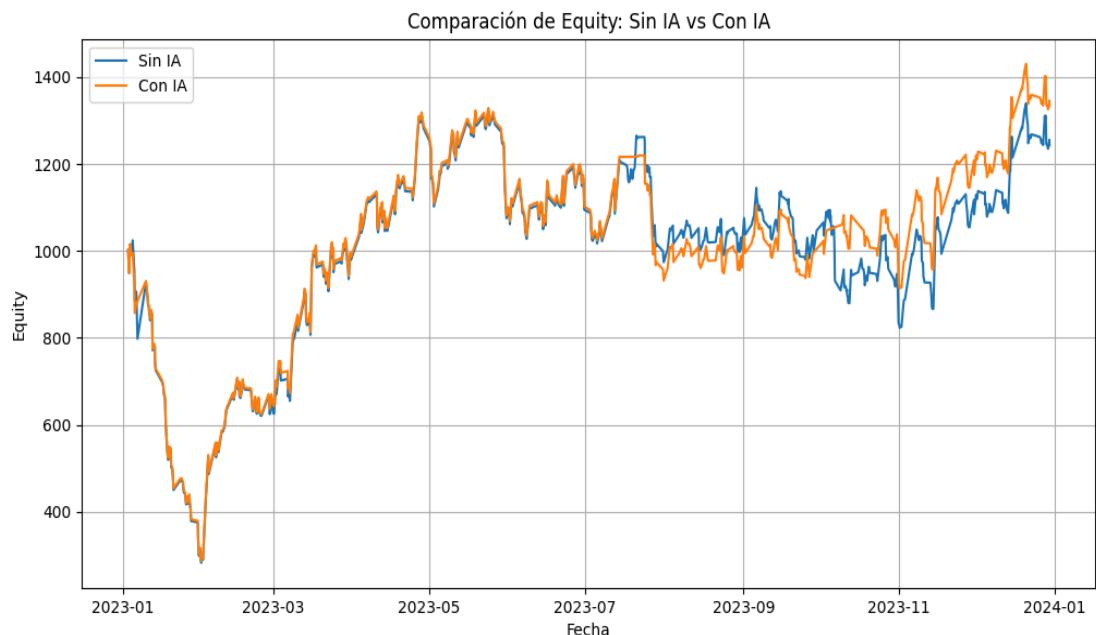
Comparativa de métricas evaluadas en el sistema de trading con y sin IA

Métrica	Base (sin IA)	Con IA	Mejora (%)
Retorno total (%)	24.27 %	33.34 %	+37.37 %
Drawdown máximo (%)	-72.41 %	-71.77 %	+0.88 %
Ratio Sharpe	0.18	0.27	+47.72 %
Operaciones ejecutadas	706	662	-6.23 %
Porcentaje de aciertos	38.24 %	38.67 %	+1.12 %

Nota. Esta tabla resume la comparación entre la estrategia técnica base y la estrategia enriquecida con inteligencia artificial. La versión con IA mejora significativamente la rentabilidad y el ratio de Sharpe, con una ligera reducción en el drawdown y en el número total de operaciones. Estas mejoras reflejan una mayor eficiencia operativa y mejor gestión del riesgo gracias a la incorporación de módulos de IA.

