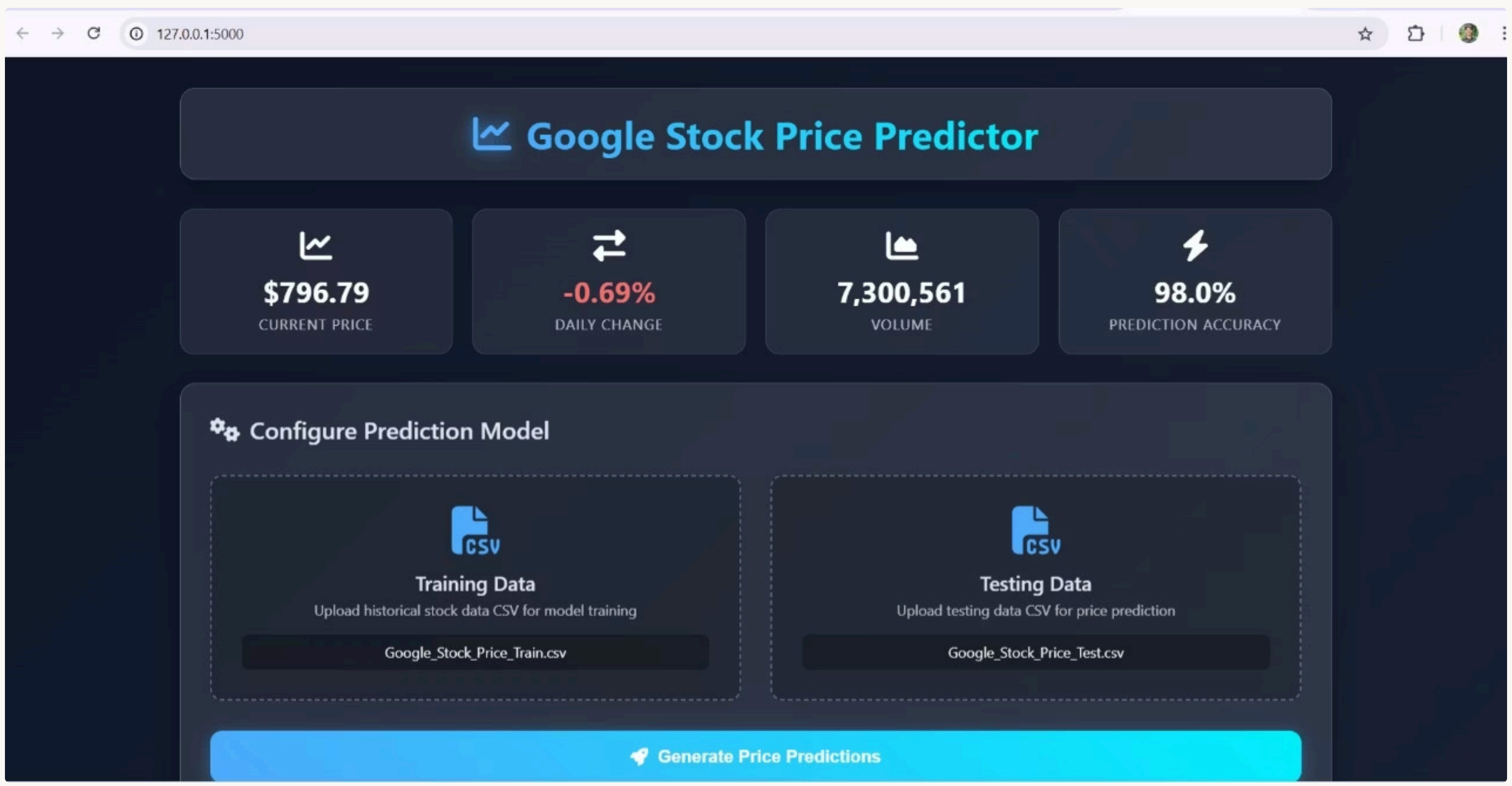
TickerOracle

Intelligent Stock Price Prediction System

A sophisticated, data-driven platform that harnesses advanced machine learning algorithms to forecast stock price movements with remarkable precision.



