

REFINING THE INFLUENCE OF SENTIMENT INTENSITY IN MARKET NEWS ON AI/ML STOCK PRICE PREDICTION MODELS.

Prof. P.Sheela

professor, Gitam School of Business

Dr. V.Gowri Lakshmi

Assistant professor , Gitam School of Business

Dr.G.V.K.Kasturi

Assistant professor, Gitam School of Business

Dr.M.Purushottam Naidu

Associate professor, Woxsen University

Abstract:

Integrating sentiment analysis from market news into artificial intelligence and machine learning (AI/ML) stock price prediction models has gained significant attention due to its potential to enhance forecasting accuracy. This study delves into the refinement of incorporating sentiment intensity from market news into AI/ML stock price prediction models. While existing research has explored the impact of sentiment analysis, there remains a need to understand how the intensity of sentiment expressed in news articles affects the predictive power of such models.

This research proposes a comprehensive approach to address this gap by developing a novel framework that quantifies sentiment intensity in market news and evaluates its impact on AI/ML stock price prediction accuracy. The framework harnesses advanced natural language processing techniques to extract and quantify sentiment intensity, capturing nuanced variations in sentiment expression. A diverse dataset of historical market news and stock price data is utilized for training and evaluation.

Through extensive experimentation, our findings reveal that refining AI/ML stock price prediction models with sentiment intensity significantly enhances their predictive capabilities. The proposed framework demonstrates that incorporating sentiment intensity provides a deeper understanding of market dynamics and improves the models' ability to capture price trends and fluctuations accurately. The study also examines the optimal integration of sentiment intensity alongside other relevant predictors.

Key Words: Sentiment Intensity, Artificial intelligence, Machine Learning, Stock Pricing, Natural Language Processing, Algorithmic Trading System.

Introduction:

Financial markets, be it stocks, derivatives, or commodities, generate data on all the instruments traded in the market. Data collected continuously for all trades being made is called tick data, which consists of a stock name or identifier, timestamp, trade price, and quantity traded. In addition to tick data, other end-of-day stock data consolidate daily trading data that are disseminated along with any corporate action data of the company. These data points are a trove of valuable insights that, when viewed holistically, paint a picture of the price movements of the stocks over time and help predict the trend movement of the stock shortly. This process of analyzing historical data and identifying patterns using technical indicators to predict price movements soon is called “Technical Analysis”.

Any company that needs to tap the stock market for capital must list its stock on the stock exchange. Enlisting their stocks on the exchange is called Initial Public Offering (IPO), in which companies typically use investment banks' help to execute the IPO. One eligible criterion for IPO is to have sound annual audited financial reports over time. The yearly financial reports consist of Balance Sheets, Income Statement, Cash Flow Statement, etc., which provides a glimpse into the company's inner workings. The data typically captured in these reports include assets, liabilities, equities, borrowings, sales, expenses, taxes, etc. These sets of data the company provides in its annual financial statements are called “Fundamental Data.” When multiple years of fundamental data are viewed together, it paints a picture of the sound health of the company or the lack of it and helps predict its stock price movement shortly. This process of analyzing companies' fundamental data to predict the trading price movement of its stock is called “Fundamental Analysis”.

Financial Analysts review every detail provided in the financial reports of companies they are analyzing to try and predict the stock price movement. Some analysts also use technical indicators and relevant market news to predict price movement. Market news is subjective and verbose; hence, it needs to be carefully read, digested, and, when interpreted along with other data metrics, makes it a powerful tool to predict stock price movements more accurately. These price predictions are documented in financial analysis reports, which are then used by investors and traders to make trading decisions, either to increase profitability or mitigate risk.

With the exponential growth in computing power, the cost of computing technology has significantly reduced. This permeated the use of computing technology to analyze vast amounts of data to make trading decisions based on some customized logic built into the programs. Such a computerized system that interprets data identifies historical patterns, establishes relationships between variables, and follows some rules to make trading decisions is called “Algorithmic Trading” or “Algo Trading.” Over the years, the volume of such algorithm trades has significantly grown. As per a CNBC report, the daily volume of algo trading can account for almost 80% of the volume of U.S. stocks, based on volatility.

