Intersect, Vol 17, No 1 (2023)

Artificial Intelligence in the Stock Market: Quantitative Technical Analysis, Model Weight Optimization, and Financial Sentiment Evaluation to Predict Stock Prices

Emin Cilingiroglu
Canyon Crest Academy

Abstract

This research paper demonstrates the implementation of artificial intelligence in predicting the stock market. In doing so, it focuses on stock price prediction through A.I. models and machine learning algorithms to maximize profit potential, improve investments, and eliminate risk. Potentially lucrative company stocks and shares have attracted investors as well as general interest in the stock market for decades (Malky, 1973, p. 269), leading more people to try to forecast the rise or fall of market prices. However, industry volatility and the seemingly unpredictable nature of the stock market have led many buyers to invest impulsively or make poor purchasing decisions like selling or buying shares at the wrong times. The paper outlines the training and testing of linear regression and neural network models on collected, classified data to generate accurate predictions. The program also utilizes natural language processing (NLP) through a deep learning model with transformer-produced sentence embeddings, allowing the algorithm to consider relevant socioeconomic and sociopolitical news to produce predicted prices at an even higher level of accuracy. These models achieved average prediction errors of 0.12% for Amazon's stock prices, 0.13% for Google's stock prices, and 0.07% for Microsoft's stock prices on the testing data sets, over a two-week testing period. This paper ultimately evaluates existing prediction methods and builds on robust machine learning systems to offer more efficient estimation models.

Background

Although it is more than 400 years old, the recent explosions of cryptocurrency, NFTs, and other forms of digital assets have sparked a sudden interest in the stock market. The stock exchange's average daily volume, which has more than doubled since 2019 to total $38.3 million in contracts (*Trading & Data*, 2023), attests to this renewed interest. The market also saw a boom in daily trading volumes with buyers looking to take advantage of low interest rates and the availability of commissionfree transactions ("Board of Governors of the Federal Reserve System," 2020). Moreover, the economic uncertainty of the COVID-19 pandemic led to a surge in retail investor participation in the stock market as people looked for alternative investment opportunities (Lush *et al.*, 2021).

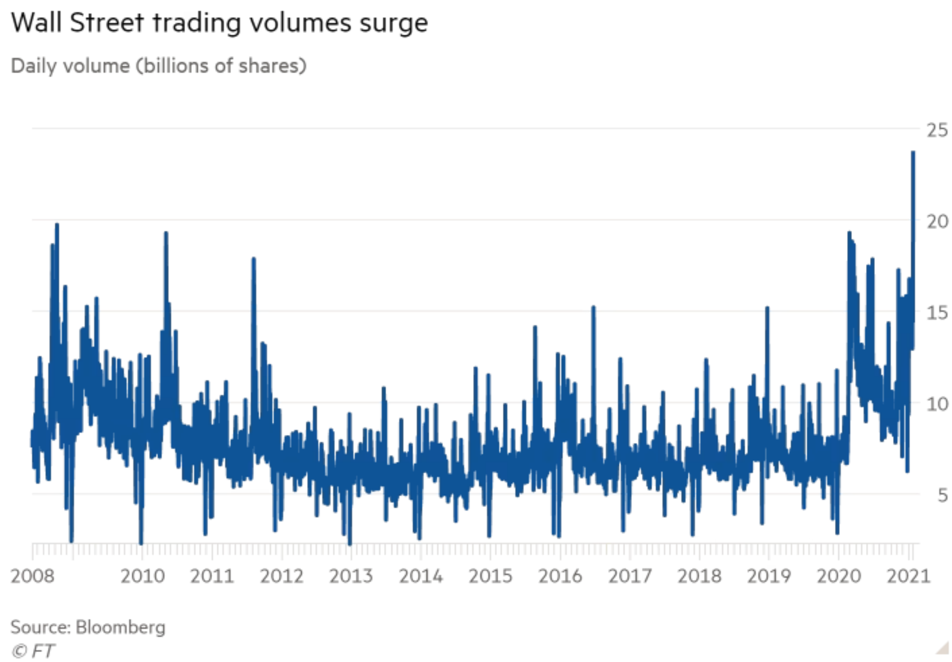

FIGURE 1: Daily Trading Volume on Wall Street Market (Kantor, Lewis, & Stafford, 2021).

Nevertheless, the driving force behind any interest in the stock market is the potential to make huge profits relatively quickly. Subsequently, investors have constantly been trying to develop or modify different methods for predicting the stock prices of public companies, especially as the influx of new investors could signal increased liquidity in the market. Most buyers have relied on traditional approaches such as calculating the moving average with technical analysis (Chadda & Yadav, 2023), fundamental analysis (Elleuch & Lofti, 2009), or simple heuristics.

