VIETNAM NATIONAL UNIVERSITY, HANOI

INTERNATIONAL SCHOOL

GRADUATION PROJECT

Investing Decision Support System: Stock Portfolio Optimization using Machine Learning

Student's name

Lê Tuấn Sơn - 21070340

Hanoi - Year 2025

VIETNAM NATIONAL UNIVERSITY, HANOI

INTERNATIONAL SCHOOL

GRADUATION PROJECT

Investing Decision Support System: Stock Portfolio Optimization using Machine Learning

SUPERVISOR: PhD. Nguyễn Doãn Đông

STUDENT: Lê Tuấn Sơn

STUDENT ID: 21070340

SUBJECT CODE: INS4011

MAJOR: Business Data Analytics

Hanoi - Year 2025

Acknowledgement

I want to express my sincere gratitude to everyone who has supported and guided me throughout my journey in completing this graduation thesis in the major of Business Data Analytics.

First, I would like to express my appreciation to my thesis advisor, Ph.D. Nguyen Doan Dong, for his guidance, insightful feedback, and continuous support at VNUIS. His mentorship has been both inspiring and instrumental throughout this project. His expertise and encouragement have significantly shaped this project and greatly enhanced my academic growth.

Special thanks also to my classmates and friends for their constructive feedback, lively discussions, and unwavering support during tough times. Your camaraderie made this academic journey truly memorable and rewarding.

Finally, my deepest gratitude goes to my family, whose endless love, patience, and encouragement have provided me the strength and motivation to pursue and complete this milestone. Their belief in my abilities has been my most significant source of inspiration. I am incredibly grateful to my parents for their love, care, and sacrifices in educating and preparing me for my future.

Thank you all for being an integral part of my journey and making this accomplishment possible.

Table of Contents

List of Figures

List of Tables

Abstract

As AI prediction techniques and machine learning continue to improve, accurate stock price forecasting has become a valuable tool for enhancing portfolio selection and informed investment decisions. Deep learning, when paired with traditional financial theory, marks a significant shift in quantitative finance, particularly in markets like Vietnam, where most models struggle to comprehend the complexities of market volatility and pricing.

For this project, I developed a two-stage framework for optimizing stock portfolios in the Vietnamese market. The framework uses neural network predictions, sentiment analysis with large language models (LLMs), signal processing, hyperparameter tuning, and a coherent risk measure optimization. The project included two stages: Stock price prediction and Portfolio selection. In the first stage, I processed around 15,000 Vietnamese financial news articles to generate sentiment scores using Google's Gemini API. These scores, ranging from -1 to 1, reflect the market emotions and news that may affect stock movements. I also denoised the VN100 price series using a Savitzky-Golay filter to smooth the price trend. These sentiment scores and denoised price series, along with technical indicators, were inputs for an LSTM neural network. Then, the LSTM's hyperparameters were optimized using a random search to maximize predictive accuracy and prevent overfitting. For the second stage, I employed the mean Conditional Value-at-Risk (CVaR) model, an optimization framework, to determine the optimal stock allocation strategies. I considered multiple key factors, including potential returns from LSTM predictions, prediction accuracy measured by Mean Prediction Accuracy (MPA) metrics, individual stock growth rates, and overall risk assessment measures, to select the stock list.

The experimental results show the effectiveness of the proposed framework across multiple performance metrics. The LSTM model achieves an average test Mean Prediction Accuracy (MPA) of 0.9837 and an R^2 of 0.8962. Furthermore, the portfolio optimization strategy delivered exceptional returns, achieving annualized returns ranging from 6.24% (90% confidence) to 9.11% (99% confidence) over the 29-month backtesting period from January 2023 to June 2025.

Chapter 1. Introduction

With the advancement of artificial intelligence technology, accurately forecasting stock market trends and formulating effective investment portfolios have emerged as central priorities in investment research (C.-H. Wang et al., 2024). In the Vietnamese stock market, fintech adoption has begun to stimulate the use of artificial intelligence and big data analytics in trading and investment (Nga, 2024). However, Vietnam's market experiences rapid capital inflows and high volatility, making accurate forecasting both challenging and essential. (P.-C. Wang & Vo, 2025).

Simultaneously, numerous retail and institutional investors continue to depend on heuristics or rudimentary mean-variance allocation methods. Stock market data is often regarded as a time series, characterized by volatility, nonlinearity, non-stationarity, and high noise (C.-H. Wang et al., 2024). Whereas, Traditional time-series methods (e.g., ARIMA or GARCH) and basic machine-learning models often underperform in this context (Sheikh, 2024). They fail to capture changes or nonlinear patterns of investor behavior. Moreover, many investors still rely on technical indicators, such as moving averages or the RSI, which do not explain for shifts in market sentiment. Stock trends, however, are influenced by numerous factors, including interest rates, inflation rates, and financial news (Li et al., 2019). As a result, investors face two core challenges: high-frequency fluctuations ("noise") often obscure genuine trend information in the VN100 stock list, and Models that rely solely on price usually miss how sudden changes in news sentiment can signal big rallies or sell-offs, leaving portfolios vulnerable to unexpected losses. Furthermore, the Vietnamese market has high volatility; traditional forecasting and portfolio allocation methods often fall short, resulting in difficulties in selecting and combining different assets to achieve the optimal outcome in terms of risk and return.

However, artificial intelligence methods have demonstrated significant potential in stock prediction (C.-H. Wang et al., 2024). Recently, some Fintech companies have provided services to investors, such as user-friendly artificial intelligence investment platforms that help investors manage their investment portfolios. Additionally, some investment applications offer free basic stock trading services, as well as real-time and continuous personal financial news updates in the investment process. The application of modern technology has created a more level field for all investors in the stock market (Nga, 2024). By integrating historical stock data, sentiment analysis of market emotions, and macroeconomic indicators, AI models offer accurate predictions of stock prices. Comparative studies have found that deep learning models outperform traditional models in processing large-scale data and complex tasks (C.-H. Wang et al., 2024). Moreover, Portfolio optimization problems also require robust, algorithmically driven portfolio optimization frameworks that not only aim for higher returns but also systematically manage and mitigate the downside risks inherent in such dynamic markets. Markowitz (1952) proposed a Mean-Variance (MV) model that seeks to maximize the expected return while limiting the variance, or reduces the variance given a minimum expected return (C.-H. Wang et al., 2024). Afterward, Value at Risk (VaR) emerged as an alternative risk metric, estimating the maximum loss within a specified range under normal market conditions. Thus, the mean-VaR model began to be applied in portfolio optimization (C.-H. Wang et al., 2024). However, VaR has some drawbacks, including non-subadditivity, lack of convexity, and limited information on tail losses. To address these limitations, Conditional Value at Risk (CVaR) was subsequently introduced as a consistent risk measurement, adhering to principles such as translation invariance, positive homogeneity, and monotonic additivity (Staino et al., n.d.).

In summary, this project will implement an Investing Decision Support System using a two-stage approach, incorporating AI-powered stock price prediction and mean-CVaR portfolio allocation models. Firstly, I will collect the stock prices for the VN100 stocks list and the financial news to calculate the sentiment score from 2023 to 2025. Then, in the **first stage**, we will employ a Savitzky–Golay (SG) filter to denoise raw price series and reveal latent trend structures. Next, we will extract daily sentiment scores using an LLM model from nearly 15000 Vietnam financial news articles. After that, these denoised prices and sentiment scores will be fed into a Long Short-Term Memory (LSTM) network to predict the future stock price. To maximize prediction accuracy, we will tune the LSTM's hyperparameters using random search. In the **second stage**, the return forecasts generated by the Long Short-Term Memory (LSTM) model are integrated into a mean-Conditional Value-at-Risk (mean-CVaR) optimization framework. Monthly, the optimizer determines weights that effectively balance expected returns, forecast confidence, and downside exposure elements that are essential for managing drawdowns during Vietnam's episodic market corrections. Finally, the method was backtested from 2023 to 2025, demonstrating its performance across all confidence levels. Total returns ranged from 24.47% (90% confidence interval) to 37.09% (99% confidence interval) over the period, with annualized returns of 6.24% to 9.11%.

Chapter 2. Literature Review

2.1. Traditional Stock Prediction Method

In stock market prediction problems, researchers usually aim to produce two types of outputs: future stock values and stock price movements (P.-C. Wang & Vo, 2025). This dual approach addresses the varied needs of market participants, from algorithmic traders needing precise price targets to portfolio managers looking for optimal entry and exit points (Fama & French, 2004).

Previously, stock market predictions have relied on traditional statistical and econometric models, such as the ARIMA and GARCH models, which are key tools in quantitative finance that have been used for a long time. These methodologies operate under assumptions of linearity and stationarity in financial data, constructing their predictive capabilities based exclusively on historical price patterns and statistical relationships. These models rely on the linearity and stationarity of the data, utilizing historical data to predict future stock prices. They analyze factors like trends, seasonality, and cyclicality to make these predictions. However, traditional approaches have significant limitations when it comes to modeling complex, non-linear relationships (C.-H. Wang et al., 2024). They struggle to capture market shifts, handle shocks, and adapt to the impact of external factors, such as macroeconomic and geopolitical events. These limitations reduce their predictive power and reliability in highly volatile markets. (Cont, 2001a; C.-H. Wang et al., 2024).

The ARIMA (Autoregressive Integrated Moving Average) model represents one of the most important and theoretically grounded approaches in time series analysis and financial forecasting (Hyndman & Athanasopoulos, 2018). The ARIMA model, developed by Box and Jenkins in 1974, employs lagged methods to provide a reliable prediction model. The fundamental principle of ARIMA models is that future values can be expressed as a linear combination or weighted mixture of past observed values and past prediction errors, creating a framework that captures both autoregressive tendencies and moving average components within the data structure (Hamilton, 1994). In summary, the ARIMA model is influenced by three distinct variables (p, d, q), where p represents the auto-regressive variables, d represents the number of differences, and q represents the moving average variable (P.-C. Wang & Vo, 2025). ARIMA models often fail to capture the nonlinear behavior inherent in financial time series, as they assume stationarity and linear dependence, making them ineffective for modeling series that exhibit nonlinear patterns or regime shifts (P.-C. Wang & Vo, 2025).

The Generalized Autoregressive Conditional Heteroskedasticity (GARCH) model represents a significant advancement in statistical modeling, specifically engineered and optimized to analyze, understand, and forecast the volatility patterns in financial time series data, particularly in equity, currency, and commodity markets (Bollerslev, 1986; Engle, 1982).GARCH is a statistical model specifically designed to analyze and forecast volatility in time series data, particularly in financial applications. Unlike traditional models that assume constant error variance (homoskedasticity), GARCH models treat changing variance over time (heteroskedasticity) as something to be modeled and predicted (Alamu & Siam, 2024). This capability makes GARCH models valuable for risk management applications, pricing, and volatility forecasting in finance (Alexander, 2001). However, GARCH models require more computational resources than simpler alternative approaches, necessitating significant processing and advanced optimization algorithms for parameter estimation, especially when working with large datasets or high-frequency financial data (Hansen & Lunde, 2001; Y. Huang et al., 2021a). GARCH assumes error terms follow a normal distribution, which often fails for real-world financial returns that typically show heavy tails, skewness, excess kurtosis, and non-normal characteristics, leading to model misspecification and inaccurate risk assessments (Mutinda & Langat, 2024).

2.2. Machine Learning, Deep Learning in Stock Prediction

The stock market is characterized by volatility, nonlinearity, non-stationarity, and high noise (C.-H. Wang et al., 2024), which creates a challenge for traditional prediction models that frequently suffer from inadequate accuracy and insufficient precision for practical investment applications (Cont, 2001a). These intrinsic market characteristics make it exceptionally difficult for individual and institutional investors to effectively analyze and predict market trends using conventional approaches that rely solely on technical analysis frameworks, including company financial charts, broad market indices, fundamental analysis metrics, and qualitative textual information derived from diverse sources such as financial news blogs, newspaper reports, analyst recommendations, and social media sentiment (Bollen et al., 2011). It is challenging for investors to analyze and predict market trends based on all the technical analysis, including company charts, market indices, and textual information such as news blogs or newspapers, as well as traditional methods. The limitations of traditional econometric methods in capturing these complex market dynamics have prompted researchers to explore and investigate new artificial intelligence and machine learning approaches designed to identify, learn, and predict markets automatically (Atsalakis & Valavanis, 2009; Li et al., 2019). Subsequently, machine learning models such as SVM, RF, KNN, and others emerged (C.-H. Wang et al., 2024).

2.2.1. Classical Machine Learning Approach

Kumbure et al. (2022) investigated the application of machine learning techniques to stock market prediction from 2000 to 2019. This research analyzed the markets and stock indices used in predictions, 2173 unique predictor variables, and various machine learning techniques (Kumbure et al., 2022).

Support Vector Machines, initially developed by Vapnik, have proven particularly effective in financial applications due to their ability to handle high-dimensional data and their robust performance in noise and outliers (Cortes & Vapnik, 1995; Tay & Cao, 2001). The SVM approach constructs optimal hyperplanes that maximize the margin between different classes, making it well-suited for binary classification problems such as predicting whether stock prices will increase or decrease (W. Huang et al., 2005). Some studies have shown that SVMs have a significant impact on stock market prediction compared to traditional methods. Kim (2003) showed that SVM outperformed back-propagation neural networks in predicting the direction of daily stock price changes for the Korean Stock Price Index. Breiman introduced Random Forest algorithms, which have gained traction in financial forecasting due to their ensemble learning approach, which combines multiple decision trees to reduce overfitting and improve accuracy (Khaidem et al., 2016).

The Random Forest methodology is valuable in stock prediction because it handles numerical and categorical variables, automatically selects features, and provides insights into variable importance, making it ideal for diverse datasets with technical indicators and fundamental analysis metrics (Ballings et al., 2015). Patel et al. (2015) demonstrated that Random Forest models outperformed artificial neural networks and SVM in predicting stock price movements across Indian stock market indices.

The k-Nearest Neighbors algorithm, despite its conceptual simplicity, has shown remarkable effectiveness in financial time series prediction due to its non-parametric nature and ability to capture local patterns in data without making strong assumptions about underlying distributions (Fernández-Rodríguez et al., 2000). Moreover, Chakravorty and Elsayed (2025) examine the effectiveness of algorithms like decision trees, random forests, support vector machines (SVM) with different kernels, and K-Means Clustering using a dataset of Tesla stock transactions (Chakravorty & Elsayed, 2025). The research states that Support Vector Machines (SVMs) with the Radial Basis Function (RBF) kernel and Random Forest are the most effective models for predicting stock prices using insider trading data (Chakravorty & Elsayed, 2025).

Moreover, ML methods can be aggregated; Huang et al. (2021b) used the machine learning algorithms: FNN, ANFIS, and RF, which are trained to predict the quarterly relative return of each of the 70 stocks (Y. Huang et al., 2021a). The results show that all three machine learning methods they experimented with are capable of constructing stock portfolios that outperform the market without any input of expert knowledge, if fed with enough data (Y. Huang et al., 2021b). By applying feature selection and aggregating the different algorithms, the aggregated model achieves a Portfolio Score of 0.759 and -0.335 for the "Buy" and "Sell" portfolios, respectively (Y. Huang et al., 2021b). Moreover, Illa et al. (2022) proposed a pattern matching approach integrating Random Forest (RF) and Support Vector Machine (SVM) to develop stock price predictors and optimize trading strategy profitability. Their method combines Random Forest in handling feature interactions and robust predictions with SVM to find optimal decision boundaries, enhancing prediction accuracy and improving risk-adjusted returns.

