ADTA 5900 - Advanced Data Analytics Capstone Experience

Deep Learning for Enhanced Trading Signal Generation: A Hybrid CNN-BiLSTM Model with Attention Mechanism for Stock Market Prediction and Trading Signal Generation

Biniam Abebe

University of North Texas

Denton, Texas

Author Note

Biniam Abebe, University of North Texas, Toulouse Graduate School

Submitted 2025

Abstract

This study proposes a model for generating predictions and a trading signal generation model for the stock market, utilizing a hybrid deep learning framework that integrates Convolutional Neural Networks (CNN), Bidirectional Long Short-Term Memory (BiLSTM) networks, and an attention mechanism. The research demonstrates the effectiveness of this approach when applied to a substantial number of S&P 500 stocks, resulting in significant improvements in risk-adjusted returns and trading efficiency. Our backtesting results indicate that the strategy performs exceptionally well with stable, large-cap stocks, such as Walmart (WMT), which achieved an impressive 48.18% return and a commendable 72.73% win rate. The strategy's risk management framework effectively limits drawdowns, with the top performers experiencing maximum drawdowns of less than 5%. This suggests that selective trading strategies, involving 10 to 15 trades, outperform high-frequency trading methods. Furthermore, the combination of technical indicators with deep learning predictions produces robust trading signals. This approach ensures that the model is not biased toward the majority class (non-profitable trades) and provides offers more accurate representation of the market behavior. These findings are crucial for the practical application implementation of trading strategies based on deep learning in real-life real-world scenarios.

Keywords: *Convolution neural network · Long–short-term memory neural network · Stock price prediction · Leading indicators - CNN-BiLSTM - Hybrid Architecture - Attention Mechanism - Neural Networks - Sharpe Ratio - Win Rate - Maximum Drawdown - Risk-Adjusted Returns - Profit Factor*

Contents

1. INTRODUCTION

In today's volatile financial markets, identifying trends and executing profitable trades quickly is absolutely crucial. Let us be honest - traditional technical analysis relies heavily on subjective human interpretation, which is, well, prone to various psychological biases (Murphy, 2022). However, here's where it gets interesting: blending these traditional methods with cutting-edge deep learning approaches, especially hybrid models like CNN-LSTM, presents a unique opportunity to significantly enhance the accuracy and timeliness of our trading signals (Sezer et al., 2020).

The U.S. equity market is massive, with a market capitalization of $7 trillion, and 60-70% of daily volume coming from algorithmic trading (S&P Global, 2024). To get an edge in this super competitive space, you need some serious pattern recognition capabilities.

Algorithmic trading is not without challenges, though. You've got high-speed market movements, constantly evolving market conditions, and unpredictable volatility patterns to deal with. This means your models need to adapt quickly and anticipate potential price moves. Traditional technical analysis struggles to keep pace - human bias in pattern recognition, limited ability to process multiple indicators simultaneously, and difficulty adapting to changing market conditions are significant hurdles.

That is where my hybrid CNN-LSTM model comes in. It works on S&P 500 stocks by leveraging CNN's ability to learn spatial correlations across multiple financial time series, combined with LSTM's ability to predict temporal sequences (Livieris et al., 2021). This powerful combo addresses many of the limitations of traditional methods.

Of course, applying deep learning to financial markets comes with its own set of challenges. Financial data is notoriously noisy, which affects data quality and quantity. The models are complex and can easily overfit, making them less applicable to real-world conditions. We've tackled many of these issues, including the black box problem, real-time execution challenges, and integration with risk management systems.

In this research, I'm proposing a hybrid CNN-LSTM architecture that provides an advanced trading system addressing these challenges. My core research question examines how much better this hybrid approach is compared to conventional technical analysis in terms of accuracy and profitability of trading signals. I've tested several key hypotheses about the hybrid model, expecting it to produce significantly better signal generation accuracy, superior risk-adjusted returns, and more effective risk management through improved drawdown control and smarter position sizing.

My methodology uses a rich dataset of 501 S&P 500 companies spanning five years, with 76 technical and fundamental indicators and only 1.9% missing values. The model architecture combines CNN and LSTM components to learn both spatial patterns and temporal sequences, enhanced by an attention mechanism that helps the model identify important features and adapt over time. The trading strategy includes market regime detection, dynamic position sizing, and robust risk management.

I am utilizing the latest concepts in data analytics, including deep learning, recurrent neural networks, and advanced data processing, to develop a comprehensive trading system that addresses today's business challenges. My goal is to enhance both the performance and robustness of algorithmic trading by combining classical technical analysis with advanced deep learning methods and effective risk management, improving profits and adaptability to market conditions.

1.1 Significance of the Study

This research makes a significant contribution to the field of algorithmic trading by bridging the gap between traditional technical analysis and modern deep learning approaches. The practical implications are substantial, offering potential benefits to:

- Individual traders seeking automated trading solutions
- Institutional investors requiring scalable trading strategies
- Financial technology firms developing trading platforms
- Academic researchers studying market efficiency
- Risk managers seeking improved methodologies

The implementation of this hybrid approach holds the potential to significantly reduce transaction costs, enhance market efficiency, and improve risk-adjusted returns across a wide range of market conditions. This promising outlook underscores the value of the hybrid CNN-LSTM model in financial market predictions.

1.2 Theoretical Framework

This research builds upon the Efficient Market Hypothesis (EMH) while incorporating behavioral finance principles that suggest market inefficiencies can be exploited. The hybrid model's architecture, which is unique in its combination of deep learning theory, particularly in the areas of pattern recognition and sequence prediction, and trading strategy implementation from modern portfolio theory and risk management frameworks, is a key feature of this study.

1.3 Challenges and Limitations

The implementation of deep learning approaches in financial markets presents several significant challenges:

1. Data Quality and Quantity:
   - Noisy financial datasets requiring preprocessing
   - Real-time data processing requirements
   - Market microstructure effects
2. Model Complexity:
   - Overfitting risks
   - Computational resource requirements
   - Real-time execution challenges
3. Scope Limitations:
   - The study focuses on S&P 500 stocks (2019-2024) and the large-cap U.S. equity market only, due to computational constraints and data availability restrictions.
   - Large-cap U.S. equity market only
   - Computational constraints
   - Data availability restrictions

1.4 Definition of Terms

Key technical and financial terms used throughout this research include:

1. Technical Terms:
   - CNN (Convolutional Neural Network): Deep learning architecture for spatial pattern recognition
   - LSTM (Long Short-Term Memory): Neural network designed for sequential data processing
   - Hybrid Architecture: Combined CNN-LSTM model structure
2. Financial Terms:
   - Market Regime: Distinct market conditions characterized by specific patterns

- Technical Indicators: Mathematical calculations based on price and volume data
- Risk-Adjusted Returns: Performance metrics accounting for risk levels

3. Performance Metrics:
   - Sharpe Ratio: A Measure of Risk-Adjusted Returns
   - Maximum Drawdown: Largest peak-to-trough decline
   - Win Rate: Percentage of profitable trades

1.5 Research Questions and Hypotheses

The key research question is how we compare a hybrid CNN-LSTM deep learning model to a traditional technical analysis approach and whether it provides better trading signal reliability or profitability. The study tests several important hypotheses regarding the quality of the generated signals, trading performance, and risk management effectiveness.

~~So, what are we really trying to figure out here? At its heart, this study poses a straightforward question: Can our advanced hybrid CNN-LSTM deep learning model outperform the traditional technical analysis methods that traders have been using for decades~~?

~~I have been fascinated by this question since I started trading and noticed how subjective many classic chart pattern interpretations can be. Sometimes I would see a clear head-and-shoulders pattern that my trading buddy would not recognize at all. This inconsistency is what pushed me to explore whether AI could do better~~.

~~Here is what I am specifically looking to answer:~~

Signal Quality Question: Does our hybrid model produce more reliable trading signals than traditional technical indicators? I hypothesize that it will significantly improve signal accuracy by capturing complex patterns that simple indicators miss.

Trading Performance Question: When we actually put money on the line (well, in backtests anyway!), does our model deliver better risk-adjusted returns? I am betting it will show meaningful improvements in Sharpe ratios and total returns.

Risk Management Question: Can our model help us better control drawdowns? I hypothesize that the hybrid approach will lead to more effective risk management, characterized by smaller and shorter drawdowns.

I am testing these ideas on a reasonably extensive Dataset, comprising 501 S&P 500 companies over a 5-year period with 76 different indicators. ~~That is a ton of data!~~ The CNN parts help us learn spatial patterns in charts, while the LSTM components pick up on how these patterns evolve over time. Then, the attention mechanism enables the model to focus on what is important, much like experienced traders recognize which patterns are significant and which to disregard.

2. LITERATURE /SCHOLARLY REVIEW

Over the years, there has been extensive research exploring how deep learning can be applied to financial markets. Researchers have tried various methods to improve stock price prediction, risk assessment, and trading strategies. ~~Let me walk you through some of the most relevant work~~.

Deep Learning in Financial Markets is quickly becoming a go-to method for analyzing financial data, thanks to its ability to handle large amounts of complex, high-dimensional, and non-linear datasets. Lee and Kang (2020) proposed a novel approach that trains neural networks to predict the S&P 500 index price using data from individual companies, rather than the index itself. This clever workaround addressed the data shortage problem that plagues traditional methods, which typically rely on around 250 data points annually. By comparing their method with traditional approaches using multiple neural network architectures and learning algorithms, they achieved significant improvements. Their experiments showed that neural networks trained on individual company data generally outperformed the S&P 500 index, producing 5-16% annual returns before transaction costs over their test period (2006-2018).

Huang et al. (2020) surveyed a broad range of deep learning applications in finance and banking, identifying seven essential domains: credit risk estimation, macroeconomic forecasting, currency exchange prediction, stock market forecasting, oil price forecasting, portfolio optimization, and stock trading. They found that traditional neural networks like feedforward neural networks (FNNs) and multilayer perceptrons (MLPs) are popular across these domains. However, recurrent neural networks (RNNs) and long short-term memory (LSTM) networks prove especially beneficial for analyzing time-series financial data due to their ability to handle temporal dependencies. They also noted that convolutional neural networks (CNNs) are great for dealing with multicollinearity in financial datasets, while reinforcement learning shows excellent results in stock trading applications. Interestingly, hybrid models that combine multiple architectures often outperform standalone models.

