Sentiment Analysis and Stock Trend Prediction

Team Members: Zhipeng Luo, Chihshen Hsu, Qi Zhang

Motivation

- **Supervised Learning**

Is the stock movement predictable? This is the core question we target to understand and explore along the project. According to Random Walk Hypothesis(Burton Malkiel,1973), the stock prices evolve according to a random walk and thus cannot be predicted. The efficient market hypothesis also states that asset prices reflect all available information, which implies the market can't be predicted since market price should only react on new information (Wikipedia). The Model Thinker that "the reason stock prices might follow a random walk pattern is that smart investors identify the therefore eradicate patterns." (Page, 2018). However, as you see, a lot of people disagree with this opinion. Considering that human behavior is inherently emotional and irrational, information such as yesterday's closing price, historical stock price trend, moving averages, volume, etc. can affect investors' buying and selling behavior. That's why the term, Irrational Exuberance, is so popular and widespread, which refers to unfounded market optimism that lacks a read foundation of fundamental valuation, but instead rests on psychological factors (POTTERS, 2021).

Generally, prediction methods can be categorized into two groups, fundamental analysis and technical analysis. Fundamental analysis focuses on finding the true value of a stock, which is regarded more as a long-term strategy. In contrast, technical analysis collects all the quantitative information to identify trading signals and capture the movement patterns of the stock market. Due to its short-term nature, technical analysis can be affected by news (Li, 2022).

In this project, our team will focus on technical analysis direction, collecting both financial data and news data to explore the possibility of stock market prediction in three aspects by different machine learning models, including binary classification of the close price up and down, close price prediction and volatility prediction by regression methods.

- **Unsupervised Learning**

How to quantify the emotions from the news? This is the question we hope to understand by applying Unsupervised learning for news data to do sentiment analysis. Based on the studies in Behavioral Financer (Davidson et al., 2003) , emotion has a substantial role in investment decision making. The reason to apply unsupervised learning method for sentiment analysis is initiated from the assumption that market sentiment can affect the performance of overall market and individual stocks. Market sentiment can affect the performance of overall market and individual stocks. Investors may decide to buy, wait, or sell stocks because of the volatility of sentiment. In this case, news plays a key role in influencing sentiment (Kostolany, 1961). Suppose the release of a positive news may strengthen investors' confidence to make a decision to buy or continue to hold, while a negative news may cause investors to sell some of their holdings or even short options.

In this project, we will collect the headlines of the news data corresponding to the same period of the stock movement. After applying the unsupervised learning method for sentiment analysis, the quantitative emotions, including “Positive”, “Neutral” and “Negative”, will be combined with the financial data to verify the impact of the sentimental features.

Data Source

- **Supervised Learning**

On January 3, 2022, there was breaking news attracting the eyes from all over the world that Apple Inc. became the first company to hit $3 trillion market value (Hochreiter & Schmidhuber, 1997), which means the value of the company was greater than the GDP of the UK at that moment. As Apple is so popular globally, we are very interested in whether Apple stock movement is predictable or not. In addition, to verify the prediction capability, we would pick another two stocks to have a cross validation and analysis.

The financial data mainly comes from yfinance API. Through the Ticker module, the stock data, including daily open price, close price, highest price, lowest price, Volume, Dividends and Stock Splits, can be easily accessed in a pythonic way, returning a pandas.DataFrame dataset. In this project, we pick three companies, Apple, Microsoft and Tesla with specific time period corresponding to the news data we can get. The overall introduction of the data source is shown in the table 1 below.

- **Unsupervised Learning**

In the unsupervised machine learning part, our data mainly comes from the websites of news media, such as CNBC, Barron. We use selenium webdriver to simulate the sliding behavior on CNBC's website, and in the process, we obtain the html data with news hyperlinks, headlines, time, description, author, ticker, source, etc., and store these data into csv files.

Table 1：
Introduction of data source

	Case_1	Case_2	Case_3
Company	Apple Inc.	Microsoft orporation	Tesla Inc.
Ticker	AAPL	MSFT	TSLA
Stock Data Format		pandas.DataFrame	
Stock Data Records	1774	772	772
Stock Data Time periods (UTC)	2015.01.01 – 2022.01.17	2018.12.24 – 2022.01.17	2018.12.24 – 2022.01.17
Data Source	yfinance API (https://pypi.org/project/yfinance/)		
Stock News Searching Keywords	AAPL, apple	MSFT, microsoft	TSLA, tesla
News Media		CNBC	
News Format		Html to csv	
News Records	44401	5301	6411
News Time periods (UTC)	2014/12/31 – 2022/01/17	2018/12/19 – 2022/01/28	2018/12/24 – 2022/01/28

Data Manipulation

After collecting the original stock data, we use ta-lib, an API interface for technical stock analysis, which utilizes stock price and volume data to generate 10 new features and indicators that are valuable for analysis, including WMA(Weighted Moving Average), EMA(Exponential Moving Average) and so on. The introductions of all the indicators are included in Appendix.