2.2.2. Deep Learning Approach

Deep learning architectures have a significant impact on financial prediction, offering the capability to understand the model's complex, nonlinear relationships and temporal dependencies that traditional machine learning struggles to capture effectively. Deep learning models, particularly those developed for processing sequential data, have demonstrated superior performance in capturing the intricate patterns, long-term dependencies, and subtle market dynamics that characterize contemporary financial markets. Jiang (2021) conducted a systematic review of deep learning models for stock market trend prediction, categorizing data sources like price, fundamental, and alternative data. The research studies various neural network architectures, from feedforward to transformer models, and analyzes their performance metrics (Jiang, 2021). They also tackled implementation challenges and concerns about reproducibility.

Convolutional Neural Networks (CNNs), which were initially developed for image processing, have been applied in innovative ways to financial time series analysis. In this context, price charts and technical indicators are treated as two-dimensional images that can be analyzed for pattern recognition (Sezer et al., 2020). Research by Hoseinzade and Haratizadeh (2019) showed that CNNs can effectively extract meaningful features from candlestick charts and technical indicators, outperforming traditional machine learning methods in predicting stock price movements. The spatial invariance properties of CNNs make them well-suited for identifying recurring patterns in financial charts, regardless of their temporal position. Tu (2025) proposed a novel hybrid architecture that leverages the complementary strengths of CNNs and Transformers. They integrated CNNs for capturing short-term dependencies and Transformers for modeling long-term trends within financial time series which is applied to predict stock price movements for the S&P 500. The results show that the model outperformed all baseline methods across both 2-class and 3-class classification tasks, capturing both short-term and long-term dependencies in financial time series data (Tu, 2025).

Recurrent Neural Networks (RNNs), along with Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRUs), are the efficient approaches for analyzing sequential financial data. They effectively process time-dependent information and recall past events, tackling the challenge of capturing temporal dependencies in financial time series, where current prices stem from intricate interactions of historical events, market sentiment, and economic conditions over various time horizons.

Therefore, a comparable approach was employed by Nabipour et al. (2020), which utilized decision trees, boosting techniques, and Long Short-Term Memory (LSTM) networks to classify the stock market segment of the Tehran Stock Exchange. Additional studies have explored hybrid approaches that combine multiple machine learning techniques to leverage their complementary strengths. Nelson et al. (2017) investigated the integration of technical analysis indicators with LSTM networks, demonstrating improved prediction accuracy when technical indicators are used as input features. In conclusion, the research showed that the LSTM model outperformed the other techniques, which improves the price forecasting of the stock (Tuan et al., 2022).

2.3. Long Short-Term Memory in Stock Prediction

Long Short-Term Memory (LSTM) networks have emerged as a prominent deep learning architecture for stock prediction due to their exceptional ability to effectively manage, process, and learn from sequential data with complex temporal dependencies (Gers et al., 2000; Hochreiter & Schmidhuber, 1997). Stock market data is inherently time-series and has complex temporal relationships. It often exhibits multiple challenging characteristics, including extreme volatility with sudden price changes, inherent nonlinearity in the formation of prices, persistent non-stationarity with changing statistical properties over time, and high-frequency noise resulting from market microstructure effects. Traditional econometric and statistical methods often struggle to effectively and accurately handle these characteristics. (Cont, 2001b; Mantegna et al., 2000).

2.3.1. LSTM Foundation and Architecture

LSTM, an extension of Recurrent Neural Networks (RNN), addresses issues such as vanishing and exploding gradients, enabling effective long-term dependency learning in sequential data. It is a type of neural network designed to analyze and understand data with long-term dependencies, as there are fundamental computational problems, such as vanishing and exploding gradients, that severely limit the ability of conventional recurrent neural networks (RNNs) to capture long-term dependencies in sequential data (Pascanu et al., 2013). The architectural design of an LSTM network follows a multi-layered structure that includes an input layer for preprocessing data, one or more hidden layers with core LSTM memory cells, and an output layer that generates predictions based on processed information. The hidden layer, the heart of the LSTM architecture, contains memory cells and gate units that work together to store, update, and forget information over time selectively. The architecture of hidden layers may also incorporate conventional hidden units, which enhance computational capacity and serve as inputs for both gate units and memory cells, thereby establishing a complex, interconnected network of information processing pathways. All computational units within the network, with the prominent exception of gate units, which fulfill specialized control functions, sustain directed connections that act as inputs to all units in subsequent layers, thereby facilitating a comprehensive information flow structure that enables advanced pattern recognition and temporal modeling capabilities. Recognition and temporal modeling capabilities (Greff et al., 2017; Hochreiter & Schmidhuber, 1997).

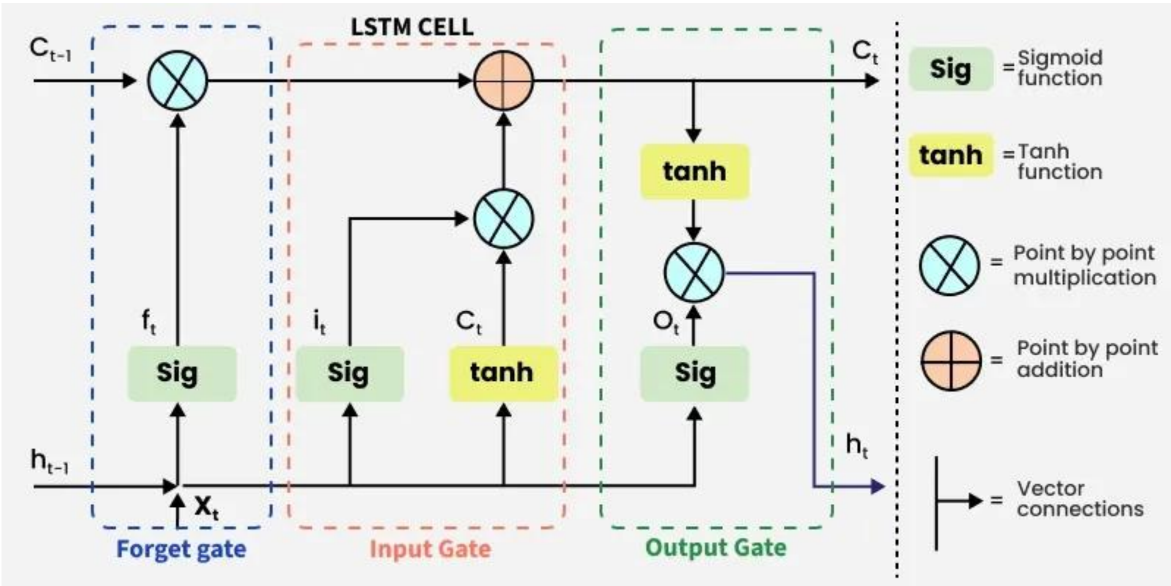


*Figure 2.1. Structure of LSTM (*What Is LSTM - Long Short Term Memory?*, 19:24:13+00:00)*

2.3.2. Gate Mechanisms and Information Processing

An LSTM network typically consists of an input layer, a hidden layer, and an output layer, as illustrated in Figure 1. The (fully) self-connected hidden layer contains memory cells and corresponding gate units (for convenience, we refer to both memory cells and gate units as being located in the hidden layer). Unlike traditional RNNs, which rely on a single neural network layer utilizing the hyperbolic tangent (tanh) activation function, LSTM networks incorporate a more sophisticated control mechanism comprising three specialized logistic sigmoid gates and one tanh layer, each serving specific functions in the information processing pipeline (Hochreiter & Schmidhuber, 1997).

The **forget gate** mechanism is a key part of the LSTM's memory management. It removes information from the cell state that's no longer relevant or useful for current predictions. The output from the forget gate is processed through a sigmoid activation function, which produces a binary-like result ranging from 0 to 1. Values close to 0 indicate that the information should be discarded, while values approaching 1 signal that the information should be preserved for future computations. This selective forgetting mechanism is valuable in financial applications, where market conditions and economic cycles can change significantly, making historical patterns obsolete or misleading. Fundamental changes that render historical patterns obsolete or misleading.

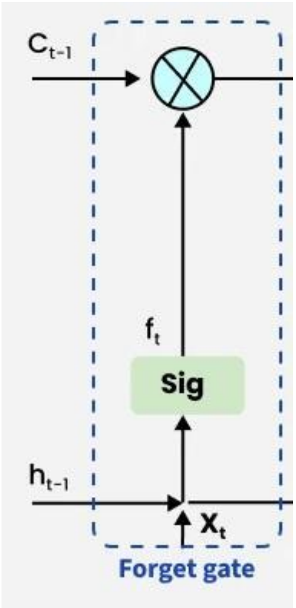


*Figure 2.2. LSTM forget gate (*What Is LSTM - Long Short Term Memory?*, 19:24:13+00:00)*

The **input gate** plays a vital role in identifying valuable new information into the cell state. This effectively updates the network's memory with relevant current data. The input gate works by first creating a candidate vector using the tanh activation function. This function generates outputs between -1 and +1, covering all possible values from the previous hidden state and the current input. Next, the values from this vector are multiplied by the regulated values from the input gate. This results in the selective incorporation of only the most valuable and relevant information into the updated cell state.

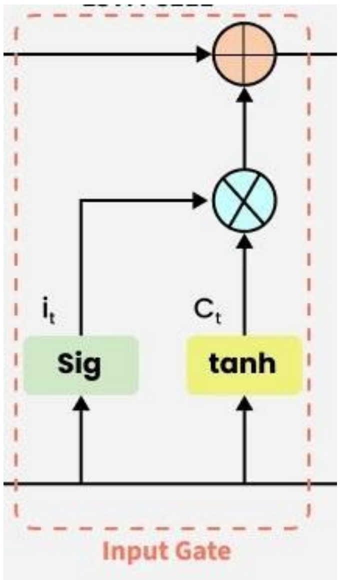


*Figure 2.3. LSTM input gate (*What Is LSTM - Long Short Term Memory?*, 19:24:13+00:00)*

The **output gate** performs the critical function of extracting and presenting useful information from the current cell state as the network's output, effectively controlling what information flows to subsequent layers and external outputs (Gers et al., 2000). The output gate mechanism operates through a multi-step process: first, the tanh function generates a normalized vector from the current cell state; next, the sigmoid function processes and regulates information filtered from both the previous hidden state (h_{t-1}) and current input (x_t). Finally, the vector and regulated values multiply for output to the next cell.

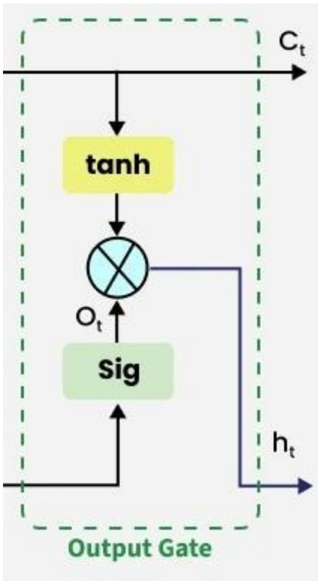


*Figure 2.4. LSTM output gate (*What Is LSTM - Long Short Term Memory?*, 19:24:13+00:00)*

2.3.3. Applications and Studies in Financial Prediction

Due to its structures and effectiveness in handling time-series data, LSTM is often used in analyzing time series data, such as stock market prices (Gülmez, 2023). Recent studies have shown LSTM's superior performance in stock price prediction tasks compared to classical models like Support Vector Machines (SVM), Random Forests (RF), and k-Nearest Neighbors (KNN), particularly due to its ability to handle larger-scale and more complex data effectively (C.-H. Wang et al., 2024).

In the literature, many studies have applied LSTM to predict stock prices. Chen et al. (2021) addressed the fundamental limitation of typical offline neuro-fuzzy systems, which require comprehensive training data to capture all possible system behaviors. This is a challenging requirement in financial applications, due to the dramatic and unpredictable change in data distributions that characterize volatile market periods. The authors developed a technique integrating a neuro-fuzzy system with the Hammerstein-Wiener model into a five-layer network. The model's efficacy was evaluated across three financial stock datasets, demonstrating superior performance compared to existing neuro-fuzzy systems and the conventional Hammerstein-Wiener model. Li et al. (2019) investigated the predictability of investor sentiment in the Chinese stock market using online user-generated content for sentiment analysis, comparing text classification algorithms, price forecasting models, time horizons, and information update systems. Their study employed a comparative methodology that examined various text classification algorithms, different price forecasting models, multiple time horizons ranging from intraday to monthly predictions, and information update systems to determine optimal configurations for sentiment-based stock prediction. Elminaam et al. (2024) have compared three different LSTM architectures: Vanilla LSTM, Stacked LSTM, and Bidirectional LSTM, which revealed that the Bidirectional LSTM consistently outperformed the other variants across multiple datasets, particularly when optimized with the RMSprop algorithm. Furthermore, it shows that while LSTM models are generally robust in handling stock price data, the specific characteristics of the BiLSTM architecture make it especially powerful in capturing the complex, dual-influenced trends common in stock market data (Elminaam et al., 2024).

Rezaei (2021) presented hybrid algorithms, including the CEEMD-CNN-LSTM and EMD-CNN-LSTM architectures. These techniques utilize advanced signal processing methods, such as Complete Ensemble Empirical Mode Decomposition (CEEMD) and Empirical Mode Decomposition (EMD), to extract meaningful features and reduce noise, thereby improving the accuracy of predicting one-step-ahead stock prices. By combining the strengths of signal decomposition techniques for noise reduction and feature extraction with the pattern recognition capabilities of Convolutional Neural Networks (CNNs) and the temporal modeling strengths of LSTM networks, these hybrid approaches excel. Moreover, Mehtab and Sen (2020) have proposed five deep learning-based regression models for the prediction of NIFTY 50 index values. Their propositions included two CNN models and three LSTM models, which were built, optimized, and then tested on the daily index values of NIFTY 50. While all the models exhibited high levels of accuracy in their forecasting performance, the univariate encoder-decoder convolutional LSTM was found to be the most accurate model. Additionally, Zhang et al. (2023) proposed a more advanced LSTM network architecture that incorporates residue-driven support vector regression features. This creates a sophisticated hybrid system that leverages the complementary strengths of both neural network and kernel-based learning approaches. Their methodology uses a genetic algorithm optimization framework to systematically optimize key model parameters, ensuring the development of superior network structures that deliver satisfactory experimental results across multiple real-world financial datasets.

2.4. Stock Portfolio Optimization

2.4.1. Portfolio Optimization Foundations and Classical Approaches

Portfolio management is an analytical process of selecting and allocating a group of investment assets in which the portion of allocated investment is persistently changed to optimize expected return and risk tolerance (Markowitz, 1952). This fundamental concept has evolved into one of the most important theoretical frameworks in modern finance, providing the mathematical and conceptual foundation for institutional asset management, individual investment strategies, and risk management frameworks worldwide.

The Markowitz mean-variance (MV) model, first developed in 1952, serves as the foundation of portfolio theory, a widely used and recognized framework in portfolio management (Chaweewanchon & Chaysiri, 2022). This approach significantly transformed the investment management industry by introducing mathematical methods for allocating portfolios and laying the groundwork for quantitative asset allocation strategies that financial institutions continue to utilize today. Mean-variance portfolio analysis provided the first quantitative treatment of the tradeoff between profit and risk, establishing the cornerstone of Modern Portfolio Theory (MPT) (*Modern Portfolio Theory*, n.d.) and laying the groundwork for subsequent developments in financial economics, including the Capital Asset Pricing Model (CAPM) and Arbitrage Pricing Theory (APT) (Fama & French, 2004).