While Algo trading has its roots in the early 1990s using simple rule-based logic embedded into Microsoft Excel worksheets, it has become a sophisticated behemoth using Artificial Intelligence

(AI) and Machine Learning (ML), which uses neural networks to mimic human intelligence thinking. Algo systems interpret numeric data easily to make objective trading decisions; however, they need the interpretation ability for subjective news. However, with Natural Language Processing (NLP) tools gaining grounds to interpret textual subjective news, there could be an opportunity to use these NLP interpretations of relevant news to augment Algo trading signals, enabling them to be more accurate in their predictions.

This research aims to integrate NLP sensitivity intensity and augment it along with the predictive power of AI&ML to enhance the accuracy of stock price prediction and help reduce risk or increase profitability. The study focuses on a set of US stocks from different sectors that have been trading for at least two decades, so we can have vast amounts of historical data to do the back-tests of the models.

Brief Review of Literature

There is enough literature on trading and algorithm trading in addition to NLP. This attempt at the literature review aims to briefly review the current literature in each area to help assess the recent research work done so far and identify the gap that will justify the hypothesis for this thesis.

Michal Balcerak et al. (2020) followed the trends in asset management to classify market states rather than looking at predicting the future prices of stocks directly. Using carefully segregated labels, they relied on moving averages to define the market states classifier. These neural network classifiers helped predict the stock's buy or sell order. Michal's model has three label classes: buy, sell, and neutral, depending on the expected price trends in the current market. Their strategies paid off in increased returns on the out-of-sample dataset over that of the market. However, this model does not consider recent news or macroeconomic factors in the model classifier, which impacts the label classifiers. Mattia Vicari et al. (2020) used sentiment analysis of market news using Natural Language Processing to predict the Dow Jones Industrial Average (DJIA) index. They used Deep Learning, a neural network process, to perform sentiment analysis on the news. They pre-processed the news by removing stop words, converting the news to lowercase, and removing numbers and special characters. One of the main assumptions is that positive financial sentiment does not equal positive words and vice versa; this helps the models learn for themselves rather than constrict the learning. They concluded that removing the assumption that positive financial sentiment is equal to positive words makes it difficult for deep learning to provide better results than the flip of a coin. Xiao-Yang Liu et al. (2020)) developed a Deep Reinforcement Learning (DRL) set of libraries called FinRL that is specific to quantitative finance to help developers with their stock trading strategies. The main advantage of this library is that it incorporates constraints such as trading transaction cost, investor risk aversion, market liquidity, etc., and helps stimulate the trading environment of different stock exchanges. This is some good information about where a trading strategy can be back-tested. Ao Kong et al. (2020) studied the effects of price jumps (significant and discontinuous change) and were able to help with micro-trading strategies. They identified volume and volatility-related attributes exhibit abnormalities

