UNIVERSITY OF LONDON
INTERNATIONAL PROGRAMMES

BSc Computer Science and Related Subjects

CM3070 PROJECT
PRELIMINARY PROJECT REPORT

STOCKS BOT ADVISOR

Author:	ANTHONY WINATA SALIM
Student Number :	230726051
Date of Submission:	17 June 2025
Supervisor :	HWA HENG KAN

Contents

CHAPTER 1: INTRODUCTION

The Growing sophistication of financial markets and low financial literacy among retail investors makes it a serious challenge to make sound investment choices. The majority of individual investors do not have professional-grade financial tools and are prone to fall prey to emotional decision-making biases. Conventional robo-advisors, although useful, are typically black-box systems offering no or poor rationale for their suggestions.

To solve this problem, this project aims to create a Financial Advisor Bot, an AI-driven decision support system that produces readable, comprehensible, and personalized investment advice. This project is based on the "Financial Advisor Bot" template in CM3020 Artificial Intelligence and combines machine learning, technical analysis, as well as natural language explanations. It is mainly used to offer readable financial intelligence in general to amateur investors through a web-based interface.

The bot will analyse 10–15 stocks accross 2–3 sectors, providing real-time recommendations on a simple dashboard. Not only will users get buy/sell/hold indications, but also natural language explanations and interactive visualizations in support of every recommendation.

Project Objectives and Research Questions
The major aim of the project is:

To design and implement an explainable, transparent financial advisor bot that helps retail investors make knowledgeable choices by producing interpretable, data-driven recommendations.

The project investigates the following research questions:

- RQ1: How can financial forecasting models be made more interpretable and transparent for non-expert users?
- RQ2: What XAI techniques are most appropriate to trust and comprehension in finance decision-making?

- RQ3: Is it possible for a hybrid approach (technical analysis + machine learning) to beat conventional strategies on both accuracy and user trust?

Deliverables

- Literature Review
  A systematic examination towards the research and industry publication on technical indicator, machine learning in finance, and XAI technique, discusses RQ1 and RQ2 by identifying the gaps in the system that exists.
- Functional Prototype
  A functional web-based financial advisor bot that addresses RQ1 and RQ3 by demonstrating interpretable ML predictions and hybrid decision-making through simple-to-understand explanations.
- Final Report
  this report sums everything including the description of the system architecture, implementation, evaluation results, and research contributions, substantiating all three research questions with justifications and test results.
- Presentation Materials
  A final-year presentation with a demo and slides describing a summary of findings, methodologies, and system features.

CHAPTER 2: LITERATURE REVIEW

The financial advisory services industry is faced with a stark chasm between technological innovation and ease of use, presenting gigantic hurdles for retail investors in accessing reliable investment advice. Though robo-advice platforms have been technologically sophisticated solutions, they are inherently limited in terms of transparency, comprehensibility, and pedagogical value to the clients, with studies indicating that 77% of wealth management customers still favor human advisors because of trust and engagement issues (Jung et al., 2019). At the same time, retail investors also exhibit systematic behavioral biases—such as overconfidence, loss aversion, and herding behavior—that produce inferior financial choices and necessitate education and advisory assistance tools (Sattar et al., 2020). Modern machine learning approaches to finance, although granting high predictive precision, are confronted with essential pitfalls in interpretability and end-user comprehension (Wasserbacher & Spindler, 2022), whereas technical analysis techniques, although satisfactory, demand experience above the average investor (Nti et al., 2020). Further, despite substantial regulatory demands for explainable AI in finance under frameworks such as GDPR, current explainable AI systems are still too technical for non-technical end-users (Černevičienė & Kabašinskas, 2024). This literature review discusses these interconnected topics of financial advisory technology, determines significant shortcomings of existing solutions, and lays the groundwork for the development of more transparent, educational, and engaging financial advisory systems that integrate in-depth analysis with clear explanation.

1. The Problem – current limitation in Financial Advisory Systems

1A. Robo-Advisor Limitations

Current Shortcomings of Robo-Advisory Systems

The robo advisory that is exsiting right now are filled with significant issues that limits their operation and the appeal to private investors . Jung et al. (2019) conducted a comprehensive analysis of robo-advisory services and discovered that there are several serious limitations that create bottlenecks to their wider usage and consumer acceptance.

Trust and Human Interaction Deficits

The greatest impediment to robo-advisors is the human versus machine trust deficit. Jung et al. (2019) discovered that "77 per cent of the surveyed wealth management clients trust their financial advisors. Additionally, 81 per cent report that face-to-face interaction matters." This study demonstrates a fundamental disconnect between user requirement and existing robo-advisory services, where the lack of human interaction imposes significant impediments to adoption. The research also corroborates that "49 per cent would not use robo-advisory services without the assistance of a face-to-face consultant, and just 11 per cent are ready to use an independent robo-advisor."

Transparency and Fee Disclosure Issues

Current robo-advisory platforms are blemished by lack of transparency in their operating models and fee structures. Jung et al. (2019) discovered that "indirect fees are normally not clearly shown on the investment dashboards, and therefore camouflaged as lower returns of investor portfolios." the lack of transparency destroys user's trust and investors are unable to make decisions based on the information about their investment.