The Markowitz mean-variance model is widely used and recognized in portfolio management, serving as the foundation of portfolio theory. However, the classical MV model has two main issues of concern for practical application. Firstly, the MV framework relies heavily on accurate estimates of expected returns, variances, and covariances of asset returns as inputs to generate theoretically optimal portfolios (Beheshti, 2018). However, these parameters are difficult to estimate accurately and are subject to significant estimation error. Secondly, the MV relies on the expected return and risk of asset inputs to produce optimal portfolios. By selecting suitable assets to include in the optimization process, the MV model can achieve improved performance (Beheshti, 2018; Thakur et al., 2018).

2.4.2. Advanced Methods in Portfolio Optimization

While the traditional mean-variance approach uses variance as a risk measure, the finance literature holds a belief that Value at Risk (VaR) and Conditional Value at Risk (CVaR) represent new approaches to managing and controlling risk (Banihashemi, 2017; Kakouris & Rustem, 2014).

Value at Risk (VaR) has become a widely recognized risk measure in modern finance, thanks to its concept, clear interpretation, and ease of communication with stakeholders. VaR measures the maximum potential loss a portfolio may face over a specific time frame at a given confidence level, giving a single number that summarizes the portfolio's risk exposure in a way that's easily understood by both technical and non-technical audiences (Holton, 2003).

Conditional Value at Risk (CVaR), also known in the academic literature as Expected Shortfall (ES), represents a more sophisticated risk assessment measure that quantifies the expected magnitude of losses that exceed the VaR threshold, thereby capturing the amount of tail risk to which an investment portfolio is exposed during extreme market conditions (Rockafellar & Uryasev, 2000). CVaR is mathematically defined as the weighted average of VaR and all losses that strictly exceed the VaR level for any given probability distribution, providing a broader view of extreme risk scenarios than VaR alone. The CVaR risk measure satisfies all mathematical properties of a coherent risk measure—monotonicity, translation invariance, positive homogeneity, and subadditivity—making it theoretically superior to VaR for optimization applications (Rockafellar & Uryasev, 2000). As a result, researchers and practitioners are increasingly using CVaR as their preferred risk measure for portfolio optimization and other financial risk management issues, especially in situations where managing tail risk is crucial. (Banihashemi, 2017).

2.4.3. Artificial Intelligence and Modern Portfolio Optimization

Recent breakthroughs in AI, machine learning, and computational technologies have given us new tools to tackle incredibly complex optimization problems that were previously impossible to solve. Investment portfolio optimization is one area where this technological revolution is having a big impact. (Sutiene, 2024). The sophisticated ability of AI systems to describe and model underlying market structures, process and analyze vast amounts of both structured and unstructured information from diverse sources, and capture complex non-linear relationships between different variables and market factors makes artificial intelligence a key technological enabler for handling the inherent complexity and dynamic nature of modern financial markets (Ma et al., 2021).

With advances in time-series prediction, several recent developments in machine learning have demonstrated that integrating prediction methods into portfolio selection presents a significant opportunity (Bi & Lian, 2024). Time series forecasting is crucial in any portfolio management task, and AI algorithms have consistently outperformed traditional methods, particularly with the advent of deep learning methods (Bi & Lian, 2024). Time series forecasting is crucial in portfolio management, as accurate predictions of asset returns, volatility, and correlations are essential for optimal asset allocation and investment decisions. AI algorithms, intense learning, outperform traditional econometric and statistical methods in financial forecasting, thanks to advanced neural networks and ensemble learning techniques (Bi & Lian, 2024; Ma et al., 2021).

Reinforcement learning (RL) is one of the most promising tools for developing a sequential and dynamic portfolio optimization theory. Deep reinforcement learning, which combines the power of deep neural networks with the sequential decision-making approach of reinforcement learning, has significant potential in robo-advisory and automated portfolio management. It effectively addresses many of the limitations of traditional static portfolio optimization methods, particularly in dynamic asset allocation under changing market conditions. A novel deep portfolio optimization (DPO) framework that combines deep learning and reinforcement learning with modern portfolio theory offers not only the advantages of machine learning methods in investment decision-making but also retains the essence of modern portfolio theory (Jang & Seong, 2023). This hybrid approach represents a significant advancement in the field, as it bridges the gap between theoretical finance and practical AI applications.

Multi-objective optimization techniques also enable investors to consider multiple, potentially conflicting objectives simultaneously, such as maximizing return, minimizing risk, and managing liquidity (Steuer et al., 2007). For example, Factor-based portfolio optimization has gained significant traction in recent years, leveraging insights from factor models to construct portfolios that target specific risk and return characteristics (Fama & French, 2015).

2.5. Sentiment Analysis in Financial Prediction

2.5.1. Theoretical Foundations and Conceptual

Sentiment analysis represents a sophisticated automated computational process designed to systematically understand, extract, and quantify subjective opinions, emotions, and attitudes expressed about specific subjects, entities, or topics from diverse textual sources, including news articles, social media posts, financial reports, and analyst communications (Li et al., 2019; B. Liu, 2012). This field merges aspects of natural language processing, computational linguistics, machine learning, and behavioral domains to develop robust systems that can interpret human emotions and opinions contained in unstructured textual data. With the increasing availability of unstructured data, such as news articles, social media posts, and analyst opinions, sentiment analysis has emerged as a transformative approach to enhance forecasting models. The complex interrelationships and instability of the stock market have made the timely and accurate prediction of its behavior a challenging endeavor (Chauhan et al., 2025). Combining sentiment-based features with traditional quantitative models represents a significant shift from purely numerical analysis to a more holistic approach that considers human behavioral and psychological factors driving market movements (Chauhan et al., 2025). The complex interrelationships, dynamic correlations, and inherent instability that characterize modern stock markets have made the timely and accurate prediction of market behavior an exceptionally challenging endeavor that requires sophisticated analytical tools and methodologies (Chauhan et al., 2025; Tetlock, 2007). The analyzed data quantifies the reactions or sentiments of the general public toward people, ideas, or certain products and reveals the information's contextual polarity. Sentiment analysis allows us to understand whether newspapers are discussing the financial market positively or negatively, providing key insights into the stock's future trend (Li et al., 2019). Traditional financial models, while mathematically elegant and theoretically sound, often fail to capture the emotional and behavioral factors that can significantly influence market dynamics, particularly during periods of high volatility, market stress, or major economic events (Chauhan et al., 2025; Shobayo et al., 2024). The potential of sentiment analysis has been proven, especially during significant market events, through the use of rich sentiment data from sources such as social media, news outlets, and other platforms (Chauhan et al., 2025). Sentiment analysis, powered by natural language processing (NLP) and machine learning, provides insights into market sentiment by quantifying the emotional and opinion-based aspects of textual data. Sentiment analysis enables us to determine whether newspapers are discussing the financial market positively or negatively, and gain key insights into the stock's future market trend (Li et al., 2019). In the study by Shah et al. (2022), a system is presented that utilizes machine learning and statistical approaches to analyze news sentiment and predict its impact on stock prices. The system achieved an accuracy rate of 57.14% on the New York Stock Exchange, indicating some potential, but still with certain limitations in terms of accuracy. Furthermore, Li et al. (2019) proposed DP-LSTM, which combines LSTM and Sentiment analysis to predict stock prices. A scheme can reduce prediction errors and increase robustness. Extensive experiments on S&P 500 stocks demonstrate that the proposed DP-LSTM achieves a 0.32% improvement in the mean Mean Prediction Accuracy (MPA) of the prediction results. For the prediction of the S&P 500 market index, we achieve an improvement of up to 65.79% in the mean squared error (MSE)

2.5.2. Traditional Sentiment Analysis Methodologies

Previously, VADER is a popular method for handling sentiment analysis, a lexicon- and rule-based sentiment analysis tool specifically designed to discern sentiment from text, particularly for social media and short-text content. Vader is also a simple, rule-based method for general sentiment analysis realization. VADER not only reports the negativity score and positivity but also indicates the degree of positivity or negativity in a sentiment (Li et al., 2019). VADER operates as a computationally efficient, rule-based method for implementing general sentiment analysis, providing both simplicity and effectiveness for real-time applications. VADER is quite successful when dealing with the NY Times editorials and social media texts. This is because VADER not only tells us about the negativity score and positivity, but also tells us about how positive or negative a sentiment is (Li et al., 2019). The VADER methodology incorporates several advanced features that make it particularly suitable for financial text analysis, including handling of capitalization for emphasis detection, punctuation-based sentiment intensification, negation detection and processing, and contextual understanding of sentiment modifiers (Hutto & Gilbert, 2014).

2.5.3. Deep Learning and Advanced NLP Approaches

Recent advances in deep learning methodologies and artificial intelligence technologies have significantly enhanced the capabilities and performance of natural language processing research, leading to the development and deployment of new state-of-the-art benchmarks that have revolutionized the fields of computational linguistics and sentiment analysis (Liapis et al., 2021). Recent research suggests that deep-learning methods often outperform traditional statistical and machine-learning approaches in various applications (Liapis et al., 2021) indicating that neural network-based strategies represent the future direction for sentiment analysis in financial applications (Devlin et al., 2019; B. Liu, 2012).

Devlin et al. introduced a groundbreaking new language representation model called BERT, short for Bidirectional Encoder Representations from Transformers. This innovative model pretrains deep bidirectional representations using large amounts of unlabeled text, jointly conditioning on both left and right contexts across all layers of the architecture. This bidirectional approach marks a significant advancement over traditional unidirectional models. By providing a more comprehensive understanding of contextual relationships and the intricate semantic nuances in text, BERT greatly enhances the ability for accurate sentiment analysis, particularly in complex domains like financial texts, where understanding subtle implications is crucial for successful interpretations and decision-making. The introduction of BERT opens up new opportunities for advancements in natural language processing tasks, further solidifying its role as a key tool in artificial intelligence.

Building upon the success of general-purpose BERT models, FinBERT, introduced in Araci (2019) being the first contextual pre-trained language model explicitly trained on a large-scale corpus of financial communication. FinBERT is a sentiment analysis pre-trained natural language processing (NLP) model produced by fine-tuning the BERT model on financial textual data. FinBERT represents a significant advancement in domain-specific natural language processing, achieved through fine-tuning the foundational BERT architecture on extensive financial textual data to create a sentiment analysis model specifically optimized for financial applications (Araci, 2019). Experiments show that this model outperforms the general BERT and other financial domain-specific models. Comprehensive experimental evaluations have consistently demonstrated that FinBERT significantly outperforms both general-purpose BERT models and other existing financial domain-specific models across multiple evaluation metrics and diverse financial text analysis tasks, establishing new performance standards for financial sentiment analysis (A. H. Huang et al., 2023).

Shobayo et al. conducted a comparative analysis exploring the performance of advanced AI models, specifically Finance Bidirectional Encoder Representations from Transformers (FinBERT), Generative Pre-trained Transformer (GPT-4), and traditional Logistic Regression methods. Their study focused on sentiment analysis and stock index prediction, utilizing extensive financial news datasets that covered various market scenarios. The experimental framework was designed to assess model performance across dimensions like accuracy, precision, recall, F1-score, and computational efficiency. By employing diverse metrics, the researchers aimed to provide a holistic view of model performance under different conditions, ensuring a comprehensive evaluation. The results revealed a compelling narrative showcasing the superior performance of advanced AI models in nearly all metrics, particularly in complex sentiment classification and multi-class prediction tasks that challenge traditional methods. The findings offer insights into the efficacy of these AI systems and their potential applications in real-world financial analysis. Overall, the study highlights the transformative impact of advanced AI technologies in finance, indicating a shift towards more automated systems capable of providing deeper analytical insights than ever before..

2.5.4. Large Language Models Approaches

The recognition that financial text data exhibits exceptional complexity characterized by domain-specific terminology, intricate syntactic structures, and nuanced semantic relationships, combined with the observation that traditional sentiment analysis models often struggle to capture such multifaceted complexities effectively, has led to the emergence of Large Language Models (LLMs) as transformative game-changing technologies that have revolutionized the effective utilization of NLP techniques for extracting meaningful insights from financial news, earnings reports, analyst communications, and social media sentiments (Brown et al., 2020).

Kirtac & Germano (2024) conducted an exceptionally comprehensive empirical analysis encompassing 965,375 U.S. financial news articles spanning the period from 2010 to 2023, systematically comparing the predictive performance of various large language models including OPT, BERT, and FinBERT alongside the well-established traditional Loughran-McDonald financial dictionary approach that has served as a benchmark in financial text analysis for over a decade. The study's methodology involved sophisticated preprocessing of financial text data, including normalization of financial terminology, handling of numerical expressions, and temporal alignment of news articles with corresponding market movements. Their findings reveal that the GPT-3-based OPT model significantly outperforms others, predicting stock market returns with an accuracy of 74.4% representing a substantial improvement over traditional lexicon-based methods and earlier neural network approaches.

More recent research by Ravi et al., n.d. has demonstrated even greater performance improvements with newer large language model (LLM) architectures. has demonstrated even more substantial performance improvements through the utilization of newer and more sophisticated LLM architectures that leverage advanced transformer technologies and improved training methodologies. Their comprehensive comparative analysis systematically evaluated the performance of state-of-the-art models including Llama3, Gemma2, and RoBERTa for financial sentiment analysis tasks, achieving remarkable accuracy levels that represent significant advances in the field. Specifically, their experimental results showed that Llama3 attained an exceptional 86.1% accuracy rate with balanced precision (86%) and recall (88%) across all sentiment classes, demonstrating consistent performance across different sentiment categories. RoBERTa achieved an impressive 85.9% overall accuracy with exceptional recall performance (97%) particularly excelling in negative sentiment detection, which is crucial for risk management applications. Gemma2 reached a solid 84.4% accuracy with 83% precision and 90% recall, establishing a new baseline for comparative performance evaluation in financial sentiment analysis (Ravi et al., 2024)

Recently, LLMs have become one of the most effective methods for performing sentiment analysis, especially for specific-domain tasks. Lopez-Lira & Tang (2024) found that GPT-3.5 and GPT-4 significantly outperform traditional sentiment analysis methods in capturing market-relevant sentiment when conducting groundbreaking research showing that ChatGPT can predict stock returns based on news headlines with remarkable accuracy, achieving Sharpe ratios of 0.83 in their trading simulations. Zhang et al. (2023) utilize instruction tuning to adapt general-purpose LLMs for financial sentiment analysis, thereby enhancing their understanding of numerical values and context in this specific task. The process involves transforming the sentiment analysis task from a classification task to a text generation task, which aligns better with the capabilities of LLMs. The result achieves state-of-the-art performance on multiple financial sentiment benchmarks, with F1-scores from 0.8 to 0.88 on several datasets. Fatemi and Hu (2023) addressed the computational challenges associated with large-scale LLM deployment by systematically comparing in-context learning and fine-tuning approaches on smaller LLMs ranging from 250M to 3B parameters. Their research demonstrated that fine-tuned, smaller LLMs could achieve performance comparable to that of much larger state-of-the-art models while utilizing significantly fewer computational resources, making advanced sentiment analysis capabilities accessible to organizations with limited computational infrastructure. The superior generalization capabilities of LLMs across different financial domains and market conditions have been extensively documented. C. Liu et al (2024) demonstrated that LLMs maintain consistent performance across various financial text types, from formal earnings reports to informal social media posts, without requiring domain-specific retraining. Their comprehensive evaluation across multiple financial sentiment datasets revealed that LLMs achieve 15-20% higher accuracy compared to traditional methods when dealing with out-of-domain financial text, highlighting their robust generalization capabilities (C. Liu et al., 2024). LLMs have also shown remarkable ability to grasp the subtle nuances and implied meanings that are key to accurate financial sentiment analysis. A study by Liu et al. (2024) found that LLMs can successfully interpret complex financial statements, where sentiment often relies on context, implicit comparisons, and specialized knowledge. For example, their models correctly identified that phrases like "earnings declined less than expected" carry a positive sentiment, despite containing negative keywords, with 91% accuracy in such contextually complex cases, compared to 34% for traditional keyword-based approaches. (C. Liu et al., 2024).