By sentiment analysis for the news, we have positive, neutral and negative scores of each piece of news. We will do DataFrameGroupBy.aggregate to calculate the daily mean score in case there are multiple pieces of news for each company in one day. Then combine the sentimental results to the financial data to verify the impact of market emotions for stock prediction.

Because of the implementation of deep learning models, a good practice is to do data normalization for the input data. In this project, we use MinMaxScaler to transform features by scaling to a given range. Specifically, for keras LSTM model, the input data has to be transformed to be a 3D tensor with shape [batch, timesteps, feature].

According to our goals, there are three kinds of target labels we need to prepare. For the binary classification prediction about whether the close price will go up or down in the next day, we label the positive as 1 and others as 0 based on the differences of the close price on each two consecutive days. To predict the actual close value, we make a target feature named as "tomorrow close" for the training of machine learning models. Finally, the daily close-price volatility is calculated for training and prediction.

Methods & Evaluation

- **Supervised Learning**

- The direction of stock price movement is hardest to be predicted

In the first task to predict the stock movement direction, there are three machine learning model, LR (Logistic Regression), SVM (Support Vector Machines) and RF (Random Forest), and one deep learning model, LSTM (Long Short-Term Memory), implemented for all three companies. To evaluate the performance of binary classification task, we use four kinds of scores, including accuracy, precision, recall and f1.

The example results of the scores for Apple stock prediction without scaling transformation for the training data is summarized in the table 2 below. For all the results from our models, no matter with or without training data transformation, are no better than random guess. The three models, LR, SVM and RF, has no improvement when combining the emotion data generated by Bert, which is the unsupervised learning method that would be introduced in details later.

However, there is 4% improvement by adding emotion values in training dataset for LSTM prediction from 47% to 51%. LSTM is an artificial recurrent neural network (RNN) architecture (Sepp Hochreiter; Jürgen Schmidhuber (1997) used in the field of deep learning which are extremely powerful time-series models. Even though it is still far from satisfaction. It provides us the motivation to further validate the impact of emotions on LSTM models for stock condition prediction in another two tasks.

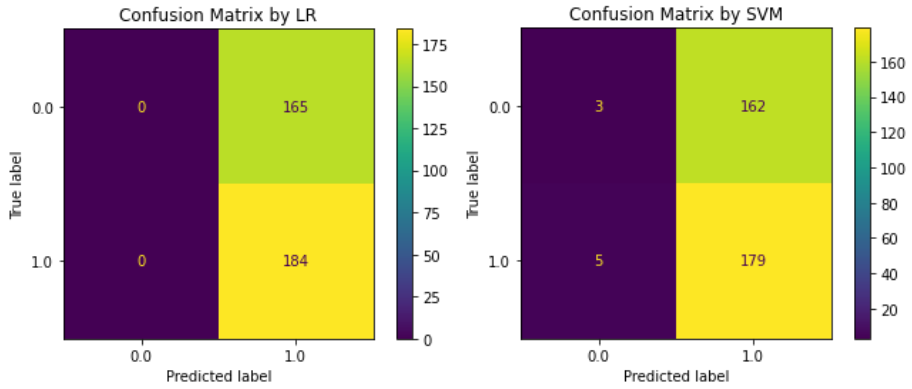
In addition, an interesting finding from confusion matrix shown in Figure 1 is that LR

and SVM are highly preferred to give a bull signal of the market even the models are trained by balanced data set.

Table 2 :
Apple - Binary classification evaluation without scaling

	LR	SVM	RF	LSTM	LSTM trained with emotions
Accuracy	0.53	0.52	0.53	0.47	0.51
Precision	0.26	0.45	0.52	\	\
Recall	0.5	0.5	0.52	\	\
F1	0.35	0.36	0.5	\	\

Figure 1:
Confusion matrix by LR & SVM



- Market Emotions are valuable for close price prediction

In this task to predict the stock close price of the next day, LSTM and ARIMA (Autoregressive Integrated Moving Average) are implemented and evaluated by the metrics of MAE (Mean Absolute Error) and RMSE (Root Mean Square Error).