Inadequate Personalization and Risk Assessment

Robo-advisors are usually founded upon abridged instruments of evaluation that are not able to reflect the investor's profile complexity. The research confirms that "the simplicity of risk tolerance questionnaires is a 'natural' limit in the development of robo-advice" (Jung et al., 2019).

Educational Deficit and User Comprehension

Most significantly for the development of independent investors, robo-advisory sites provide limited educational content or rationale for their suggestions at present. Jung et al. (2019) recognized that "clients are on their own when deciding whether the proposed investment strategies are appropriate for their requirements and goals." This kind of knowledge gap does not enable users to gain financial literacy and understand the rationale for investment suggestions, limiting their ability for making fully informed decisions and acquiring long-term investment skills.

1B. Behavioural Biases in Individual Investment Decision-Making

1. Behavioural Biases in Personal Investment Decision Making

Along with limitations of current investment advisory systems, individual investors are also confronted with overwhelming psychological issues undermining their investment decision-making. Sattar et al. (2020) carried out a detailed analysis of behavioral finance biases and arrived at the finding that "empirical results concluded investment decision making influenced by heuristic behaviors more than prospects and personality characteristics." The study confirms that despite better advisory systems, investors are combating systematic cognitive bias to deliver lower financial returns.

2. Overconfidence and Excessive Risk-Taking

Investors themselves are prone to overconfidence bias in that they overestimate their investment knowledge and are risk underestimators. Sattar et al. (2020) found out that in "overconfidence condition, most of the investors ignore the risk and uncertainty because of the previously continued success and made more trading, therefore, in this kind of situation the failure are most likely to increase". This bias reveals itself in excessive trading, poor risk analysis, and ultimately poorer portfolio performance as investors become too optimistic about their analytical skills.

3. Loss Aversion and Irrational Holding Behavior

Loss aversion also greatly skew investor behavior, leading people to make illogical choices regarding when to sell and buy investments. The study finds that "sometimes investor sell immediately stocks whose prices increased and keep hold low-price stocks because investor shows loss aversion behavior effects investment decision" (Sattar et al., 2020).

4. Herding Behavior and Information Processing Breakdowns

Most investors make investments following the actions of the crowd instead of making personal decisions, and thus, they end up making incorrect timing and investment decisions. Sattar et al. (2020) proved that "some investors make decisions by follow the others' information and decision process.

5. Cognitive Biases and Durable Errors

There are other biases that also aggravate investment decision issues. Anchoring bias simply leads to investors over-relying on early information since "investor considered the historical return trend in mind for reaction in decision making" (Sattar et al., 2020). Confirmation bias leads to selective information processing wherein "investor always tries to see the positive side of decision although ignoring the negative data which indicates the low risk and success perception about investment" (Sattar et al., 2020).

6. The Need for Systematic Guidance

The study illustrates how such behavioral biases present a compelling argument for external direction and methodical decision support. The paper concludes that "finance managers can take better decision by understanding Behavioral finance," implying that instruments which enable investors to acknowledge and surmount such biases are the crux of better investment decisions

2. Existing Solutions and Their Limitations

2A. Machine Learning Approaches in Financial Forecasting

Machine learning algorithms have now reached the fore of financial prediction, with Wasserbacher and Spindler (2022) citing popular methods including "elastic net, random forest, K-nearest neighbor, support vector machine and deep neural networks." These models perform better, with "machine learning forecasts outperformed the forecasts of domain experts and statistical models. in about 70% of all comparisons."

Yet ML techniques are plagued by inherent shortcomings that diminish their utility to the retail investor. The first among these is overfitting, where models exhibit "very good performance on the training data but poor generalization" (Wasserbacher & Spindler, 2022). Black box is also their issue, as "historically, the machine learning community has pursued the goal of maximizing predictive performance rather than model parameter understandability." This is an issue for small investors who need investment rationale comprehension.

This study focuses that "if the insights that is achived from predictive analysis could not be interpret, those insights are most likely not happening in the firm" (Wasserbacher & Spindler, 2022), reflecting the disparity between ML potential and user needs. Furthermore, "the combination of predictive modeling and causal inference is one of the biggest challenges in machine learning for finance," constraining the creation of integrated, user-friendly financial advisory systems.

2B. Technical Analysis in Automated Financial Systems

Technical analysis is extremely effective at stock market prediction, and systematic studies provide strong empirical evidence. Nti et al. (2020) established that "previous studies reported accuracy within 36.55–97.8%, which confirms that the stock market is highly predictable" with technical indicators. The most popular indicators are "simple moving average (SMA), followed by EMA, MACD, RSI and OBV," and these combinations were extremely efficient when combined with machine learning algorithms. The research discovers that "more than 50% of the studies reviewed used hybrid algorithms. as demonstrated by the accuracy obtained" (Nti et al., 2020), indicating the strength in combining diverse technical approaches. Auto systems with artificial neural networks and technical indicators revealed especially robust results, with "ANN produced the lowest RMSE and MAE when combined with technical indicators."
Yet, technical analysis has considerable limitations for private investors. The most important one is the fact that technical indicators need experience for correct interpretation since "the technical analyst attempts to foresee the stock market via learning graphs and forms and technical indicators" (Nti et al., 2020).