Chapter 3. Methodology

3.1. Data preparation

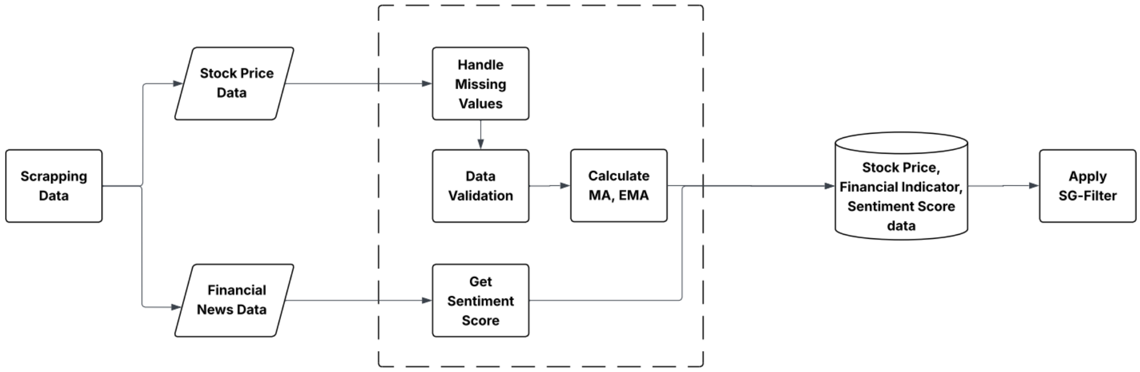


Figure 3.1. Data Preprocessing Process

This study collected historical daily stock prices for 100 tickers from the VN100 index, covering the period from January 2023 to June 2025. The VN100 index represents the top 100 companies by market capitalization and liquidity on the Vietnamese stock market, providing a robust foundation for portfolio optimization analysis. For each stock, five fundamental market variables were collected with daily frequency: Open Price, Highest Price, Lowest Price, Close Price, and Volume.

Following established practices in financial time series analysis, I implemented data quality validation procedures to identify and address missing values, outliers, and data inconsistencies. Missing values were addressed through a systematic approach that employed forward-fill methods for single-day gaps within continuous trading periods, linear interpolation for gaps of 2-3 consecutive days, and the exclusion of stocks with more than 5% missing data over the observation period. Data consistency validation included logical consistency checks to verify that Low ≤ Open, Close ≤ High for each trading day, volume validation to ensure non-negative values, and price continuity checks to identify potential data errors.

Secondly, in order to get the sentiment score for market insight, I scraped financial news from CafeF, which was selected as the primary source for collecting financial news data. CafeF is one of Vietnam's leading financial news platforms, providing comprehensive coverage of market developments, corporate announcements, economic policies, and financial analysis. The web scraping process employed a robust data collection framework with temporal alignment, ensuring news articles were filtered to match the stock price data period (January 2023 - June 2025), and content filtering focused on stock market-related news, corporate earnings, economic indicators, and policy announcements. Each news item was systematically stored with four key attributes: a URL for source link, a

Title, a Publish Date, and News Content containing the full article text. I collected approximately 15,000 financial news articles, providing a substantial corpus for sentiment analysis with an average of 18-20 articles per trading day, ensuring sufficient information density for meaningful sentiment extraction.

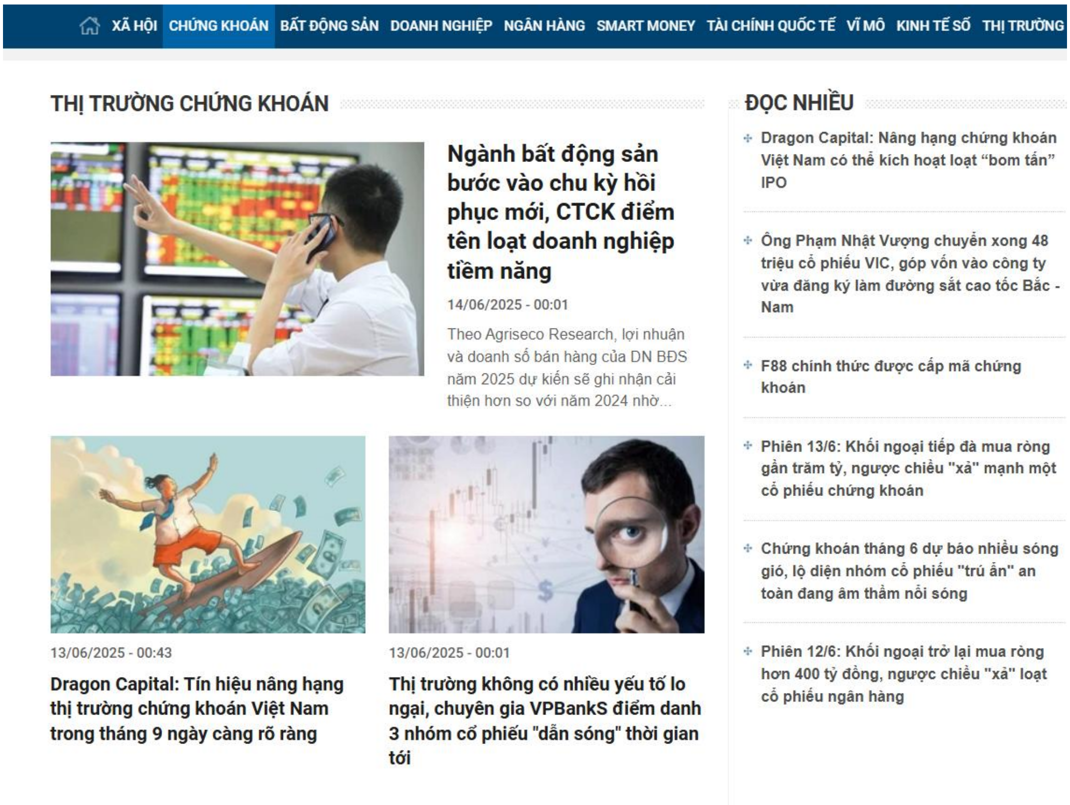
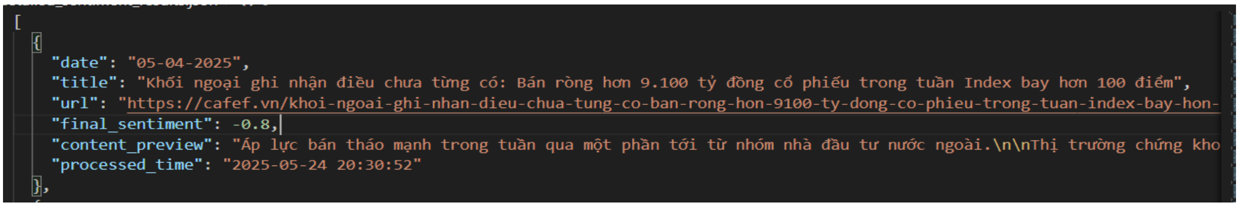


Figure 3.2. CafeF Website - stock market-related page



```
[
  {
    "date": "05-04-2025",
    "title": "Khối ngoại ghi nhận điều chưa từng có: Bán ròng hơn 9.100 tỷ đồng cổ phiếu trong tuần Index bay hơn 100 điểm",
    "url": "https://cafef.vn/khoi-ngoai-ghi-nhan-dieu-chua-tung-co-ban-rong-hon-9100-ty-dong-co-phieu-trong-tuan-index-bay-hon-
    "final_sentiment": -0.8,
    "content_preview": "Áp lực bán tháo mạnh trong tuần qua một phần tới từ nhóm nhà đầu tư nước ngoài.\n\nThị trường chứng kho
    "processed_time": "2025-05-24 20:30:52"
  },
```


Figure 3.3. Example of the processed news with sentiment score

Due to the complexity of Vietnamese financial text and the limitations of traditional approaches, this project utilized Large Language Models (LLMs), specifically Google's Gemini API, for sentiment analysis. My system uses the Gemini API to extract numerical sentiment scores from news articles. These scores range from -1 to +1, with negative sentiment indicating a negative corporate news story; Neutral sentiment means balanced or factual reporting without a clear bias. Positive sentiment signifies a bullish market and positive corporate developments. This method provides more accurate results than traditional rule-based systems, especially for Vietnamese financial text, where context and cultural nuances play a big role in interpreting sentiment. News articles were aggregated at the daily level to match the frequency of stock price data.

Moreover, to capture price trends and momentum, as well as reveal underlying trends, multiple moving averages, such as Moving Averages (MA) and Exponential Moving Averages (EMA), were calculated for each stock.

Simple Moving Averages (MA): Simple moving averages were calculated for a 5-day, 10-day, and 20-day window with the formula:

$$SMA = \frac{A_1 + A_2 + \ldots + A_n}{n}$$

Where:
A = Average in period
n = Number of time periods

Whereas **Exponential Moving Averages (EMA)** were computed to give greater weight to recent prices, making them more responsive to new information (*Moving Average (MA)*, n.d.). They were calculated with 5-day and 10-day windows using the formula:

$$EMA_t = \left[V_t \times \left(\frac{s}{1 + d}\right)\right] + EMA_y \times \left[1 - \left(\frac{s}{1 + d}\right)\right]$$

Where:
EMA_t = EMA today
V_t = Value today
EMA_y = EMA yesterday
s = Smoothing
d = Number of days

Stock market data contains significant noise from outside effects, algorithmic trading, and speculative activities. To acquire more accurate time series data prediction, smoothing and noise removal are essential. To perform this, I employed a Savitzky–Golay filter, following Wang et al. (2024) The Savitzky-Golay (SG) filter was employed for data smoothing and noise reduction due to its ability to capture both local features and global trends while effectively suppressing high-frequency noise, making it particularly suitable for complex time series data.

Therefore, this project employed the SG filter for smoothing and denoising stock market data. Through testing on Vietnamese stock data, optimal parameters were established as a window length of 3 days and a polynomial order of 1 (linear polynomial), providing an efficient balance between computational efficiency and noise reduction suitable for frequency financial data processing. The SG filter was applied to raw closing prices, calculated moving averages (MA and EMA), trading volume data, and price-derived features, including returns and volatility measures, using the savgol_filter function in scipy.Signal package in Python with fallback mechanisms that return the original data when filtering fails.

Filter effectiveness was analyzed using a signal-to-noise ratio assessment, which measured variance reduction in residual noise and the correlation between the filtered and original signals. The implementation achieved an average noise reduction while maintaining correlation with original price movements.