Ever since its formulation by Charles Dow as the Dow Theory in the 1800s, technical analysis has detailed a trading framework that prioritizes trading activity and the volume movements of stocks over time, enabling investors to recognize market trends, cycles, and support and resistance levels (Hayes, 2022). Furthermore, technical analysis assumes that fluctuations in the prices of financial instruments, such as securities, are important indicators of short-term trading signals when evaluated with appropriate investing benchmarks as well as the interplay of supply and demand. Drawing on the developed inferences of Efficient Markets Hypothesis (EMH), technical analysis is also driven by the notion that any relevant information about a security, whether fundamental data, market psychology, or technical indicators, is already reflected in its price. The backbone of technical analysis applies a “systematic, graphical approach to identify patterns of historical trading prices, and then formulate predictions that may generate abnormally strong returns...graphs are the primary instruments of TA” (de Souza *et al.*, 2018). Thus, the use of graphs and statistical data supplements the detection of trends, cash flows, and events affecting business strategy by revealing benchmarks and moving averages. By calculating the moving average of a stock’s price, investors create a constantly updated average price that also mitigates random, short-term fluctuations for that stock (Fernando, 2023). However, due to the reliance on past price data or the potential to produce false signals in volatile markets, moving averages can produce inaccurate trading evaluations and incorrect market trends. A study conducted by Arjun Chaddha and Shilpa Yadav (2023) on the profitability and reliability of moving average combinations over a 10-year period concluded that for long-term, short-term uptrends, and short-term sideways trends, there was “no moving average whose percentage return was significantly higher than market value. Thus, investors should never use moving averages in the long-term, a short-term uptrend, or a short-term sideways trend.” Consequently, although technical analysis and moving averages have been used by investors for years, their vulnerabilities to lagging indicators and market instability validate the use of more efficient, reliable stock prediction methods (Zhang, 2020).

Conversely, investors use fundamental analysis to evaluate a company’s fiscal position along with economic and market trends down to an intrinsic value. Hence, fundamental analysis revolves around any macroeconomic and microeconomic condition that may affect the intrinsic value (Segal, 2023). Unlike technical analysis, which depends on historical prices and assumes an all-representative stock price, fundamental analysis involves assessing annual reports, cash flow statements, or any other current data and metrics that examine the company’s financial health. The emphasis placed on a company’s underlying value through financial analysis is aimed at highlighting that company’s future growth prospects. As explained by Jaouida Elleuch and Trabelsi Lotfi (2009), “screens based on historical financial signals can shift the distribution of returns earned by an investor by separating eventual winners stocks from losers...results show that fundamental signals have a positive and significant correlation with future earnings performance.” The main benefit of fundamental analysis stems from traders developing a comprehensive evaluation of a company’s fiscal health, allowing investors to make more informed investment decisions. Nonetheless, fundamental analysis is limited by its reliance on often subjective assumptions and neglection of market sentiment, as “your judgements are subjective, as is your definition of a fair value. You may need to use different criteria to evaluate different groups, which will be time consuming" (Petrusheva & Jordanoski, 2016). Therefore, the weaknesses of fundamental analysis in maintaining objective estimations require the use of other prediction methods as well, like technical analysis, to accurately predict stock prices.

Existing stock prediction methods fail to provide a complete understanding of market trends, further underscoring the need for more effective, adaptive forecasting techniques (Zhang, 2020). This research project applies a newer, more sophisticated and efficient prediction method known as quantitative technical analysis, which entails a mathematical approach to producing accurate predictions for stock prices by measuring signals through statistical modeling. This idea will be discussed in a subsequent section. Naturally, stock market investors would be interested in a program working to predict stock prices, determining the strategic value of any given company or investment. Such an algorithm would also attract the attention of buyers interested in both short-term and long-term gains, whether short selling company stock or making capital investments. An optimized stock prediction model would also affect the financial policies of publicly traded companies with regards to their market capitalization and value (Zhang, 2020).

The increasingly elusive economic environment, coupled with unnerving studies showing that around 80% of investors lose money in the stock market as they have no knowledge of how the market works (Gillham, 2023), stress not only the dire situation of the market, but also the lack of financial responsibility amongst most investors. Usually clueless about relevant monetary and industry information, most buyers make poor investment decisions and maintain erroneous perceptions of disciplined approaches to purchasing on the market.

Fortunately, this project has developed a program with the use of A.I. models to assist buyers in making better investments through accurate, calculated predictions. These models solve the issues of financial responsibility by guiding investors and minimizing risk, a necessity in navigating the volatile stock market. The paper will thereby explore the use of technology and artificial intelligence to accurately predict the stock price of publicly traded companies on a given day, providing an in-depth explanation of the programming behind each of the models.

Data Sets and Libraries

The program consists of numerical databases, including prediction history, stock prices, and measured error data. The A.I. models are trained on this data to recognize market patterns with price fluctuations in comparing their predicted and actual outcomes. These databases access a company's stock prices at the open and close of the market, trading volume, dividends, and stock splits over a five-year period. Since the program imports the pandas, NumPy, and yfinance software libraries, data is then selected with the pandas package and initialized in an empty array through the NumPy package. Additionally, the algorithm holds an up-to-date library that allows users to input a predefined set of dates on which to run the models. To generate predictions, the models must evaluate and modify multiple different categories of the selected data. First, the user sets the amount of time that the models will take the stock price data from, whether the last five days or the last five years. The models then import this data from Yahoo Finance, through the yfinance library, to analyze the stock prices at the market's open and close for each day within the given time frame. The models then split the given values into random sets of training, validation, and testing data.

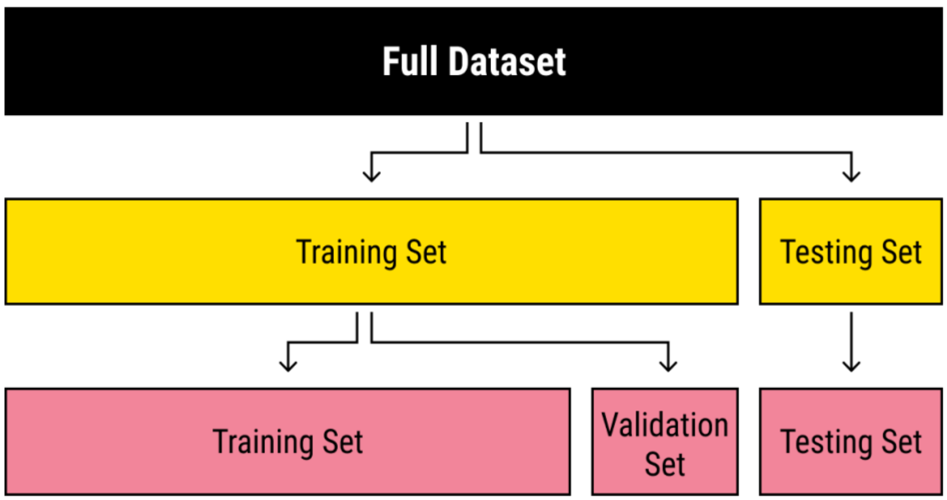

FIGURE 2. A Model's Division of Data into Training and Testing Data (Sydorenko, 2021).

