Stock Price Trend Prediction Model Based On Machine Learning

Shaowei Lin U17616063

Linsw@bu.edu

Introduction

With the advent of computational intelligence, finance has seen an increasing shift towards machine learning algorithms designed to predict the future trends of individual stocks or specific portfolios. Hybrid models that incorporate ARIMA with support vector machines, for instance, have shown promise. Machine learning methods have the distinct advantage of being able to identify nonlinear relationships between features without requiring any prior knowledge of the data. Artificial Neural Networks (ANNs), in particular, have demonstrated significant learning capabilities with large datasets.

In recent years, deep learning models have gained increasing attention for their superior learning abilities and prediction accuracy. The deep residual neural network, a specific type of convolutional neural network, has proven to be exceptionally effective. It allows for the addition of hundreds or even thousands of layers to the network without significantly increasing the training time, thus notably improving accuracy in fields like image recognition.

Stock Price Forecasting: Methodologies Explored

Stock price prediction remains a labyrinthine subject requiring multifaceted approaches for accurate forecasting. This paper zeroes in on three prevalent paradigms: Fundamental Analysis, Technical Analysis, and Time Series Forecasting Methods.

Fundamental Analysis

Macroeconomic Variables

Fundamental analysis forms its bedrock on macroeconomic parameters—global economic landscapes, national fiscal policies, and monetary stances—to name a few. These elements ostensibly dictate the long-term trajectory of stock prices.

Financial Health of the Company

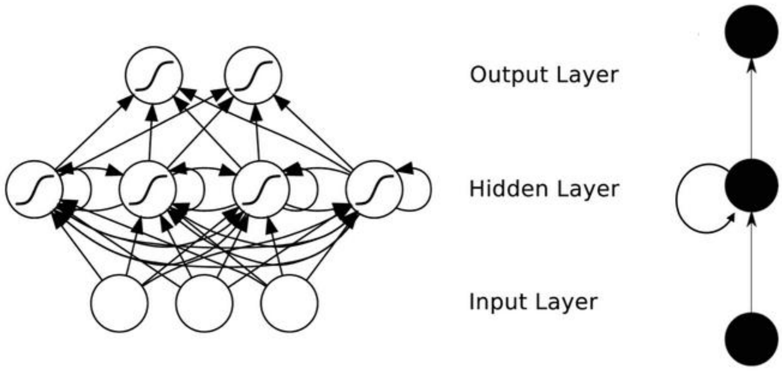
This method entails a rigorous examination of a firm's financial metrics, including market capitalization, P/E ratios, P/B ratios, and cash flows.

Industry-specific Trends

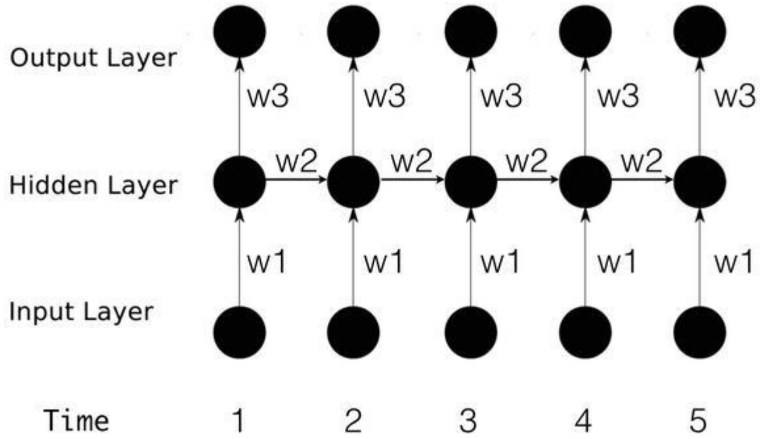
Additionally, it seeks to forecast how sectoral trends might influence the stock prices of companies operating within that industry.

RNN Neural Network Modeling Principles

Recurrent Neural Networks (RNN), have the ability to explicitly model time by adding self-connecting hidden layers that span points in time. In other words, the feedback from the hidden layer, in addition to going to the output, also goes to the hidden layer at the next time step, thus affecting the individual weights at the next time step.



RNNs differ from traditional multilayer perceptron machines in that they have an internal notion of temporal order, where the next time (understood as step) is affected by the present time, and unfolding the network in a temporal order allows a better description of the RNN neural network, where the network is unfolded through multiple time steps, materializing the connections in an acyclic form. Note that the weights (from input to hidden and hidden to output) are the same at each time step. Recursive networks are sometimes called deep networks, and their depth is reflected not only between inputs and outputs, but also across time steps, each of which can be thought of as a layer.



LSTM(Long Short-Term Memory)

Long Short-Term Memory (LSTM) neural networks are a special type of Recurrent Neural Networks (RNN) designed to capture long-term dependencies in data sequences. To achieve this goal, they employ a unique architecture that maintains a continuous flow of error information through special storage units. This avoids problems such as vanishing gradients and explosions, which are common in standard RNNs.

In the LSTM model, each cell contains a memory unit capable of retaining information for a long time. The flow of data in and out of this memory unit is regulated by three specialized gates: an input gate, an output gate, and an oblivion gate. Unlike traditional neurons, these gates do not output data to the network, but instead control the internal state of the memory cell by adjusting the weights.