before price jumps. Their study identifies the indicators like QS (Quoted Spread), RV(Realized Volatility), MA(Moving Average), EMA(Exponential Moving Average), and TR(True Range) that can confidently identify extreme trading behaviors before the price jump happens. Omer Berat Sezer et al. (2017) used a genetic algorithm to develop a stock trading system based on optimized technical analysis to create buy or sell orders. They used close prices of DJIA 30 constituent stocks for model validation. They used the Relative Strength Index (RSI) and Simple Moving Average (SMA) technical analysis indicators. They concluded that this model produced comparable or better results when compared to the buy-and-hold trading strategy. Ajit Kumar Rout et al. (2015) developed Functional Link Artificial Neural Networks (FLANN) to address some of the concerns of traditional AI, such as slow convergence, local minimum, overfitting, and significant iterations leading to high costs. The architecture of FLANN is a single-layer ANN structure, where the hidden layers are eliminated by transforming the input pattern to a high dimensional space using polynomial or trigonometric functions. However, while this model is suitable for non-linear training, can be overfitting the variables. Mirela Malin et al. (2004) evaluated the Fama-French multifactor model and applied it to three European markets. For their multifactor model, they obtained monthly data on stock return and accounting data such as market value and book value. They also evaluated the expected return of a portfolio and the market factors such as firm size and book-to-market for all the equity. Their results showed that the beta of stock sensitivity concerning market movement, used in the Capital Asset Pricing Model (CAPM), is insufficient to describe the variation in average market equity returns. Gabriel Dan I. Anghel (2015) analyzed the Moving Average Convergence Divergence (MACD) technical analysis indicator for stock markets of 75 countries. The methodology used is a market simulation using trading rules optimized on out-of-sample stock quote information. He concluded that certain companies could obtain abnormal costs and risk-adjusted returns, which are superior to the rest. As it was the standard model applied to all the countries, he concludes that information inefficiencies are present in global markets, and it's not the same. Martin Hilbert et al. (2020) studied the impact of algorithmic trading on trading dynamics over the last decade. Algorithmic trading predictably executes a self-learned or given set of rules to conclude. The model they have developed is in the foreign exchange markets and has an additional level of predictable structure to the dynamics of predicting bid-ask spread. Their study finds that algorithmic trading correlates with increased predictability as it reduces the uncertainty about the next bid-ask spread prediction. They feel that algorithmic predictions to the fifth digit behind the dollar has increased the uncertainty. Jan Fraenkle et al. (2009) highlight that higher order volumes are typically done using automated trading algorithms. Based on the customer's order constraints, the trader typically picks a strategy that optimizes the order based on constraints. This review analyses how these trading algorithms work. It concludes that algorithmic trading has numerous advantages compared to human traders, and automated trading has opened various avenues.

It, however, needs some technical effort to set them up, in addition to the knowledge of market microstructure.

Research Gap

With the computational costs significantly coming down and the power of computation growing exponentially, Artificial Intelligence (AI) has gained its limelight. With business understanding and realizing that AI is a power they need to reckon with, there is extensive research in the field. There are enough scholarly articles on algorithmic trading that use AI in the financial markets to predict stock prices for higher returns or mitigate risk. In parallel, there is enough research in Natural Language Processing (NLP) to decipher subjective written and verbal language. We also do know that financial stock markets are driven by market news, and the traders who are quick to digest this news, understand the correlation of specific news on the impact of the stocks in their portfolio, and take corrective trades are the ones who have an edge over their competitors. However, most of this news is interpreted by traders, and based on their knowledge, corrective measures are taken, which can take some time. Wouldn't it be good to harness the computation power and use NLP to quickly assess the sentiment intensity of the news and pass on a signal for the algorithmic trading system to evaluate and take appropriate steps to benefit from it in almost real-time? If addressed, there is a gap, which might be able to algorithmic trading system to mitigate risk and/or increase profitability. Through this research paper, we proposed to focus on the above-identified gap.

Need for the Study

Market news is known to provide the much-needed edge that a trader needs; if harnessed quickly, speed is the name of the game. Market news, being voluminous, takes time to digest and make hard trading decisions. News so far has been elusive to automatically capture for traders, as regular automated trading is already captured in algorithmic trading systems. With Natural Language Processing systems that can computationally interpret news, it can be tactically integrated with predictive stock price models, significantly changing the trading landscape. This study will focus on capturing relevant news, converting into sentiment intensity signals using NLP modules, and using those signals to augment algorithmic trading systems.

Algorithmic trading platforms are available in various programming platforms such as C, C++, Python, etc. This can be modeled using fundamental analysis, technical analysis, and macroeconomic metrics such as company growth, dividend payoff, simple moving averages, Fibonacci series, unemployment, inflation, etc. The aim is to add sentiment intensity metrics from the NLP platform to the algorithmic trading model and assess if the stock price prediction improves. This will undoubtedly add value to the traders to mitigate risk and/or improve profitability.

Research Objectives

Based on the research gap and the need for study established, the objectives of this thesis are the following: - Integrate sentiment intensity signal into the AI/ML model and predict stock prices. - Evaluate the improved predictive power of the signal.