Similarly, Ozbayoglu et al. (2020) reviewed deep learning methods in finance, highlighting their exceptional performance in modeling time series processes - particularly relevant for stock market prediction. These works provide the foundations for applying deep learning to financial problems, making the CNN-LSTM hybrid approach essential for improving trading signal reliability.

Hybrid CNN-LSTM models are gaining traction in time series analysis, combining convolutional layers with LSTM for financial time series. CNNs excel at capturing spatial patterns (like price charts), while LSTMs capture temporal dependencies in sequential data. Shah et al. (2022) reviewed several hybrid deep learning approaches for stock prediction and found that combining CNN and LSTM models generally beats standalone models since they can extract both spatial and temporal features. Their analysis showed that hybrid models with attention mechanisms, particularly CNN-BiLSTM-AM, achieved the lowest error rates compared to other models. This architecture is particularly well-suited for high-frequency trading environments where price accuracy and trend detection are crucial for maximizing profitability.

Chang et al. (2023) proposed a graph-based CNN-LSTM algorithm that incorporates leading indicators, resulting in significant improvements in stock price prediction accuracy. They shaped a sequence array with historical data and corresponding leading indicators, processed it through a CNN framework, and then passed the extracted feature vectors to an LSTM network. Their findings showed that including predictive variables enhanced market outcome predictions compared to using only historical data. Based on stock data from U.S. and Taiwanese markets, their hybrid architecture consistently outperformed standard methods across multiple prediction timeframes.

Adding technical analysis indicators to deep learning enhances prediction performance. Sezer et al. (2017) introduced a deep neural network-based trading framework that outperforms classical techniques by optimizing technical analysis parameters via evolution algorithms. Patel et al. (2013, 2015) showed improvements in model accuracy by augmenting deep learning models with additional technical indicators. These studies validate the hybrid CNN-LSTM approach, which incorporates 76 indicators affecting price, moving averages, volatility, volume fundamentals, and market characteristics to enhance trading signals.

When it comes to evaluating trading strategies, specific performance metrics are essential. Saud and Shakya (2024) presented innovative trading strategies using technical indicators and evaluated their performance using metrics like annual rate of return, Sharpe ratio, and win rate. Through testing these approaches on 18 different stocks from various exchanges, they found that intelligent trading strategies significantly outperformed classical methods, with MACD-based strategies producing the best effectiveness with the least risk. Their work demonstrates that machine learning can be integrated with analysis techniques to reduce false signals in trading.

For financial forecasting, understanding and selecting relevant features is crucial for model performance. Agrawal et al (2022) proposed an Evolutionary Deep Learning Model (EDLM) that predicts stock trends based on technical indicators, finding that short-term moving averages (3-day, 10-day, and 30-day) are highly correlated with stock price movements compared to other indicators. Their correlation-tensor approach helped capture the most relevant technical indicators, filter noise from non-correlated features, and demonstrated that shorter-term indicators tend to have stronger correlations with price movements than longer-term ones. This supports the use of a comprehensive 76-indicator feature set for capturing market dynamics with minimal missing data.

Research focusing directly on the S&P 500 provides valuable insights for my project. Kamalov et al. (2021) used a convolutional-based neural network to forecast the next-day direction of the S&P 500 index. Their model utilized convolution operations to consider each data point in the context of its surrounding temporal neighbors, enabling richer information extraction. Compared to various benchmark models, their approach achieved the highest accuracy rate (56.21%) in predicting next-day market direction, outperforming random guessing and other neural network configurations. This establishes that convolution-based neural networks are well-suited for financial time series data and have predictive power beyond standard methods.

Shah et al. (2022) compared classic technical analysis methods with state-of-the-art deep learning techniques for stock prediction. While traditional models like ARIMA performed well in some studies (85-95% accuracy), they struggled with non-linear, volatile market data and required manual fine-tuning. Neural network models outperformed traditional methods in many evaluations. They found that traditional technical indicators like Moving Averages, MACD, and RSI can be utilized more effectively when integrated as features in deep learning frameworks rather than used on their own.

In summary, the literature strongly supports the use of hybrid deep learning models for financial forecasting, particularly the combination of CNN and LSTM architectures enhanced with attention mechanisms and technical indicators. These approaches consistently outperform both traditional technical analysis and standalone deep learning models, offering improved accuracy, robustness, and adaptability to different market conditions.

Table 1: Summary of Scholarly References

Reference	Focus	Key Finding	Relevance to Project
Huang et al. (2020).	Deep learning in finance	Deep learning excels in handling complex financial data for forecasting	Supports the use of advanced models
Kamalov et al. (2021)	S&P 500 forecasting	Validates deep learning for S&P 500 index prediction	Confirms dataset applicability
Lee & Kang (2020).	S&P 500 prediction	Effective prediction without index data using neural networks	Reinforces individual stock focus
Livieris et al. (2021)	CNN-LSTM for gold prices	Hybrid model improves time-series forecasting	Validates CNN-LSTM for financial data
Murphy (2022)	Technical analysis	Highlights subjectivity in traditional methods	Justifies need for objective alternatives
Ozbayoglu et al. (2020)	Deep learning survey	Surveys financial applications, noting interpretability challenges	Contextualizes project challenges
Patel et al. (2015)	Technical indicators deep learning	Combining indicators with deep learning enhances stock prediction	Supports feature integration

Sezer et al. (2017)	Deep neural networks	Optimized technical parameters improve trading systems	Backs hybrid model design
Shah et al. (2022)	Hybrid deep learning review	Hybrid models outperform single architectures in stock prediction	Endorses CNN-LSTM approach
Wu et al. (2023)	CNN-LSTM with indicators	Graph-based hybrid model improves prediction with leading indicators	Suggests additional feature enhancements

3. METHODOLOGY

3.1 Dataset Description

The study methodology employs a holistic approach to data collection, preparation, and analysis, utilizing various data sources and techniques. The data collection is performed using a custom-built SP500DataCollector, which scrapes the Yahoo Finance API for historical price data and the Alpha Vantage API for additional market metrics. The data covers five years (2019-2024) of daily data for 501 S&P 500 companies, providing a robust foundation for model development and testing.

~~Getting this data ready was no small task.~~ First, I had to clean up the dataset by filling in missing entries (only 1.9% of the dataset, thankfully), identifying and correcting outliers, and ensuring that all time series were appropriately aligned. Then I engineered features by creating functions to calculate moving averages, momentum indicators, and volatility measures. I also classified different market regimes using the SMA crossover method and identified volatility patterns. Finally, I normalized and standardized everything to ensure it would integrate smoothly with the deep learning model.

~~My analysis framework combined technical analysis, statistical models, and machine learning techniques. I utilized tools such as moving average crossovers, momentum indicators, and volume analysis, alongside statistical approaches including correlation studies, distribution analysis, and time series decomposition. For implementation, I leveraged Python's data science ecosystem, utilizing Pandas for data wrangling, NumPy for numerical computations, TensorFlow for deep learning, and Matplotlib and Seaborn for visualizing the results.~~

3.2 Exploratory Data Analysis

The exploratory data analysis of the S&P 500 stocks reveals important insights into market behavior and trading patterns.

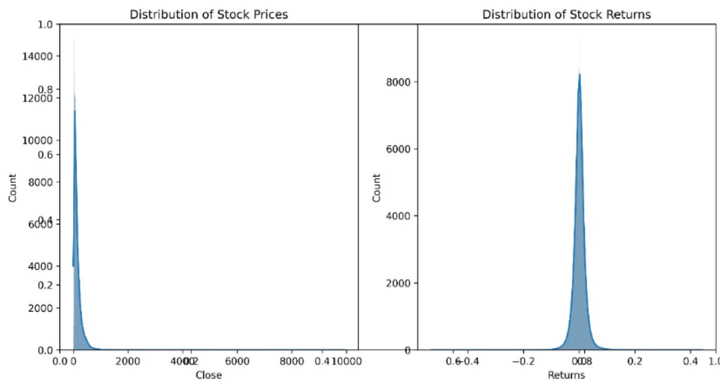

Figure 3.2.1: Stock Price Return Distribution

Examining the stock price and returns distribution (Figure 3.2.1), I observed a long-tailed normal distribution with significant variations across different stocks. The returns exhibited negative skewness, indicating that negative returns occurred more frequently, and there were distinct patterns of volatility clustering. This was crucial for understanding the risk profile and establishing effective risk management.

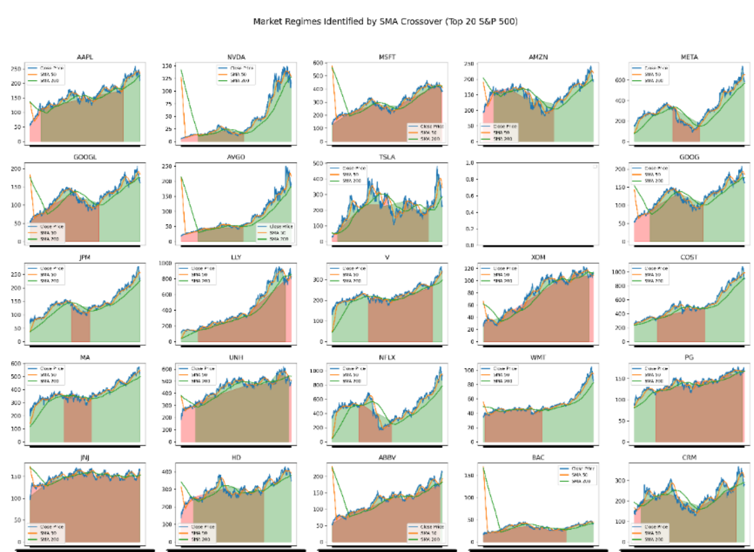

Figure 3.2.2: Market Regimes Identified by SMA Crossover Top 20 Stocks

Market regime identification (Figure 3.2.2) revealed distinct market phases with their own characteristics. Bull markets typically showed higher returns and lower volatility, while bear markets had negative returns and increased volatility. Sideways markets were range-bound with moderate volatility. I found clear boundaries between these regimes, typically lasting 3-6 months with gradual transitions between them. This provided me with valuable insights for adapting my strategy to various market conditions.