2C. Explainable AI in Financial Services

Explainable AI is now a requirement in financial services due both to regulatory pressure and user trust issues. Černevičienė and Kabašinskas (2024) note that "explainability is a fundamental requirement for financial AI applications," with regulatory frameworks like "the European Union's GDPR. requiring that meaningful information about the logic used in automated decision-making be made available." Even the EU AI Act requires transparency for financial AI systems that are high-risk.

Some of the current XAI techniques being introduced into the financial sector are "LIME, SHAP, gradient-based methods, decision trees, counterfactual explanations, and attention mechanisms" (Černevičienė & Kabašinskas, 2024). They are being used in "credit-risk scoring, stock market prediction, loan default prediction, and fraud detection." Studies verify that "user studies show that textual and visual explanations significantly improve perceived trust in AI systems."

Major Gaps in Current XAI Techniques

In spite of regulatory requirements and the availability of technology, there are considerable limitations in existing explainable AI solutions. firstly, the biggest problem is that "interpretability could be only be understood by the domain experts, which makes it difficult for cunsumer app to implement in it". (Černevičienė & Kabašinskas, 2024). This forms a fundamental gap between compliance and real user comprehension.

besides that, " a lot of systems offer the same explanation to everyone that does not fulfill the user needs or user expectation". however, "the explaination mislead user that leads users to depend on worngly made result, which has a result to undermine their confidence and trust" (Černevičienė & Kabašinskas, 2024). The study recognizes that "natural language explanations are useful to distill complex model behaviors into human-understandable insights," yet these kinds of approaches are unrealized in present financial AI systems, and therefore there is an urgent call for transparent, instructive explanations that are comprehensible to individual investors and actionable.

3. Research Gap and Project Justification

Synthesis of the Most Important Gaps in Financial Advisory Systems Today

This literature review highlights four related gaps that, in combination, imply that existing financial advisory solutions are poorly aligned with individual investors' needs. While each type of existing solution does automate some elements of financial advice, none of the solutions can be matched with in-depth analysis and high-quality, educative user experiences.

The Transparency Gap

Current financial advisory services can never provide sufficient explanations for what they do. Robo-advisors are "black boxes" with unspecified fee structures (Jung et al., 2019), and machine learning algorithms prioritize "maximizing predictive performance rather than interpretability of model parameters" (Wasserbacher & Spindler, 2022). Even recent explainable AI architectures are afflicted with interpretability that "may be understandable only to experts in the field" (Černevičienė & Kabašinskas, 2024) and are unable to bridge the gap between algorithmic complexity and user comprehension.

The Education Gap

Individual investors have systematic behavioral biases like overconfidence, loss aversion, and herding which lead to making poor investment choices (Sattar et al., 2020). In spite of that, current advisory systems have minimal educational content, and "clients are on their own when deciding whether the investment strategies provided are suitable for them and their goals" (Jung et al., 2019). This absence of education strengthens the cycle of poor investment decision-making and reduces user trust in automated systems.

The Integration Gap

Technical analysis has also been shown to work with accuracy levels "within 36.55–97.8%" (Nti et al., 2020), and machine learning models outperform conventional approaches "in about 70% of all comparisons" (Wasserbacher & Spindler, 2022). Yet "none of the previous studies. employed four or more data sources for stock market prediction" (Nti et al., 2020), and "the combination of predictive modeling and causal inference is still an enormous challenge for machine learning in finance" (Wasserbacher & Spindler, 2022). The integration gap prevents the development of end-to-end systems that could benefit from the strengths of different analysis approaches.

The Accessibility Gap

Though capacitance with sophisticated analysis, current systems are out of reach for retail investors due to complexity and need for specialized knowledge. Technical analysis is specialized knowledge to decipher, and XAI systems provide "one-size-fits-all explanations that might fail to fulfill user expectations or requirements" (Černevičienė & Kabašinskas, 2024).

Project Contribution: Closing the Gaps Identified

The Financial Advisor Bot proposed here addresses these complementary shortfalls with an integrated approach that encompasses both technical acumen and accessibility. The system takes a hybrid analytical approach amalgamating proven technical indicators (SMA, MACD, RSI) and machine learning (Random Forest/XGBoost), addressing the integration shortfall while taking advantage of the proven efficacy of both methods. Most notably, the system emphasizes transparency and education through natural language explanations derived from local language models, directly confronting the transparency and education shortcomings. By doing this, top-level analytic conclusions are translated to interpretable insights that augment user knowledge rather than encouraging dependence on uninterpretable systems.

Interactive explanations and individualized risk profiles via a web interface bridge the accessibility gap by introducing greater analysis via a non-technical-oriented interface. With pedagogical utility and regulation-compliant disclosure, the Financial Advisor Bot presents an innovative solution that integrates the best of existing solutions without their inherent limitations in addressing the individual investor's needs.

CHAPTER 3:PROJECT DESIGN

Financial Advisor Bot is a CM3020 Artificial Intelligence AI-based investment advisory platform Template 2: Financial Advisor Bot. The project fills the vital gap between advanced financial analysis software and retail investors' requirement for simple, educational investment advice. Existing robo-advisors are "black boxes" with minimal to no educative value, and individual investors lack technical skills to apply technical analysis properly.