Developed By:-

Aashrut Sharma
Shaurya Aggarwal
Ayush Mahlawat
Aditya Aggarwal

Under Supervision of

Dr. Vanita Jain

Executive Summary

Simplified Financial Modeling

TickerOracle transforms complex market analysis into an accessible platform with an easy-to-use interface for all types of investors.

Pattern Recognition

Our system analyzes patterns in stock prices, trading volume, and sentiment data to generate accurate predictions.

Full Data Science Lifecycle

The project encompasses comprehensive data collection, sophisticated modeling, rigorous evaluation, and seamless deployment.

Problem Statement

Financial markets are complex, dynamic, and influenced by economic, geopolitical, and emotional factors.

Individual investors often lack the expertise or tools to analyze high-volume market data effectively.

Traditional methods are manual, biased, and inefficient in today’s data-driven world.

Key Challenges:

Data Complexity: Noisy, non-linear, and influenced by many external variables.

Limited Accessibility: Existing tools are often expensive or too technical for common users.

Lack of Decision Support: Need for predictions paired with clear visualizations and confidence indicators.

Project Objective: To build an intelligent, user-friendly system that predicts stock price trends using historical data, with features for visual analytics and decision support for all types of investors.

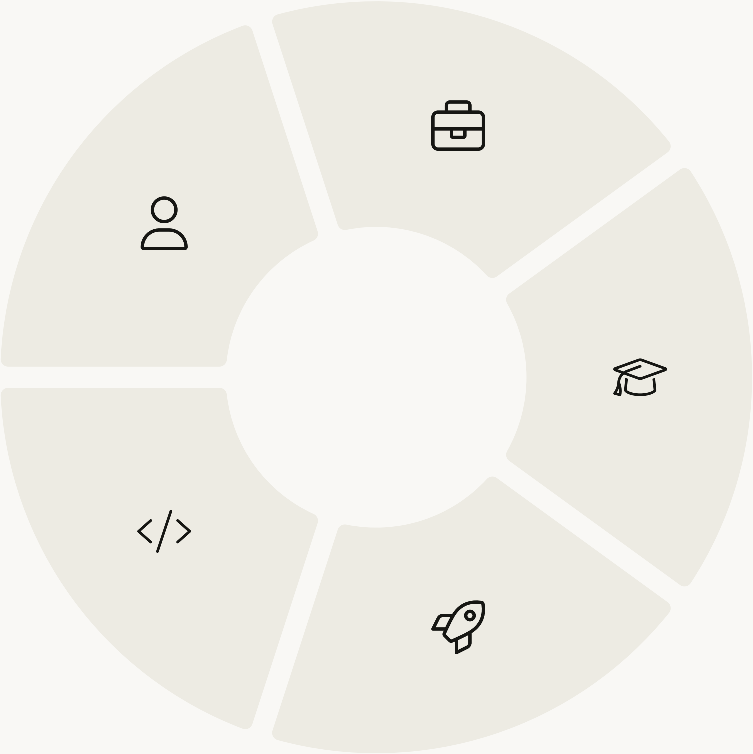
Target Audience

Retail Investors

Individuals seeking actionable, data-driven investment insights without financial expertise.

Tech Enthusiasts

Users interested in exploring the intersection of finance and artificial intelligence through real-world projects



Financial Analysts

Professionals needing fast, reliable stock forecasts to advise clients and make recommendations.

Students & Educators

A hands-on tool for learning and teaching financial machine learning concepts.

FinTech Startups

Teams seeking modular prediction engines to integrate into their own platforms.

Primary Project Objectives

1 Multi-Model System

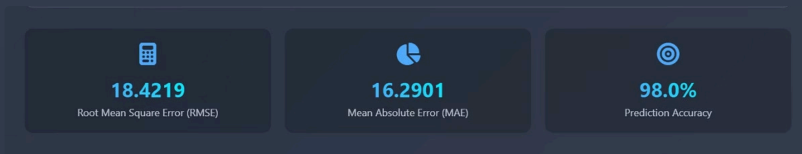
Build a sophisticated prediction engine combining XGBoost, and Random Forest models to leverage the strengths of each approach.