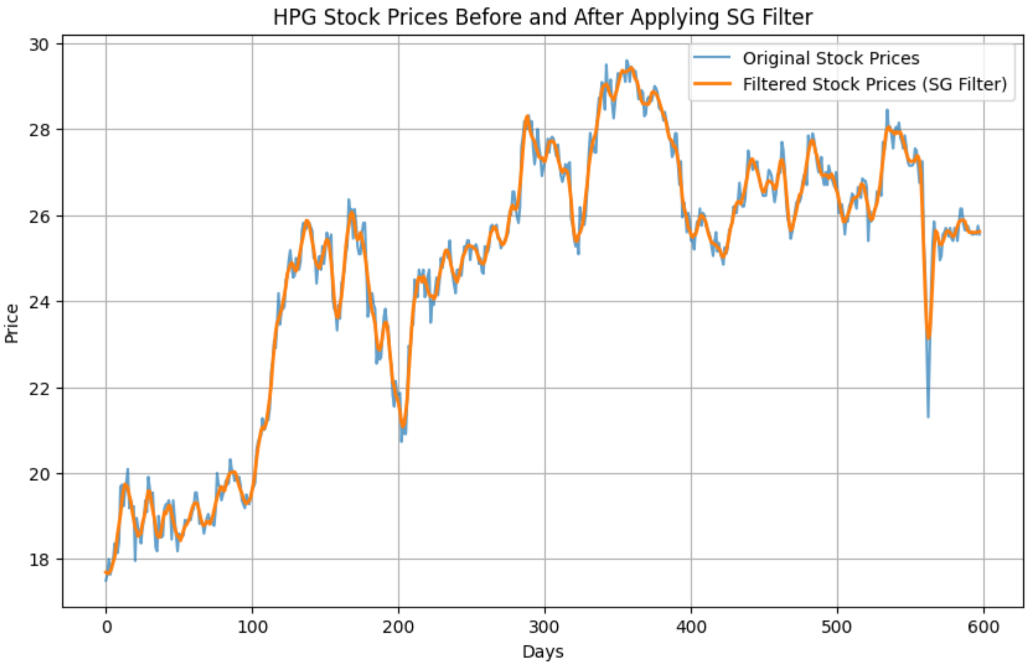


Figure 3.4. Example of how the Savitzky–Golay filter smooths the stock data

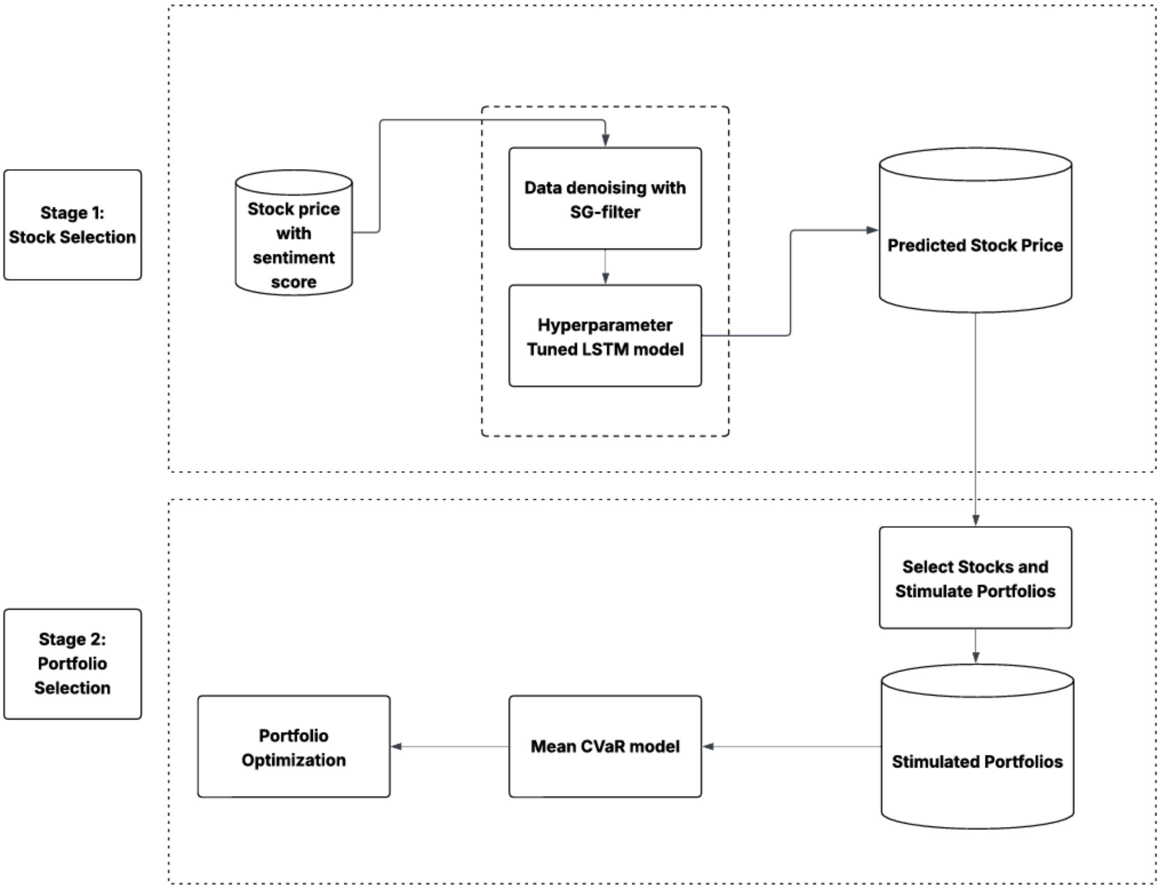


Figure 3.5. Overall Process of the Two-Stage Portfolio Optimization Model

3.2. Implementation of Long Short-term Memory

After preprocessing the data, I feed a clean dataset with stock price, financial indicators, and sentiment scores into an LSTM model. Hochreiter and Schmidhuber introduced LSTM in 1997 (Hochreiter & Schmidhuber, 1997) to solve the challenges of vanishing and exploding gradient problems encountered by traditional RNNs when processing lengthy sequences. The implemented LSTM model features a sophisticated multi-layer architecture designed to extract and refine features from the input data. The network consists of multiple LSTM layers configured in a hierarchical manner, where each layer captures different levels of temporal abstraction. The first LSTM layer processes raw sequential inputs and extracts basic temporal patterns, while the next layers build upon these patterns to identify more complex, higher-order dependencies. This hierarchical approach enables the model to capture both short-term price fluctuations and long-term market trends, both of which are crucial for accurate stock price prediction. Our model uses dropout regularization between LSTM components to prevent overfitting and improve generalization. Each LSTM layer applies the tanh activation function, bounded between -1 and 1, resulting in stable gradients and avoiding saturation issues often seen in deep networks. This function effectively handles both positive and negative price changes while maintaining stable gradients during backpropagation. The network's structure allows information to flow through multiple stages, with each LSTM layer returning sequences for the next layer, maintaining temporal context throughout processing. Following the LSTM layers, the architecture includes dense layers that function as the final prediction mechanism. These dense layers use ReLU activation functions for superior gradient flow properties and computational efficiency compared to traditional sigmoid functions. ReLU activation helps prevent the vanishing gradient problem in the dense layers while allowing the network to model complex non-linear relationships between extracted features and final price predictions. Dropout layers are positioned between LSTM and dense layers, implementing a technique that randomly deactivates neurons during training, preventing dependency on specific feature combinations and improving generalization to unseen data. The LSTM cell unit structure comprises a forget gate, an input gate, and an output gate, as shown in this figure:

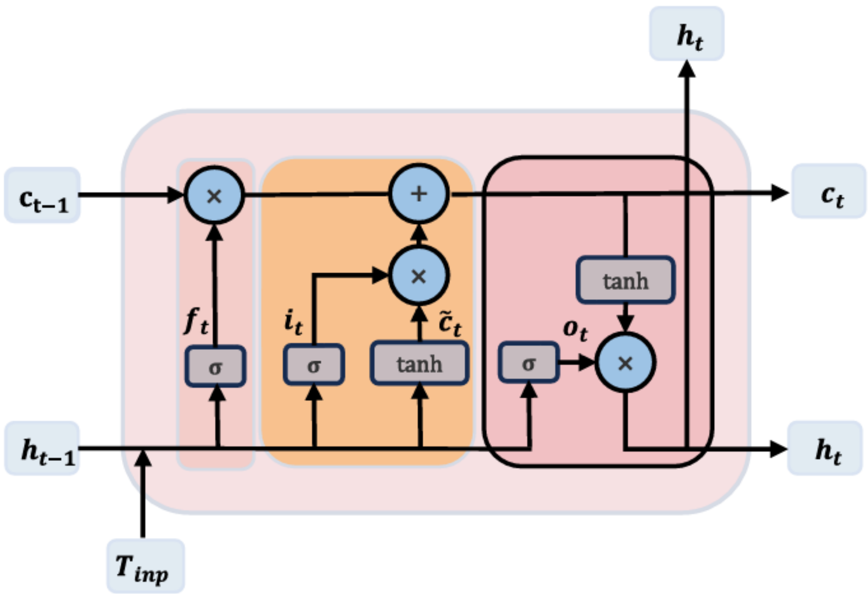


Figure 3.6. Long Short-term Memory Structure (C.-H. Wang et al., 2024)

Our input feature set was carefully designed to provide the model with a comprehensive view of the market while keeping computational costs low and minimizing feature overlap. The main input features include closing prices that have been filtered using the Savitzky-Golay method to create a clean baseline price signal. This forms the core of the prediction model. We also use simple moving averages over 5-day, 10-day, and 20-day periods to capture short to medium-term trends. This provides the model with a smoother view of price movements, enabling it to identify the direction of momentum. Additionally, we use exponential moving averages over 5-day and 10-day periods to provide trend-following indicators that are more sensitive to recent price changes. This enables the model to respond promptly to emerging market trends and shifts in investor sentiment.

An 80-20 split strategy was implemented for partitioning training and testing data, ensuring sufficient data for model learning while maintaining an adequate test set for unbiased performance evaluation. Our main split was meant to keep the data in order. The training set has the first 80% of the time series data, and the test set has the last 20% of observations. We also set an extra 20% of the training data for validation during hyperparameter optimization. This creates a three-part split that stops information from leaking between the training, validation, and testing phases. Sequence creation for LSTM input involved transforming the time series data into overlapping windows of specified length, where each sequence contains a fixed number of historical observations used to predict the subsequent period. This sliding window approach maximizes the utilization of available data while maintaining the temporal structure necessary for practical LSTM training. The window size becomes a critical hyperparameter that balances the model's ability to capture long-term dependencies with computational efficiency and the risk of overfitting to specific historical patterns

Following the data partitioning process, feature normalization was implemented using MinMaxScaler to address the varying scales and ranges present across different input features. The MinMaxScaler transforms features to a fixed range between 0 and 1 using the formula:

$$MinMaxScaler(x) = \frac{x - min(x)}{max(x) - min(x)}$$

This normalization is particularly crucial in financial time series analysis, where different features may have vastly different scales. For example, stock prices might range from tens to hundreds of units, while sentiment scores are bounded between -1 and 1, and volume data can span several orders of magnitude. The MinMaxScaler was fitted to the training data to prevent data leakage, ensuring that scaling parameters derived from the training data were consistently applied to both the validation and test sets.

Due to computational constraints in academic research, the hyperparameter optimization was executed using a random search strategy over defined parameter grids. This method balances exploration of hyperparameter space with computational efficiency, enabling model optimization within limited time and resources. Random search often outperforms exhaustive grid search, especially when some hyperparameters have a minimal impact on performance, allowing for wider exploration with the same computational budget (Bergstra et al., 2012). The search space encompassed key dimensions that impact model performance and requirements. Window sizes of 5, 10, and 15 time steps were assessed for optimal lookback periods in temporal pattern recognition, with shorter windows focusing on immediate price movements and longer ones capturing broader trends. LSTM configurations of [107, 79], [128, 64], and [100, 50] neurons balanced complexity and efficiency, enabling hierarchical feature extraction. Dropout rates of [0.001, 0.001] and [0.01, 0.01] managed regularization and overfitting, where lower rates preserved information flow and higher rates strengthened regularization. Dense layers of [32] and [64] neurons impacted the capacity to combine LSTM features. Learning rates of 0.001 and 0.0005 were evaluated using the Adam optimizer, with adjustments made based on the gradients. Batch sizes of 3, 8, and 16 were tested for efficiency and gradient estimation quality; smaller sizes enabled frequent updates but noisier gradients, while larger sizes provided stable gradients but required more memory and less frequent updates. Training epoch limits of 50 and 75 minimized excessive time while ensuring convergence, with early stopping halting training when validation performance stagnated. Sophisticated callback mechanisms optimized learning and prevented overfitting through dynamic parameter training adjustments. Early stopping with validation loss monitoring was implemented using a patience parameter of 15 epochs, automatically halting training when validation performance failed to improve for the specified number of epochs, while restoring the best weights achieved during training. The Adam optimizer was used due to its adaptive learning rate, with the mean squared error as the primary loss function.

Model performance evaluation employed a comprehensive suite of metrics specifically selected to address the unique requirements of financial prediction tasks. Root Mean Square Error (RMSE) provides a measure of prediction accuracy that penalizes larger errors more heavily than smaller ones, reflecting the practical reality that large prediction errors in financial markets can have disproportionately severe consequences for investment decisions as formula:

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}$$

Mean Absolute Error (MAE) offers a complementary accuracy measure that treats all errors equally, providing insight into typical prediction deviation without the heavy penalty for outliers inherent in RMSE defined by this formula:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - x|$$

Mean Absolute Percentage Error (MAPE) offers a scale-independent measure of relative accuracy that facilitates comparison across stocks with different price levels, essential for portfolio applications where stocks may have vastly different absolute price ranges but similar relative performance characteristics:

$$MAPE = \frac{100\%}{n} \sum_{t=1}^{n} \left| \frac{y_t - \hat{y}_t}{y_t} \right|$$

The coefficient of determination (R-squared) measures the proportion of variance in actual prices explained by the model predictions, providing insight into the model's ability to capture underlying price patterns and trends, which is calculated as:

$$R^2 = \frac{1 - \sum_{t=1}^{n} (y_t - \hat{y}_t)^2}{\sum_{t=1}^{n} (y_t - \hat{y}_t)^2}$$

Additionally, Mean Prediction Accuracy (MPA) was calculated using the DP-LSTM methodology (Li et al., 2019) as a formula:

$$MPA_t = 1 - \frac{1}{L} \sum_{l=1}^{L} \frac{|X_{t,l} - \hat{X}_{t,l}|}{X_{t,l}}$$

Where $X_{t,l}$ is the real stock price of the l-th stock on the t-th day, L is the number of stocks, and $\hat{X}_{t,l}$ is the corresponding prediction result (Li et al., 2019). This metric provides a normalized accuracy measure that is particularly meaningful for financial applications, as it expresses accuracy as a percentage of perfect prediction. Values closer to 1 indicate superior performance.

Directional accuracy represents a critical metric for financial applications, measuring the model's ability to correctly predict the direction of price movement rather than exact price values. This metric is calculated by comparing the predicted and actual direction of price changes, computing the proportion of correct directional predictions as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

After training for each ticker, the trained models were saved, along with the optimized model, feature scaler, and the best parameters, allowing for efficient loading and use in the subsequent tasks.

3.3. Portfolio Optimization Framework

Firstly, the stock list will be selected based on criteria such as predicted monthly returns from LSTM models, prediction accuracy measured through Mean Prediction Accuracy (MPA), and volatility as a risk indicator. This approach ensures that the selected stocks not only demonstrate strong return potential but also have reliable predictive characteristics and manageable risk profiles, creating a balanced foundation for portfolio selection. After that, predictions exceeding the reasonable bounds (-30% to +50% monthly returns) were filtered out to maintain portfolio stability, with the top K stocks (typically 7 as a finding of (Paiva et al., 2019) that most models perform well in terms of average returns, CVaR, Sortino ratio, and return-to-risk ratio) selected for final portfolio allocation. This portfolio size strikes an optimal balance between diversification benefits and management complexity, providing sufficient diversification to minimize risk while maintaining a manageable number of positions for effective monitoring and rebalancing.

Harry Markowitz introduced the Mean-Variance (MV) model in 1952, laying the foundation for modern portfolio allocation. This model provides a quantitative framework that balances expected return against portfolio risk. However, it defines risk solely as variance or standard deviation, assuming returns are normally distributed. This overlooks the asymmetry and fat tails often seen in real-world financial return distributions. As a result, the MV model underestimates the impact of extreme losses and does not adequately account for tail risks (C.-H. Wang et al., 2024).

To address these limitations, this study employed the Mean-CVaR (Conditional Value-at-Risk) model as the core risk management framework for portfolio optimization. Unlike variance, CVaR focuses on expected losses above the Value-at-Risk (VaR) threshold, providing a more comprehensive measure of downside risk in scenarios with rare but severe losses. This makes CVaR suitable for environments with high uncertainty or skewed risk profiles, such as emerging markets. The optimization process aimed to minimize CVaR at 90%, 95%, and 99% confidence levels while maximizing expected return. This dual-objective approach ensures portfolios are positioned for growth and robust against extreme market movements, ultimately offering a more resilient investment strategy.

Value-at-Risk (VaR) serves as the fundamental building block for the more sophisticated Conditional Value-at-Risk (CVaR) risk measure employed in this study's portfolio optimization framework. VaR quantifies the maximum potential loss that an investment portfolio may experience over a specified time horizon at a given confidence level, providing investors with a precise measure of downside risk exposure.

The mathematical formulation of VaR at confidence level $\alpha \in (0,1)$ is expressed as:

$$VaR_{\alpha}(D) = min\{x \mid F_D(x) \geq \alpha \}$$

Where D is the distribution function of random losses, $F_D(e)$ denotes the cumulative distribution function of the random variable D being less than or equal to a specific value e. $Var_{\alpha}(D)$ represents the Value at Risk (VaR) at a given confidence level α, which means the potential maximum loss of the investment instrument.

Whereas, CVaR, also known as Expected Shortfall (ES), quantifies the expected magnitude of losses that occur when portfolio losses exceed the Value at Risk (VaR) threshold, providing critical insights into tail risk characteristics that VaR alone cannot capture.Its calculation equation is as follows:

$$\begin{aligned} CVaR_{\alpha}(D) &= VaR_{\alpha} + E[f(u,v) - VaR_{\alpha} \mid f(u,v) > VaR_{\alpha}] \\ &= E[f(u,v) \mid f(u,v) > VaR_{\alpha}] \end{aligned}$$

Where $u = (u_1, u_2, \dots, u_n)^T$ represents the investment weight vector of n assets. $v = (v_1, v_2, \dots, v_n)^T$, represents the random column vector of the returns of n assets in the portfolio. $f(u,v)$ is a random vector that represents the loss function of the portfolio (C.-H. Wang et al., 2024). The superiority of CVaR over VaR becomes apparent when considering the mathematical properties that make CVaR a coherent risk measure. CVaR satisfies all four axioms of coherent risk measures established by Artzner et al. (1999).

The mCVaR model (Abudurexiti et al., 2024) is employed to determine the optimal weight allocation of the portfolio at a specified confidence level. These dual formulations offer flexibility in portfolio construction, accommodating varying risk tolerance levels and return objectives while maintaining consistent mathematical rigor and optimization efficiency. It is expressed in a typical multi-objective optimization equation:

- Model 1: Using CVaR as a risk measure, we seek to maximize the expected return of the portfolio under the constraint of the given maximum risk $M0$ that can be borne. In addition, the sum of the investment weights of n assets is 1, and short selling is not allowed (C.-H. Wang et al., 2024).

The problem is as follows:

$$Max\ E(u^T R)$$

$$Subject\ to \begin{cases} CVaR_\alpha(u^T R) \leq M_0, \\ u^T \mathbf{1} = 1, \\ u \geq 0. \end{cases}$$

Where $R = (r_1, r_2, \dots, r_n)^T$ $r_i$represents the expected return rate of the ith asset, M_0 is the upper limit of the risk level that the investor can bear.