To evaluate the sensitivity of the model to features for LSTM, we trained the model by the data with different features. One is trained only by financial data, one is trained by only sentimental data extracted from news, the last one is trained by combined data with both types of features. All the three models trained by different features are not powerful enough to give an accurate prediction of the close price next day. However, from the comparison, we find that the market emotions are valuable for some stocks. In figure 2, for apple close price prediction, even though, model trained by combined data has almost the same prediction power, the model trained by financial data performs slightly better, achieving MAE as 31.1 and RMSE as 32.9. But for Microsoft in figure 3, the emotions can improve the prediction performance with the smallest MAE as 46.0 and RMSE 48.7.

Another important parameter to be considered for the data preprocessing for LSTM is “time_step”, namely the memory length you want to keep for each epoch. Taking Apple close price prediction as example, when setting the time_step as 1, the prediction curves is almost horizontally flattened. In this project for all LSTM models, we set time_step as 10 which means we take the data of the past 10 days to predict the next day.

Actually, there is another popular method, ARIMA, for time-series prediction. ARIMA requires the stationarity of the data, meaning the statistical properties such as mean,

variance, and so on do not change over time. However, most of the real-world data, like stock data, are non-stationary by nature. This non-stationarity can be taken care of by using the Box–Jenkins ARIMA(p,d,q) approach (Makridakis & Hibon, 1997). The autoregression AR(p), order p, and moving average MA(q), order q, are determined from the analysis of the autocorrelation function. The number d indicates the number of differences applied to the time series to remove the trend. In this task, we use auto_arima model in pmdarima to pick the best parameters automatically. After fitting the model by only close price in the training dataset and predict the future with specific length. Shown in figure 4 , the model provides a linear prediction and loses the natural properties of the stock movement. Even though the MAE is only 9.4 and RMSE 11.8, we wouldn't use it to predict when the data and environment is complicated.

Figure2:
Apple Close Price Prediction by LSTM

Figure3:
Microsoft Close price prediction by LSTM

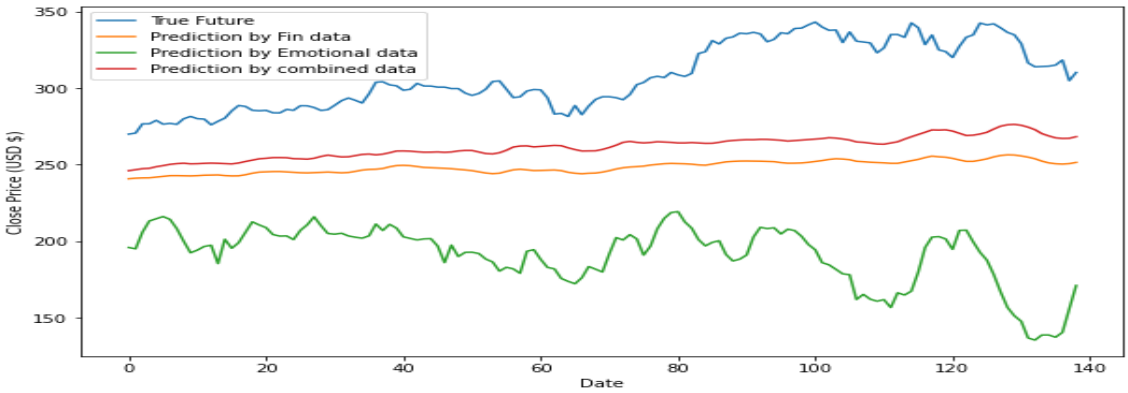
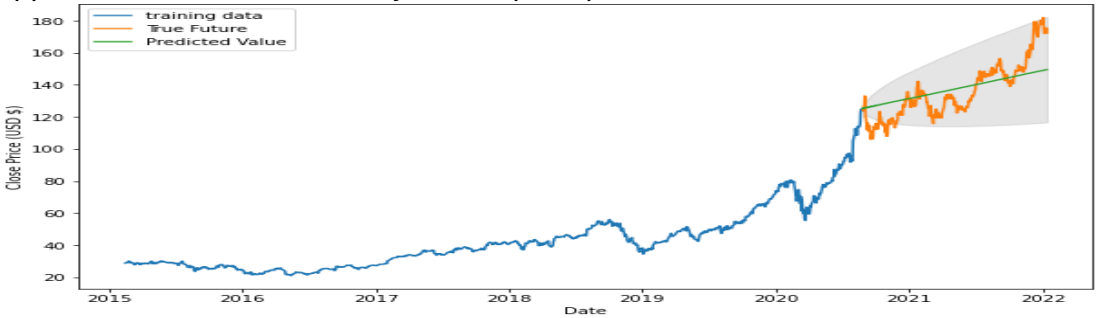