Quantitative Technical Analysis

After accessing data, the models implement qualitative technical analysis to conduct an in-depth examination of past trends and cycles as well as develop accurate price predictions from analytical systems of future stock movements. While quantitative technical analysis may seem similar to technical analysis, the two are inherently different both in procedure and application. Instead of relying on price and volume data, quantitative technical analysis depends mostly on numerical and statistical records (Lo, Mamaysky, & Wang, 2000). The quantitative framework also applies mathematical methods and probability to evaluate the financial market. In contrast to technical analysis, this approach utilizes modeling to incorporate behavioral finance concepts in calculations and process both discrete and continuous data simultaneously (Zhang, 2020). As a result, the procedure "has a much stronger fault tolerance for outliers" (Zhang 2020). The advantages of quantitative analysis are apparent in providing investors with a robust process for stock forecasting within any timeline. This is due to the controls that analysts have over the analytic procedure, meaning “the algorithm of quantitative analysis can be tweaked to generate recommendations of any timeframe” (Chanda, 2021). However, a major limitation of the quantitative system stems from its market efficiency assumption, as markets may be influenced by irrational factors. Hence, disparities in short-term forecasting can skew prediction accuracy.

Nevertheless, the strictly analytical approach of quantitative technical analysis to predicting stock prices also allows for an algorithm that is compatible with modern machine learning and A.I. protocols (Zhang, 2020), which is why this project prioritized the linear regression and Multilayer Perceptron (MLP) neural network. In essence, the impact of quantitative analysis is best explained by Andrew Lo, Harry Mamaysky, and Jiang Wang (2000), as in “the wake of recent breakthroughs in financial engineering, computer technology, and numerical algorithms, it is no wonder that quantitative finance has overtaken technical analysis in popularity—the principles of portfolio optimization are far easier to program into a computer.” The implementation of quantitative analysis techniques in this program first outlined the use of a large set of financial data and indicators like trend lines, which was obtained through the yfinance library. The program would then need to divide this data into training and testing data sets, which is already a necessary step of A.I. programming. The algorithm also requires a target variable, which will be the future stock price at the open of the market for this paper. Finally, model performances on the testing data set would be evaluated through metrics like the Mean Squared Error (MSE) and the average prediction error.

In the case of the linear regression, the NumPy matrix of stock prices at the market open would first be split into data for the model to be trained and tested on. Fitting the model to the training data, the regression’s target, or dependent, variable is the stock price at the next day’s market open. Over a one-week testing period, the regression achieved an average MSE of 2.31% for the stock prices of Google and 8.24% for Amazon, for example. The relatively high MSE values were attributed to the model only considering the last three days of stock prices at the market open, which was solved by steadily increasing throughout the testing period to expand the amount of input data. The model’s performance was also likely the result of the program struggling to find a correlation between past stock prices and the prediction column of the matrix when stock prices are affected by several different factors. The linear regression presumably experienced overfitting as well, focusing excessively on the training data itself instead of the underlying patterns to accurately generalize its results to the unseen testing data. Fortunately, this issue can be addressed through techniques like regularization and cross-validation, which minimizes overfitting by reducing the complexity of the model (Crowley & Ghojogh, 2019).

Company Name	Linear Regression Average MSE
Google	2.31%
Amazon	8.24%
Tesla	39.77%
Apple	1.48%
Microsoft	14.55%
NVIDIA	13.92%
Walmart	3.16%
eBay	2.23%

TABLE 1: The linear regression model's average MSE performance on several publicly traded companies over the one-week testing period.

Programming the MLP neural network to apply quantitative technical analysis required practically the same procedure as the linear regression. Once again, the matrix of stock prices at the market open will be split for training and testing while the data undergoes preprocessing. The main difference between the two model processes is that the neural network's architecture can be adjusted by the user, from the numbers of hidden layers and nodes to the appropriate activation functions. This project maintained the default network architecture to preserve the accuracy of the model while altering the matrix features to maintain accuracy. The default hyper-parameters also produce a fair balance between variance and bias within the model, as the neural network is neither extremely simple nor complex. Likewise, the default activation function of the MLP neural network, the rectified linear unit (ReLU), works to decrease the likelihood of the vanishing gradient problem, an issue that can affect other activation functions like the sigmoid or hyperbolic tangent (Nabeel, 2023).

The statistical modeling applied through quantitative technical analysis also facilitates the generation of multiple predictions for any one company and comparisons of opportunity costs by contrasting investment opportunities, improving model precision. Secondly, this procedure registers technical indicators like relative strength index (RSI) and Bollinger Bands as input features for the model. The neural network's feature detectors assist the model in deducing price momentum and trends as well as assessing crucial investment risks in unstable markets. As with the linear regression, the neural network's performance on the testing data was evaluated through the MSE, after the model weighed each prediction to improve accuracy. The neural network experienced an average MSE of 3.18% for the stock prices of Google and 7.62% for Amazon. While the MSE acts to distinguish outlying predictions from the training data set, even a single erroneous prediction will skew the overall error of the model (Seif, 2019), which is why the MSE is not the only metric of model performance in this paper.