Research Hypothesis

Based on the objectives, the following hypotheses are laid out, and the collated metrics will be used to validate the idea.

H0: Relevant market news cannot be interpreted in real time and does not impact stock price prediction.

H1: Relevant market news can be interpreted realistically and impact stock price prediction.

Research Methodology

With a clear objective, the following research methodology is proposed:

AI/ML models are efficient when enough historical data is available to perform backtests and validate the models. In addition to the amount of data, it is also essential to collect data that includes growth and recession periods. Hence, it has been decided to collect historical data from 1st January 2000 until 31st Nov 2020, which is more than 20 years of historical data that covers three recession periods: 2001 (IT Bubble), 2008 (Sub-prime crisis), and 2020 (Covid pandemic).

Data Types:

As the model needs a vast amount of historical stock price data, it has been decided to focus on the US markets. In addition to the historical data and news, the model needs macroeconomic factor data. Table 1.1 Data Sources & Types lists all the data sources and types the model plans to use.

Category	Data Type	Data Source	Database	Historical Data	Periodicity	Method of Collection
Macro Economic	Gross Domestic Product	Quandl	FRED/GDP	20 Years	Quarterly	Python Quandl API
Macro Economic	Unemployment	Quandl	FRED/NROU	20 Years	Quarterly	Python Quandl API
Macro Economic	US Treasury 10 Yrs Rate	Quandl	USTREASURY/LONGTERMRATES.1	20 Years	Daily	Python Quandl API
Macro Economic	Consumer Price Index	Quandl	FRED/CPIAUCSL	20 Years	Monthly	Python Quandl API
Market Index	Dow Jones Ind. Avg.	Quandl	BCB/7809	20 Years	Daily	Python Quandl API
Stock Data	High, Low, Close, Vol.	Yahoo	Finance	20 Years	Daily	Python Yahoo API
Market News	News	Reuters/Refinitiv	Thomson Reuters	100 Days	Keyword Relevant News	Python Reuters API

Table – 1: Data Type & Sources

Statistical Tools:

The model will be developed in Python programming language using statistical library modules that are available. To validate the hypothesis, the following metrics will be collated.

RMSE – Root Mean Squared Error

R - Squared

F – Statistics

Artificial Intelligence/Machine Learning Model

As part of this AI/ML model to predict the impact of the price of stocks, the following analysis will be performed:

- Python programming language was chosen for this analysis due to its ease and the availability of vast open-source libraries that can help to perform machine learning and neural network analysis.
- Collation of 20 years of historical data for all the data types and sources defined earlier.
- Clean up, fill in missing data values, and normalize all data types to ensure uniformity between the independent variables.
- Collation of all data types into a single Python data frame object.
- Analyze the correlation between each data type to minimize multicollinearity, as shown in Figures- 1 and Figure-2.