Figure4:
Apple Close Price Prediction by ARIMA(3,1,2)



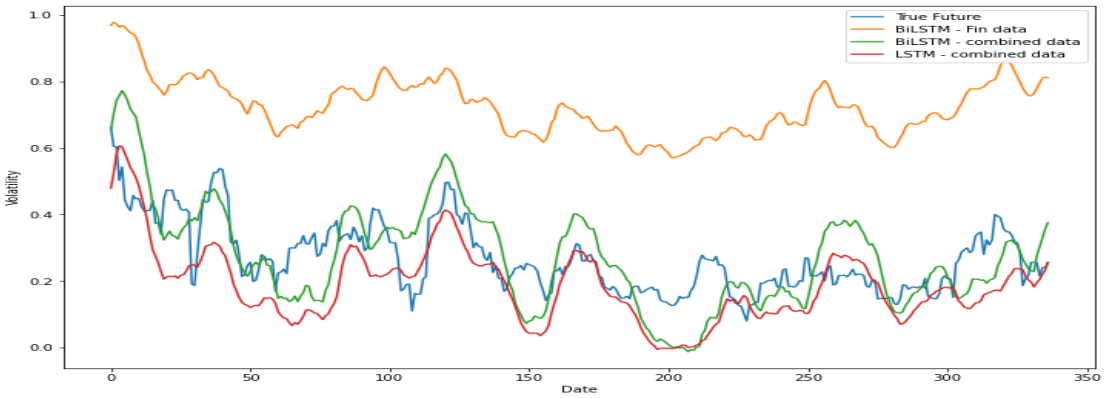
- Stock market volatility prediction performs the best by BiLSTM

Volatility is a statistical measure of the dispersion of returns for a given security or market index (POTTERS, 2021). Volatility refers to the amount of risk related to the amount a person has invested on the stock. Say higher the volatility higher the risk, the price of the stock may go either high or low. In this task, besides LSTMs, we tried Bidirectional LSTMs which enable additional training by traversing the input data twice. In some cases, BiLSTM performs better than LSTM (Siami-Namini et al., 2019).

From figure 5 about the prediction for Apple stock volatility, we can find that BiLSTM trained by only stock data can't provide reasonable predictions. However, if the data combined with emotions, BiLSTM performs the best with MAE as 0.09 and RMSE as 0.11. LSTM models trained with combined data can also achieve certain high-level prediction accuracy with MAE as 0.10 and RMSE as 0.12. An important tradeoff that needs to be considered is between accuracy and overfitting. In this task, the early-stop mechanism is designed during the model training and monitored by "value loss".

To verify the conclusion, we also tried the same methods on Microsoft and Tesla. The prediction results for the stock volatility of both companies are very unstable and unsatisfied. We can infer that the main reason is the insufficient amount of training data. Because of the problem of crawling news, the training records and test records are only 574 and 137 for both companies. Another interesting finding is from stationery testing. By Augmented Dickey–Fuller test, Apple stock volatility has stationary property which Microsoft and Tesla don't have. The impact of stationary property for RNN-based model can be studied further.

Figure 5:
Apple volatility prediction by BiLSTM and LSTM



- **Unsupervised Learning**

- Method Introduction

After crawling the stock market news for the specific company, the two methods, Vadar (Hutto & Gilbert, 2014) and Bert (Araci, 2019; Devlin et al., 2018), are utilized for sentiment analysis to generate three emotion features, including "Positive", "Neutral" and "Negative", for each piece of news.

The main process of Vadar is to first convert the words in the sentence into tokens, then convert the emotion of each word into lexical features, and classify them according to the degree of emotion, and finally calculate the emotional intensity of the content under lexical and rule-based. As a result, we directly use the functions in nltk

to calculate and eventually output three kinds of results: positive, neural, and negative.

Bert also first transforms the text in the sentence into token, and then uses the mechanism of seq2seq and self-attention for each word to infer the relationship between before and after, and finally completes the classification.