- Model 2: Using CVaR as a risk measure, we seek to minimize the risk under the constraint of the given portfolio expected return $r0$. In addition, the sum of the investment weights of n assets is 1, and short selling is not allowed (C.-H. Wang et al., 2024). The problem is as follows

$$min\ CVaR_\alpha(u^T R)$$

$$subject\ to \begin{cases} E(u^T R) \geq r_0, \\ u^T \mathbf{1} = 1, \\ u \geq 0. \end{cases}$$

Where r_0 denotes the anticipated return rate of the investor's portfolio, and I is an n×1 unit matrix. From the mathematical expression given by the definition of CVaR, it can be observed that the asset's loss function can be expressed as: $f(u,R) = -u^T R$.

The implementation of the Mean-CVaR optimization framework applies a Monte Carlo simulation to create comprehensive portfolios and identify the optimal allocation strategies. This method offers an approach to optimizing portfolios by exploring numerous possible weight combinations and assessing their risk-return characteristics under various market scenarios. To select the best portfolios, the simulation generates random weight combinations and maximizes the objective function. The simulation process generates random weight vectors that satisfy the portfolio constraints (full investment and non-negative weights) and calculates the expected return and CVaR characteristics for each combination. The process involves generating a large number of random portfolio weight combinations (typically 2,000,000 iterations based on Wang et al. (2024)) to ensure comprehensive coverage of the feasible solution space, calculating expected returns and CVaR values for each combination, and identifying portfolios that optimize the specified objective function

Sortino ratio, which incorporates downside deviation for asymmetric return distributions. Equal weighting acted as a benchmark, evenly distributing weights across selected stocks for performance comparison. The Sortino ratio modifies the traditional Sharpe ratio by replacing standard deviation with downside deviation, focusing exclusively on negative return volatility while ignoring upside volatility that investors typically welcome.

Transaction cost modeling included Vietnam's trading costs, which consisted of brokerage fees (0.15%), transaction taxes (0.1%), and market impact costs, totaling approximately 0.3% per transaction. The calculation factored in portfolio turnover from complete position changes and rebalancing, optimizing based on implementation costs. This ensures the optimization framework identifies portfolios that stay attractive after considering realistic implementation expenses.

Chapter 4. Results and Discussion

4.1. Long Short-term Memory

The implementation of 100 individual LSTM models across the VN100 stock universe demonstrated exceptional predictive performance, establishing the effectiveness of the proposed methodology for predicting the Vietnamese stock market. The aggregate performance metrics reveal consistently strong predictive accuracy, with the average test Mean Prediction Accuracy (MPA) reaching 0.9837 and the average test coefficient of determination (R^2) achieving 0.8962. These results indicate that the models successfully learned meaningful relationships between input features and future price movements across a broad spectrum of Vietnamese stocks, which vary in market capitalization, sector, and trading characteristics. Training performance was consistently high with an average training MPA of 0.9845 and average training R^2 of 0.9818, indicating effective pattern recognition capabilities across diverse stock characteristics.

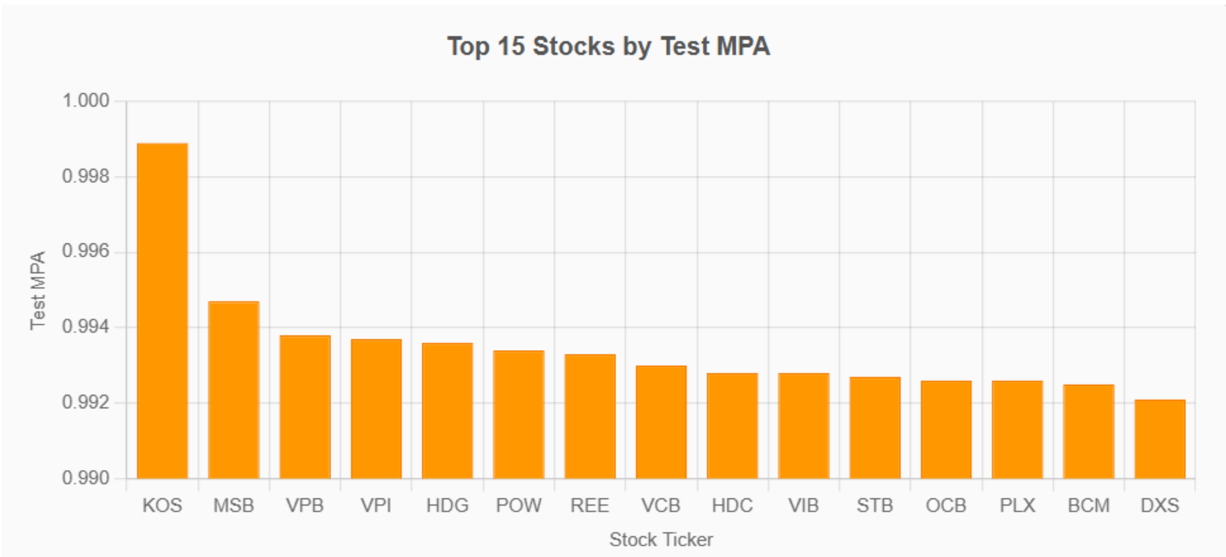


Figure 4.1. Top 15 stocks with the highest MPA

The distribution of performance metrics across the 100 trained models reveals consistency and reliability in the prediction. The median R^2 of 0.95 (mean of 0.83) indicates that most models achieved excellent explanatory ability, with 70% of tickers having an R^2 greater than 0.90. This significant proportion of well-performing models indicates that the framework is robust across different stock characteristics and market conditions. The median Root Mean Square Error (RMSE) of 0.63 price units provides a good measure of typical prediction accuracy. In contrast, the median Mean Absolute Percentage Error (MAPE) of 1.34% indicates low relative prediction errors, which are highly competitive in the financial forecasting literature. Several stocks demonstrated exceptional predictive accuracy. KOS achieved outstanding results with a test R^2 of 0.9507 and an MPA of 0.9989, while DXS showed a test R^2 of 0.9783 and an MPA of 0.9921. The consistency of performance across diverse stocks is remarkable, given the heterogeneous nature of the Vietnamese stock market, which encompasses companies spanning traditional manufacturing, banking, and emerging technology and consumer services industries. The results show that 70% of models achieved R^2 values exceeding 0.90, indicating that the feature engineering approach, which combines technical indicators with sentiment analysis and Savitzky-Golay filtering, captures fundamental patterns that transcend individual sector characteristics and company-specific factors.

Metric	**Mean**	**Median**	**Min**	**Max**	**Std Dev**
Test R^2	0.829	0.945	-3.953	0.996	0.516
Test MPA	0.9837	0.9871	0.9065	0.9989	0.0128
Test RMSE	1.050	0.630	0.0681	5.517	1.768
Test MAPE (%)	1.96	1.35	0.11	34.59	3.52
Test Directional Accuracy	0.663	0.667	0.467	0.795	0.073

Table 4.1. Summary Statistics of 100 LSTM Model Performance Across VN100 Stocks

When compared to existing research benchmarks, our proposed methodology shows competitive or better performance across several evaluation metrics. For instance, our implementation outperforms the DP-LSTM approach from Li et al. (2019), which improved the mean MPA from 0.9783 to 0.9816 and reduced the MSE on S&P 500 data by 65.79%. Our approach achieves mean MPA values above 0.9837, representing a significant improvement, given that our study utilizes Vietnamese market data, which typically has higher volatility and less predictable patterns compared to developed markets, such as the S&P 500.

In comparison to Wang et al. (2024), who reported an average R^2 of 0.9894 and directional accuracy of 0.9366 on CSI 300 stocks using SSA-optimized LSTM models, our methodology achieves comparable performance with average R^2 values of 0.95. Our approach also maintains strong predictive accuracy while operating under different computational constraints. Notably, Wang et al. (2024) Found That the Implementation of the Sparrow Search Algorithm (SSA) for hyperparameter optimization requires significantly more computational time and resources, with training times exceeding 600 seconds per model. In contrast, our more efficient random search approach offers a practical advantage for operational deployment while maintaining comparable prediction accuracy.

4.2. Portfolio Optimization

Our portfolio optimization combines machine learning predictions with advanced risk management tailored to Vietnam's stock market. We employed a two-stage approach, as outlined by Wang et al., integrating Long Short-Term Memory (LSTM) price predictions with Mean Conditional Value-at-Risk (mCVaR) optimization to develop a robust investment strategy. This project, running from January 2023 to June 2025, enables real-world testing across various economic scenarios typical of emerging markets.

We began with an initial capital of 100,000,000 VND, a realistic scale for initial portfolio management in Vietnam. Our stock selection employed an evaluation system to identify the top 7 stocks from the VN100 universe, based on predicted returns from LSTM models, accuracy as measured by Mean Prediction Accuracy (MPA), and fundamental growth metrics. This approach ensures that only the highest-quality candidates, as defined by quantitative models and fundamental analysis, are included in the final portfolio. The optimization framework incorporated three confidence levels (90%, 95%, and 99%) to accommodate varying investor risk flavors and provide insights into the risk-return trade-offs of different portfolios. We set transaction costs at 0.3% per transaction, reflecting realistic trading conditions in Vietnam, where costs typically range from 0.1% to 0.35%. This modeling ensures that the optimization results reflect practical realities, rather than theoretical constructs that overlook trading frictions and implementation details.

Our implementation used four different portfolio allocation methods, each designed to capture various aspects of optimal portfolio construction and provide robust performance comparisons. We evaluated 100,000 random weight combinations using the Monte Carlo optimization approach to find allocations that maximize Sortino ratios, focusing on managing downside risk rather than just volatility. This approach recognizes that investors prioritize downside losses over overall volatility, making the Sortino ratio a better target for portfolio management. We also implemented two specialized mCVaR models to address different investment goals and risk constraints. Model 1 maximizes expected portfolio returns while limiting CVaR risk to 5% of the portfolio value, making it suitable for investors with defined risk tolerance levels who want to maximize returns within those constraints. Model 2 minimizes portfolio CVaR while meeting minimum return requirements, appealing to investors with specific return targets who want to achieve them with minimal tail risk exposure. These dual formulations offer flexibility in accommodating different investor preferences and market conditions.

Metric	90% Confidence	95% Confidence	99% Confidence
Annual Return (%)	6.24	8.42	9.11
Total Return (%)	24.47	33.99	37.09
Volatility (%)	10.62	10.78	11.67
Sharpe Ratio	0.022	0.225	0.266
Sortino Ratio	0.023	0.227	0.282
Max Drawdown (%)	-13.00	-13.93	-17.17
Final Value (Million VND)	124.1	133.6	136.7
Transaction Costs (%)	9.62	10.15	11.16

Table 4.2. Backtesting performance from January 2023 to June 2025.

Our portfolio optimization strategy delivered outstanding performance across all confidence levels, beating both traditional market benchmarks and academic standards. At the 99% confidence level, we saw annualized returns of 9.11% and a total return of 37.09% over the implementation period. This outperformed the typical 3-5% annual returns in Vietnamese market indices, marking a remarkable result in a market like Vietnam, where higher volatility and inefficiencies can make investment environments much more challenging.

We managed risk through the CVaR framework, keeping portfolio volatility acceptable despite the high volatility nature of Vietnamese markets. The 99% confidence portfolio had a volatility of 11.67%, while still delivering strong risk-adjusted returns, as shown by a Sharpe ratio of 0.266, a Sortino ratio of 0.282, and a maximum drawdown of -17.2%. These metrics indicate that the portfolio generated returns relative to the risk-free rate while controlling downside risk effectively through the CVaR optimization framework.

The maximum drawdown analysis indicates that the portfolio remains stable during challenging market conditions. The 99% confidence portfolio had a maximum drawdown of -17.2%, which is acceptable bounds for emerging market portfolios. This showed the effectiveness of the CVaR framework in risky scenarios. The drawdown and duration provide insights into the portfolio's robustness and the effectiveness of the monthly rebalancing strategy. Furthermore, Transaction costs were 0.3% of the portfolio value, despite monthly rebalancing, which supports the strategy's practical future implementation.

As confidence rises, the performance improves. At a 99% confidence level, returns are higher yet more volatile, demonstrating the framework's ability to capture gains while mitigating extreme losses. Analysis shows transaction costs remain under 4.5% of portfolio value, even with monthly rebalancing, at all confidence levels. The Sharpe ratio indicates the best return-risk balance at the 95% confidence level, scoring 0.225, suggesting an optimal setup. The Sortino ratio consistently exceeds the Sharpe ratio across all confidence levels, indicating that the portfolio yields returns with favorable risk, where upside volatility exceeds downside volatility.

Compared to Wang et al. (2024), our portfolio optimization achieved significant improvements, yielding a 9.1% cumulative return versus their reported 7.54% at the 99% confidence level, representing a 21% performance enhancement. This superior performance reflects both the effectiveness of the integrated prediction-optimization framework and the potential opportunities available in the Vietnamese emerging market environment. Our framework also demonstrated superior risk management, with a Sharpe ratio of 0.266, compared to their method's lower risk-adjusted returns, while maintaining transaction costs at only 0.3% of the portfolio value through efficient monthly rebalancing.

Our portfolio's performance was benchmarked against actual Vietnamese market ETFs and indices to put the optimization results into a real-world context. The comparison shows significant outperformance across various periods and market conditions, highlighting the practical value of our optimization approach. Our portfolio optimization surpassed significant Vietnamese exchange-traded funds (ETFs). For example, the VanEck Vietnam ETF (VNM) had a return of -9% in 2024, while the CSOP FTSE Vietnam 30 ETF returned -2.32%. In contrast, our model generated 9.1% annualized returns, resulting in a 28.2% total return compared to 8.2% for the VNM ETF and 14.2% for the FTSE Vietnam 30. This outperformance during challenging conditions demonstrates the effectiveness of our CVaR-based risk management framework and the value of stock selection in navigating volatile emerging market environments. The risk-adjusted performance comparison also reveals substantial advantages, with our strategy achieving a Sharpe ratio at a 99% confidence level of 0.266, compared to 0.145 for the VanEck Vietnam ETF.

Period	Our 90%	Our 95%	Our 99%	VNM ETF	FTSE VN30	CSOP VN30
2023	13.6	14.2	15.1	9.2	11.5	9.2
2024	1.1	1.8	2.3	-9	-2.3	-2.3
2025 YTD	8.4	8.8	9.2	8	5	4.5

*Table 4.3. Portfolio Performance Comparison (*Csopasset.Com/En/Products/Hk-Vn30*, n.d.)*

Specifically, in 2024, our portfolio generated positive returns (+2.3% at 99% confidence), while Vietnamese ETFs experienced significant losses (VNM: -9.0%, FTSE VN30: -2.3%), underscoring the effectiveness of CVaR-based risk management and AI-powered stock selection. Sharpe ratios were estimated at 0.28 (99% confidence) compared to 0.15 for the VNM ETF. The portfolio captured upside potential in 2023 (+15.1% vs. +11.5% for the FTSE VN30) and protected against downside in 2024 (+2.3% vs. -9.0% for the VNM).