Company Name	Neural Network Average MSE
Google	3.18%
Amazon	7.62%
Tesla	42.05%
Apple	2.40%
Microsoft	12.46%
NVIDIA	12.03%
Walmart	2.78%
eBay	1.08%

TABLE 2: The neural network's average MSE performance on several publicly traded companies over the one-week testing period.

Model Weight Optimization

Once the models have quantified data about a company's stock prices at the market open, they adapt the weights, or learnable parameters, of the training data to compare weight accuracies to the random values of the testing set of data at the end of the process. Both models subsequently experience their first learning curves with the data, optimizing parameters to improve accuracy. These learning curves are important to understanding the relationship between model performance and the amount of training data it has evaluated, identifying instances of overfitting or underfitting. In the case of underfitting, a model has been trained on limited data, which in turn indicates that the model has not been able to grasp patterns or trends in the data set (Barla, 2023). The model therefore cannot accurately predict target variables in the testing data set and will struggle to iteratively improve its performance. Overfitting, meanwhile, describes excessive training on the data, where the model is simply memorizing the training set instead of learning generalizable patterns (Barla, 2023). As a result, the model's accuracy when training will be significantly higher than its accuracy on the new testing data. While learning curves emphasize model performance and precision, they also give insights into model bias through the bias-variance tradeoff. The bias-variance tradeoff accounts for the learning curves of models with either a high bias or variance ("Bias-Variance Tradeoff," 2020). This project aims to balance bias and variance well as reduce overfitting and underfitting by developing models that continuously modify their own nodes. In doing so, the models apply the Multiplicative Weights Method, an algorithmic technique that assigns initial weights to each model base and progressively adjusts these weights according to each model's respective performance. The multiplicative structure essentially solves the problem of prediction by optimizing model bases, allowing the program to iteratively decide which successful models should be combined.

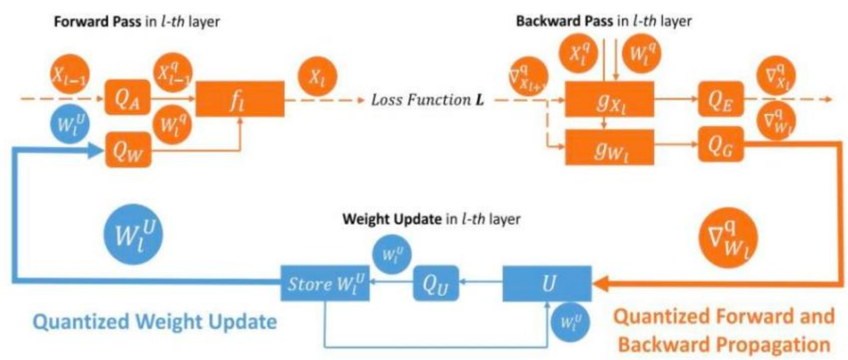

FIGURE 3: The Multiplicative Weights Method in model training with a logarithmic number system (Anandkumar *et al.*, 2022).

The Multiplicative Weights Method ensures a balance between different decisions and utilization of the most favorable choice based on all available information. Since each model is taking raw stock data as input, this approach streamlines the process of weighing each price by rewarding high-performing model bases as well as decreasing the weights of inaccurate bases. As stated by Sanjeev Arora, Elad Hazan, and Satyen Kale (2012), to appreciate the multiplicative weights method, one must "consider the naïve strategy that, in each iteration, simply picks an expert at random...Suppose now that a few experts clearly outperform their competitors. This is easy to spot as events unfold, and so it is sensible to reward them by increasing their probability of being picked in the next round."

In terms of the algorithm, programming this project's models to apply the Multiplicative Weights Method is similar to implementing quantitative technical analysis. One of the main benefits of this strategy is that it is applicable to many different machine learning models, assuming that said programs have some notion of a loss function. During training, the multiplicative weights method simply oversees the model adjusting its weights to minimize the loss function (Anandkumar *et al*, 2022). Now, the algorithm has established weights within both models, which allow the program to test each model on all the days outlined in the given time frame. As they did before, the models adapt based on their accuracies and inaccuracies. The models will then evaluate the other's precision. After an iteration, these weights will be used to combine the predictions of each model into one final prediction, also giving the percentage error for the two models. At this point, the algorithm has analyzed stock price history as well as its own prediction history, compared its predictions with actual outcomes, identified market patterns, and weighed its models on performance to output a predicted stock price.

Linear regression, one of the two models generating a prediction, finds a direct correlation based on patterns in the stock price's history. The linear model, although utilizing fewer parameters than the neural network, initializes the exact significance of each daily stock price, which then becomes the weight of that day (Chandola, 2022). These weights are in turn determined by frequencies and irregularities in the stock price history, meaning an outlying stock price for a certain day is weighed less because it is considered an abnormality. The model also uses a loss function to measure the difference between the predicted and actual prices of the training data, which will be the MSE function for this paper. Moreover, the model employs optimization algorithms like gradient descent during training to adjust the coefficients of weights. Gradient descent details updating the model weights by shifting them towards the negative gradient of the MSE to increase the accuracy of each prediction (Gupta, 2023).