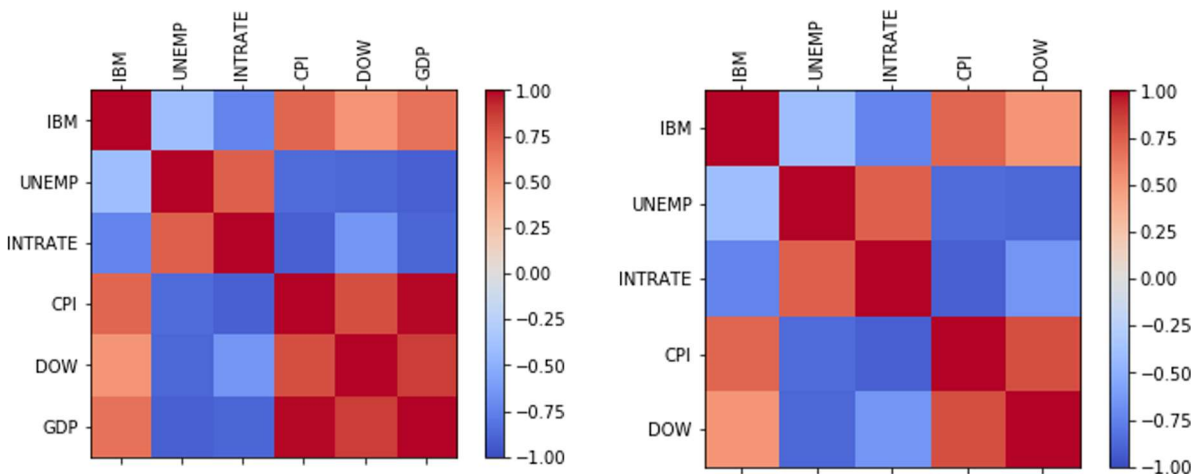


Figure-1: Correlation heatmap between dependent & independent variables

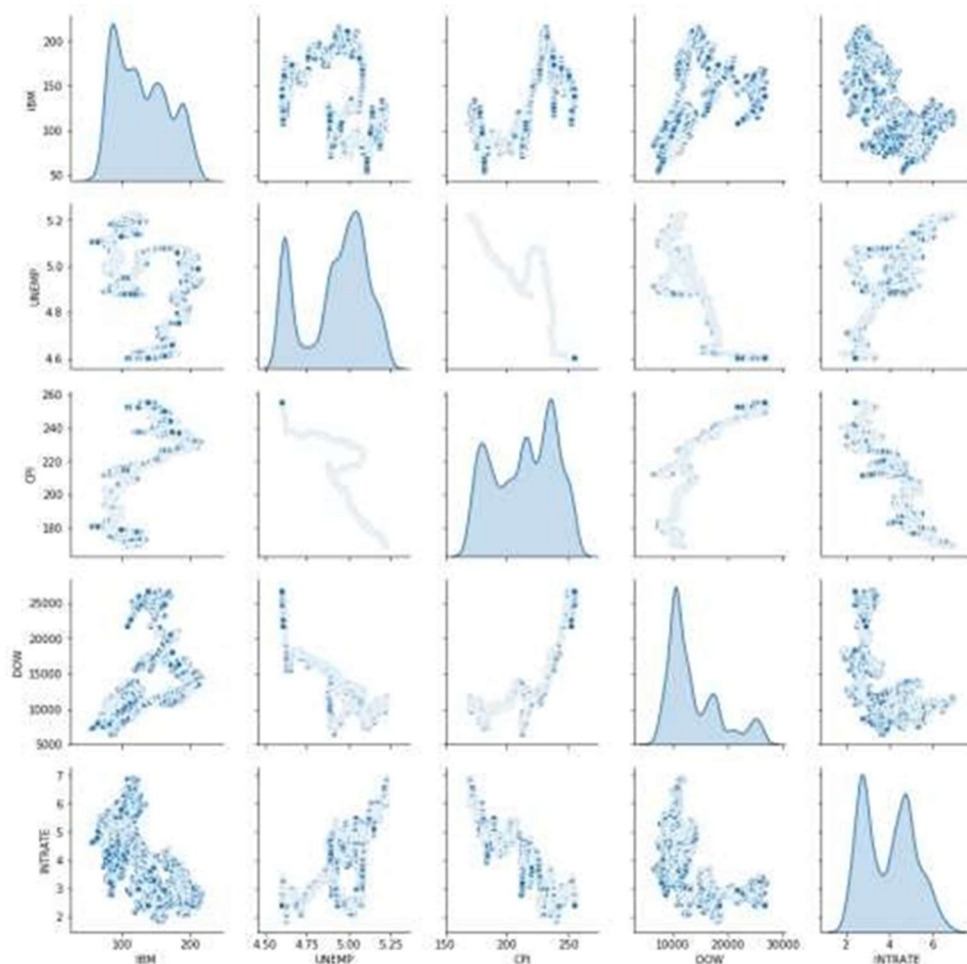


Figure-2: Correlation pair plots between dependent & independent variables

- News was collected, sensitivity intensity was collated day-wise using a VADER (Valence Aware Dictionary and sentiment Reasoner) sensitivity analyzer, and a news sensitivity intensity vector was generated.
- Two data frames will be generated: one with news and one without news integrated into it.
- The data frame was split between training and validation sets at a ratio of 0.80 to 0.20 to ensure adequate data for the model to learn.
- Once the models are built, they will be trained using the training dataset that includes the historical stock prices.
- After training, the trained model will fit with the test dataset without the stock prices.
- The model will then predict the stock prices of the test dataset.
- This prediction is then compared with the actual stock prices in the test dataset.