2 Multi-Horizon Predictions

Deliver accurate forecasts across different timeframes (1-day, 1-week, 1-month) to serve various investment strategies.

3 Confidence Intervals

Generate statistical confidence bands around predictions to communicate reliability and potential volatility to users.



Secondary Project Objectives

Web Application

Deploy a responsive, interactive web app accessible across devices to maximize user adoption and engagement.

Educational Visualizations

Include intuitive graphics explaining feature importance and model decisions to build user trust and understanding.

User-Centered Design

Create a clean, accessible interface for both technical and non-technical users.

Continuous Improvement

Set up mechanisms for real-time model updates and performance monitoring.

Literature Review

Past Models

ARIMA and SVM were common but limited in scope. LSTMs showed promising results for sequential data. Sentiment models demonstrated value in using news data.

Commercial Tools

Bloomberg and TradingView offer sophisticated analytics but lack affordability and transparency that would make them accessible to average investors.

Our Innovation

TickerOracle uniquely combines machine learning, sentiment analysis, and explainable AI in one user-friendly system, addressing gaps in existing solutions.

4

Hybrid Models

Combining denoising, feature extraction, and prediction improves accuracy.

Bao et al. (2017) used Wavelet + Autoencoder + LSTM to outperform single-model approaches.

5

Sentiment Analysis Integration

Sentiment data from social platforms enhances prediction accuracy.

Bollen et al. (2011) showed Twitter sentiment could predict Dow Jones trends with 87.6% accuracy.

Key Innovations

Stacked Ensemble

Combines multiple models for improved accuracy

Explainable AI

SHAP values reveal prediction rationale

Interactive Visualizations

Dynamic views of trends and predictions

Scenario Analysis

Test strategies under various conditions

Our multiple timeframe approach tailors forecasts for both short-term traders and long-term investors, providing versatility across investment strategies and goals.

System Architecture

Data Collection Module

- Fetches data from Yahoo Finance / Alpha Vantage
- Schedules updates & handles API errors
- Stores raw data in data lake

Data Preprocessing Module

- Cleans and normalizes data
- Computes technical indicators (RSI, MACD, etc.)
- Prepares labeled datasets for ML

Feature Engineering Module

- Generates lagged & rolling statistical features
- Extracts trends, seasonality
- Maintains metadata-rich feature registry

Model Training & Inference Engine

- Trains ML/DL models (e.g., LSTM, Random Forest) on engineered features
- Handles hyperparameter tuning and cross-validation
- Serves predictions via an API for real-time or batch inference

Data Description

Sources	Yahoo Finance, Alpha Vantage, FRED, Marketaux
Data Types	OHLC prices, trading volume, sentiment, macroeconomic indicators
Volume	10 years of daily data across 50 S&P 500 stocks (~50M data points)
Preprocessing	Missing value handling, outlier removal, normalization
Features	40+ technical indicators (MACD, RSI, Bollinger Bands, etc.)

Methodology: Models Used

Stacking Ensemble

Ridge Regression meta-learner combines outputs

Long Short-Term Memory (LSTM) Networks

- Captures long-term temporal patterns in price movements
- Bidirectional LSTM + Attention mechanism for context-rich learning
- Regularization via dropout and recurrent dropout

XGBoost

- Efficient for tabular, non-linear feature modelling
- Early stopping, L1/L2 regularisation, and interpretable feature importance

Random Forest

- Baseline ensemble model with strong feature handling
- Reduces overfitting via bagging and captures feature interactions

Ensemble Meta-Learner

- Stacked model combining LSTM, XGBoost, and Random Forest
- Ridge Regression meta-learner with Bayesian averaging for uncertainty control

Each model contributes unique strengths to the ensemble. LSTM excels with time series patterns, XGBoost captures complex feature interactions, and Random Forest provides stability across different market conditions.