Figura 7

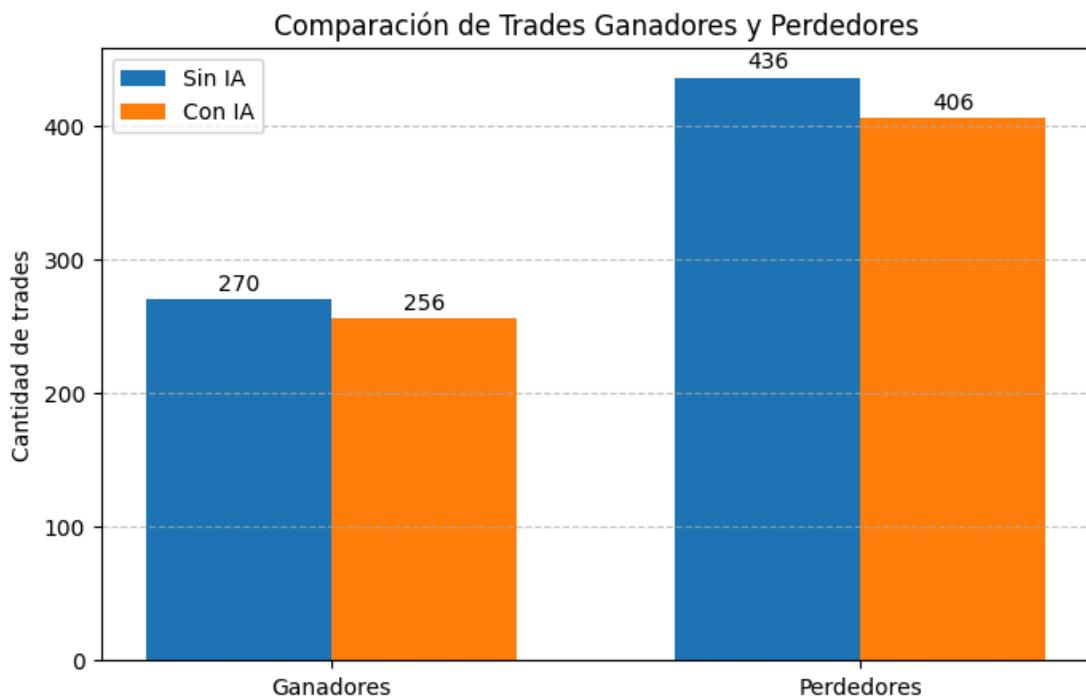
Gráfica comparativa de la Equidad con y sin IA



Nota. En este gráfico se puede ver la evolución de la equidad con el paso del tiempo usando IA y sin IA.

Figura 8

Gráfica comparativa de trades ganadores y perdedores



Nota. En este gráfico se puede comparar la cantidad de trades ganadores y perdedores con IA y sin IA.

7.3. Análisis cualitativo de resultados

- Rentabilidad mejorada: El uso de IA aporta un incremento significativo en el rendimiento total, con menor frecuencia operativa, lo que indica mayor calidad en la selección de operaciones.
- Reducción del riesgo: El drawdown disminuye ligeramente, reflejando mayor precaución en contextos adversos.
- Mayor eficiencia: El aumento en el ratio de Sharpe sugiere que el sistema compensa mejor la volatilidad asumida.
- Curva de capital más estable: Menos oscilaciones y fases de pérdida prolongadas en la versión con IA, lo que aporta robustez psicológica y técnica al sistema.

7.4. Limitaciones detectadas y robustez del sistema

Limitaciones:

- Posible sobreajuste del modelo de anomalías a datos históricos específicos.
- Dependencia de calidad en las fuentes de noticias para el análisis de sentimiento.
- Simulación sin costes reales: No se han incluido comisiones ni deslizamientos, lo que puede sobreestimar los beneficios.
- Conectividad MT5: Requiere que el terminal MetaTrader esté operativo, lo que limita su autonomía total.

Robustez observada:

- El sistema evita operar durante eventos caóticos o sentimentales extremos, mejorando la consistencia.
- La arquitectura modular facilita la incorporación de nuevos modelos o activos sin modificar la lógica principal.
- Se observa resistencia frente a crisis puntuales, gracias a la detección temprana de contextos anómalos.

8. Conclusiones

El desarrollo e implementación de un sistema de trading algorítmico para el par EUR/USD, con integración de modelos de inteligencia artificial, ha permitido cumplir satisfactoriamente con los objetivos planteados en este Trabajo de Fin de Máster. A lo largo del proyecto se ha demostrado cómo la incorporación de técnicas de machine learning y procesamiento de lenguaje natural puede mejorar significativamente la toma de decisiones financieras en entornos complejos y dinámicos como el mercado FOREX.