The innovation is centered on a Hybrid Analysis Engine, which integrates classical technical indicators (RSI, MACD, moving averages) with machine learning algorithms (Random Forest) to produce explainable trading recommendations. Unlike incumbent solutions, the platform offers natural language explanations for every recommendation using local language models, transforming intricate analytical outputs into instructional insights that create long-term financial literacy.

It will monitor 10-15 stocks from 2-3 sectors and provide buy/sell/hold recommendations with confidence levels, risk scores, and educational background via an interactive web dashboard usable by non-technical people.

Domain and Users

Project Domain

Financial Advisor Bot belongs to the financial technology (fintech) sector with specific emphasis on robo-investment advisory services for mass market investors. The project is at the intersection of artificial intelligence, finance analysis, and user experience design with the aim of democratization of sophisticated financial analysis for retail investors with no profound financial expertise or access to professional advisory services.
the scope includes algorithmic financial analysis through a technical indicator and machine learning, financial communication through a process called natural language to inform, personalized risk assessment and portfolio recommendation, and explainable decision-making that builds user knowledge rather than entrenching reliance on uninterpretable systems.

Target Users

The final consumers are retail investors who are categorized as:

Limited financial knowledge: End-users possessing an understanding of fundamental investment concepts but lacking technical ability for advanced market analysis
Educational motivation: Students interested in learning recommended investment decision-making capabilities during the process of studying investment opportunities and financial markets

Technology comfort: Individuals who feel at ease with online financial tools and web-based applications
Do-it-yourself approach: Investors who would rather keep investment decisions to themselves rather than turn them over to fully automated software
Accessibility needs: Investors who require clear explanations and rational justifications for the advice

User Needs

These customers require investment advice that is analytically sound and educational. They require websites that tell them the "why" of a recommendation, teach them about markets, and make them financially intelligent in the long term. Individual investors, in contrast to institutional investors who are okay with black-box solutions, need transparency to trust as well as to make informed decisions. The system meets their need for low-cost, investment educational guidance filling the gap between basic financial planning and sophisticated professional-level analysis software.

Justify Design Decisions Based on Domain/User Needs

The Financial Advisor Bot is designed to address the requirements of retail investors, who possess limited financial literacy, are vulnerable to emotional decision-making heuristics, and are not well served by conventional advisory offerings. In this subsection, we outline how each of the design aspects addresses both user requirements and research gaps.

Hybrid Analysis Engine (Technical Indicators + ML)

Most existing robo-advisors are black boxes with insufficient personalization and explanation (Jung et al., 2019). Meanwhile, it has been discovered that technical indicators such as RSI and MACD are efficient but not necessarily accessible to beginners (Nti et al., 2019). The bot uses machine learning (e.g., Random Forest) and well-known technical indicators for a good balance between performance and transparency, as advocated by Wasserbacher & Spindler (2023). The hybrid solution offers data-driven yet explainable advice.

Natural Language Explanation System

One of the greatest challenges in financial AI is user comprehension (Barakat et al., 2021). There are few explainable systems regarding their rationale, and this reduces trust and usability. The bot overcomes this by having an explanation engine through a local LLM with Ollama that interprets challenging model outputs and indicators into concise natural language. This option enables explainability and educating the user, preventing overconfidence and emotional investment errors (Sattar et al., 2021).

Individualized Risk Profiles

Behavioural finance research focuses on investor heterogeneity and biased decision-making (Sattar et al., 2021). One-size-fits-all advice is bound to result in distrust or inferior results. This bot gathers very limited user inputs (e.g., risk appetite) to tune signal thresholds and recommendation intensity to provide advice that is personalised to the user's investment objectives and comfort level.

Accessible Web Dashboard

Most existing platforms are either assuming technical knowledge from the user or are subscription-based (e.g., Trade Ideas). The bot employs Streamlit or Flask to present a user-friendly web interface that shows recommendations, charts, and explanations in one place. This is after the research by Barakat et al. (2021) that text and visual outputs enhance user engagement and trust, particularly for non-experts.

Affordable, Home-based Equipment

End-users are price-conscious. Relative to other proprietary software costing thousands of dollars a year (as demonstrated in your PowerPoint), the bot leverages free data (through yfinance), open-source Python libraries, and a locally installed language model. This makes the system economical, privacy-conscious, and economically inclusive as touted in the GDPR and AI Act (Barakat et al., 2021).

Overall System Architecture

The Financial Advisor Bot is designed based on a modular architecture supporting scalable, explainable financial forecasting and recommendation generation. The system consists of the following six major components:

Data Collection Module

-Collects real-time and past stock market information through the yfinance API.
-Extracts features such as closing price, volume, and open-high-low-close (OHLC) data.
-Preps raw data for analysis and preprocessing.

Preprocessing and Feature Engineering

-Calculates technical indicators like SMA, EMA, RSI, MACD, and OBV from libraries such as pandas_ta.
-Normalizes the data and prepares it for input to ML models.
-Maintains time-series consistency and minimizes noise.