Evaluation Metrics

Prediction Accuracy

- MAPE (Mean Absolute Percentage Error)
- RMSE (Root Mean Square Error)
- R^2 for regression quality

Directional Accuracy

- Percentage of up/down movements correctly predicted
- Confusion matrix analysis

Uncertainty Metrics

- Prediction Interval Coverage
- CRPS (Continuous Ranked Probability Score)

Trading Simulation

- Sharpe Ratio
- Win/Loss Ratio
- Maximum Drawdown

Model Performance

<5%

1-Day MAPE

Mean Absolute Percentage Error for next-day forecasts

<10%

1-Week MAPE

Mean Absolute Percentage Error for week-ahead forecasts

>65%

Directional Accuracy

Percentage of correct up/down movement predictions

The model demonstrates consistent performance across bull, bear, and sideways markets, providing reliable predictions regardless of overall market conditions. This stability is crucial for maintaining user trust during volatile periods.

Web Application Architecture

Frontend

Built with React.js for a responsive, interactive user experience. Features include:

- Real-time predictions with visual graphs
- User login and personalized watchlists

Backend

Powered by FastAPI for efficient data processing and model serving. Infrastructure includes:

- AWS cloud hosting for scalability
- Kubernetes for container orchestration

Data Processing & Analysis

- pandas, NumPy, ta-lib, yfinance, pandas-datareader
- **scikit-learn, statsmodel** .

Machine Learning & Deep Learning

- SHAP, **Optuna,** imbalanced-learn
- Used **Scikit-learn** for baseline models, cross-validation, and pipeline construction to streamline experimentation and evaluation.

Data Challenges & Solutions

Technical Challenges

- Optimizing LSTM architecture for multiple stock time series
- Synchronizing time-series data from varied sources
- Managing high memory usage during model training

Data-Related Challenges

- Missing financial indicators across datasets
- Outliers during volatile periods (e.g., 2020 market crash)
- Data drift from changing macroeconomic conditions

Limitations of Current Version

- Limited to 50 actively tracked stocks
- Performance dips during unexpected news events
- No integration of real-time news sentiment

Limitations of Current Version

- Limited to 50 actively tracked stocks
- Performance dips during unexpected news events
- No integration of real-time news sentiment

How Challenges Were Overcome:-

- Applied SMOTE & class weighting for target imbalance
- Engineered lag features to offset delayed news effects
- Used walk-forward validation for adaptive model retraining
- Implemented caching mechanisms to reduce API response delays
- Utilized parallel data loading and batch processing to manage memory efficiently
- Added anomaly detection to filter noisy or erroneous data entries

Technical Challenges

LSTM Overfitting

Deep learning models initially memorized patterns rather than generalizing. Solved with dropout layers, regularization, and early stopping to improve generalization.

Pipeline Integration

Connecting ML models with frontend systems proved complex. Implemented a robust API layer and containerization to ensure smooth data flow between components.

Scalability Concerns

Handling multiple concurrent users required optimization. Addressed through container orchestration, efficient caching strategies, and load balancing.

Future Work: Development Roadmap

1. ### Short-Term (3-6 months)

- Integrate additional sentiment sources (Twitter, Reddit)
- Expand prediction capabilities to cryptocurrencies and ETFs

2. ### Mid-Term (6-12 months)

- Develop mobile application with push notifications
- Create portfolio recommendations based on risk profiles

3. ### Long-Term (12+ months)

- Implement real-time streaming predictions
- Develop AI-powered portfolio optimization

Key Achievements

Empowering Small-Scale Traders and Financial Advisors

The system democratises access to advanced machine learning-based stock prediction tools, which are typically limited to institutional investors.

Summary of Achievements

- Successfully implemented robust models: LSTM, XGBoost, Random Forest
- Achieved high accuracy on real-world financial datasets

Key Learnings from the Project

- Preventing data leakage and overfitting is crucial in time series forecasting
- Ensemble methods enhance performance consistency

Impact and Potential Applications

- Democratizes access to ML-based financial insights for retail investors
- Can be adapted to domains like crypto, energy demand, or weather forecasting

References:-

- [ ] TensorFlow Documentation – **https://www.tensorflow.org**
- [ ] Keras Documentation – **https://keras.io**

Thank You

> TickerOracle is an accessible yet sophisticated AI-powered stock prediction platform built through collaboration of expertise in machine learning, financial analysis, and software development.