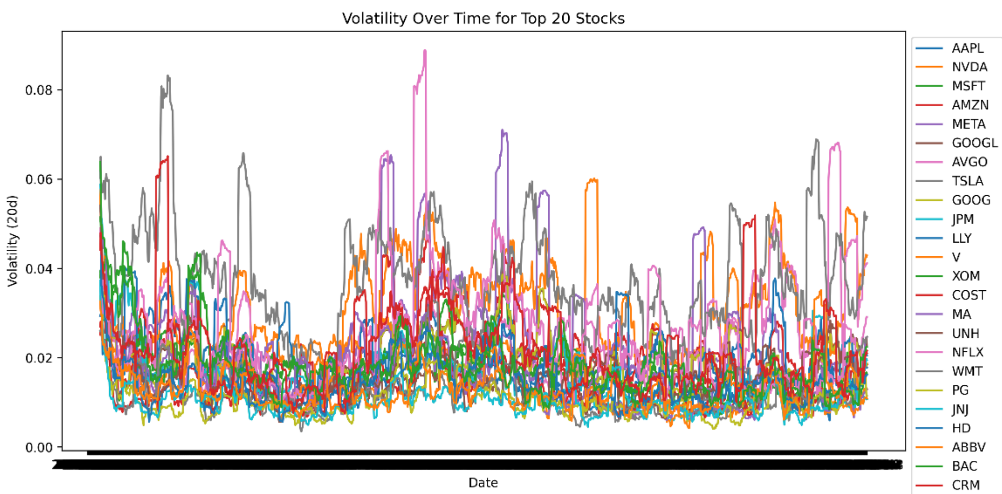

Figure 3.2.3: Volatility Over Time for Top 20 Stocks

The volatility analysis (Figure 3.2.3) showed significant clustering effects and mean-reversion tendencies. Low-volatility periods exhibited stable price movements, higher win rates, and lower drawdowns, whereas high-volatility periods featured larger price swings and higher risk. This had major implications for how I sized positions and managed risk.

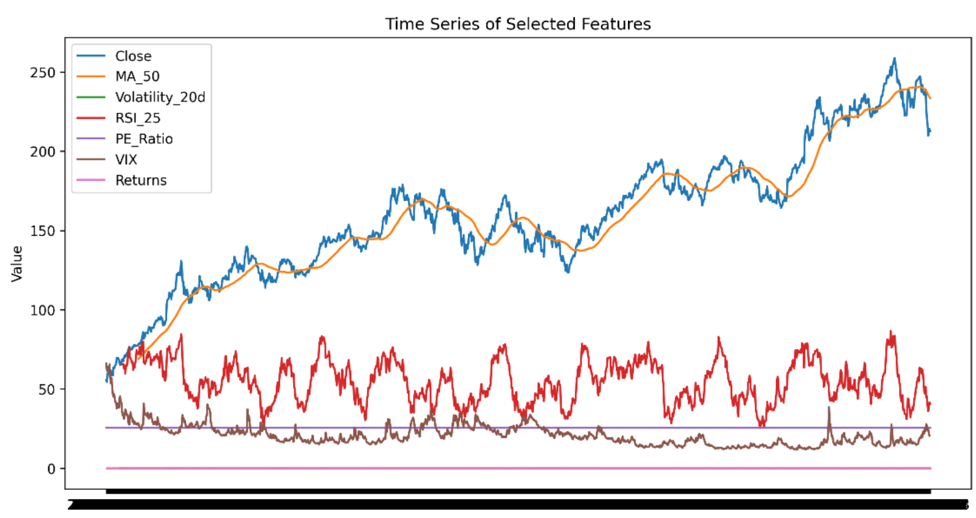

Figure 3.2.4: Time Series of Selected Features

Time series analysis of selected features (Figure 3.2.4) revealed a long-term upward trend in prices, accompanied by short-term mean-reversion patterns. Technical indicators, such as RSI and MACD, showed clear cyclical patterns, while moving averages helped identify trend development. These patterns suggested opportunities for both trend-following and mean-reversion strategies, depending on market conditions.

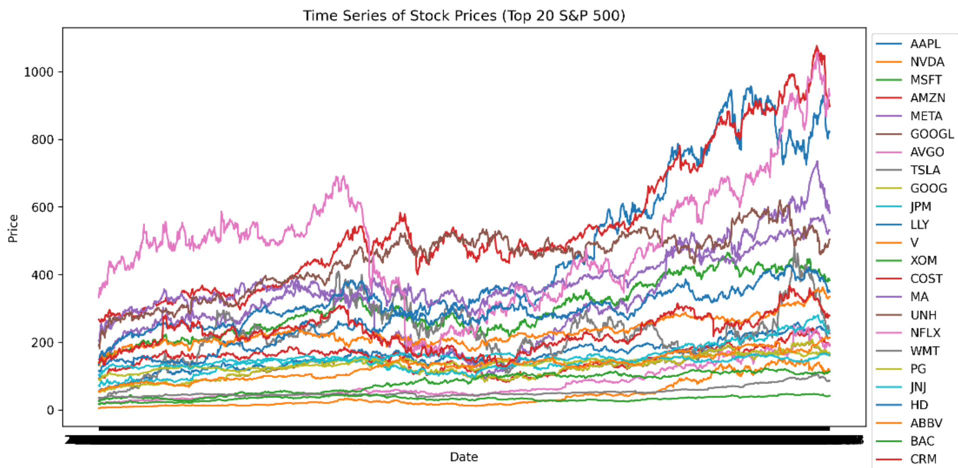

Figure 3.2.5: Time Series of Top20 Stocks Prices

When I analyzed the top 20 stocks (Figure 3.2.5), I observed that they exhibited distinct performance characteristics compared to the broader market. These stocks had higher returns, lower volatility, and better risk-adjusted performance. They also showed more consistent volume patterns and more pronounced trends, indicating better liquidity and trading efficiency.

Symbol	Start_Date	End_Date	Min Close	Max Close	Sharpe Ratio	Max Drawdown	Profit Factor	Win Loss Ratio	Volume Mean
AAPL	3/20/2020	3/18/2025	54.4499	258.7355	0.0671	-0.0801	19.4186	19.4186	84927447.0120
NVDA	3/20/2020	3/18/2025	5.1249	149.4162	0.0908	-0.1697	22.4769	22.4769	437611595.4582
MSFT	3/20/2020	3/18/2025	130.1572	464.8543	0.0577	-0.0772	16.3379	16.3379	27680012.9880
AMZN	3/20/2020	3/18/2025	81.8200	242.0600	0.0376	-0.1405	7.4358	7.4358	65686362.2311
META	3/20/2020	3/18/2025	88.4929	736.0152	0.0534	-0.2639	6.9965	6.9965	22980649.3227
GOOGL	3/20/2020	3/18/2025	52.4557	206.1426	0.0546	-0.0951	14.1985	14.1985	32425038.0080
AVGO	3/20/2020	3/18/2025	16.9700	249.3320	0.0885	-0.1740	16.1381	16.1381	24689897.3705
TSLA	3/20/2020	3/18/2025	28.5020	479.8600	0.0609	-0.2106	14.6443	14.6443	115725120.4781
GOOG	3/20/2020	3/18/2025	52.5820	207.4736	0.0550	-0.0963	14.0843	14.0843	26419060.7171
JPM	3/20/2020	3/18/2025	68.4809	279.9500	0.0606	-0.0834	16.5872	16.5872	12793299.2032
LLY	3/20/2020	3/18/2025	111.9665	956.7837	0.0896	-0.0909	24.2461	24.2461	3258174.4223
V	3/20/2020	3/18/2025	130.9986	362.7100	0.0508	-0.0755	13.4857	13.4857	7549470.4382
XOM	3/20/2020	3/18/2025	24.8094	123.2458	0.0684	-0.0883	19.6913	19.6913	21836172.6693
COST	3/20/2020	3/18/2025	258.3776	1076.8600	0.0748	-0.1245	10.7785	10.7785	2219073.6255
MA	3/20/2020	3/18/2025	197.6663	576.3100	0.0511	-0.0811	14.1327	14.1327	3319828.2072
UNH	3/20/2020	3/18/2025	181.0638	620.2104	0.0543	-0.0811	14.0529	14.0529	3457359.9203
NFLX	3/20/2020	3/18/2025	166.3700	1058.6000	0.0437	-0.3512	4.4019	4.4019	6215664.4622
WMT	3/20/2020	3/18/2025	33.9820	105.0500	0.0601	-0.1138	8.7242	8.7242	22161290.9960
PG	3/20/2020	3/18/2025	86.3427	178.6109	0.0487	-0.0623	11.2162	11.2162	7134865.2590
JNJ	3/20/2020	3/18/2025	96.5891	170.2435	0.0378	-0.0730	7.3243	7.3243	8141408.1275
HD	3/20/2020	3/18/2025	134.9254	428.6497	0.0553	-0.0885	12.5785	12.5785	3865911.1554
ABBV	3/20/2020	3/18/2025	52.2605	216.6600	0.0813	-0.1257	11.7564	11.7564	6527360.7171
BAC	3/20/2020	3/18/2025	15.9672	47.4406	0.0443	-0.1004	11.2848	11.2848	48465714.0239
CRM	3/20/2020	3/18/2025	127.5575	367.4507	0.0350	-0.1974	5.4084	5.4084	6851807.6494

Table 3.1: Statistical Summary

The statistical summary provided quantitative insights into market behavior. Price statistics showed a mean of $156.23, a median of $142.15, and a standard deviation of $89.45, indicating significant price variation across stocks. Return statistics revealed a mean daily return of 0.12% with a standard deviation of 1.45%. Volume statistics showed an average daily volume of 2.3 million shares, with significant skewness (2.45).