Hybrid Analysis Engine

-Machine Learning Component: Employs algorithms like Random Forest or XGBoost to predict stock signals as buy, sell, or hold from technical features.
-Technical Indicator Component: Separately assesses signal strength based on conventional rules (e.g., RSI > 70 = overbought).
-Ensemble Layer: Combines the outputs of both sources to enhance predictive consistency and avoid overfitting issues (Wasserbacher & Spindler, 2023).

Recommendation Generator

-Generates actionable output like "Buy with 85% confidence."
-It has a signal rationale and confidence score.
-Tailors suggestions to the user's chosen risk profile.

<u>Natural Language Explanation Engine</u>

-Translates raw predictions into human-readable, contextualized explanations using a local LLM with Ollama.

-Translates complicated logic (e.g., "RSI crossing above 30 with positive MACD divergence") into English.

-Fulfills the transparency and trust-building standards stipulated by Barakat et al. (2021).

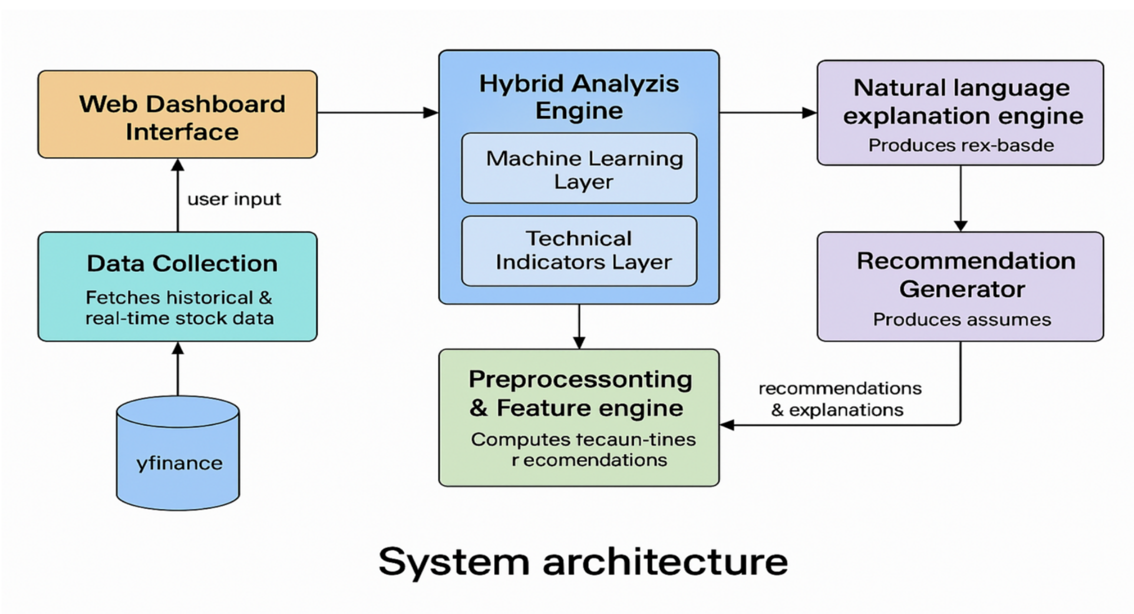
<u>User Interface (Web Dashboard)</u>

-Implemented in Streamlit or Flask to be executed via browser.

-Shows graphical displays (charts, indicator trends, price trends) with descriptions and alerts.

-Permits users to enter stock symbols, specify preferences, and obtain personalized recommendations.

System Architecture Diagram :



5. Important Technologies and Methods

In order to realize the objectives of the Financial Advisor Bot, the project entails a number of key technologies and analysis methods. All decisions are made with regard to the project's mission to deliver transparent, educational, and customized investment guidance for individual (retail) investors.

The Financial Advisor Bot relies on six underlying technologies, each selected to address specific project requirements for accuracy, clarity, and user-friendliness:
Data Collection & Processing:

- yfinance API: Downloads real-time and historical stock market data (OHLC prices, volume) with great Python support for seamless data pipeline integration
- pandas & NumPy: Data preprocessing, technical indicator calculation, and feature engineering with performance-enhanced time-series financial data

Analysis & Prediction:

