

AI-Enabled Automated Trading System: Trend Analysis, Entry/Exit Strategies and Risk Controls

Shlok Desai, Shravani Chavan, Pallavi Shinde, Vaishnavi Kurumkar, Prof. Rupali Adhau.

Students, Computer Engineering

Indira College of Engineering and Management, Pune, India

Abstract: *In today's highly volatile financial markets, human-driven trading often leads to inconsistent results due to emotional biases, inadequate market analysis, and delayed reactions to price fluctuations. Many beginner and even intermediate traders rely on social media tips or instincts rather than factual market data, which frequently results in poor decisions and significant financial losses. To overcome these limitations, this project proposes an AI-powered Trading System designed to automate trade analysis, strategy formation, and decision execution using artificial intelligence and machine learning models.*

The system uses real-time and historical market data to forecast possible price movements and automatically execute trades based on preset strategies, including profit-target and stop-loss mechanisms. Through the incorporation of AI-based algorithms, the system ensures emotionless trading by removing the influence of impulsive decisions, greed, or fear, all of which tend to impede profitability. By providing traders with a quick and dependable decision-support tool, trend forecasting, real-time execution, and portfolio performance monitoring, it further increases efficiency.

Data collection, preprocessing, predictive modeling, and automated trading execution are some of the AI components that make up the system's architecture, which enables it to dynamically adjust to changes in the market. Before entering actual markets, novices should practice trading strategies with virtual currency to reduce risk and build confidence.

All things considered, the suggested AI Trading System represents a significant advancement in intelligent, data-driven, and self-governing trading. It speeds up decision-making in addition to lowering manual intervention. By combining state-of-the-art technologies such as machine learning. With the use of automation and predictive analytics, this system seeks to make trading profitable and accessible for both new and seasoned traders in the quickly changing financial landscape..

Keywords: AI-powered trading system, automated trading, machine learning models, real-time market data, historical data analysis, emotionless trading, trading automation, data-driven decision making, market trend prediction, virtual trading

I. INTRODUCTION

In today's rapidly evolving digital economy, financial markets have become highly complex, data-driven, and unpredictable, making traditional human-based trading increasingly inefficient. With the volume of market data growing exponentially and market movements occurring within milliseconds, manual trading struggles to keep pace. Human traders often face limitations such as emotional bias, delayed decision-making, and restricted analytical capacity, which frequently lead to inconsistent results and financial losses. These challenges have paved the way for a more intelligent, automated, and data-centric approach to trading.

Machine learning (ML) and artificial intelligence (AI) have become game-changing technologies in a number of industries, with the financial sector being one of the most affected. AI is faster and more accurate than humans at processing large datasets, finding hidden patterns, and making trades. The financial sector can improve decision accuracy, risk management, and consistent performance—even in extremely volatile markets—by incorporating artificial intelligence (AI) into trading systems. Artificial intelligence (AI)-powered trading systems use historical and



real-time data to predict price movements, identify lucrative opportunities, and execute trades with little assistance from humans. The difficulties that both novice and seasoned traders encounter serve as the impetus for creating an AI trading system. Due to their lack of technical expertise, many novice traders make poor trading decisions by following social media trends, gut feelings, or emotional whims. Tracking market indicators and fluctuations requires constant monitoring, which even experienced traders find difficult. Decisions are frequently influenced by emotional stress, fear, and greed, which can result in rash choices or lost opportunities. By automating the entire decision-making process and minimizing human error, an AI-based solution offers a trustworthy, methodical, and objective substitute.

The inconsistent and subjective nature of human judgment in trading is the main issue this project attempts to solve. The AI Trading System seeks to substitute rational, data-supported, real-time strategies for irrational and delayed decisions. Its goals are to analyze past and present market data, use machine learning models to forecast future price trends, integrate stop-loss and profit-target mechanisms for efficient risk management, and design an intuitive user interface that makes intelligent trading tools accessible to even novice traders.

All things considered, by combining automation, predictive analytics, and intelligent decision-making, the AI Trading System marks a substantial breakthrough in contemporary financial technology. In a market that is becoming more and more volatile, it helps traders to minimize risks while operating with more accuracy, assurance, and consistency.