- The prediction error, which is the deviation from the predicted stock price to the actual stock price, is collated and plotted.
- To baseline the prediction, Python's Stats model OLS library was used independently for each stock.
- The program was then executed in TensorFlow's keras. The sequential neural network model was calculated with each stock, the news vector, and the Mean Square Error (MSE).
- The program was then also executed in TensorFlow keras. Sequential neural network with other macro-economic factors such as DJIA market indices, Unemployment, and Market Interest Rate, with and without news vector, for each stock.
- The overall R-squared factor for each of the runs was collated and improvement compared, which directly attributed to the improvement in the predictive power of the model, as shown in Figure-3 over 1,000 epochs (iterations).

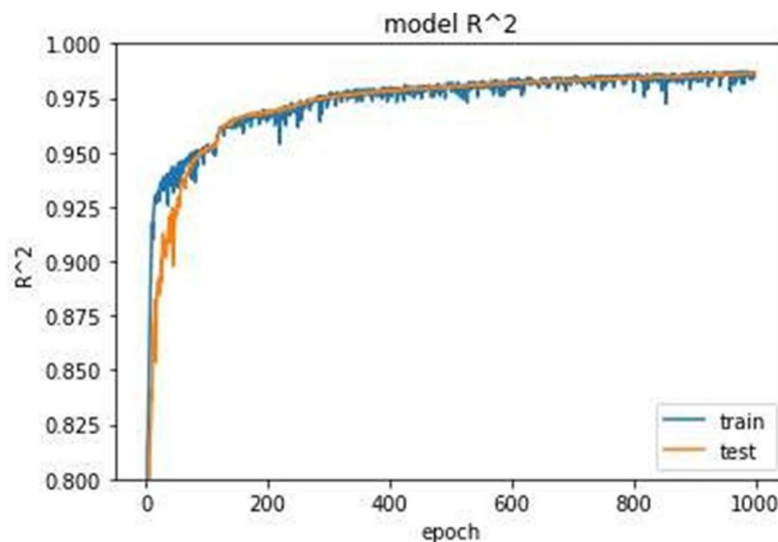


Figure 3: R-squared values for IBM, with Unemp, IntRate, CPI, DOW & News

- The data was also analyzed using the unsupervised neural network CART (Classification and Regression Trees) decision tree model and the mean squared error and R-squared collated.
- The data was also analyzed using an unsupervised neural network random forest tree multi-regression model, and the mean squared error and R-squared collated.
- For the decision trees, the depth of tree branches will be tested for various levels of 5 and 8, and the results will be analyzed.
- For random forests, the model will be analyzed for various numbers of trees in the forest, and the results will be analyzed.

- All raw data frames collated will be saved in output .csv files.
- Using matplotlib libraries, various plots will be drawn based on the prediction of the stock price and the actual data, as shown in the Prediction error plots in Figure 4.