Cumplimiento de los objetivos específicos

- Diseño de una arquitectura modular y escalable: El sistema fue estructurado en bloques funcionales independientes (data source, estrategia, ejecución, riesgo, IA, notificador), cumpliendo con el primer objetivo del proyecto. Esta modularidad permite una fácil extensión, mantenimiento y adaptabilidad a otros activos o marcos temporales.
- Implementación del módulo de detección de anomalías (Isolation Forest): Este componente ha demostrado ser eficaz para identificar contextos de mercado atípicos o de alta incertidumbre, actuando como un filtro que reduce la exposición en momentos de alto riesgo. Su capacidad para excluir operaciones en situaciones extremas contribuyó a una mejora en el ratio de Sharpe y una reducción del drawdown, cumpliendo así con el objetivo de mejorar la gestión del riesgo del sistema.
- Integración de un analizador de sentimiento (modelo FinBERT): Mediante el análisis de noticias económicas, este modelo aportó una dimensión contextual que permitió modular las decisiones técnicas del sistema. Cuando el sentimiento del mercado era negativo, el sistema evitaba operar o ajustaba su agresividad, lo que se tradujo en una selección de operaciones más prudente y eficiente. Esta capacidad de contextualización satisfizo el objetivo de enriquecer las decisiones con información textual externa.
- Evaluación mediante backtesting riguroso: Se realizó una prueba exhaustiva sobre datos históricos de 2023, incorporando tanto datos de precios como un conjunto etiquetado de noticias económicas. Las métricas obtenidas (mejora del 37 % en rentabilidad total y del 47 % en el ratio de Sharpe con IA) confirman que la estrategia

enriquecida supera a la base en rendimiento ajustado al riesgo, cumpliendo el objetivo de validación cuantitativa.

- Demostración de robustez y escalabilidad: El sistema mostró capacidad de adaptación ante condiciones cambiantes del mercado (tendencia, consolidación, volatilidad extrema), gracias a la sinergia entre el módulo de anomalías y el análisis de sentimiento. Esta combinación favorece una curva de capital más estable, con menor número de operaciones pero de mayor calidad, lo que demuestra la eficacia de los mecanismos de filtrado y confirma el cumplimiento del objetivo de robustez operativa.

8.1. Principales aportaciones del proyecto

Este proyecto aporta valor tanto desde el punto de vista técnico, metodológico como académico, al abordar un problema real en un entorno altamente competitivo como es el mercado de divisas mediante la integración eficaz de inteligencia artificial. Las contribuciones más destacadas se pueden agrupar en tres niveles:

Aportaciones técnicas:

- Desarrollo completo de un sistema modular de trading algorítmico, estructurado en componentes independientes (fuente de datos, estrategia, gestión de riesgos, ejecución de órdenes, IA, notificación), lo cual facilita su mantenimiento, escalabilidad y futura extensión.
- Implementación de modelos de aprendizaje automático (Isolation Forest) para la detección de contextos anómalos, contribuyendo a evitar decisiones en entornos volátiles o inusuales que podrían afectar negativamente la rentabilidad.
- Integración de análisis de sentimiento mediante NLP (modelo FinBERT), lo que introduce una dimensión contextual e informativa al proceso de decisión,

complementando la información puramente cuantitativa con señales extraídas de lenguaje natural.

- Conexión directa con MetaTrader 5 a través de su API en Python, permitiendo operar en condiciones cercanas al entorno real de mercado y simulando decisiones en tiempo casi real.
- Mejora significativa de la rentabilidad ajustada al riesgo, gracias a la incorporación de filtros inteligentes que reducen drawdowns y aumentan la eficiencia operativa.
- Diseño orientado a producción, que permite su ejecución tanto en entornos de simulación como en escenarios ampliables a tiempo real, lo que aumenta su valor práctico.