II. LITERATURE SURVEY

I. AI-Powered Stock Market Prediction Using Deep Learning” (Shah, 2019):

The application of deep learning models—more especially, Long Short-Term Memory (LSTM) networks—for stock price prediction is presented in this paper. The author concentrated on examining stock price time series data to find obscure trends that might be mature enough to make quick trading decisions. Historical price data, volume trends, and sentiment analysis of financial news were used to train the system. When compared to more conventional machine learning models like SVM and linear regression, the model performed better and produced more accurate price forecasts.

II. Financial Forecasting Through Reinforcement Learning Algorithms” (Deng, 2020):

Deng put forth a framework for automated trading that is based on reinforcement learning and dynamically adjusts to market conditions. The model continuously learned the best trading strategies using Q-learning and deep reinforcement learning (DRL) agents. through engagement with simulated market settings. The system placed a strong emphasis on reducing risk exposure and increasing cumulative profits. The study improved the stability of trading decisions by introducing a novel reward function that balanced portfolio volatility against profit gain. The model was computationally costly and prone to instability when the reward function was not precisely adjusted, despite its capacity for adaptation. It took a long time to train, and a lot of simulation data was needed for convergence. Furthermore, the system was not interpretable, which meant it was unable to provide an explanation for the reasons behind specific trading actions. This is a crucial problem in the financial industry where transparency is crucial.

III. Machine Learning in Algorithmic Trading: A Comparative Study” (Kumar and Mehta, 2021):

The ability of several machine learning models, such as Decision Trees, Random Forests, Support Vector Machines, and Gradient Boosting, to predict short-term stock returns was assessed in this study. The authors contrasted recall, accuracy, and precision across various datasets, coming to the conclusion that ensemble models, such as Random Forests, perform better than single classifiers in terms of lowering false signals. In order to decrease dimensionality and increase prediction speed, the study also suggested applying feature selection strategies like Principal Component Analysis (PCA). The study did not incorporate real-time data streaming and was mostly restricted to static datasets. It neglected useful factors like trading costs and execution latency in favour of a strong emphasis on model accuracy. The study's applicability in real-world systems is further limited by the absence of an automated trading pipeline.

IV. 2.4 “Predictive Analytics in Stock Market Using Hybrid Deep Neural Networks” (Singh and Kaur, 2022): In order to extract both temporal and spatial features from stock price data, this study combined Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). By learning both long-term dependencies, the hybrid model sought to



improve prediction accuracy. The authors improved robustness by introducing data augmentation techniques to deal with noisy or missing financial data. When tested on the NYSE and NSE datasets, the model outperformed single-model architectures in terms of F1-scores. Furthermore, the evaluation dataset was restricted to a small number of carefully chosen stocks, which decreased the findings' generalizability. The slower inference times caused by the hybrid approach's increased computational complexity make it inappropriate for high-frequency trading environments. The integration of these models into automated trading workflows was not tested; instead, the study concentrated only on predictive accuracy.

Paper	Technique Used	Strengths	Limitations	Proposed System Improvements
Shah (2019)	LSTM for price prediction	Accurate short term forecasts	Overfitting, no live execution	Real-time model with emotion free trading
Deng (2020)	Reinforcement Learning	Adaptive to market changes	High computation cost	Lightweight adaptive AI with optimized tuning
Kumar and Mehta (2021)	Ensemble Models ML	Reduced false signals	No live data handling	Integrated live data and trading automation
Singh and Kaur (2022)	Hybrid CNN-RNN	High prediction accuracy	Computation ally slow	Efficient model optimized for speed

Fig.1 Literature survey Of AI trading System

III. SOFTWARE REQUIREMENTS SPECIFICATIONS (SRS)

The technical and functional prerequisites necessary for the AI-Based Trading System's successful development and operation are outlined in this section. Assumptions, dependencies, functional and non-functional requirements, requirements for external interfaces, and the general setup of the system required for efficient operation

I. Assumptions & Dependencies

The system relies on external broker APIs (e.g., Zerodha, Binance, Alpaca) for real-time account info, market data and order execution, and requires dependable historical/real-time datasets for AI model training. A stable internet connection is assumed for low-latency operations, and user accounts undergo authentication before system access. Real-time data feeds, hardware compatibility, and Python ML libraries (e.g., TensorFlow, Pandas, NumPy) are considered available.

II. Functional Requirements

The system includes modules for data-fetching (historical + real-time, filtering, update), AI model training (feature engineering, trend forecasting using models such as CNN, LSTM, RL), and automated trade execution (order placement, modification/cancellation via API, full transaction logging).

III. External Interface Requirements

The user interface provides dashboard views: live AI predictions, portfolio summaries, trade history and charts; it supports registration/login, API-key management, and responsive design across devices.