A. Self-attention: It is an improvement on attention, as attention is a mechanism to cut sentences into repr. (Shaw et al., 2018), while self-attention also "pays attention" and retrieves semantic information from other elements in the same sequence. This token information is then combined into contextual information and treated as its own repr. to generate the output.

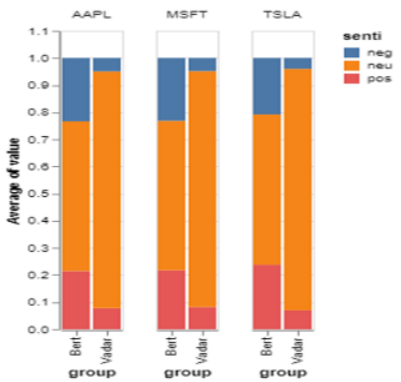
B. seq2seq: It is an improvement on the traditional encoder-decoder, where the encoder-converted hidden state is passed to the decoder only for the final result (Cho et al., 2014), while seq2seq passes the result of each stage to the decoder. This allows the model to verify the contextual content before and after making inferences.

Bert's model is more complex with 110 million parameters in its 11-layer architecture. Due to the computational power limitation, we use transfer learning, in which the hidden layer uses the trained Bert layer. The final output layer is fine-tuned and converted into a sentimental classification task, and the final output is positive, neural, and negative.

- Evaluation

The following chart compares the average of two different sentiment analysis models for news from 2021/01/01 to 2021/12/31. The following figure 6 shows the average of Vadar's and Bert's evaluations of the three stocks, we find that Vadar is more neutral in all three stocks, with lower positive and negative evaluations, while Bert is more able to give positive and negative evaluations.

Figure 6：
Vadar & Bert evaluations in different stocks



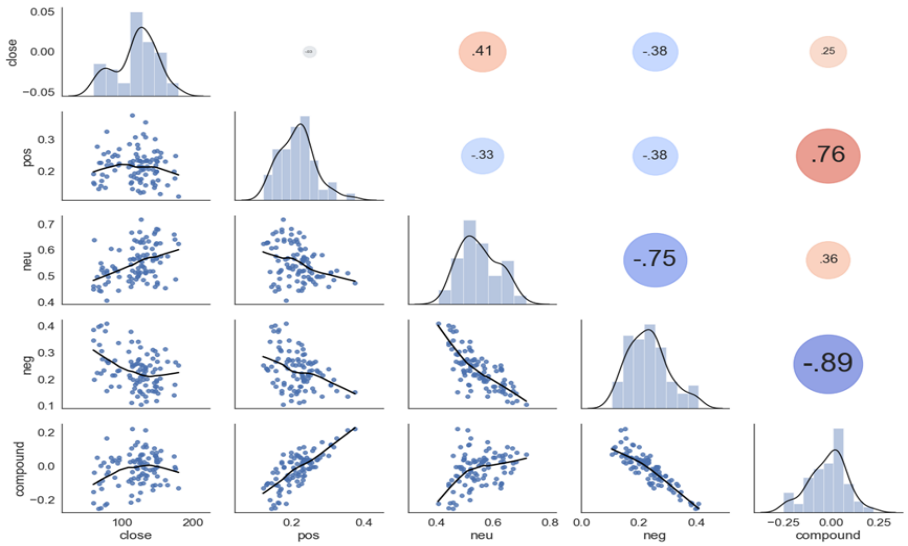
We need to further compare the correlation coefficients between the two models and stock prices. In the table 3 below, we find that the results of Bert's evaluation have a higher correlation coefficient with stock market closing prices compared to Vadar's results, so in this case, Bert outperforms Vadar's performance. On the other hand, we can find that the correlation between stock market prices for positive news is lower and the correlation between negative news and neutral news is higher, and the trend is the same in both models.