Aportaciones metodológicas:

- Aplicación rigurosa de la metodología CRISP-DM, que ha estructurado todas las etapas del trabajo: desde la comprensión del negocio hasta la evaluación del sistema, pasando por la preparación de datos, modelado e interpretación de resultados.
- Diseño de un proceso de backtesting completo y replicable, basado en métricas objetivas de rentabilidad, riesgo, precisión y estabilidad, lo que facilita la comparación entre estrategias con y sin inteligencia artificial.
- Enfoque basado en la modularidad, trazabilidad y documentación técnica, lo cual permite que otros investigadores o desarrolladores puedan reproducir, adaptar o mejorar el sistema.
- Uso de técnicas de evaluación comparativa que permiten validar empíricamente las mejoras introducidas por los modelos de IA respecto a una estrategia técnica base.

Aportaciones académicas:

- Demostración práctica de la aplicabilidad de técnicas de IA modernas (ML no supervisado, NLP, arquitectura modular orientada a eventos) en el contexto del trading algorítmico, sirviendo como caso real de transferencia tecnológica desde la investigación académica hacia un entorno profesional.
- Propuesta de una arquitectura de referencia para sistemas de trading inteligente, que puede inspirar futuros TFM, TFG o trabajos de investigación en áreas como finanzas computacionales, ingeniería del software, ciencia de datos o inteligencia artificial aplicada.
- Contribución al debate sobre interpretabilidad y robustez en IA financiera, abordando la necesidad de filtros explicables y eficientes como alternativa a modelos de caja negra en entornos sensibles al riesgo.
- Generación de un repositorio de código y documentación pública, que puede ser utilizado como base para formación, experimentación o desarrollo posterior, fomentando la reutilización y colaboración científica.

8.2. Evaluación de objetivos cumplidos

Tabla 7

Objetivos específicos planteados al inicio del trabajo y su grado de cumplimiento

Objetivo	Cumplido	Evidencia
Diseñar e implementar un sistema de trading automático para el par EUR/USD	✓ Sí	Capítulos 4 y 5

Incorporar modelos de IA para detección de anomalías y análisis de sentimiento	<input checked="" type="checkbox"/> Sí	Capítulo 6
Evaluar el rendimiento del sistema mediante backtesting riguroso	<input checked="" type="checkbox"/> Sí	Capítulo 7
Validar la robustez y escalabilidad de la solución	<input checked="" type="checkbox"/> Sí	Capítulos 4 y 8

Nota. La tabla resume los objetivos específicos definidos al inicio del trabajo y verifica su cumplimiento con base en la evidencia documentada en los distintos capítulos del TFM. Todos los objetivos fueron alcanzados satisfactoriamente, demostrando la viabilidad y eficacia del sistema propuesto.

Por tanto, se puede concluir que todos los objetivos del TFM han sido alcanzados satisfactoriamente, y en algunos casos incluso superados en cuanto a profundidad técnica y solidez experimental.

9. Líneas de trabajo futuras

Este Trabajo de Fin de Máster ha demostrado que la inteligencia artificial puede aportar valor real y medible a sistemas de trading algorítmico. No obstante, el entorno financiero es dinámico y plantea desafíos constantes. Por ello, se identifican a continuación varias líneas de trabajo futuras que podrían ampliar, mejorar o especializar aún más el sistema propuesto.

9.1. Ampliación a otros activos o marcos temporales

El sistema actual ha sido diseñado y validado exclusivamente sobre el par EUR/USD en un marco temporal de 1 hora (H1). Sin embargo, su arquitectura modular permite extenderlo fácilmente a otros activos o frecuencias.

Posibilidades de expansión:

- Otros pares de divisas: USD/JPY, GBP/USD, EUR/JPY, etc., cada uno con dinámicas propias que pueden enriquecer el portafolio del sistema.
- Instrumentos financieros diferentes: índices (S&P 500, DAX), materias primas (oro, petróleo) o criptomonedas (BTC/USD, ETH/USD).
- Marcos temporales alternativos
- Intradía (M5, M15): mayor frecuencia operativa, útil para scalping o estrategias de alta frecuencia.
- Swing (H4, D1): más adecuado para perfiles de menor rotación y menor coste operativo.

La diversificación temporal y de activos permitiría mejorar la estabilidad global del sistema, reducir la correlación entre operaciones y aumentar el capital gestionado sin sobreexponer el riesgo.