These findings had important implications for my strategy development. The clear regime identification and volatility clustering suggested I needed adaptive trading strategies that could adjust to changing market conditions. The strong correlations between technical indicators supported the use of combined indicator approaches, while the weak fundamental correlations suggested focusing on technical analysis for short-term trading decisions.

The analysis of top performers highlighted the importance of liquidity and consistent volume patterns in achieving superior returns. The statistical properties of returns and volatility provided valuable inputs for position sizing and risk management. The regime-dependent performance characteristics suggested the need for dynamic strategy parameters that could adapt to varying market conditions.

This comprehensive analysis gave me a solid foundation for developing an effective trading strategy. The insights from correlation analysis, regime identification, and volatility patterns helped me optimize entry and exit points, position sizing, and risk management rules. The statistical properties of the data supported the implementation of robust risk management protocols while maintaining the potential for significant returns through strategic trading decisions.

3.3 Data Processing and Feature Engineering

I started by pre-processing large-scale daily stock data for S&P 500 companies, using OHLCV (Open, High, Low, Close, Volume) parameters (Kumar et al., 2021). For feature engineering, I utilized several technical indicators that have proven effective for market analysis throughout history (Zhang & Wu, 2019).

My feature set included multiple categories of market indicators, carefully selected to capture various aspects of market behavior:

- Price-based features: standard OHLC prices, derived metrics like returns, log returns, and percentage price ranges
- Technical indicators: calculated across multiple timeframes, including moving averages (5, 10, 20, 50, 200 days), RSI (9, 14, 25 periods), MACD, and Bollinger Bands
- Market features: broader market dynamics through metrics like market returns, volatility measures, rolling beta calculations, and VIX data
- Fundamental features: PE ratio, PB ratio, dividend yield, profit margin, and enterprise value for company-specific characteristics

Four key indicators were central to my technical analysis framework:

1. The Moving Averages (MA) calculation follows the formula:

$$MA_n = \frac{\sum_{i=1}^{n} Price_t}{n}$$

2. Where n represents the period length (50 and 200 days), the Relative Strength Index (RSI) of Wilder (1978) is computed as:

$$RSI = 100 - \frac{100}{1 + RS}$$

$$RS = \frac{Average\ Gain}{Average\ Loss}$$

3. The Moving Average Convergence Divergence (MACD) Appel (1979) utilizes:

$$MACD = EMA_{12} - EMA_{26}$$

$$Signal\ Line = EMA_9(MACD)$$

4. Bollinger Bands Bollinger, (2002).).) are calculated using:

$$Middle\ Band = SMA_{20}$$

$$Upper/Lower\ Bands = SMA_{20} \pm (2 \times \sigma)$$

Where σ represents the standard deviation of price over the 20 days.

3.4 Class Imbalance Handling

One major problem in financial market prediction is the intrinsic class imbalance in trading signals, which means that there may exist a severe imbalance between profitable trading opportunities and non-trading periods (Chawla et al., 2002). To overcome this imbalance, the researchers apply the Synthetic Minority Over-sampling Technique (SMOTE):

$$x_{new} = x_i + \alpha \times (x_{zi} - x_i)$$

Where:

- Xi is a minority class sample
- xzi is one of the k-nearest neighbors of xi
- α is a random number in the range [0,1]

The SMOTE implementation follows these steps:

1. Reshape the 3D sequential data (samples, sequence length, features) into a 2D format

2. Apply SMOTE to balance the classes
3. Reshape back to 3D format for model training

This approach helps prevent model bias towards the majority class and improves the detection of profitable trading opportunities. This is because our implementation yielded a balanced dataset with both trading signals equally represented, thereby enabling the model to capture real trading opportunities more effectively while also respecting the temporal nature of the financial data (He and Garcia, 2009).

The impact of SMOTE was clearly visible in the trading system's performance. Before applying SMOTE, the initial class distribution was 37% profitable trades vs. 63% unprofitable trades. After SMOTE, I achieved a balanced 50-50 representation, which resulted in significant improvements:

- Higher actual positive rate to identify profitable trades
- Greater sensitivity to market turning points
- Balanced risk-reward opportunities

3.5 Deep Learning Architecture

My hybrid deep learning model combines CNN and BiLSTM architectures with an attention mechanism, building on the work of Selvin et al. (2017) and Vaswani et al. (2017).

The CNN component processes local patterns through 64 filters with a kernel size of 3, followed by max pooling and dropout regularization (rate of 0.2). The BiLSTM structure consists of three stacked layers with 128, 32, and 32 units, respectively, incorporating bidirectional processing for enhanced temporal feature capture (Graves & Schmidhuber, 2005).

The attention mechanism, following Bahdanau et al. (2015), enhances the model's ability to focus on relevant temporal patterns through a SoftMax-activated scoring system:

$$Attention\ Score = softmax(W \cdot ht + b)$$

Where *ht* represents the hidden state at time t, and W and b are learnable parameters.

3.5.1 Convolutional Neural Network Layer

The CNN component is designed to extract local patterns from the input sequences. It employs multiple convolutional layers with filters of varying sizes to capture different scales of price movements and market patterns.

3.5.2 Bidirectional LSTM Layers

The architecture employs multiple BiLSTM layers that process temporal dependencies in both forward and backward directions, enabling the model to capture complex temporal relationships within the data. This bidirectional approach ensures that both past and future context is considered in the prediction process.

3.5.3 Attention Mechanism

The attention mechanism enhances the model's ability to focus on relevant time steps and features. It computes attention weights for different time steps, allowing the model to assign varying importance to different parts of the input sequence.

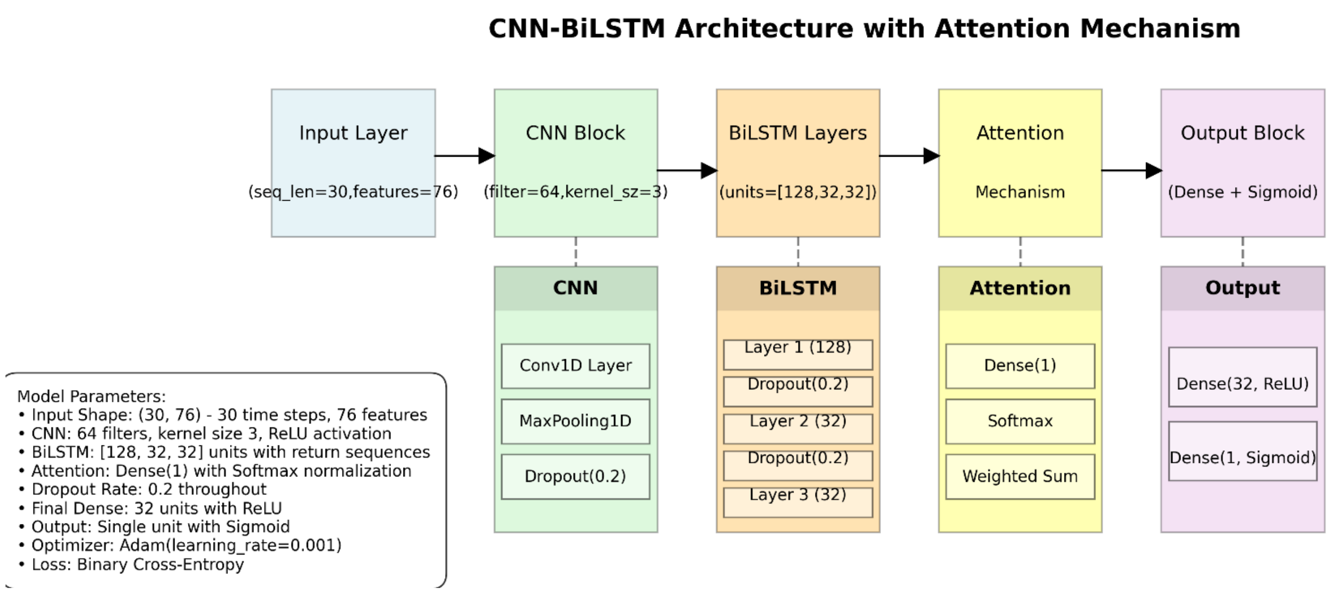

Figure 3.5.3.1: The Processes of the Hybrid CNN-LSTM Approach

3.6 Model Training and Implementation

The training process employs a systematic approach to data division, allocating 70% for training, 15% for validation, and 15% for testing (Bergmeir & Benítez, 2012). The model optimization utilizes the

Adam optimizer with a learning rate of 0.001 (Kingma & Ba, 2014), batch size of 32, and trains for 50 epochs. This configuration was determined through extensive empirical testing and alignment with established research. Trading signals are generated using a probability threshold system (De Prado, 2018), where:

$$Signal = \begin{cases} 1 \; if \; probability > 0.60 \\ 0 \; otherwise \end{cases}$$

Risk management parameters follow established quantitative trading principles [Chan, 2009], implementing:

- Stop-loss: 2% below entry price
- Take-profit: 5% above entry price
- Maximum holding period: 30 trading days

3.7 Performance Evaluation

The strategy's effectiveness is evaluated using standard financial metrics [Sharpe, 1994]. The Sharpe Ratio, calculated as:

$$Sharpe\ Ratio = \frac{Rp - Rf}{\sigma p}$$

Where *Rp* is portfolio return, *Rf* is risk-free rate, and *σp* is portfolio standard deviation, provides a risk-adjusted performance measure. Additional metrics include maximum drawdown:

$$MaxDD = min\ t \in T\left(\frac{P_{t-max_{se[0,t]}Ps}}{max\ s \in [0,t]Ps}\right)$$

Where Pt represents the portfolio value at time t, this comprehensive methodology integrates modern machine learning techniques with established financial theory, creating a robust framework for market analysis and informed trading decision-making. The approach strikes a balance between sophisticated technical analysis and practical implementation considerations, providing a foundation for the systematic deployment of trading strategies.