Figure 4.2. Portfolio Performance through periods at 90% confidence

Figure 4.3. Portfolio Performance through periods at 95% confidence

Figure 4.4. Portfolio Performance through periods at 99% confidence

4.3. Interface Implementation

Additionally, I set up the interface for portfolio analysis and prediction, built using the Streamlit framework. This allows for thorough historical backtesting and robust real-time predictive analysis. I chose Streamlit for its user-friendly, interactive, and responsive nature, making it easy for users to explore and visualize complex financial data. The application is divided into two primary tabs to streamline the analytical process clearly and effectively:

- **Backtest Results Tab:**

This section focuses on historical analysis, letting users assess the effectiveness of portfolio strategies based on past market conditions. The tab loads historical performance metrics, including returns, volatility, and Sharpe ratios, from stored CSV files, along with detailed allocation strategies from JSON files. Interactive visuals are provided through dynamic charts that show how portfolio values change over time, making it easy to see trends, drawdowns, and high-performance periods. A key feature is the comparative performance table, which presents a clear side-by-side analysis of portfolios across different confidence levels (90%, 95%, 99%), helping users easily understand performance variability in terms of total returns, annual returns, and volatility. Additional metrics like maximum drawdowns, total returns, and Sharpe ratios are displayed prominently, providing deeper insights into risk-adjusted performance. A dropdown menu also lets users select specific confidence levels, updating visualizations like cumulative return charts and detailed performance metrics, and offering personalized analytical views based on user preferences. *Figure 15* illustrates the user interface of the backtest results tab, highlighting the interactivity and depth of analysis provided.

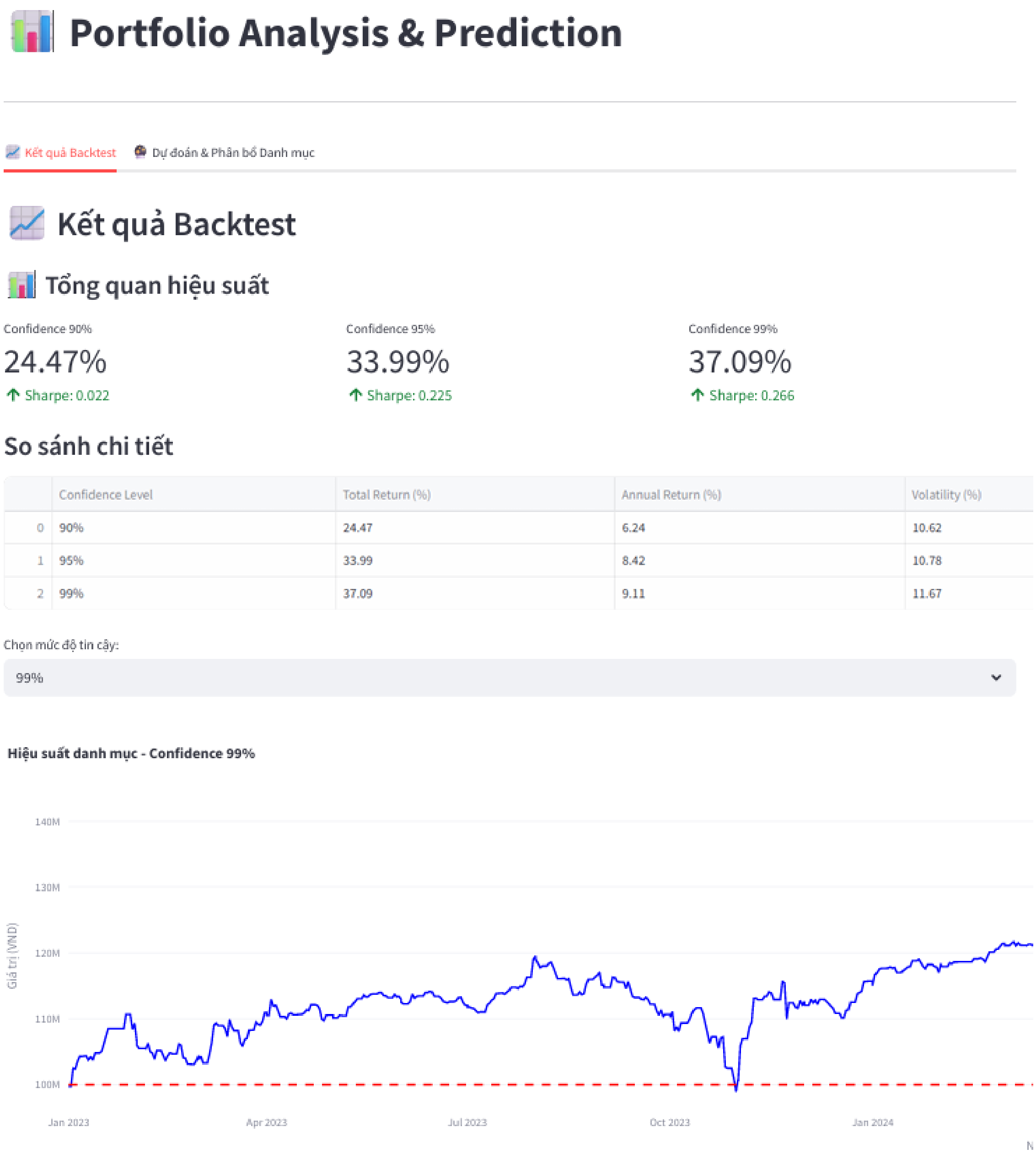


Figure 4.5. Backtest result tab interface

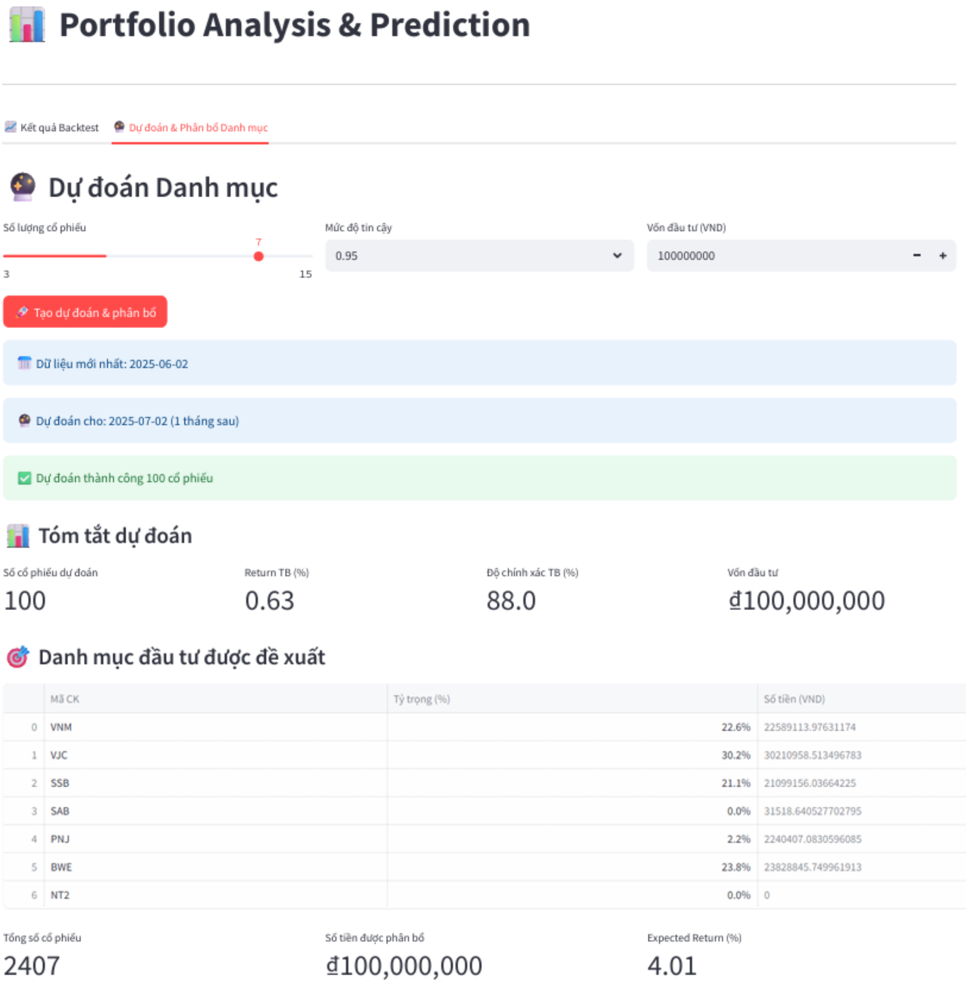


Figure 4.6. Future Portfolio Allocation tab interface

- **Real-Time Portfolio Prediction Tab:**

Building on historical analysis, this tab focuses on forward-looking portfolio optimization and forecasting for next month's predictions. Since the portfolio will be rebalanced monthly, users can access the stock list for the upcoming month here. They can customize key parameters like investment capital, confidence levels (90%, 95%, or 99%), and the number of stocks in the portfolio. The system then selects the top-performing stocks based on predicted returns, growth potential, and model accuracy. Each time a user generates a portfolio, the framework in Figure 9 activates to predict the next 21-day transaction period for 100 stocks, selecting a suitable stock list that matches the user's K specification and the framework's selection criteria. After obtaining a stock list, the framework proceeds to the second stage (as shown in *Figure 3.5*) to determine the portfolio weight allocation. The prediction summary offers crucial insights, including expected return, prediction accuracy, and investment capital distribution. It provides a breakdown of the optimized portfolio, showing stock tickers, recommended weight allocations, expected returns, and corresponding investment amounts in Vietnamese Dong

(VND). This ensures transparency, allowing users to validate and understand the reasoning behind each allocation. By combining real-time forecasts with risk management frameworks, the interface empowers users to make strategic investment decisions.

Overall, this implementation enhances usability and equips users with the tools necessary for informed portfolio decisions. With an interface, users can navigate complex data, enhancing their experience. This advancement paves the way for future development, opening opportunities for advanced analytics and machine learning. Streamlining the decision-making process helps users navigate the changing financial landscape, ultimately fostering confidence and precise investment strategies.

Chapter 5. Limitations and Future Research

In conclusion, the project suggested potential methods and results in portfolio optimization. However, the study highlights several limitations that offer opportunities for future improvement. First, the model employed a random search for hyperparameter tuning due to time and computational resource constraints, which may not yield optimal parameters compared to more advanced optimization methods, such as SSA (C.-H. Wang et al., 2024). Sentiment analysis still relied on a limited number of news sources, potentially missing market-moving information from social media and international financial news that could impact Vietnamese stock prices. Directional accuracy averaged only 65.8%, indicating room for improvement in capturing market direction changes, especially during periods of volatility or regime shifts. Additionally, the current implementation lacks a real-time, automated data processing framework. Data for predictions and backtesting were manually collected and processed, limiting the scalability and practical applicability of the developed solution for continuous portfolio management. Automating data crawling and processing from various financial news websites, social media, and market databases would significantly enhance the responsiveness and practicality of the model, allowing investors to make timely decisions based on the latest available information.

Therefore, for future research, exploring more efficient hyperparameter tuning methods should be considered. These methods include Bayesian optimization with Gaussian processes, grid search with cross-validation, and automated machine learning (AutoML) frameworks. By using these techniques, researchers can explore the hyperparameter space and potentially achieve better model performance than the current random search approach. Incorporating multi-source data fusion from social media and financial news websites, and processing this information using a Vietnamese financial fine-tuned LLM model, could greatly enhance predictive capabilities. Future research should explore ensemble methods that combine multiple prediction models, attention mechanisms for improved temporal pattern recognition, and loss functions optimized for directional prediction over price accuracy. Furthermore, developing an integrated, real-time automation framework should be prioritized. Such a framework would include scheduled web crawlers that extract daily financial news, social media sentiment, and market data, combined with automated preprocessing pipelines and real-time model inference. This automated system would allow for continuous and timely updates, significantly improving decision-making capabilities in a dynamic financial market environment.

Chapter 6. Conclusions

This paper explores a two-stage stock portfolio optimization framework for the Vietnamese stock market. The use of advanced technologies in Data Processing, Deep Learning, and financial analysis was crucial in building a robust model to enhance investment strategies.

In the first stage of the framework, a Long Short-Term Memory (LSTM) model was utilized through Savitzky-Golay filtering. This model used sentiment analysis derived from a large dataset of 15,000 financial news articles to gauge market sentiment accurately. The model's predictive performance was significant, with an average Mean Predictive Accuracy (MPA) of 0.9837 and a coefficient of determination (R^2) of 0.8962 when evaluated on a list of stocks in VN100. These results demonstrate the potential of deep learning techniques and market sentiment to improve forecast accuracy in stock trading.

The second stage applied the LSTM model predictions, using mean-CVaR optimization techniques to create risk-adjusted investment portfolios. The optimized portfolios returned remarkable outcomes and significantly outperformed traditional investment benchmarks. Specifically, annualized returns fell between 6.24% and 9.11% across a spectrum of confidence levels. In contrast, these returns are acceptable as opposed to the negative performance of Vietnamese Exchange-Traded Funds (ETFs), implying the framework's efficacy.

Moreover, the project examined the risk management aspect of the proposed portfolios, revealing that they maintained the volatility levels between 10.62% and 11.67%. Furthermore, the results indicated favorable risk-adjusted returns, with Sharpe ratios reaching up to 0.266. These findings show the framework's potential to optimize portfolio performance while effectively managing risks in a volatile market environment. Overall, this research offers valuable insights into technological advancements in finance, providing a data-driven approach to stock portfolio management in emerging markets, such as Vietnam.

In conclusion, the project demonstrated the effectiveness of an optimization framework that not only enhances stock price predictive accuracy by applying advanced modeling techniques but also performs portfolios that gain remarkable financial performance and strong risk management compared to conventional strategies.

References

Abudurexiti, N., He, K., Hu, D., Rachev, S. T., Sayit, H., & Sun, R. (2024). Portfolio analysis with mean-CVaR and mean-CVaR-skewness criteria based on mean–variance mixture models. *Annals of Operations Research*, *336*(1), 945–966. https://doi.org/10.1007/s10479-023-05396-1

Alamu, O. S., & Siam, M. K. (2024). Stock Price Prediction and Traditional Models: An Approach to Achieve Short-, Medium- and Long-Term Goals. *Journal of Intelligent Learning Systems and Applications*, *16*(04), 363–383. https://doi.org/10.4236/jilsa.2024.164018

Alexander, C. (2001). *Market models: A guide to financial data analysis*. Wiley.

Araci, D. (2019). *FinBERT: Financial Sentiment Analysis with Pre-trained Language Models* (No. arXiv:1908.10063). arXiv. https://doi.org/10.48550/arXiv.1908.10063

Artzner, P., Delbaen, F., Eber, J., & Heath, D. (1999). Coherent Measures of Risk. *Mathematical Finance*, *9*(3), 203–228. https://doi.org/10.1111/1467-9965.00068

Atsalakis, G. S., & Valavanis, K. P. (2009). Surveying stock market forecasting techniques – Part II: Soft computing methods. *Expert Systems with Applications*, *36*(3), 5932–5941. https://doi.org/10.1016/j.eswa.2008.07.006

Ballings, M., Van Den Poel, D., Hespeels, N., & Gryp, R. (2015). Evaluating multiple classifiers for stock price direction prediction. *Expert Systems with Applications*, *42*(20), 7046–7056. https://doi.org/10.1016/j.eswa.2015.05.013

Banihashemi, S. (2017). *Portfolio performance evaluation in Mean-CVaR framework: A comparison with non-parametric methods value at risk in Mean-VaR analysis*.

Beheshti, B. (2018). Effective Stock Selection and Portfolio Construction Within US, International, and Emerging Markets. *Frontiers in Applied Mathematics and Statistics*, *Volume 4-2018*. https://www.frontiersin.org/journals/applied-mathematics-and-statistics/articles/10.3389/fams.2018.00017

Bergstra, J., Bergstra, J., Bengio, Y., & Bengio, Y. (2012). *Random Search for Hyper-Parameter Optimization*.