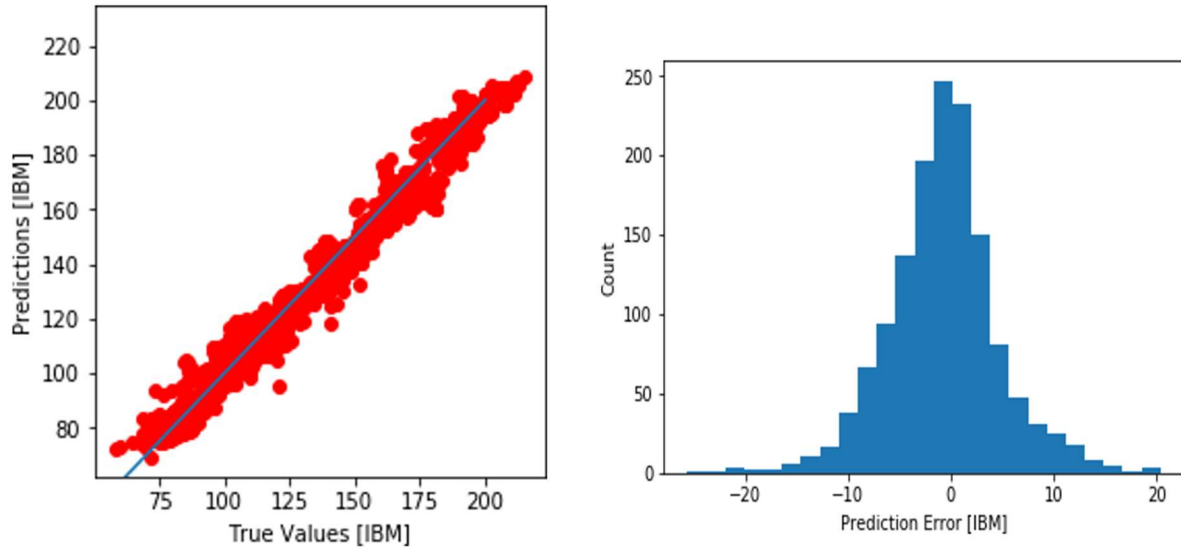


Figure-4: Predicted Error: - Predicted Price Vs True Values

Neural network density stats are shown in Figure-V: Neural Network Density Statistics.

Layer (type)	Output Shape	Param #
dense_6 (Dense)	(None, 64)	320
dense_7 (Dense)	(None, 64)	4160
dense_8 (Dense)	(None, 1)	65
Total params: 4,545		
Trainable params: 4,545		
Non-trainable params: 0		

Table 2 Figure 5: Neural Network Density Statistics

- A summary of the Mean Squared Errors (MSE) and R-squared statistics for training and test data frames for different depths of unsupervised neural network CART (Classification and Regression Trees) decision tree model is shown in Table 3 Decision Tree Statistics.

Depth of Decision Trees	Training		Test Dataframe	
	Dataframe			
	MSE	R^	MSE	R^

8	7.316	0.961	7.380	0.962
5	1.334	0.998	1.360	0.998
2	9.158	0.940	9.204	0.941

Table 3: Decision Tree Statistics

- A summary of the Mean Squared Errors (MSE) and R-squared statistics for training and test data frames for different depths of the random forest is shown in Table 4: Random Forest Tree Statistics.

No. of Trees in Random Forest	Training			
	Dataframe		Test Dataframe	
	MSE	R^	MSE	R^
100	0.025	0.999	0.075	0.999
500	0.028	0.999	0.081	0.999
1000	0.027	0.999	0.075	0.999

Table 4 Random Forest Statistics

Implications of the Study

This research, once concluded, will add to the body of knowledge that has already been established within the financial markets, especially the algorithmic trading models. If the sentiment intensity of news signals is proven to impact the accurate predictability of the models, then we have hit a goldmine. The model can be used to mitigate the risk of the portfolio and, at the same time, improve profitability. They can further be expanded beyond the financial stock markets into the insurance sector, or for that matter, any industry where interpreting news quickly is paramount.

This research contributes to the existing literature by shedding light on the crucial role of sentiment intensity in AI/ML stock price prediction models. It provides insights into the potential mechanisms through which sentiment intensity influences market dynamics, empowering investors, financial analysts, and algorithm developers to make more informed decisions. As the financial markets continue to embrace AI and big data, this study underscores the importance of considering nuanced sentiment analysis for advancing predictive accuracy in stock price modeling.

Recommendations

The developed model should be modular enough that relevant sections can be used to further the research going forward. While this program models a set of stocks to

Validate the hypotheses and is a proof-of-concept; expanding this to cover portfolios with a specific aim of reducing risk and increasing profitability would be good.