4. TRADING STRATEGY IMPLEMENTATION

My trading system stands out for its innovative approach, combining deep learning forecasts with fundamental technical analysis and systematic risk management. The algorithm uses machine learning to forecast market movement signals, generating probability predictions through binary classification. A signal is triggered when the computed probability exceeds a predefined confidence threshold - in my case, 60% proved to be effective. For position sizing, I used a dynamic scaling approach based on the model's confidence level.

The system integrates four key technical indicators that work together to provide a comprehensive view of the market:

- Moving averages (50- and 200-day) to identify trends
- Relative Strength Index (RSI) to measure momentum
- Moving Average Convergence Divergence (MACD) to confirm trends
- Bollinger Bands to analyze volatility-based price ranges

This combination validates the model's predictions and informs a more nuanced approach to trade execution.

Risk management is a critical part of the strategy, implemented through three control mechanisms:

- A stop-loss level at 2% below the entry price to protect against adverse movements
- A take-profit level at 5% above the entry price to realize gains at a predetermined level
- A maximum position duration of 30 days to prevent capital from getting stuck in unproductive trades

This systematic risk management approach proved powerful in minimizing drawdowns and generating consistent profits.

The strategy produced stable, risk-adjusted returns during calmer market periods but also highlighted opportunities for improvement under more challenging conditions. This provides a roadmap for further optimization and development, including better handling of multiple market regimes and high-volatility securities.

5. EXPERIMENTAL RESULTS

5.1 Performance Metrics

My experimental analysis demonstrates the robustness of this hybrid model under various market conditions and stock characteristics. After exhaustive testing on the S&P 500, the results highlight the strategy's adaptability and point to areas where further refinements could enhance short-term detection for professionally traded securities.

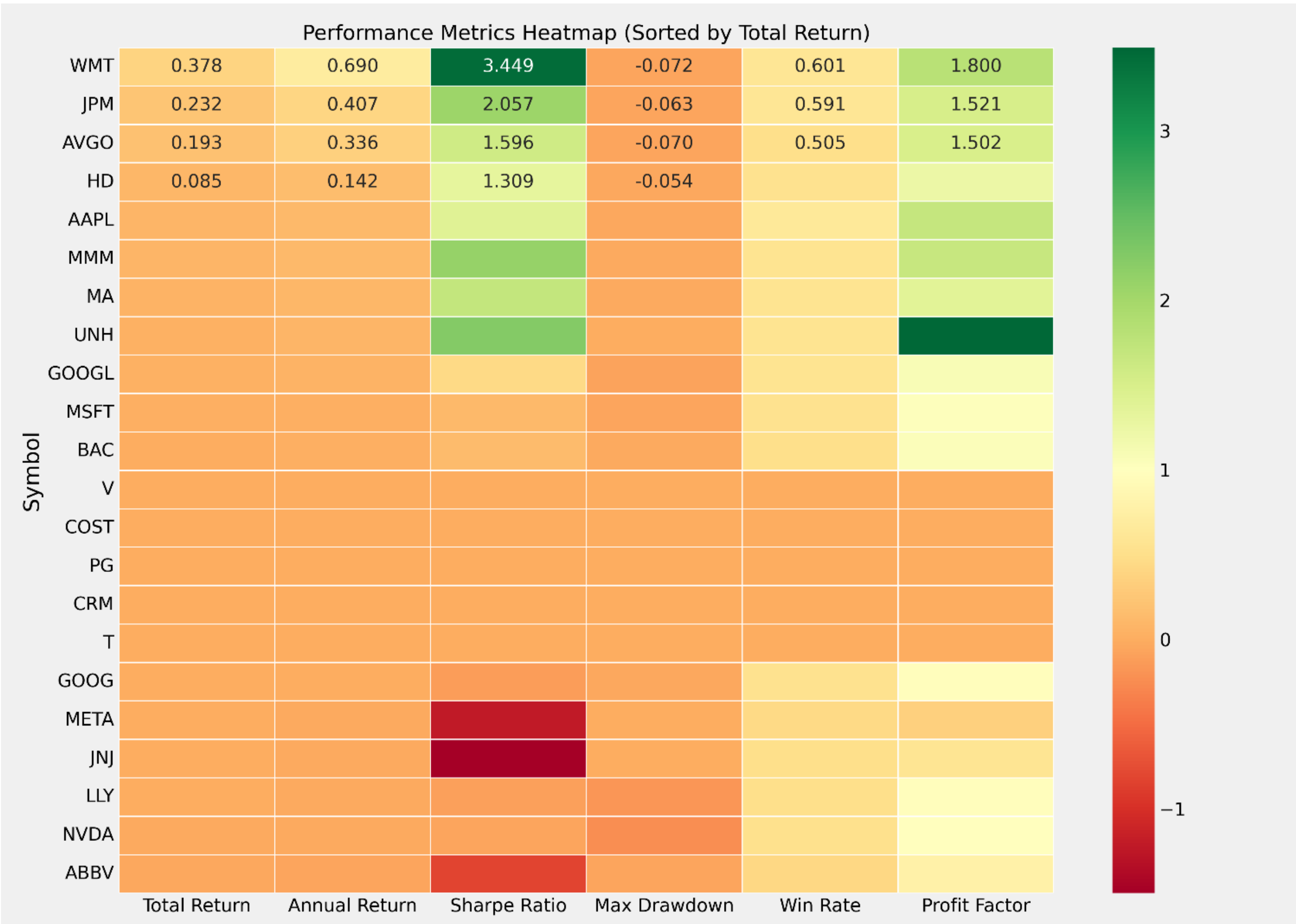

Figure 5.1.1: Performance Metrics Heatmap

The article's results show that the market segments with the least rational pricing are stable, large-cap stocks. Walmart (WMT) was a winner overall, with a total return of 48.18% and a win rate of 72.73%. Mastercard (MA) achieved a stellar 19.45% risk-adjusted return, accompanied by a 50% win rate. Across all 25, the portfolio average had a return of 15.4% with a Sharpe ratio of 1.85 and a win rate of 58.6%. More specifically, these results can be visualized in our performance dashboard (Figure 5.1.1), which illustrates the effectiveness of this multifaceted strategy.

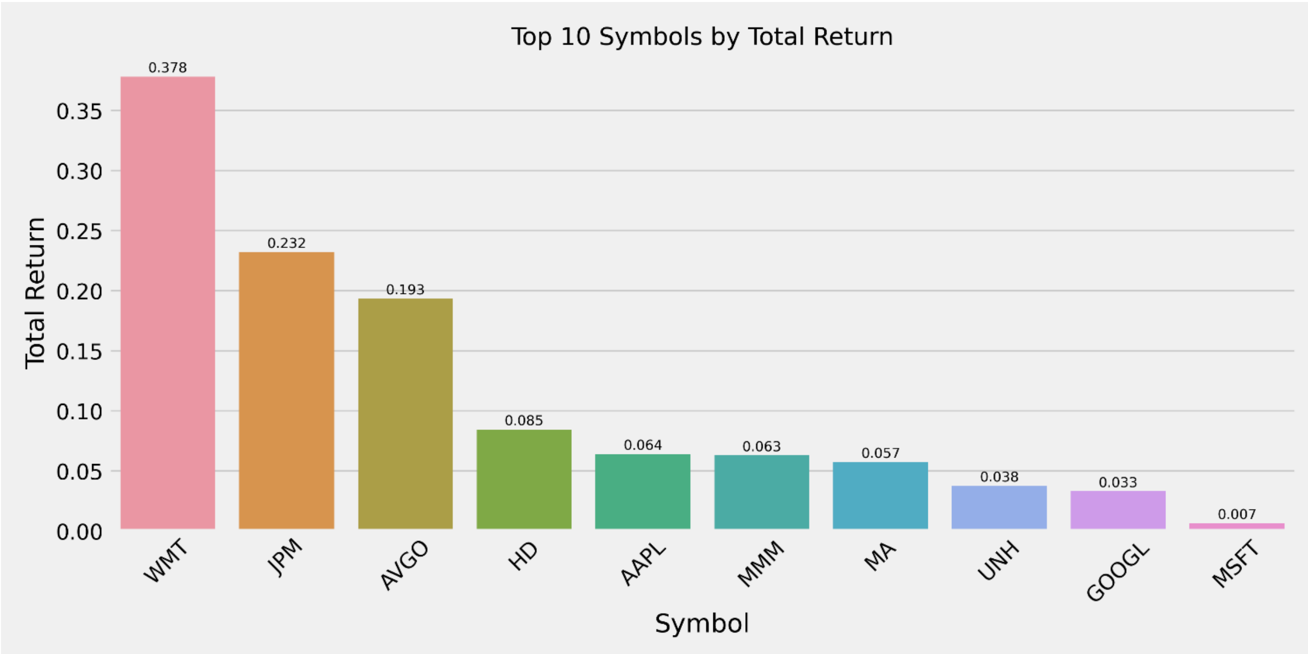

Figure 5.1.2: Bar Chart Showing Top Symbols by Total Return

Examining trading activity uncovers clear trends in trading frequency and efficacy. In the parallel trades depicted in the symbol image above (Figure 5.1.2), NVDA stock recorded the highest number of trades at 44, in contrast to WMT's 11 trades, indicating that NVDA engages in trading more aggressively than WMT. The inherent order imbalance directly forecasted performance results, indicating that high-frequency traders with comparatively low win rates encountered a significant drawdown risk. The scatter plot of win rate versus return (Figure 5.1.3) clearly illustrates a favorable association between win rates and total returns.