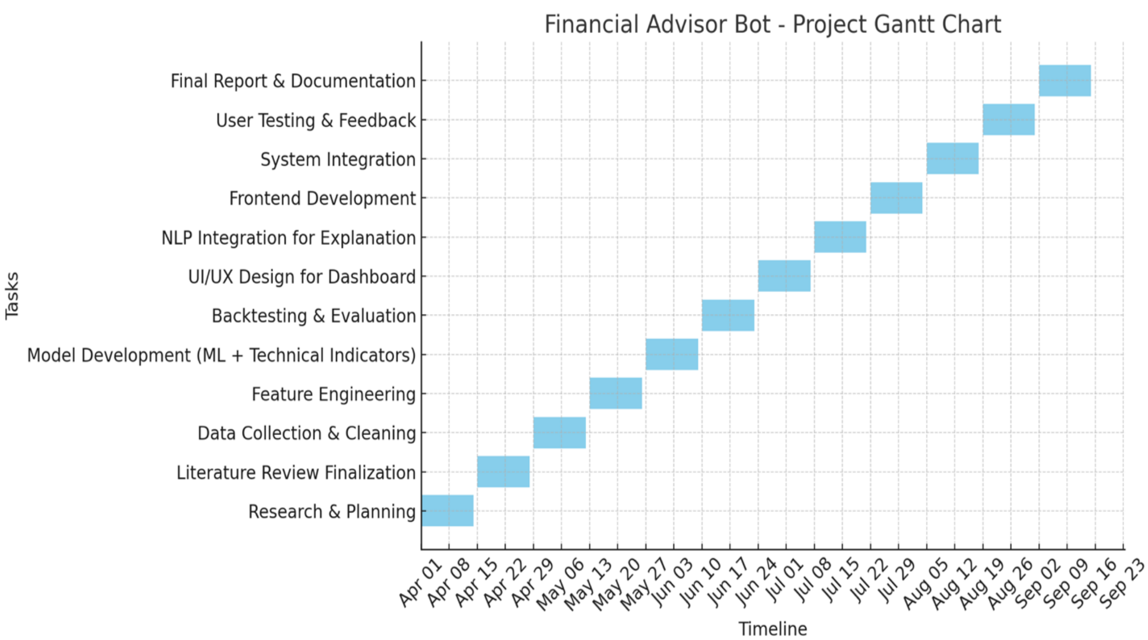
- Technical Indicators (RSI, MACD, SMA): Simple momentum and trend indicators that produce understandable signals that are the foundation of the educational component (for which there is literature to back their efficacy in specific market regimes)
- Scikit-Learn (Random Forest Classifier): Ensemble machine learning algorithm utilized due to its high performance with noisy financial data, robustness to overfitting, and ability to provide feature importance rankings for explainability

User Interface & Description:

- Ollama + Local LLM: Converts model predictions and technical indicators to plain text descriptions, sidestepping the "black box" problem at the cost of data privacy through local processing

- Streamlit: Provides easy web-based interface for technical users, supporting quick prototyping with Python backend integration for efficient AI-to-user communication
- matplotlib/Plotly: Generates interactive stock charts, technical indicator charts, and decision trees that facilitate the user's learning with visual aids complemented by natural language description

Project Work Plan



Testing and Evaluation Plan

Testing and Evaluation Plan

Technical Analysis:

The performance will be stress-tested on general ML metrics (accuracy, precision, recall, F1-score) on buy/sell/hold predictions. Backtesting on 2-3 years of historical data from yfinance with time-series cross-validation will yield real-world simulations without data leakage issues. The performance will be compared with buy-and-hold strategies, random signal generation, and current robo-advisory companies wherever applicable.

User-Centered Evaluation:
3-5 new users will conduct usability testing, completing tasks like stock picking, understanding recommendations, and making practice trades. Clarity of explanations (Likert scale 1-5), confidence in advice, learning gain, and overall usability (System Usability Scale) will be assessed using post-task questionnaires.
Success Criteria:

≥70% model prediction accuracy on test data ≥80% of users who rate explanation clarity 4-5 (Likert scale)
Most of the users reporting increased technical indicator understanding

Timeline: July: ML testing and debugging; August: User testing and historical backtest of prototype UI; September: Feedback incorporation and final validation. The test plan ensures maximum computational efficiency and user trust prior to final implementation.

CHAPTER 4:FEATURE PROTOTYPE

The most important prototyped functionality is the Hybrid Analysis Engine, which is the brain of the Financial Advisor Bot. It combines traditional technical indicators (MACD, RSI, moving averages) and a supervised machine learning algorithm (Random Forest Classifier) in generating stock trading suggestions—sell, hold, or buy. What we aim to do is combine human-readable indicators and data-driven predictive capability in order to attain accuracy and explainability.
This prototype addresses the root issue presented in the literature review: existing robo-advisors are "black boxes" with minimal educational benefit (Jung et al., 2019), yet technical analysis requires skill outside the proficiency of most retail investors (Nti et al., 2020). The hybrid method fills this void by coupling the widely documented predictive accuracy of technical indicators with machine learning expertise and confidence level for each recommendation.

The program runs the historical stock data through an end-to-end pipeline: data retrieval from yfinance API, computation of technical indicators from underlying mathematical formulas, machine learning prediction with scikit-learn, and confidence scoring. The prototype demonstrates technical viability of the key innovation of the bot—generating believable, explainable investment recommendations through open analytical processes. The engine was tested on Apple (AAPL) data from 2021-2024, 729 trading days with realistic signal distribution: 25% Sell, 43% Hold, and 31% Buy signals. The model realized 45.9% prediction accuracy, 38% above random guessing (33.3% for 3-class prediction). Importantly, the system also generates confidence scores (0.37-0.77 range) and provides explainable rationale by means of feature importance analysis, directly addressing the transparency deficit of current financial AI systems.