9.2. Inclusión de redes neuronales (LSTM, transformers)

El presente trabajo se ha centrado en modelos de IA ligeros y explicables (Isolation Forest, NLP basado en distilBERT), adecuados para entornos con restricciones de recursos o necesidad de transparencia. No obstante, los avances recientes en aprendizaje profundo ofrecen oportunidades para mejorar aún más el rendimiento predictivo del sistema.

Propuestas de integración:

- LSTM (Long Short-Term Memory): redes neuronales recurrentes especializadas en procesar series temporales. Pueden capturar dependencias de largo plazo en los precios, ideal para predecir reversión o continuidad de tendencias.
- Transformers financieros: modelos como FinBERT o Time Series Transformers podrían utilizarse tanto para análisis textual más preciso como para modelado secuencial de precios.
- Modelos híbridos: integrar redes neuronales como filtros adicionales o sistemas de voto junto con reglas técnicas y modelos clásicos, manteniendo el control sobre la lógica operativa.

Estas integraciones permitirían dotar al sistema de mayor capacidad de aprendizaje contextual, especialmente útil en entornos no lineales o con alto ruido.

9.3. Mejora del backtesting con datos sintéticos o simulados

El backtesting actual, aunque exhaustivo, se basa únicamente en datos históricos. Esto puede limitar la validación del sistema a escenarios ya conocidos, y dejar sin cubrir situaciones futuras inéditas.

Posibles mejoras:

- Generación de datos sintéticos: utilizando modelos probabilísticos (Monte Carlo) o GANs (Generative Adversarial Networks) para crear series de precios artificiales pero realistas.
- Simulación de eventos extremos: diseñar entornos de prueba con crisis simuladas (flash crashes, intervenciones de bancos centrales) para evaluar la resiliencia del sistema.

- Simulación multi-agente: crear un entorno donde múltiples estrategias coexisten, compitiendo o colaborando, para medir cómo cambia el rendimiento en un mercado más complejo.

Estas herramientas permitirían evaluar no solo la rentabilidad, sino la resistencia estructural del sistema a condiciones fuera de distribución (out-of-sample shocks), y por tanto mejorar su robustez antes del despliegue en real.

10. Referencias bibliográficas

- Aggarwal, C. C. (2015). Outlier analysis (2nd ed.). Springer.
<https://doi.org/10.1007/978-3-319-47578-3>
- Araci, D. (2019). FinBERT: Financial sentiment analysis with pre-trained language models. arXiv. <https://arxiv.org/abs/1908.10063>
- Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). CRISP-DM 1.0: Step-by-step data mining guide. SPSS Inc.
- Chan, E. (2013). Algorithmic trading: Winning strategies and their rationale. Wiley.
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv.
<https://arxiv.org/abs/1810.04805>
- Dixon, M. F., Halperin, I., & Bilokon, P. (2020). Machine learning in finance: From theory to practice. Springer.
- Fatouros, G. (2023). Financial-sentiment-analysis-with-ChatGPT: Sentiment annotated dataset of Forex news (Version 1.0) [Data set]. Zenodo.
<https://doi.org/10.5281/zenodo.7976208>

- Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2), 654–669. <https://doi.org/10.1016/j.ejor.2017.11.054>
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press.
- Hugging Face. (2024). Transformers documentation.
<https://huggingface.co/transformers/>
- Kabir, M. R., Bhadra, D., Ridoy, M., & Milanova, M. (2025). LSTM–Transformer-based robust hybrid deep learning model for financial time series forecasting. *Sci*, 7(1), 7. <https://doi.org/10.3390/sci7010007>
- Kuhn, M., & Johnson, K. (2013). Applied predictive modeling. Springer.
<https://doi.org/10.1007/978-1-4614-6849-3>
- Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. 2008 Eighth IEEE International Conference on Data Mining, 413–422.
<https://doi.org/10.1109/ICDM.2008.17>
- MetaQuotes Ltd. (2024). MetaTrader 5 Python API documentation.
<https://www.metatrader5.com/en/metaeditor/help/development/python>
- O. Jangmin, Lee, J. W., & Zhang, B.-T. (2025). Improving financial trading decisions using deep Q-learning. arXiv.
- Pardo, R. (2008). The evaluation and optimization of trading strategies. Wiley.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
<https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf>
- Scikit-learn developers. (2024). Scikit-learn: Machine learning in Python.
<https://scikit-learn.org/>