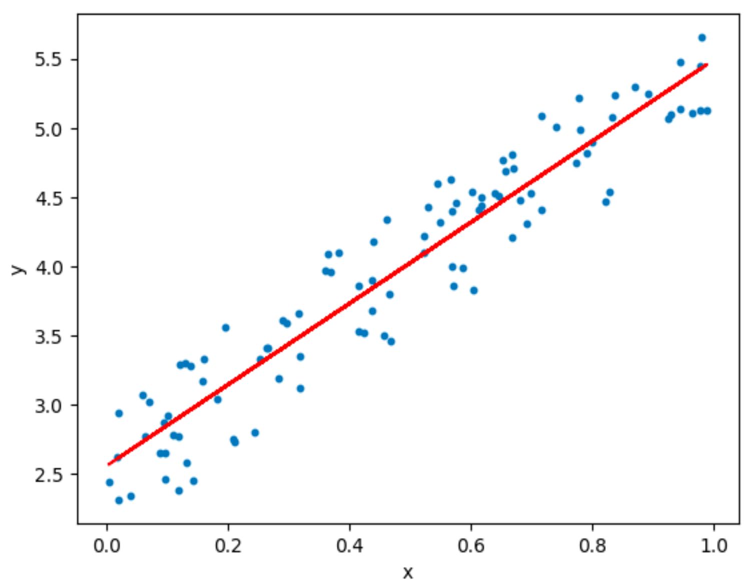

FIGURE 4: A demonstration of a regression model evaluating a data set by calculating a correlation to predict an outcome through Python (Agarwal, 2018)

The neural network is another model used in the program to create predictions with a weight system. However, neural networks differ from linear regression in that they have more parameters and are also able to find nonlinear patterns, which means that they can perform nonlinear fitting (Kumar, 2005). Therefore, the neural network can detect frequencies in both accurate and inaccurate predictions in iterating through the data set to measure the algorithm's accuracy on a given day. The neural network then analyzes patterns between days where the model was relatively inaccurate and future dates, adapting its predictions to account for random fluctuations in the stock price. The model is also able to recognize seasonal patterns and weekly changes in stock price history to ultimately learn and generate an increasingly accurate prediction. The default activation function, the rectified linear unit (ReLU), also works to decrease the likelihood of the vanishing gradient.

The neural network adjusts its weights during training, through a process known as optimization, of which back-propagation is the most common algorithm (Mujtaba & Sowgath, 2022). The back-propagation program involves adapting the weights of a neural network by converging the error down from the output layer to the input layer with a loss function and the chain rule of calculus. In doing so, the algorithm computes the gradient of each individual weight to "produce the local minima report" (Mujtaba & Sowgath, 2022).

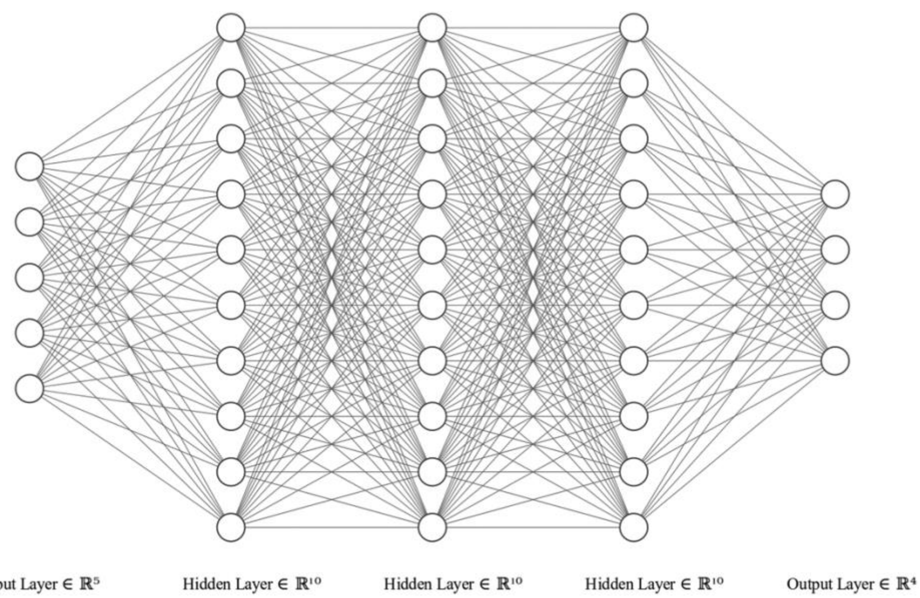

FIGURE 5: Diagram of a neural network with weighted, hidden layers and complex parameters (Pang, Nijkamp, & Wu, 2022)

In addition to the Multiplicative Weights Method, the project utilizes the boosting model to improve the accuracies of the linear regression and neural network. The boosting model is an algorithm that operates as a collaborative system to assess model performance and optimize the predictions of the two models. This algorithm works to reduce model errors in predictive data analysis by increasing the weights of samples with more significant errors (Saini, 2021). Weights are also assigned based on model performance, meaning that the models with the best predictions will have greater influence over the final output. Additionally, the linear model and neural network can adapt to each other's analyses and predictions so that the neural network can access the specialized weighting system of the linear regression while the linear model can evaluate and apply the neural network's results ("Boosting in Machine Learning," 2023). Ultimately, the boosting model drives the optimization of generated predictions by refining model performance and accuracy.