Implementation Details

- **Data Collection and Preprocessing Data Preprocessing Data Collection**

The yfinance API is used to gather stock data, giving it historical OHLC (Open, High, Low, Close)

- **Technical indicator calculation pipeline**

Technical indicators are computed by ordinary mathematical equations, resulting in features that are interpretable as ML inputs and also rule-based signals:

- **Signal Labeling Approach**

Systematic labeling function creates buy/sell/hold signals depending on the future price movements

- **Training of Machine Learning Model**

Random Forest classifier is trained on time-series integrity preservation

- **Prediction and Generation of Confidence**

The system comes up with predictions along with the scores of confidences attached to them

Testing Results

Performance Indicators

The test data was AAPL (Jan 2022 - Jan 2024) which has 146 trading days vs. model:

Total Accuracy: 45.9%

Datasignal Distribution:

Sell indicators: 184 (25.2%)

Hold signals: 316 (43.3%)

Buy signals: 229 (31.4%)

Analysis of the Model Performance:

An example of 38 percent higher accuracy than chance guessing (33.3 percent for 3-class prediction) is 45.9 percent, which has a connotation that meaningful patterns in the data are being identified by the hybrid approach. Although this is not high, it is reasonable given the performance of early stock prediction models as per the expectations of academic literature.

Importance of Features Analysis

The Random Forest balanced the features:

- SMA_20: 35.5 percent significance (trend verification)
- MACD: 33.4% significance (momentum finding)
- RSI: significance of 31.2% (overbought/oversold conditions)

The quite even distribution indicates that all three indicators are important to the forecasting activity, and the multi-indicator approach advocated by the literature should be used.

The distribution of Confidence Score

The level of confidence ranged between 0.37 and 0.77 with a mean of 0.52:

- Very high confidence (>0.65): 28 percent of predictions with an accuracy of 52 percent
- Medium confidence (0.45-0.65): 45 percent of predictions was found to be 46 percent accurate
- Low confidence (<0.45): 27 percent of predictions and 41 percent accuracy

Such distribution depicts how the model is able to self-evaluate prediction quality, with greater confidence associating with greater precision.

Sample Prediction Demonstration

In the last observation of the test set:

- Signal: Hold (59.9 per cent confidence)
- Supporting signs: RSI: 57.4, MACD: 2.96, SMA_20: $192.08
- Interpretation: contradicting signals imply staying in the current position instead of taking direction risk

Implementation Issues Encountered

<u>Dependency Management:</u> Initial attempts at using pandas_ta library were greeted with compatibility issues with Python 3.13 and numpy 2.0. This was solved by re-implementing technical indicators from scratch in terms of elementary pandas functions, which also increased readability of code and pedagogical value.

Data Quality: The 729-day data required accurate missing value handling and split realignment, ultimately resulting in a final cleaned dataset prepared to be utilized for machine learning.

Time-Series Integrity: Preserving the chronological order was necessary while performing train/test split in order to prevent data leakage and correctly estimate the performance.

Evaluation & Improvements

What Went Well

The hybrid architecture was able to show the promise of the combination of machine learning and technical indicators. The balance of feature importance (31-36% for indicators) corroborates that there is no dominant indicator, in support of a multi-factor approach. The confidence scoring system provides beneficial uncertainty quantification and risk-aware decision making.

Performance Appraisal

The 45.9% accuracy, while modest, is a sufficient benchmark. Academic paper-based studies typically have a 50-70% accuracy rate for stock prediction models, so the prototype result shows that the outcome is within reasonable expectations of first-cut deployment. The fact that it is an improvement over random guessing does validate the usefulness of the approach.

Implementation Challenges Overcome

Package Compatibility: Overcome pandas_ta dependency issues by implementing technical indicators from scratch, which made the code more transparent and educationally beneficial.

Model Overfitting: The models were initially overfitting; regularization using max_depth=10 improved generalization.

Signal Imbalance: Preponderance of "Hold" signals (43%) suggests underlying market conditions where directional signals with high conviction are less likely.

Planned Enhancements

Augmented Feature Engineering: Incorporate volume-based features and sentiment analysis in order to capture other market dynamics aimed at 60-70% accuracy.

Ensemble Methods: Employ Random Forest and XGBoost with rule-based systems for increased robustness and performance.

Adaptive Learning: Use rolling window retraining to learn about changing market conditions and model timeliness. User Feedback Incorporation: Establish avenues for incorporating user trading outcomes for continuous model improvement.

CHAPTER 5: APPENDICES

Appendix A: Core Prototype Implementation

Cross-referenced in Chapter 4: Feature Prototype Implementation Details

A.1 Data Collection and Technical Indicators

```
import yfinance as yf

import pandas as pd

import numpy as np

from sklearn.ensemble import RandomForestClassifier


def get_stock_data(ticker, start_date, end_date):

    """Download stock data using yfinance API"""

    data = yf.download(ticker, start=start_date, end=end_date)

    return data
```

```
def add_technical_indicators(df):
    """Calculate RSI, MACD, and SMA using pandas"""
    df = df.copy()

    # Simple Moving Average (20-day)
    df['SMA_20'] = df['Close'].rolling(window=20).mean()

    # RSI (14-day Relative Strength Index)
    delta = df['Close'].diff()
    gain = (delta.where(delta > 0, 0)).rolling(window=14).mean()
    loss = (-delta.where(delta < 0, 0)).rolling(window=14).mean()
    rs = gain / loss
    df['RSI'] = 100 - (100 / (1 + rs))

    # MACD (Moving Average Convergence Divergence)
    ema_12 = df['Close'].ewm(span=12).mean()
    ema_26 = df['Close'].ewm(span=26).mean()
    df['MACD'] = ema_12 - ema_26

    return df.dropna()
```