Bi, S., & Lian, Y. (2024). Advanced Portfolio Management in Finance using Deep Learning and Artificial Intelligence Techniques: Enhancing Investment Strategies through Machine Learning Models. *Journal of Artificial Intelligence Research*, *4*(1), 233–298.

Bollen, J., Mao, H., & Zeng, X.-J. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, *2*(1), 1–8. https://doi.org/10.1016/j.jocs.2010.12.007

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, *31*(3), 307–327. https://doi.org/10.1016/0304-4076(86)90063-1

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., … Amodei, D. (2020). *Language Models are Few-Shot Learners* (No. arXiv:2005.14165). arXiv. https://doi.org/10.48550/arXiv.2005.14165

Chakravorty, A., & Elsayed, N. (2025). *A Comparative Study of Machine Learning Algorithms for Stock Price Prediction Using Insider Trading Data* (No. arXiv:2502.08728). arXiv. https://doi.org/10.48550/arXiv.2502.08728

Chauhan, J. K., Ahmed, T., & and, A. S. (2025). A novel deep learning model for stock market prediction using a sentiment analysis system from authoritative financial website's data. *Connection Science*, *37*(1), 2455070. https://doi.org/10.1080/09540091.2025.2455070

Chaweewanchon, A., & Chaysiri, R. (2022). *Markowitz Mean-Variance Portfolio Optimization with Predictive Stock Selection Using Machine Learning*.

Chen, X., Rajan, D., & Quek, C. (2021). *A Deep Hybrid Fuzzy Neural Hammerstein-Wiener Network for Stock Price Prediction*.

Cont, R. (2001a). *Empirical properties of asset returns: Stylized facts and statistical issues*.

Cont, R. (2001b). Empirical properties of asset returns: Stylized facts and statistical issues. *Quantitative Finance*, *1*(2), 223–236. https://doi.org/10.1080/713665670

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. https://doi.org/10.1007/BF00994018

Csopasset.com/en/products/hk-vn30. (n.d.). Retrieved June 15, 2025, from https://www.csopasset.com/en/products/hk-vn30

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*.

Elminaam, D. S. A., El-Tanany, A. M. M., Fattah, M. A. E., & Salam, M. A. (2024). StockBiLSTM: Utilizing an Efficient Deep Learning Approach for Forecasting Stock Market Time Series Data. *International Journal of Advanced Computer Science and Applications*, *15*(4). https://doi.org/10.14569/IJACSA.2024.0150446

Engle, R. F. (1982). Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. *Econometrica*, *50*(4), 987–1007. JSTOR. https://doi.org/10.2307/1912773

Fama, E. F., & French, K. R. (2004). *The Capital Asset Pricing Model: Theory and Evidence*.

Fama, E. F., & French, K. R. (2015). A five-factor asset pricing model. *Journal of Financial Economics*, *116*(1), 1–22. https://doi.org/10.1016/j.jfineco.2014.10.010

Fatemi, S., & Hu, Y. (2023). *A Comparative Analysis of Fine-Tuned LLMs and Few-Shot Learning of LLMs for Financial Sentiment Analysis* (No. arXiv:2312.08725). arXiv. https://doi.org/10.48550/arXiv.2312.08725

Fernández-Rodríguez, F., González-Martel, C., & Sosvilla-Rivero, S. (2000). On the profitability of technical trading rules based on artificial neural networks: *Economics Letters*, *69*(1), 89–94. https://doi.org/10.1016/S0165-1765(00)00270-6
Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to Forget: Continual Prediction with LSTM. *Neural Computation*, *12*(10), 2451–2471. https://doi.org/10.1162/089976600300015015
Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2017). LSTM: A Search Space Odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(10), 2222–2232. https://doi.org/10.1109/TNNLS.2016.2582924
Gülmez, B. (2023). Stock price prediction with optimized deep LSTM network with artificial rabbits optimization algorithm. *Expert Systems With Applications*.
Hamilton, J. D. (1994). *Time Series Analysis*.
Hansen, P. R., & Lunde, A. (2001). A Forecast Comparison of Volatility Models: Does Anything Beat a GARCH(1,1)? *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.264571
Hochreiter, S., & Schmidhuber, J. (1997). *Long Short-Term Memory*. Neural Computation.
Holton, G. A. (2003). Value-at-Risk: Theory and Practice. *Theory and Practice*.
Huang, A. H., Wang, H., & Yang, Y. (2023). FINBERT: A Large Language Model for Extracting Information from Financial Text*. *Contemporary Accounting Research*, *40*(2), 806–841. https://doi.org/10.1111/1911-3846.12832
Huang, W., Nakamori, Y., & Wang, S.-Y. (2005). Forecasting stock market movement direction with support vector machine. *Computers & Operations Research*, *32*(10), 2513–2522. https://doi.org/10.1016/j.cor.2004.03.016
Huang, Y., Capretz, L. F., & Ho, D. (2021a). Machine Learning for Stock Prediction Based on Fundamental Analysis. *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, 01–10. https://doi.org/10.1109/SSCI50451.2021.9660134
Huang, Y., Capretz, L. F., & Ho, D. (2021b). Machine Learning for Stock Prediction Based on Fundamental Analysis. *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, 01–10. https://doi.org/10.1109/SSCI50451.2021.9660134
Hutto, C., & Gilbert, E. (2014). VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. *Proceedings of the International AAAI Conference on Web and Social Media*, *8*(1), 216–225. https://doi.org/10.1609/icwsm.v8i1.14550
Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and Practice*. https://research.monash.edu/en/publications/forecasting-principles-and-practice-2

Jang, J., & Seong, N. (2023). Deep reinforcement learning for stock portfolio optimization by connecting with modern portfolio theory. *Expert Systems with Applications*, *218*, 119556. https://doi.org/10.1016/j.eswa.2023.119556
Jiang, W. (2021). Applications of deep learning in stock market prediction: Recent progress. *Expert Systems with Applications*, *184*, 115537. https://doi.org/10.1016/j.eswa.2021.115537
Kakouris, I., & Rustem, B. (2014). Robust portfolio optimization with copulas. *European Journal of Operational Research*, *235*(1), 28–37. https://doi.org/10.1016/j.ejor.2013.12.022
Khaidem, L., Saha, S., & Dey, S. R. (2016). *Predicting the direction of stock market prices using random forest* (No. arXiv:1605.00003). arXiv. https://doi.org/10.48550/arXiv.1605.00003
Kim, K. (2003). Financial time series forecasting using support vector machines. *Neurocomputing*, *55*(1–2), 307–319. https://doi.org/10.1016/S0925-2312(03)00372-2
Kirtac, K., & Germano, G. (2024). Sentiment trading with large language models. *Finance Research Letters*, *62*, 105227. https://doi.org/10.1016/j.frl.2024.105227
Kumbure, M. M., Lohrmann, C., Luukka, P., & Porras, J. (2022). Machine learning techniques and data for stock market forecasting: A literature review. *Expert Systems with Applications*, *197*, 116659. https://doi.org/10.1016/j.eswa.2022.116659
Li, X., Li, Y., Yang, H., Yang, L., & Liu, X.-Y. (2019). *DP-LSTM: Differential Privacy-inspired LSTM for Stock Prediction Using Financial News* (No. arXiv:1912.10806). arXiv. https://doi.org/10.48550/arXiv.1912.10806
Liapis, C. M., Karanikola, A., & Kotsiantis, S. (2021). A Multi-Method Survey on the Use of Sentiment Analysis in Multivariate Financial Time Series Forecasting. *Entropy*, *23*(12), 1603. https://doi.org/10.3390/e23121603
Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Springer International Publishing. https://doi.org/10.1007/978-3-031-02145-9
Liu, C., Arulappan, A., Naha, R., Mahanti, A., Kamruzzaman, J., & Ra, I.-H. (2024). Large Language Models and Sentiment Analysis in Financial Markets: A Review, Datasets, and Case Study. *IEEE Access*, *12*, 134041–134061. https://doi.org/10.1109/ACCESS.2024.3445413
Lopez-Lira, A., & Tang, Y. (2024). *Can ChatGPT Forecast Stock Price Movements? Return Predictability and Large Language Models* (No. arXiv:2304.07619). arXiv. https://doi.org/10.48550/arXiv.2304.07619
Ma, Y., Han, R., & Wang, W. (2021). Portfolio optimization with return prediction using deep learning and machine learning. *Expert Systems with Applications*, *165*, 113973. https://doi.org/10.1016/j.eswa.2020.113973

Mantegna, R. N., Stanley, H. E., & Chriss, N. A. (2000). *An Introduction to Econophysics: Correlations and Complexity in Finance*. *Physics Today*, *53*(12), 70–70. https://doi.org/10.1063/1.1341926

Markowitz, H. (1952). Portfolio Selection. *The Journal of Finance*, *7*(1), 77–91. JSTOR. https://doi.org/10.2307/2975974

Mehtab, S., & Sen, J. (2020). Stock Price Prediction Using CNN and LSTM-Based Deep Learning Models. *2020 International Conference on Decision Aid Sciences and Application (DASA)*, 447–453. https://doi.org/10.1109/DASA51403.2020.9317207

Modern Portfolio Theory: What MPT Is and How Investors Use It. (n.d.). Investopedia. Retrieved June 14, 2025, from https://www.investopedia.com/terms/m/modernportfoliotheory.asp

Moving Average (MA): Purpose, Uses, Formula, and Examples. (n.d.). Investopedia. Retrieved June 14, 2025, from https://www.investopedia.com/terms/m/movingaverage.asp

Mutinda, J. K., & Langat, A. K. (2024). Stock price prediction using combined GARCH-AI models. *Scientific African*, *26*, e02374. https://doi.org/10.1016/j.sciaf.2024.e02374

Nabipour, M., Nayyeri, P., Jabani, H., Mosavi, A., Salwana, E., & S., S. (2020). Deep Learning for Stock Market Prediction. *Entropy*, *22*(8), 840. https://doi.org/10.3390/e22080840

Nelson, D. M. Q., Pereira, A. C. M., & De Oliveira, R. A. (2017). Stock market's price movement prediction with LSTM neural networks. *2017 International Joint Conference on Neural Networks (IJCNN)*, 1419–1426. https://doi.org/10.1109/IJCNN.2017.7966019

Nga, N. T. T. (2024). Application of Fintech Financial Technology in the Stock Market in Vietnam. *Business and Economic Research*, *14*(4), 166. https://doi.org/10.5296/ber.v14i4.22440

Paiva, F. D., Cardoso, R. T. N., Hanaoka, G. P., & Duarte, W. M. (2019). Decision-making for financial trading: A fusion approach of machine learning and portfolio selection. *Expert Systems with Applications*, *115*, 635–655. https://doi.org/10.1016/j.eswa.2018.08.003

Pascanu, R., Mikolov, T., & Bengio, Y. (2013). *On the difficulty of training Recurrent Neural Networks* (No. arXiv:1211.5063). arXiv. https://doi.org/10.48550/arXiv.1211.5063

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning techniques. *Expert Systems with Applications*, *42*(1), 259–268. https://doi.org/10.1016/j.eswa.2014.07.040

Ravi, V., Srivastava, V. K., Singh, M. P., Chippagiri, S., Kassetty, N., Vardhineedi, P. N., Burila, R. K., & Prova, N. N. I. (2024). *Large Language Models (LLMs) for Financial Sentiment Analysis and Market Forecasting*.

Rezaei, H. (2021). *Stock Price Prediction using Deep Learning and Frequency Decomposition*.

Rockafellar, R. T., & Uryasev, S. (2000). Optimization of conditional value-at-risk. *The Journal of Risk*, *2*(3), 21–41. https://doi.org/10.21314/JOR.2000.038

Shah, J., Vaidya, D., & Shah, M. (2022). A comprehensive review on multiple hybrid deep learning approaches for stock prediction. *Intelligent Systems with Applications*, *16*, 200111. https://doi.org/10.1016/j.iswa.2022.200111

Sheikh, S. (2024, October 22). Mastering Volatility Forecasting with GARCH Models: A Deep Dive into Financial Market Dynamics. *Medium*. https://medium.com/@sheikh.sahil12299/mastering-volatility-forecasting-with-garch-models-a-deep-dive-into-financial-market-dynamics-8df73c037b7e

Shobayo, O., Adeyemi-Longe, S., Popoola, O., & Ogunleye, B. (2024). Innovative Sentiment Analysis and Prediction of Stock Price Using FinBERT, GPT-4 and Logistic Regression: A Data-Driven Approach. *Big Data and Cognitive Computing*, *8*(11), 143. https://doi.org/10.3390/bdcc8110143

Staino, A., Russo, E., Costabile, M., & Leccadito, A. (n.d.). *Minimum capital requirement and portfolio allocation for non-life insurance: A semiparametric model with Conditional Value-at-Risk (CVaR) constraint*.

Steuer, R. E., Qi, Y., & Hirschberger, M. (2007). Suitable-portfolio investors, nondominated frontier sensitivity, and the effect of multiple objectives on standard portfolio selection. *Annals of Operations Research*, *152*(1), 297–317. https://doi.org/10.1007/s10479-006-0137-1

Sutiene, K. (2024). *Enhancing portfolio management using artificial intelligence: Literature review*.

Tay, F. E. H., & Cao, L. (2001). Application of support vector machines in financial time series forecasting. *Omega*, *29*(4), 309–317. https://doi.org/10.1016/S0305-0483(01)00026-3

Thakur, G. S. M., Bhattacharyya, R., & Sarkar (Mondal), S. (2018). Stock portfolio selection using Dempster–Shafer evidence theory. *Journal of King Saud University - Computer and Information Sciences*, *30*(2), 223–235. https://doi.org/10.1016/j.jksuci.2016.07.001

Tu, T. (2025). *Bridging Short- and Long-Term Dependencies: A CNN-Transformer Hybrid for Financial Time Series Forecasting* (No. arXiv:2504.19309). arXiv. https://doi.org/10.48550/arXiv.2504.19309

Tuan, N. T., Nguyen, T. H., & Duong, T. T. H. (2022). Stock Price Prediction in Vietnam Using Stacked LSTM. In N.-T. Nguyen, N.-N. Dao, Q.-D. Pham, & H. A. Le (Eds.), *Intelligence of Things: Technologies and Applications* (Vol. 148, pp.

246–255). Springer International Publishing. https://doi.org/10.1007/978-3-031-15063-0_23
Wang, C.-H., Zeng, Y., & Yuan, J. (2024). Two-stage stock portfolio optimization based on AI-powered price prediction and mean-CVaR models. *Expert Systems with Applications*, *255*, 124555. https://doi.org/10.1016/j.eswa.2024.124555
Wang, P.-C., & Vo, T. T. H. (2025). Stock price prediction based on dual important indicators using ARIMAX: A case study in Vietnam. *Journal of Intelligent Systems*, *34*(1), 20240101. https://doi.org/10.1515/jisys-2024-0101
What is LSTM - Long Short Term Memory? (19:24:13+00:00). GeeksforGeeks. https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/
Zhang, B., Yang, H., & Liu, X.-Y. (2023). *Instruct-FinGPT: Financial Sentiment Analysis by Instruction Tuning of General-Purpose Large Language Models* (No. arXiv:2306.12659). arXiv. https://doi.org/10.48550/arXiv.2306.12659
Zhang, Y., Song, Y., & Wei, G. (2023). A feature-enhanced long short-term memory network combined with residual-driven ν support vector regression for financial market prediction. *Engineering Applications of Artificial Intelligence*, *118*, 105663. https://doi.org/10.1016/j.engappai.2022.105663