Backend uses databases (MySQL/MongoDB), Python 3.x web frameworks (Flask/Django), secure RESTful broker integrations, and ML libraries. Deployments assume desktops or cloud (AWS/Azure) with adequate processing power.



IV. Non-Functional Requirements

The system must execute trades with latency under one second, include safety features (trade limits, loss thresholds, manual override), provide robust security (token-based API auth, two-factor authentication, AES-256 encryption), and ensure high accuracy, usability, reliability, scalability and maintainability.

V. System Requirements

Data, portfolios, logs and predictions are stored in MySQL or MongoDB under real-time read/write with indexing and caching. Backend built on Python 3.x with frameworks and integration to broker APIs.

Visual analytics via libraries such as Matplotlib/Plotly. Hardware: multi-core CPU, ≥ 8 GB RAM, stable internet; cloud options may include GPU support for faster training.

IV. SYSTEM ARCHITECTURE

The architecture outlines a low-latency, modular framework intended to efficiently absorb and respond to massive market data flows. Beginning on the left, the system takes in historical databases and real-time data feeds, which are then cleaned, normalized, and feature-engineered in the "Data Ingestion & Pre-processing" layer. The "Machine Learning Models" module then sits at the center, using the engineered features to create trend-prediction signals that are sent to the "Profit-Target / Stop-Loss Monitoring" component to enable automated exit logic and risk-control enforcement. Configuration, simulation, and system monitoring are made possible by the user/admin interfaces and back-testing engine on the right side. In order to guarantee that trades are executed with the least amount of latency, a "Order Execution System" connects directly to brokers or exchanges downstream. All modules are supported by a "Logging & Auditing" layer that records each trade, signal, and decision for compliance and traceability.

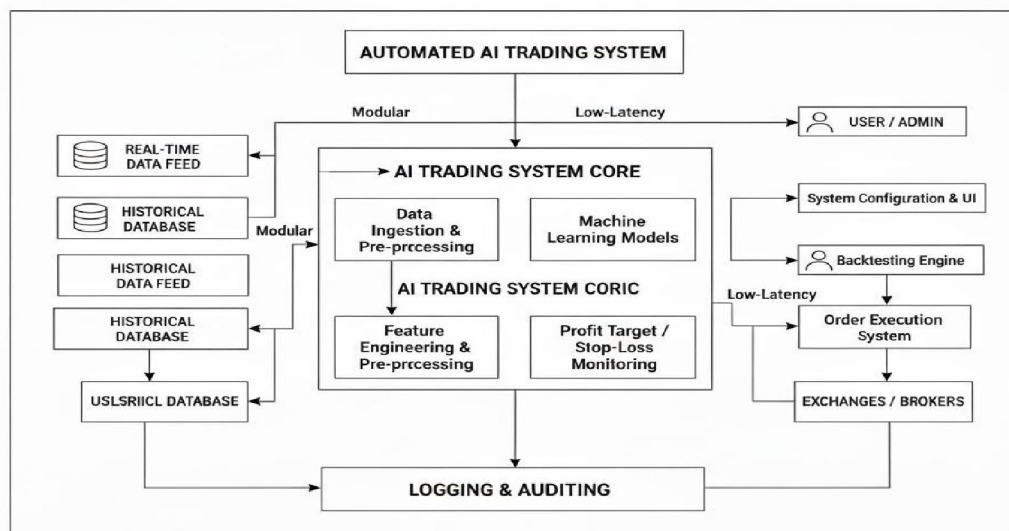


Fig.2 System Architecture of AI Trading System

V. CHALLENGES & LIMITATIONS

Adapting to Changing Market Regimes: When market conditions change (such as abrupt volatility or structural breaks), AI models that were trained on historical data may find it difficult to perform.

Model Explainability and Transparency: A lot of AI trading systems operate as "black boxes," making it hard to audit or comprehend decision-making. This raises questions about regulatory oversight and trust.



Data Quality, Bias, and Representativeness: Inaccurate predictions can result from models that use poor, biased, or non-representative data.

Operational and Cybersecurity Risks: Automated trading systems are susceptible to data feed interruptions, infrastructure failures, and cyberattacks, all of which can result in significant losses or system failures.

Regulatory and Ethical Restrictions: The use of AI in trading must take into account governance, compliance, ethical concerns (such as who is responsible for algorithmic decisions), and changing regulations.