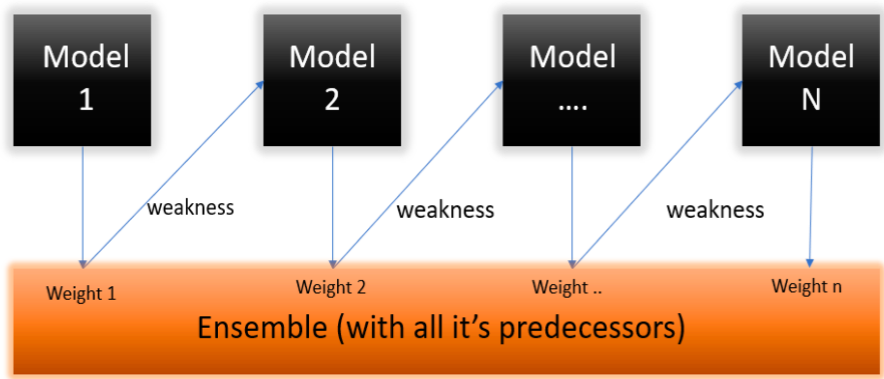

FIGURE 6: Diagram of a boosting model and an adaptive ensemble technique (Saini, 2021)

Finally, the program will implement a game theory framework to perfect the weights of each model. Game theory is a subfield of mathematics that analyzes how models apply reinforcement learning algorithms to solve the issues of conflict and cooperation (Silva, 2019). This method details assessing the impacts of each model's performance on the other and working with the boosting algorithm to merge each model, generating the most accurate prediction possible ("Game Theory," 2022).

While traditional machine learning models have struggled with evaluating dynamic scenarios and creating accurate long-term stock price predictions, the neural network and linear model act interdependently to ensure reliability and adapt to current market trends (Gülen, 2023). The flexibility of the neural network as well as the simple weights of the linear regression work in unison to create an optimized prediction. Similarly, the boosting model and game theory improve model performances by reducing bias as well as reaching maximum output in the strategic setting. In particular, the boosting method mitigates bias and variance within the machine learning ensemble while the game theory base allows the linear model and neural network to evaluate the accuracies of different predictions. Game theory can also analyze the potential behaviors of other market factors, such as buy-sell trends, to boost the decision-making process of each model.

Financial Sentiment Evaluation

The increasing influence of social media on the stock market and financial policies, as seen with the GameStop short squeeze in 2021, signifies the societal shifts that have given rise to, for example, cryptocurrency stocks (Bishop, 2021). The ever-growing impact of significant socioeconomic and sociopolitical events on stock prices, which has also been fueled by the rise of online social networking platforms, further highlights the need to consider market sentiment and relevant news when making price predictions (Gillham, 2023). Thus, the program also analyzes financial sentiment by associating positive and negative values to labeled sentences through a transformer model that utilizes the fundamental concepts of natural language processing (NLP). Natural language processing is a field of artificial intelligence studying the interaction between computers and humans in terms of language and speech. This specialization mainly aims to bridge the gap between words and numbers, as humans "can extract the meaning out of text effortlessly, but this is not the case for a computer" (Gilloz *et al.*, 2020). Natural language processing combines techniques from computer science, machine learning, and linguistics to understand human language by embedding text as well as producing representative relation between the words in a sentence. The first process of NLP is tokenization, which involves breaking down and simplifying text into individual words or clauses, known as tokens. According to Gilloz and others (2020), the most used data sets with transformer models contain approximately 4 billion tokens. These tokens are essential for identifying relevant features and characteristics of the text. In the case of financial sentiment analysis for this paper, a data file with almost 6000 sentences of market news and social media posts will serve as the evaluated sentiment. Another important procedure within the larger task of natural language processing, and this project specifically, is sentiment analysis. Sentiment analysis outlines the classification of opinion expressed in any given text through machine learning algorithms and detects positive or negative connotations within that text. For the purposes of this paper, the available financial sentiment data will be divided into tokens and converted to a vector of real numbers, containing word semantics and similarities.

To begin, the program employed the pandas package and imported a transformer framework to map text for embedding. This transformer structure, from the SentenceTransformers package, enables the vectorization of text associated with positive, negative, or neutral labels. The use of vectors solves two problems of sentiment analysis: preserving the order of the text and evaluating idiomatic expressions. As the program will transform every single word into a vector to then be processed by the neural network, the neural network will also take the corresponding vector of the previous word as input. Now, the neural network can essentially process a vector composed of each individual word and any previous words in the sequence, maintaining sentence order and structure. This method will also solve the issue of expressions and idioms that might confuse an NLP model, as the context of each word will be preserved in the corresponding vector.

Sentence	Sentiment
The GeoSolutions technology will leverage Benefon 's GPS solutions by providing Location Based Search Technology , a Communities Platform , location relevant multimedia content and a new and powerful co	positive
$ESI on lows, down $1.50 to $2.50 BK a real possibility	negative
For the last quarter of 2010 , Componenta 's net sales doubled to EUR131m from EUR76m for the same period a year earlier , while it moved to a zero pre-tax profit from a pre-tax loss of EUR7m .	positive
According to the Finnish-Russian Chamber of Commerce , all the major construction companies of Finland are operating in Russia .	neutral
The Swedish buyout firm has sold its remaining 22.4 percent stake , almost eighteen months after taking the company public in Finland .	neutral
$SPY wouldn't be surprised to see a green close	positive
Shell's $70 Billion BG Deal Meets Shareholder Skepticism	negative
SSH COMMUNICATIONS SECURITY CORP STOCK EXCHANGE RELEASE OCTOBER 14 , 2008 AT 2:45 PM The Company updates its full year outlook and estimates its results to remain at loss for the full year .	negative
Kone 's net sales rose by some 14 % year-on-year in the first nine months of 2008 .	positive
The Stockmann department store will have a total floor space of over 8,000 square metres and Stockmann 's investment in the project will have a price tag of about EUR 12 million .	neutral

FIGURE 7: A sample of the 6000-sentence file of labeled news posts to undergo financial sentiment evaluation through the transformer model.