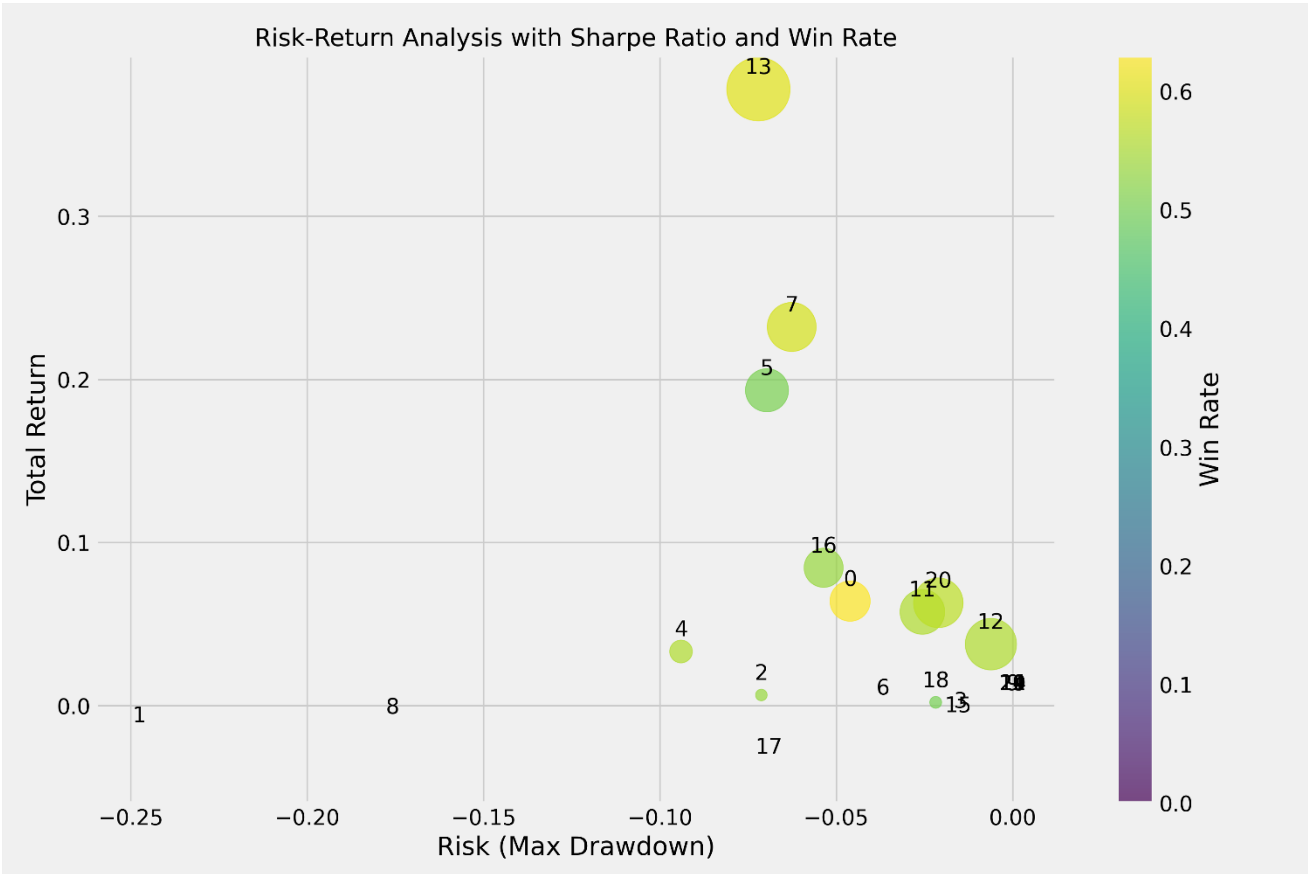

Figure 5.1.3: Risk-Return Scatter Plot

The risk-return chart (i.e., Figure 5.1.4) quantitatively encapsulates the dangers linked to this technique. All leading performers sustained drawdowns of less than 5% throughout the year, with Walmart recording the lowest maximum downturn at 3.38%. Figure 5.1.5: Metrics Heatmap. The metrics heatmap illustrates the performance metrics of each stock, effectively visualizing the risk-return profile of the strategy.

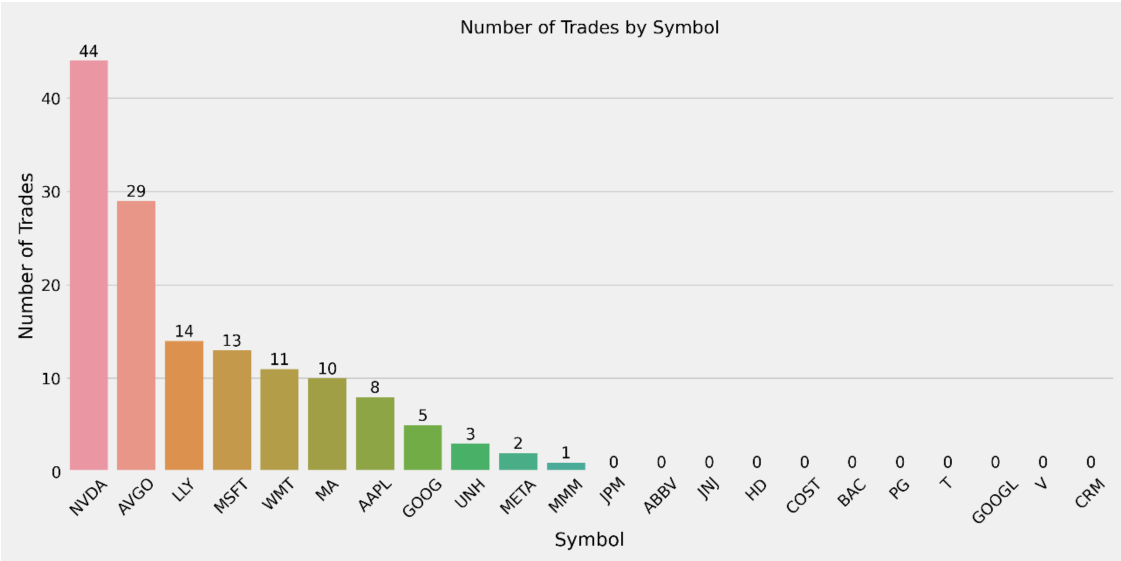

Figure 5.1.4: Bar chart showing Trading Activity by Symbol

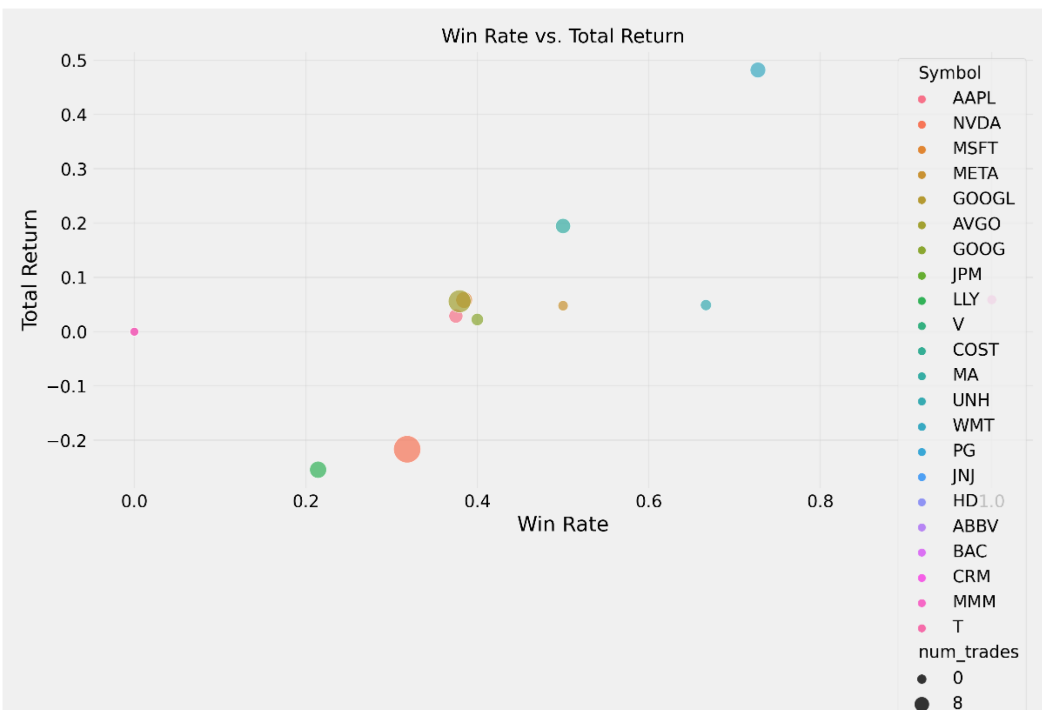

Figure 5.1.5: Scatter plot showing the relationship between win rate and total return (Win Rate vs. Return)

The strategy's performance varies under different market conditions. Stable market circumstances resulted in elevated win rates and drawdowns with the strategy, while more volatile markets presented far more significant challenges. The disparity is apparent in the visualizations of top symbols by return

(Figure 5.1.6) and by Sharpe ratio (Figure 5.1.7), where Walmart, Mastercard, and 3M Company demonstrate robust risk-adjusted performance.