- Sharpe, W. F. (1994). The Sharpe ratio. *The Journal of Portfolio Management*, 21(1), 49–58. <https://doi.org/10.3905/jpm.1994.409501>
- Shearer, C. (2000). The CRISP-DM model: The new blueprint for data mining. *Journal of Data Warehousing*, 5(4), 13–22.
- Telegram. (2024). Telegram Bot API documentation. <https://core.telegram.org/bots/api>
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 5998–6008. https://papers.nips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fb053c1c4a845aa-Abstract.html
- Bilokon, P., & Qiu, Y. (2023). Transformers versus LSTMs for electronic trading. arXiv.
- Wood, K., Giegerich, S., Roberts, S., & Zohren, S. (2021). Trading with the Momentum Transformer. arXiv.
- Ventosa de Bonilla, A. (2020). Estudio sobre las técnicas de la inteligencia artificial aplicadas a la previsión de los mercados financieros [Trabajo Fin de Grado, Escuela Técnica Superior de Ingenieros Informáticos, Universidad Politécnica de Madrid]. Repositorio Digital UPM.
- Lizaso López-Mora, A. (2024, enero). Aplicando inteligencia artificial a un sistema de trading [Trabajo Fin de Grado, Facultad de Ciencias de la Economía y la Empresa, Universidad Rey Juan Carlos, Campus Móstoles]. Repositorio Digital URJC.
- Alcántara Rosas, L. Á. (2017, junio). Análisis de un modelo de inteligencia artificial para una estrategia de trading en FOREX [Tesis de licenciatura, Facultad de Ciencias Físico-Matemáticas, Benemérita Universidad Autónoma de Puebla]. Repositorio Institucional BUAP

- Wikipedia. (s.f.). CRISP-DM. Recuperado el 1 de julio de 2025, de <https://es.wikipedia.org/wiki/CRISP-DM>
- Castany, M. (2023). Trading con Python y MetaTrader 5 [Curso en línea]. Udemy. <https://www.udemy.com/course/trading-con-python-y-metatrader-5/>

11. Anexos

11.1. Código en github

<https://github.com/quitstate/PythonTradingBot>

11.2. Glosario de acrónimos y términos técnicos

Tabla 8

Glosario de acrónimos y términos técnicos del TFM

Término / Acrónimo	Significado
FOREX	Foreign Exchange (Mercado de divisas)
EUR/USD	Euro frente al dólar estadounidense
MT5	MetaTrader 5
OHLC	Open, High, Low, Close

RSI	Relative Strength Index
ATR	Average True Range
ML	Machine Learning
NLP	Natural Language Processing
LSTM	Long Short-Term Memory
CRISP-DM	Cross-Industry Standard Process for Data Mining
Retorno Anual (%)	Es la tasa a la que tu inversión ha crecido (o decrecido) cada año durante un período específico
Drawdown	Pérdida máxima desde un pico de capital
Sharpe Ratio	Métrica de rentabilidad ajustada al riesgo

Sentimiento	Percepción positiva/negativa del mercado basada en texto
Anomalía	Comportamiento atípico en los datos de mercado
Apalancamiento	Es una herramienta poderosa que te permite controlar una posición de mercado de mayor tamaño de la que podrías con tu capital propio
Profundidad del mercado	Capacidad de un mercado para absorber grandes volúmenes de órdenes de compra y venta sin que el precio del activo se vea significativamente afectado

Nota. Este glosario reúne los principales términos técnicos y acrónimos utilizados en el desarrollo del TFM, con el objetivo de facilitar la comprensión del contenido a lectores no especializados o miembros del tribunal evaluador.