The division of available data into sentiment and sentence categories also makes the process of evaluating and transforming text much easier. The program creates new, empty arrays where sentences and sentiment labels will be stored accordingly. The algorithm then iterates through every sentence and stores the corresponding positive, negative, or neutral label into a unique array. At the end of the sequence, the array of appended labels is converted to a vector with the SentenceTransformer model. This model is a deep learning system that generates embeddings for financial sentiment, capturing semantic meaning into a feature vector. These vectors will allow the linear regression and neural network to generate enhanced price predictions.

The embedded vectors then undergo feature concatenation, combining with the database of previous stock prices along a two-dimensional axis into a single representation set for the two models. The program's multimodal interface ensures a more comprehensive integration of information from the different feature types, capturing any potential interactions between sentiment values and historic performance. With the NumPy package, a preprogrammed feature architecture applies column-based horizontal concatenation of the price matrix and sentiment vectors to create a simpler two-dimensional input matrix. The linear regression and neural network are naturally able to process this concatenated tensor, optimizing their predicted outputs in the process.

Results

The linear regression model and neural network were used during the two-week testing period for Microsoft, Amazon, and Google stock price predictions. The tables below show the predicted prices on two random dates during this period. For comparison, the actual stock prices for the specified dates are also listed. Both models also achieved a relatively low average prediction error, with the neural network performing exceptionally well throughout the testing.

	Linear Regression	Neural Network	Actual Price
Microsoft	$253.50	$253.31	$252.46
Amazon	$95.13	$94.91	$94.28
Google	$91.99	$91.45	$90.09

TABLE 3: The predicted stock prices for Microsoft, Amazon, and Google on February 27, 2023, by the linear regression and neural network, with the actual stock prices listed as well.

	Linear Regression	Neural Network	Actual Price
Microsoft	$250.10	$250.76	$250.76
Amazon	$93.76	$93.80	$93.87
Google	$89.75	$89.85	$90.16

TABLE 4: The predicted stock prices for Microsoft, Amazon, and Google on February 29, 2023, by the linear regression and neural network, with the actual stock prices listed as well.

	Linear Regression Error	Neural Network Error
Microsoft	0.54%	0.29%
Amazon	0.62%	0.47%
Google	1.83%	1.38%

TABLE 5: The average percent errors for both the linear regression and the neural network across the two-week testing period for the stock prices of Microsoft, Amazon, and Google.

Conclusion

This project draws its successes from the innovative weighting techniques, adaptive learning structures, and novel systems that were implemented. To verify the accuracies of the linear regression and neural network, the models were tested on other companies as well, producing a surprisingly low average prediction error of 0.41% for Nike's stock prices and 0.59% for Disney's stock prices. It is important to note that the average model prediction error and the MSE represent different metrics of performance. Although these models were successful in predicting stock prices and achieved low average percent errors for their forecasts, there are more advanced or complex machine learning models that have been specifically developed for application in market prediction. The implementation of these deep learning models remains a priority for generating increasingly accurate stock price predictions. Nevertheless, this paper aims to challenge basic A.I. systems in solving complex, real-world problems through analytical methods and robust mathematical schemes. As machine learning algorithms and models progress, these systems can be updated to account for political, cultural, and socioeconomic trends in major industries around the world to generate predictions at an even higher accuracy. The almost-limitless applications of A.I. prediction methods in helping investors reduce risk while maximizing profits stress the need for such technologies in the unpredictable financial market.

References

Agarwal, A. (2018, October 5). *Linear Regression using Python*. Medium. Retrieved from https://towardsdatascience.com/linear-regression-using-python-b136c91bf0a2

Arora, S., Hazan, E., & Kale, S. (2012, May 1). The Multiplicative Weights Update Method: A Meta Algorithm and its Applications. *Theory of Computing*, *8*(1), 121–164. https://doi.org/10.4086/toc.2012.v008a006

Barla, N. (2023, April 19). Overfitting vs Underfitting in Machine Learning: Everything You Need to Know [web log]. Retrieved 2023, from https://neptune.ai/blog/overfitting-vs-underfitting-in-machine-learning.

Bias-Variance Trade off - Machine Learning. GeeksforGeeks. (2020, June 3). Retrieved from https://www.geeksforgeeks.org/ml-bias-variance-trade-off/

Bishop, J. (2021, December 2). *From GameStop to Tesla - how social media is driving the stock market*. Maddyness UK. Retrieved from https://www.maddyness.com/uk/2021/12/02/from-gamestop-to-tesla-how-social-media-is-driving-the-stock-market/

Chaddha, A., & Yadav, S. (2022, August 31). Examining the predictive power of moving averages in the stock market. *Journal of Student Research*, *11*(3). https://doi.org/10.47611/jsrhs.v11i3.3382

Chanda, S. (2022, December 20). *Fundamental vs Technical vs Quantitative Analysis*. Jarvis. Retrieved from https://jarvisinvest.com/jarvis-library/fundamental-vs-technical-vs-quantitative-analysis/

Chandola, D., Mehta, A., Singh, S., Tikkiwal, V. A., & Agrawal, H. (2022, August 1). Forecasting directional movement of stock prices using Deep Learning. *Annals of Data Science*. https://doi.org/10.1007/s40745-022-00432-6

de Souza, M. J., Ramos, D. G., Pena, M. G., Sobreiro, V. A., & Kimura, H. (2018, February 24). Examination of the profitability of technical analysis based on Moving Average Strategies in BRICS. *Financial Innovation*, *4*(1). https://doi.org/10.1186/s40854-018-0087-z

Elleuch, J., & Lotfi, T. (2009, August). *Fundamental Analysis Strategy and the Prediction of Stock Returns*. Retrieved from https://www.researchgate.net/publication/255608724_Fundamental_Analysis_Strategy_and_the_Prediction_of_Stock_Returns.