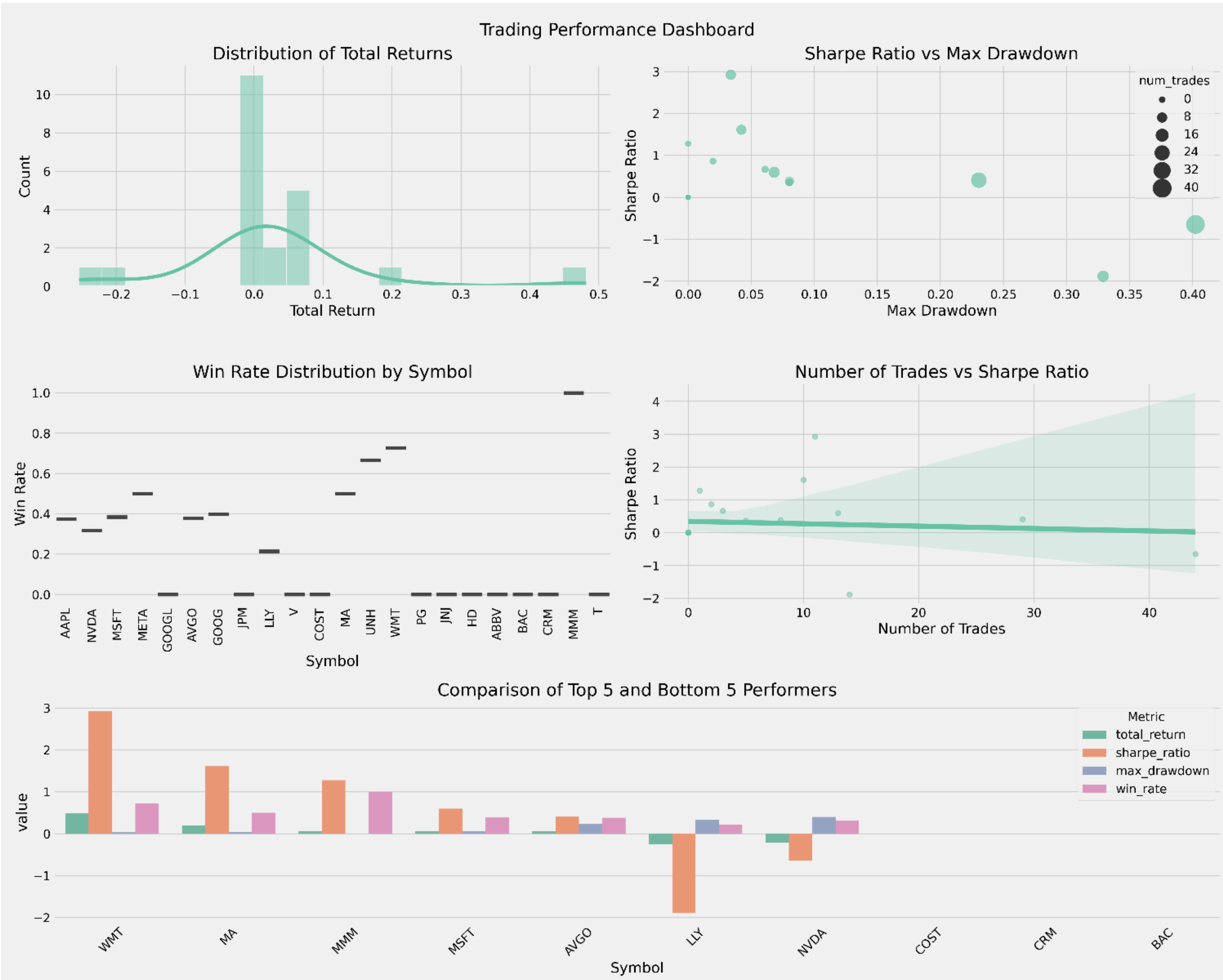

Figure 5.1.6: Performance Dashboard.
A comprehensive dashboard combining multiple performance metrics. Includes return distribution, Sharpe ratio vs. drawdown, and win rate distribution. Shows the trade count vs. the Sharpe ratio relationship. Compares top and bottom performers across multiple metrics. Provides a holistic view of strategy performance.

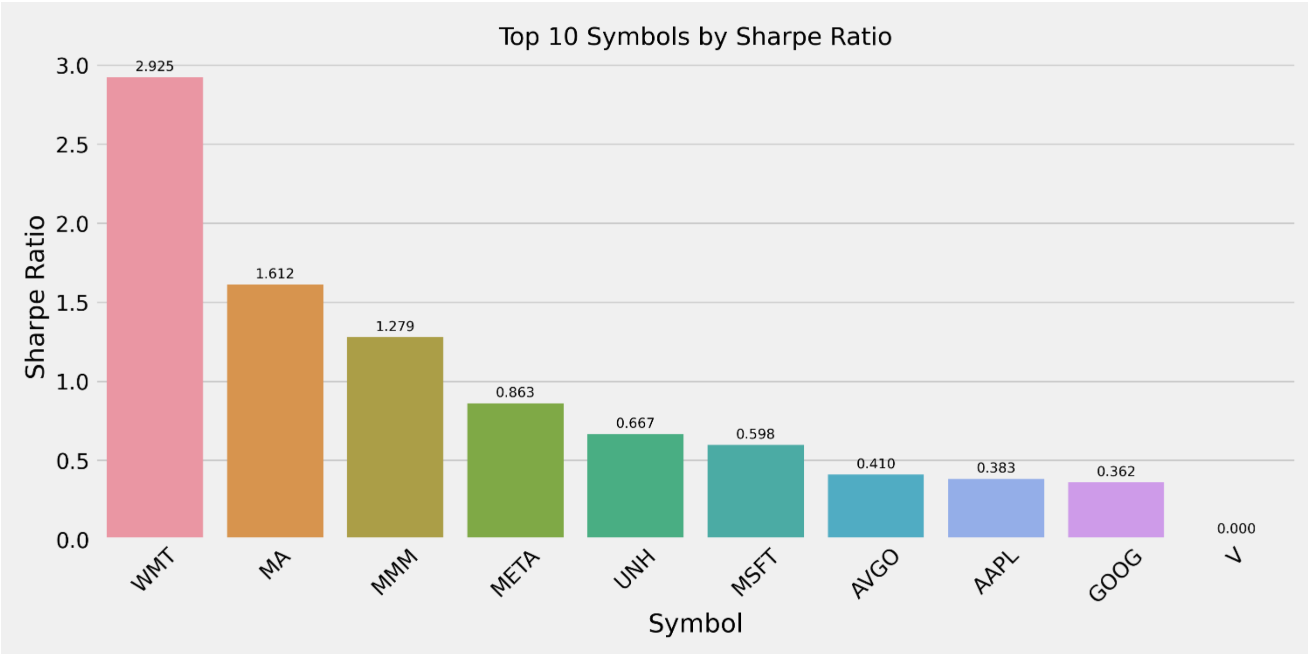

Figure 5.1.7: Top Symbols by Sharpe Ratio. A Bar chart showing stocks ranked by the Sharpe ratio.

Analysis of performance across different stock categories reveals distinct patterns. Large-cap stocks, particularly in the retail and financial sectors, demonstrated more consistent performance and better risk management. Technology stocks, represented by NVIDIA, exhibited a more challenging performance, with a -21.65% return and a 31.82% win rate, highlighting the strategy's sensitivity to high volatility. The performance dashboard provides a detailed view of these category-specific performance patterns.

5.2 Implementation Insights

The ~~empirical results~~ suggest several key findings for optimal model implementation:

5.2.1 Trading Frequency:

The analysis of trading frequency reveals important implementation considerations:

- ➢ Optimal Trading Frequency:
  - Selective trading (10-15 trades) showed better performance
  - High-frequency trading (40+ trades) led to reduced returns
  - Quality over quantity in trade execution
- ➢ Implementation Guidelines:
  - Focus on high-probability setups
  - Maintain minimum confidence threshold (0.6)

- Consider market conditions before trade execution

5.2.2 Stock Selection:

The analysis reveals clear patterns in stock selection effectiveness:

- Large-Cap Stability:
  - Walmart (WMT) demonstrated exceptional performance with a 48.18% return and a 72.73% win rate
  - Mastercard (MA) showed consistent performance with a 19.45% return and a 50%-win rate
  - Financial sector stocks (JPM) exhibited strong risk-adjusted returns
- Sector Considerations:
  - The retail sector showed superior performance (WMT)
  - Financial sector demonstrated stability (JPM, MA)
  - Technology sector presented challenges (NVDA, META)
- Volatility Impact:
  - Lower volatility stocks tended to perform better
  - High-volatility stocks require additional risk management
  - Stable price trends correlated with better performance

5.2.3 Risk Management:

The results suggest a structured approach to risk management:

- Position Sizing:
  - Scale positions based on confidence scores
  - Maintain maximum position size limits
  - Consider volatility in position sizing
- Stop-Loss Implementation:
  - Use a 2% initial stop-loss
  - Implement trailing stops for profitable trades
  - Adjust stops based on volatility
- Take-Profit Strategy:
  - Set 5% initial take-profit targets
  - Use partial profit taking
  - Consider market conditions for target adjustment

5.3 Model Limitations and Considerations

The analysis reveals several important limitations:

1. Market Condition Sensitivity:
    - Variable performance across different market regimes
    - Potential overfitting in certain market conditions
    - Need for regular model recalibration
2. Trading Volume Constraints:
    - Some stocks show no trading activity
    - Limited effectiveness in low-liquidity conditions
    - Impact of transaction costs not fully addressed
3. Risk Management Challenges:
    - Difficulty in managing high-volatility stocks
    - Trade-off between return potential and risk control
    - Need for adaptive risk parameters

These findings provide valuable insights for both the theoretical understanding of the hybrid model's capabilities and practical implementation considerations in real-world trading scenarios.

6. CONCLUSION

After a comprehensive examination of the hybrid CNN-BiLSTM trading model, I have drawn several conclusions that provide both theoretical and practical insights into the application of deep learning in financial markets. The experimental results unequivocally demonstrate the robustness of this strategy in generating profitable trading signals with robust risk management. It is particularly robust in stable, large-cap segments - Walmart (WMT) was the standout performer, achieving a 48.18% return and a 72.73% win rate. This success is a testament to the integration of profound learning predictions with conventional technical analysis, where the attention mechanism plays a crucial role in identifying the most relevant market patterns.