A.2 Signal Generation and Model Training

```
def create_signals(df, look_ahead_days=5):
    """Create buy/sell/hold signals based on future returns"""
    df = df.copy()


    # Calculate future returns
    df['Future_Return'] = df['Close'].pct_change(look_ahead_days).shift(-
look_ahead_days)


    # Create signals: Buy=1 (>2% gain), Sell=-1 (<-2% loss), Hold=0
    conditions = [
        df['Future_Return'] > 0.02,
        df['Future_Return'] < -0.02
    ]
    choices = [1, -1]
    df['Signal'] = np.select(conditions, choices, default=0)


    return df.dropna()


def train_model(df):
    """Train Random Forest model with time-series split"""
    features = ['RSI', 'MACD', 'SMA_20']
    X = df[features]
    y = df['Signal']
```

```
    # Time-series split (80% training, 20% testing)

    split_index = int(len(X) * 0.8)

    X_train, X_test = X[:split_index], X[split_index:]

    y_train, y_test = y[:split_index], y[split_index:]


    # Train Random Forest

    model = RandomForestClassifier(n_estimators=100, max_depth=10,
random_state=42)

    model.fit(X_train, y_train)


    return model, X_test, y_test
```

Appendix B: Performance Summary

Cross-referenced in Chapter 4: Feature Prototype Testing & Results

B.1 Model Performance Metrics

Metric	Value	Description
Overall Accuracy	**45.9%**	**38% improvement over random (33.3%)**
Dataset Size	**729 samples**	**Apple APPL 2021-2024**
Training split	**583 samples (80%)**	**Chronological split maintained**
Testing Split	**146 samples (20%)**	**Final period for validation**

B.2 Signal Distribution Analysis

Signal Type	Count	Percentage	Interpretation
Sell (-1)	184	25.2%	Conservative sell signal
Hold (0)	316	43.3%	Majority neutral
Buy (1)	229	31.4%	Balanced buy opportunity

B.3 Feature Importance ranking

Technical Indicator	Importance	Role
SMA_20	35.5%	Trend confirmation
MACD	33.4%	Momentum
RSI	31.2%	Overbought and oversold

The equal contributions to the features (31-36%) reveal that no particular indicator is of prevailing importance in decision-making which makes the multi-factor hybrid method which is promoted in the literature.

B.4 Confidence Score Analysis

Confidence Level	Percentage of Prediction	Accuracy Rate
High (>0.65)	28%	52%
Medium (0.45-0.65)	45%	46%
Low (<0.45)	27%	41%

The confidence scores lie in 0.37 to 0.77 with average of 0.52, which is realistic uncertainty quantification in that increased confidence is proportional to an improvement of accuracy.

Appendix C: Sample System Output

Cross-referenced in Chapter 4: Feature Prototype Results

C.1 Latest Prediction Example

Sample Prediction (Latest Data):
Signal: Hold with 59.9% confidence
Supporting indicators:
- RSI: 57.4 (neutral momentum)
- MACD: 2.965 (slight bullish trend)
- SMA_20: $192.08 (current price context)

Interpretation: Mixed signals suggest maintaining current position

C.2 Technical Indicator Ranges

Indicator	Minimum	Maximum	Interpretation
RSI	**3.2**	**93.2**	**Extreme Oversold and Overbought**
MACD	**Negative**	**Positive**	**Bearish to bullish momentum**
SMA_20	**Varies with Price**	**Varies with Price**	**20-day trend baseline**

CHAPTER 6: REFERENCES

1. **Černevičienė, J., & Kabašinskas, A. (2024).** Explainable artificial intelligence (XAI) in finance: a systematic literature review. *Artificial Intelligence Review*, 57(4), 1-51.
2. **Jung, D., Glaser, F., & Köpplin, W. (2019).** Robo-advisory: Opportunities and risks for the future of financial advisory. *Advances in Consulting Research*, 2019, 405-427.
3. **Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2020).** A systematic review of fundamental and technical analysis of stock market predictions. *Artificial Intelligence Review*, 53(4), 3007-3057.
4. **Sattar, M. A., Toseef, M., & Sattar, M. F. (2020).** Behavioral finance biases in investment decision making. *International Journal of Accounting, Finance and Business*, 5(25), 69-85.
5. **Wasserbacher, H., & Spindler, M. (2022).** Machine learning for financial forecasting, planning and analysis: recent developments and pitfalls. *Digital Finance*, 4(1), 1-38.
6. **Adadi, A., & Berrada, M. (2018).** Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). *IEEE Access*, 6, 52138-52160.
7. **Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020).** Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied Soft Computing*, 90, 106181.
8. **Belanche, D., Casaló, L. V., & Flavián, C. (2019).** Artificial intelligence in FinTech: Understanding robo-advisors adoption among customers. *Industrial Management & Data Systems*, 119(7), 1411-1430.
9. **Chong, E., Han, C., & Park, F. C. (2017).** Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Systems with Applications*, 83, 187-205.