Fernando, J. (2023, March 31). *Moving Average (MA): Purpose, Uses, Formula, and Examples*. Investopedia. Retrieved from https://www.investopedia.com/terms/m/movingaverage.asp

Game Theory. WallStreetMojo. (2022, September 22). Retrieved 2023, from https://www.wallstreetmojo.com/game-theory-2/

Ghojogh, B., & Crowley, M. (2019, May). The Theory Behind Overfitting, Cross Validation, Regularization, Bagging, and Boosting: Tutorial. Retrieved from https://www.researchgate.net/publication/333505702_The_Theory_Behind_Overfitting_Cross_Validation_Regularization_Bagging_and_Boosting_Tutorial.

Gillham, D. (2023, May 21). *Trading the Stock Market – Why Most Traders Fail*. Wealth Within. https://www.wealthwithin.com.au/learning-centre/share-trading-tips/trading-the-stock-market

Gillioz, A., Casas, J., Mugellini, E., & Khaled, O. A. (2020). Overview of the Transformer-based Models for NLP Tasks. *Proceedings of the 2020 Federated Conference on Computer Science and Information Systems*, *21*, 179–183. https://doi.org/10.15439/2020f20

Gupta, M. (2023, March 1). *Gradient descent in linear regression*. GeeksforGeeks. Retrieved from https://www.geeksforgeeks.org/gradient-descent-in-linear-regression/

Gülen, K. (2023, January 11). *Applying machine learning in financial markets: A review of state-of-the-art methods*. Dataconomy. Retrieved from https://dataconomy.com/2023/01/11/stock-prediction-machine-learning/

Hayes, A. (2023, January 13). *Technical analysis: What It Is and How to Use It in Investing*. Investopedia. Retrieved from https://www.investopedia.com/terms/t/technicalanalysis.asp

Kantor, A., Stafford, P., & Lewis, L. (2021, January 28). *US trading volumes soar past 2008 peak in reddit battle*. Financial Times. Retrieved from https://www.ft.com/content/56658052-76fe-4910-8cb7-810039753f7c

Lo, A., Mamaysky, H., & Wang, J. (2000, March). Foundations of Technical Analysis: Computational Algorithms, Statistical Inference, and Empirical Implementation. *The Journal of Finance*. https://doi.org/10.3386/w7613

Lush, M., Fontes, A., Zhu, M., Valdes, O., & Mottola, G. (2021, February). Investing 2020: New Accounts and the People Who Opened Them. Chicago; FINRA Foundation.

Malkiel, B. G. (2000). *A Random Walk Down Wall Street: Including a Life-Cycle Guide to Personal Investing*. W. W. Norton & Company, Inc.

Mujtaba, I. M., & Sowgath, M. T. (2022, July 1). Application of artificial intelligence in desalination processes. *Desalination Technologies*, 541–593. https://doi.org/10.1016/b978-0-12-813790-1.00011-6

Nabeel, M. (2023, October 10). *What is the vanishing gradient problem?* Educative. Retrieved from https://www.educative.io/answers/what-is-the-vanishing-gradient-problem

Open Market Operations. Board of Governors of the Federal Reserve System. (2023, March 22). Retrieved from https://www.federalreserve.gov/monetarypolicy/openmarket.htm

Pang, B., Nijkamp, E., & Wu, Y. N. (2019, September 10). Deep Learning With TensorFlow: A Review. *Journal of Educational and Behavioral*

Statistics, *45*(2), 227–248. https://doi.org/10.3102/1076998619872761

Petrusheva, N., & Jordanoski, I. (2016, June 5). Comparative analysis between the fundamental and technical analysis of stocks. *Journal of Process Management. New Technologies*, *4*(2), 26–31. https://doi.org/10.5937/jpmnt1602026p

Saini, A. (2021, September 15). Master the AdaBoost Algorithm: Guide to Implementing & Understanding AdaBoost [web log]. Retrieved 2023, from https://www.analyticsvidhya.com/blog/2021/09/adaboost-algorithm-a-complete-guide-for-beginners/.

Segal, T. (2023, March 17). *Fundamental Analysis: Principles, Types, and How to Use It*. Investopedia. Retrieved from https://www.investopedia.com/terms/f/fundamentalanalysis.asp

Seif, G. (2019, May 20). *Understanding the 3 most common loss functions for Machine Learning Regression*. Medium. Retrieved from https://towardsdatascience.com/understanding-the-3-most-common-loss-functions-for-machine-learning-regression-23e0ef3e14d3

Silva, M. (2019, October 26). *Game Theory to Interpret Machine Learning Models and Predictions — The Ultimate Guide*. Medium. Retrieved from https://towardsdatascience.com/the-ultimate-guide-using-game-theory-to-interpret-machine-learning-c384cbb6929

Sydorenko, I. (2021, March 18). *Machine Learning & Training Data: What You Need to Know*. Label Your Data. Retrieved from https://labelyourdata.com/articles/machine-learning-and-training-data

Zhao, J., Dai, S., Venkatesan, R., Zimmer, B., Ali, M., Liu, M.-Y., Khailany, B., Dally, W. J., & Anandkumar, A. (2022, September 22). LNS-madam: Low-precision training in logarithmic number system using multiplicative weight update. *IEEE Transactions on Computers*, *71*(12), 1–12. https://doi.org/10.1109/tc.2022.3202747

Zhang, K. (2020). *Research on Quantitative Investment Based on Machine Learning*. Atlantis Press. https://www.atlantis-press.com/article/125947104.pdf