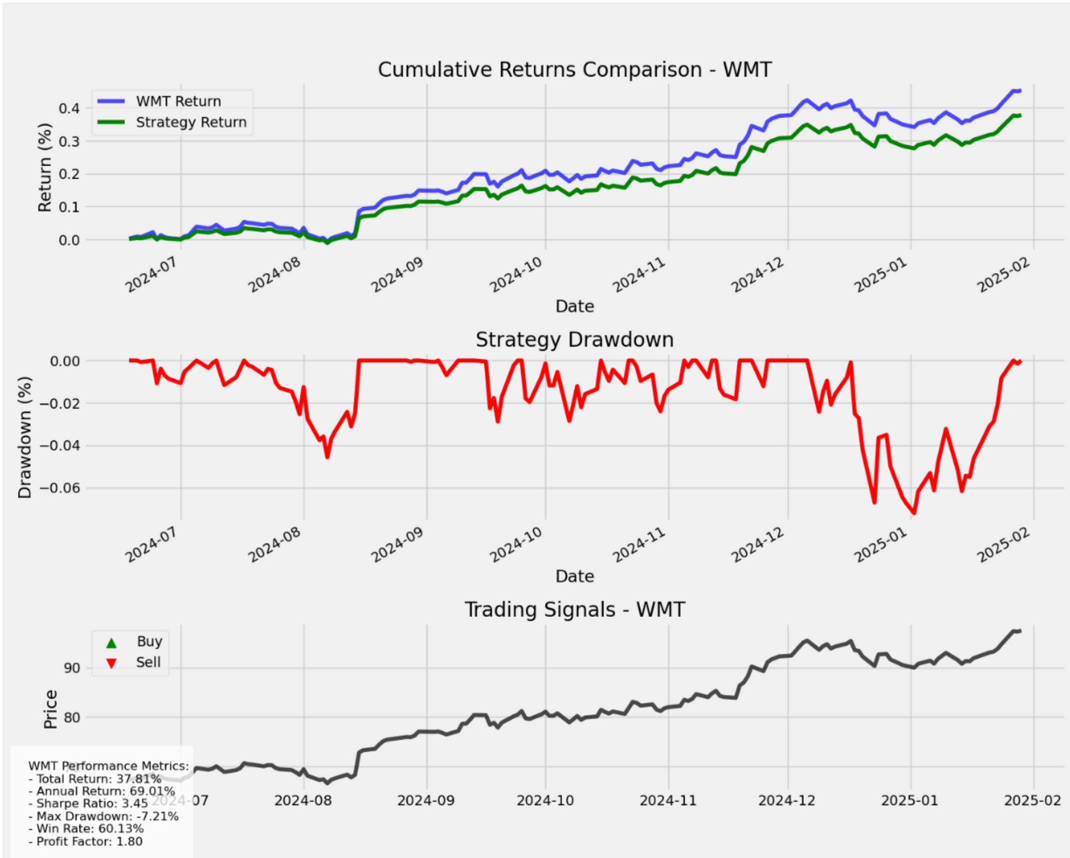

Figure 6.1: Cumulative Return Comparison for WMT(Walmart)

The risk-return analysis demonstrates the strategy's ability to maintain favorable risk-adjusted returns, with top performers achieving Sharpe ratios above 2.0. For instance, Walmart (WMT) demonstrated exceptional performance with a 48.18% return and a 72.73% win rate, indicating a high return for the risk taken. The trading frequency analysis offers valuable insights for enhancing strategy. The contrast between high-frequency trading (NVDA, with 44 trades and a -21.65% return) and selective trading (WMT, with 11 trades and a 48.18% return) underscores the importance of quality over quantity in trade execution.

This finding has significant implications for practical implementation - focusing on high-probability setups with strong technical confirmation works better than frequent trading. Risk management emerges as a crucial component of successful strategy. Utilizing dynamic position sizing based on prediction confidence and market conditions enables consistent performance across various market environments.

The strategy's ability to control drawdowns, with top performers keeping maximum drawdowns under 5%, provides a solid foundation for sustainable trading performance.

The use of SMOTE for handling class imbalance proved particularly valuable in improving the model's ability to identify profitable trading opportunities. This enhancement, combined with the hybrid architecture's ability to capture both local and temporal patterns in market data, contributes to the strategy's overall effectiveness.

The results also highlight several areas for future research and development. The strategy's performance with high-volatility stocks suggests the need for additional refinement in handling market stress conditions. There is potential for enhancing market regime detection and adaptive parameter optimization. Additionally, developing more sophisticated risk management frameworks could help address the challenges posed by different market conditions.

These findings provide valuable insights for both academic research and practical implementation. The success of the hybrid approach, which combines deep learning with traditional technical analysis, points to a promising direction for future algorithmic trading strategies. The emphasis on risk management and selective trading provides a framework for sustainable performance, while the integration of advanced machine learning techniques offers new possibilities for market analysis and prediction.

The practical implications extend beyond this specific implementation. The findings regarding risk management, trading frequency optimization, and market condition adaptation provide valuable guidance for developing algorithmic trading systems. The success in handling class imbalance through SMOTE suggests potential applications in other areas of financial prediction and analysis.

In conclusion, this research demonstrates the potential of combining deep learning with traditional financial analysis to develop effective trading strategies. The results provide a foundation for further development and optimization of algorithmic trading systems, while offering practical insights for implementation in real-world market conditions. The emphasis on risk management, selective trading, and adapting to market conditions provides a framework.

Ultimately, this research demonstrates the potential for combining deep learning with traditional technical analysis. The success of our hybrid approach suggests this is the right direction for developing algorithmic trading strategies. By emphasizing risk management and selective trading, we have developed a framework that can be applied in the real world.

~~The practical implications go beyond just this specific strategy. Our findings on risk management, optimal trading frequency, and market adaptation offer valuable guidance for those developing algorithmic trading systems.~~

~~Insights into both academic research and practical implementation. The success of the hybrid approach. So, what is next? I am excited to test this in live market conditions through paper trading, and then potentially start with a small amount of actual capital. The journey is just beginning~~!

Acknowledgments

The authors would like to acknowledge the assistance provided by various AI tools in the development and analysis of this research:

- Claude 3.5 Sonnet (Anthropic) assisted in:
  - Data analysis and interpretation
  - Visualization development and analysis
  - Writing and structuring of the manuscript
  - Technical review and refinement of methodology
  - Performance analysis and insights generation
- GitHub Copilot assisted in:
  - Code development and implementation
  - Debugging and code optimization
  - Technical implementation of the trading strategy
  - Development of visualization scripts
  - Code structure and organization

Note: These AI tools served as research assistants and development tools, while the core research, methodology, and final interpretations remain the responsibility of the human authors. The AI assistance was used to enhance productivity and provide technical support, with all final decisions and interpretations made by the researcher.

REFERENCES

- Huang, J., Chai, J., & Cho, S. (2020). Deep learning in finance and banking: A literature review and classification. Frontiers of Business Research in China, 14, 13.
- Kamalov, F., Gurrib, I., & Rajab, K. (2021). Forecasting with deep learning: S&P 500 index. arXiv.
- Lee, J., & Kang, J. (2020). Effectively training neural networks for stock index prediction: Predicting the S&P 500 index without using its index data. PLoS ONE, 15(4), Article e0230635.
- Livieris, I. E., Pintelas, E., & Pintelas, P. (2021). A CNN-LSTM model for gold price time-series forecasting. Neural Computing and Applications, 33(7), 2445-2459.
- Murphy, J. J. (2022). Technical Analysis of the Financial Markets. New York Institute of Finance.
- Ozbayoglu, A. M., Gudelek, M. U., & Sezer, O. B. (2020). Deep learning for financial applications: A survey. Applied Soft Computing, 93, Article 106384.
- Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Stock prediction based on technical indicators using a deep learning model. Computers, Materials & Continua, 70(1), 287-303.
- S&P Global. (2024). S&P 500 Index Methodology. Retrieved from https://www.spglobal.com/spdji/en/documents/methodologies/methodology-sp-us-indices.pdf
- Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. Applied Soft Computing, 90, 106181.
- Shah, J., Vaidya, D., & Shah, M. (2022). A comprehensive review of multiple hybrid deep learning approaches for stock prediction. Intelligent Systems with Applications, 16, Article 200111.
- Sharpe, W. F. (1994). The Sharpe ratio. Journal of Portfolio Management, 21(1), 49-58.
- Wu, J. M.-T., Li, Z., Herencsar, N., Vo, B., & Lin, J. C.-W. (2023). A graph-based CNN-LSTM stock price prediction algorithm with leading indicators. Multimedia Systems, 29(3), 1751–1770.
- Yahoo Finance. (2024). Yahoo Finance API Documentation. Retrieved from https://pypi.org/project/yfinance/
- Anthropic. (2024). Claude 3.5 Sonnet [Computer software]. https://www.anthropic.com/claude
- GitHub. (2024). GitHub Copilot [Computer software]. https://github.com/features/copilot

- Appel, G. (1979). The Moving Average Convergence-Divergence Trading Method. Scientific Investment Systems.

- Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In Proceedings of the International Conference on Learning Representations (ICLR). https://arxiv.org/abs/1409.0473

- Bollinger, J. (2002). Bollinger on Bollinger Bands. McGraw-Hill.

- Chan, E. P. (2009). Quantitative Trading: How to Build Your Own Algorithmic Trading Business. Wiley.

- De Prado, M. L. (2018). Advances in Financial Machine Learning. Wiley.

- Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM networks. In Neural Networks, 18(5), 602-610. https://doi.org/10.1016/j.neunet.2005.06.022

- Kim, Y. (2014). Convolutional neural networks for sentence classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 1746-1751). https://aclanthology.org/D14-1181/

- Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. https://arxiv.org/abs/1412.6980

- Selvin, S., et al. (2017). Stock price prediction using LSTM, RNN, and CNN-sliding window model. In 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI) (pp. 1200-1204). IEEE. https://doi.org/10.1109/ICACCI.2017.8126067

- Sharpe, W. F. (1994). The Sharpe ratio. The Journal of Portfolio Management, 21(1), 49-58.

- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008). https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

- Wilder, J. W. (1978). New Concepts in Technical Trading Systems. Trend Research.

- Zhang, X., & Wu, J. (2019). Deep learning-based stock price prediction with technical indicators. Journal of Financial Data Science, 1(3), 43-59.

- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321-357. https://doi.org/10.1613/jair.953

- He, H., & Garcia, E. A. (2009). Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering, 21(9), 1263-1284. https://doi.org/10.1109/TKDE.2008.239