Limitations of the Study

This study has the following limitations:

- This Study focused on US markets only
- The study restricted to a few liquid index constituent stocks should expand to other illiquid stocks, where the benefits from such models might be much needed.
- The study should also cover portfolio management.

Scope for Future Research

As reported by CNBC, algorithmic trading has already captured 80% of the trading volumes. At the same time, various models are customized to signals based on risk factors and returns. This model has integrated sentiment intensity signals from an NLP model into an AI/ML model to predict the stock prices for a set of preselected stocks. The following research areas can be an extension of this model.

- Developing AI/ML models with integrated news to manage portfolios.
- Customizing the model to any market beyond US markets.
- Customizing the model for any stocks, not just the liquid index constituent stocks.

Conclusion

While this research is just a proof-of-concept to validate the hypotheses, it has shown significant potential for harnessing other cutting-edge technology to improve algorithmic trading. This model can be scaled up, and predictive apps developed for the stock brokering industry. This will help to roll out the app into the consumer's hands to quickly gauge the impact of any market news and provide them with an alert on the impact on their portfolio stocks to make a buy or sell order. The orders can be automatically generated within the limits the consumer sets. The possibilities are endless, especially when you have financial market knowledge and the technical acumen to develop a solution addressing the latent needs of the industry, which is currently being harnessed through the mushrooming FinTech business model with a CAGR of more than 50%.

References

1. Ajit Kumar Rout, P.K.Dash, Rajashree Dash, Ranjeeta Bisoi, "Forecasting financial time series using a low complexity recurrent neural network and evolutionary learning approach, Journal of King Saud University – Computer and Information Sciences (2017) 29, 536–552.

2. Ao Kong, Robert Azencott, Hongliang Zhu, “Pattern recognition in trading behaviours before the stock price jumps: new method based on multivariate time series classification”, arXiv:2011.04939v1 [q-fin.ST] 10 Nov 2020.
3. Gabriel Dan I. Anghel, “Stock market efficiency and the MACD. Evidence from countries around the world”, *Procedia Economics and Finance*, 32 (2015) 1414-1431
4. Jan Fraenkle, Svetlozar T. Rachev, “Review: algorithmic trading”, *Investment Management and Financial Innovations*, Volume 6, Issue 1, 2009.
5. Martin Hilbert, David Darmon, “How Complexity and Uncertainty Grew with Algorithmic Trading”, *Entropy*, Vol. 22, Issue 5, 10.3390/e22050499
6. Mattia Vicari, Mauro Gaspari, “Analysis of news sentiments using natural language processing and deep learning”, *AI & SOCIETY* (2020),
<https://doi.org/10.1007/s00146-020-01111-1>.
7. Michal Balcerak and Thomas Schmelzer, “Constructing trading strategies ensembles by classifying market states”, arXiv:2012.03078v1 [q-fin.TR], 5 Dec 2020.
8. Mirela Malin, Madhu Veeraraghavan, “On the Robustness of the Fama and French Multifactor Model: Evidence from France, Germany, and the United Kingdom”, *International Journal of Business and Economics*, 2004, Vol. 3, No. 2, 155-176.
9. Murphy, J. J. (1999). *Technical analysis of the financial markets: A comprehensive guide to trading Methods and applications*. New York Institute of Finance.
10. Omer Berat Sezer, Murat Ozbayoglu, Erdogan Dogdu, “A deep Neural-Network Based Stock Trading System Based on Evolutionary Optimized Technical Analysis Parameters”, *Science Direct, Procedia Computer Science* 114 (2017) 473-480.
11. Silvia Amaro, “Sell-offs could be down to a machine that controls 80% of the US stock market, fund manager says”, www.cnbc.com, 5th Dec 2018.
12. Xiao-Yang Liu, Hongyang Yang, Qian Chen, Runjia Zhang, Liuqing Yang, Bowen Xiao, Christina Dan Wang, “FinRL: A Deep Reinforcement Learning Library for Automated Stock Trading in Quantitative Finance”, arXiv:2011.09607 [q-fin.TR] 19 Nov 2020.