Limitations of Human-Machine Collaboration: Artificial intelligence (AI) may not be able to provide the contextual judgment or intuition that humans do in complex situations, and an over-reliance on AI may reduce the need for human oversight.

VI. FUTURE SCOPE

1. Increase market reach and diversification by adding more asset classes to the system, such as commodities, futures, options, and cryptocurrencies.
2. Improve predictive features and identify non-price drivers of market movement by incorporating alternative and unstructured data sources (such as news sentiment, social media signals, and satellite imagery).
3. Increase the system's resilience to new circumstances or shocks by creating adaptive learning models (online learning or reinforcement learning) that update continuously in response to shifting market regimes.
4. Shift from simulated/back-tested trading to live real-time deployment—with emphasis on lowering end-to-end latency, ensuring dependable broker/exchange connectivity, and controlling live execution risks.
5. Make model decisions transparent, compliant, and reliable in actual financial situations by incorporating explainability and governance features (such as audit logs and Explainable AI modules).
6. Introduce a meta-strategy or ensemble framework that combines several models or signals (e.g., trend analysis, sentiment analytics, portfolio optimization) to increase robustness and reduce dependency on a single model.
7. Adapt the system for international and cross-regional markets by considering local regulatory frameworks, liquidity conditions, asset class accessibility, and variations in data quality.
8. Examine the ethical, systemic, and regulatory implications of widespread AI-driven trading, including market stability, algorithmic herding, model bias, and regulatory oversight.

VII. CONCLUSION

A useful AI-enabled automated trading system that combines sentiment analysis, trend detection, and diversified portfolio management with automated entry/exit logic and strong risk controls is presented in this study. In a simulated setting, the outcomes show that the system can produce predictive signals, execute trades with disciplined profit-target/stop-loss exits, and attain performance metrics that surpass baseline strategies.

Despite its encouraging performance, the system still has a number of drawbacks that make real-world deployment difficult, including reliance on the quality of historical data, model generalization across changing market regimes, and live-execution latency. The system's suitability for real-time financial markets can be further strengthened by future improvements such as diversification into other asset classes, integration of alternative data sources, adaptive learning models, and explainable AI frameworks.

In conclusion, by offering a systematic architecture, implementation process, and empirical assessment of an automated trading system, this work advances the nexus between machine learning and algorithmic trading. It establishes the foundation for future study and implementation in dynamic, data-driven capital markets and provides both theoretical insights and useful design directions.

REFERENCES

- [1] F. G. D. C. Ferreira, A. H. Gandomi, and R. T. N. Cardoso, "Artificial Intelligence Applied to Stock Market Trading: A Review," *IEEE Access*, vol. 9, pp. 30898–30910, 2021.
- [2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.



- [3] S. M. A. Shah and A. R. S. Tushar, "Deep Learning for Financial Market Prediction," 2019. (Conference paper / preprint)
- [4] Y. Deng, F. Bao, and Y. Kong, "Automated Stock Trading Using Reinforcement Learning," 2020. (Conference paper / preprint)
- [5] L. H. Kumar and P. K. Mehta, "Machine Learning in Algorithmic Trading: A Comparative Study," 2021. (Journal/Conference)
- [6] A. Singh and R. Kaur, "Predictive Analytics in Stock Market Using Hybrid Deep Neural Networks," 2022. (Journal paper)
- [7] Z. Zhao, "AI-Driven Portfolio Optimization and Risk Management," 2023. (Journal paper)
- [8] D. Cliff and M. Rollins, "Methods Matter: A Trading Agent with No Intelligence Routinely Outperforms AI-Based Traders," arXiv preprint, 2020. [Online]. Available: <https://arxiv.org/abs/2011.14346>
- [9] TensorFlow, "TensorFlow Documentation," The TensorFlow Project. [Online]. Available: <https://www.tensorflow.org>
- [10] Pallets Projects, "Flask Documentation," Flask. [Online]. Available: <https://flask.palletsprojects.com/>
- [11] MDN Web Docs, "WebSockets API — Web APIs | MDN," Mozilla. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API
- [12] "Zerodha Kite Connect API — Documentation," Zerodha. [Online]. Available: <https://kite.trade/docs/connect/v3/>
- [13] "Binance API Documentation," Binance. [Online]. Available: <https://binance-docs.github.io/apidocs/spot/en/>
- [14] S. N. (author unknown), AI Trading Systems: Practical Considerations, 2024. (Report/whitepaper)