The principle of operation is as follows:

Input gates determine what new information will be stored in the memory unit.

Output gates determine what information will be sent from the memory unit to the rest of the network.

The forgetting gate selectively deletes the contents of the memory cell based on the current needs of the sequence model.

Each gate functions by applying specific weights learned in the training phase to its inputs and outputs. In this way, the LSTM network can selectively remember or forget certain information, thus enhancing its ability to model complex long-term dependencies in the data.

The memory unit itself is a linear neuron with self-connectivity that helps maintain state across time steps. When the forgetting gate is activated, the memory unit maintains its existing state. Conversely, when the output of the forgetting gate is zero, the memory unit is reset, erasing its stored information.

By combining these elements, LSTM provides a powerful method for capturing patterns and dependencies in continuous data, making it useful in a variety of time series and natural language processing tasks.

Transformer Model

In finance, stock price prediction has been a challenging task because it involves a large amount of nonlinear, high-dimensional and time-dependent data. In recent years, Transformer models have gained attention due to their outstanding performance in natural language processing, and have also been gradually applied to stock prediction. Relative to traditional RNN and LSTM models, Transformer's self-attention mechanism allows it to capture long-range dependencies in time-series data, which is especially critical for understanding and predicting the complex dynamics of stock markets. In addition, Transformer's parallelized architecture allows it to efficiently process large amounts of financial data, which is especially beneficial on modern high-speed computing devices. By combining multiple financial indicators, news reports, and other relevant data sources, Transformer can provide investors and analysts with accurate and timely stock price forecasts that support more informed investment decisions.

Stock Picking Strategy

We use machine learning models to predict the probability of each stock going up or down daily. Based on these predictions, we rank stocks by their likelihood of going up and select the ten best-performing high-probability stocks to form our daily portfolio. Each stock in the portfolio is allocated an equal share of investment capital.

Market Timing Strategy

Since the performance of individual stocks is often closely related to the general market trend, our methodology also includes market timing. If the market is depressed for an extended period of time, it is unlikely that even top-performing stocks will continue to grow; conversely, when the market is strong, it is unlikely that even weaker stocks will fall sharply. To determine market conditions, we use the Transformer model to analyze the recent movement of the NASDAQ index. If the index closes above its five-day moving average, we view this as a bullish market signal, prompting us to open new positions to increase investment opportunities. Conversely, if the closing price falls below this average, we consider this as a bearish signal and sell the portfolio or maintain a short position to reduce risk.

Integrated Strategy

Our integrated strategy incorporates both stock picking and market timing methods:

First, using a machine learning model, stocks are identified and ranked according to their daily probability of going up, and the top ten are selected for inclusion in the portfolio.

Then, using the LSTM model, we assess broader market conditions through the NASDAQ index. We open or close positions based on how well the index agrees with its five-day moving average.

Finally, by integrating these two elements, we aim to more precisely target investment opportunities while controlling downside risk.

The suggestion from Chatgpt

1. Data Collection:

Gather daily stock prices and trading data for various companies.

Collect daily data for the NASDAQ index.

Obtain other relevant data associated with the stock market, if possible, such as news reports, macroeconomic data, or social media sentiment.

2. Data Preprocessing:

Clean and normalize the data to remove anomalies or errors.

Enhance the raw data with technical indicators, such as the five-day moving average.

3. Individual Stock Prediction:

Analyze the data for each stock using machine learning models, such as the Transformer or other models.

Rank the stocks based on their daily probability of increasing in value.

Select the top ten stocks for inclusion in the portfolio.

4. Market Timing Strategy:

Analyze the NASDAQ index data using the LSTM model to capture long-term market trends.

Consider the market bullish if the index closes above its five-day moving average.

Consider the market bearish if the index closes below its five-day moving average.

5. Decision Making:

In a bullish market, open new positions or increase investments, especially in the top ten ranked stocks.

In a bearish market, reduce positions, sell off the portfolio, or maintain a short position to minimize risk.

6. Strategy Integration:

Identify investment opportunities more precisely by combining predictions for individual stocks with the market timing strategy.

Periodically evaluate the effectiveness of the strategy and adjust based on market changes.

Reference

[1]. Liu, H. (n.d.). Stock Price Trend Prediction Model Based on Deep Residual Network and Stock Price Graph. 2018 11th International Symposium on Computational Intelligence and Design.

[2]. Mittermayer, M-A. "Forecasting intraday stock price trends with text mining techniques." 37th Annual Hawaii International Conference on System Sciences, 2004. Proceedings of the. IEEE, 2004

[3]. Salisu, Afees A., and Xuan Vinh Vo. "Predicting stock returns in the presence of COVID-19 pandemic: The role of health news." International Review of Financial Analysis 71 (2020): 101546.

[4]. G. Peter Zhang. Time series forecasting using a hybrid ARIMA and neural network model[J]. Neurocomputing.2003(50). 159-175

[5]. Hammda, M.alhajali. Forecasting the jordanian stock prices using artificial neural network. Intelligent Engineering Systems through Artificial Neural Networks .2012(17). 273-275