Table 3:
Vadar & Bert evaluations in different stocks

		mean	std	Pearson correlation with price
Vadar_AAPL	Positive	0.0734	0.024	-0.06
	Neutral	0.823	0.205	0.05
	Negative	0.0465	0.02	-0.1
Vadar_MSFT	Positive	0.0773	0.032	0.08
	Neutral	0.8201	0.2048	0.05
	Negative	0.0454	0.0208	-0.16
Vadar_TSLA	Positive	0.0694	0.0189	-0.02
	Neutral	0.8811	0.0917	0.07
	Negative	0.0396	0.0396	-0.15
Bert_APPL	Positive	0.2047	0.073	-0.09
	Neutral	0.5136	0.152	0.28
	Negative	0.2162	0.09	-0.33
Bert_MSFT	Positive	0.2085	0.085	0.09
	Neutral	0.5161	0.148	0.17
	Negative	0.2188	0.097	-0.26
Bert_TSLA	Positive	0.2353	0.069	-0.03
	Neutral	0.5472	0.1	0.26
	Negative	0.2077	0.083	-0.26

Because sentiment analysis is a very important input feature in our prediction, in order to examine the data distribution, we compare the size of the data distribution with the correlation coefficient in the figure 7 below to confirm whether the trend of the distribution is consistent with the correlation, and we can confirm from the figure below that the feature has the value of adding input variables.

Figure 7:
Bert sentiment analysis correlation between close price



Conclusion & Discussion

- **Supervised Learning**

One of the main findings in supervised learning part is that stock market volatility is more predictable than stock price or price movement direction. For the stock price movement direction prediction, all the models perform no better than random guess. From the confusion matrix, it is surprised that Logistic Regression and SVM models highly preferred to predict upward direction. When applying ARIMA and LSTM models for time-series data, it is a good habit to do stationery testing to understand the data natural property. By combining emotional data, it is surprised that the volatility prediction performance by BiLSTM improved a lot and binary classification by LSTM improved 4%, while no impact for Logistic Regression, SVM or Random Forest model. The critical parameter "time step" should be tested and verified for every LSTM-kind model. In data preprocessing step, data normalization is normally a good practice for deep learning models.

In case with more time and resources in the future, firstly, we would like to pick more stocks in different industries to explore the prediction by different models. It is assuming that different industry markets may response to the market emotions differently. Secondly, besides of news, it is considerable to collect market emotions from other sources, such as twitter, blogs, financial magazines, etc. Thirdly, cross validation and prediction. For example, can we predict Microsoft stock volatility by the model trained by Apple related data? It is a practice we tried but failed in this project because of insufficient amount of data that we hope to try further in the future assuming that the stock of the company can be impacted by its competitors' financial performance or market emotions.

Regarding to the ethical issues, selection bias may be encountered. In this project, we only pick the stock data of Apple, Microsoft and Tesla with certain period of time. One question we should ask ourselves is that can we still get the same conclusion when collecting the data of different time range? Except for the three companies, can the model be applied to other companies in different industries? In order to avoiding such bias, we could study more companies in different time period to verify the generalization capability. In addition, all those models which we used in the project doesn't mean we advocate them as highly reliable models that exploit the patterns in stock data perfectly. All the studies models here can't be used blindly without any human-in-the-loop for stock exchange or other investment.

- **Unsupervised Learning**

In unsupervised learning section, we learned how to crawl news from websites using selenium, understanding further about the concept of transformer in neural networks. BERT model is powerful to generate sentiment analysis stock predictions. The more surprising result is that Bert generates a sentiment classification in which positive has little effect on the stock price, but neutral and negative have more effect on the stock price. There may be two possible explanations for this. First, people are more sensitive to negative messages than to positive messages, which is like the positive-negative asymmetric proposed by Baumeister et al. (2001), where people attach more importance to negative than positive opinions generated by an object. Second, the Bert model we use does not properly cut positive and neutral evaluations, resulting in some of the news that are positive for the market being included as neutral.

Assuming we have more computational resources, perhaps we can use a more

computationally demanding model such as GPT-3 (Brown et al., 2020) or Gopher (Rae et al., 2021) for sentiment analysis, which can better examine the effect of sentiment in news headlines on stock prices. Besides of sentiment analysis, we can also try topic modeling methods for the news to understand its impact for market from different aspects.

When doing the sentiment analysis, we need to crawl news messages, which may bring burden to the server. We have followed the following principles to minimize the harm throughout the process: 1) Compliance with legal requirements such as GDPR (the European General Data Protection Regulation); 2) If a public API available, we use it to get the required data and avoid scraping; 3) We request data at a reasonable rate. I will strive to never be confused for a DDoS attack; 4) We will only save the data I absolutely need from your page; 5) We will scrape for the purpose of creating new value from the data, not to duplicate it.

Stock market prices are highly unpredictable and volatile. This means that there are no consistent patterns in the data that allow you to model stock prices over time near-perfectly. The predictive models are statistical in nature. It doesn't provide any guarantees. In the unsupervised machine learning section, we find that the semantic approach is effective for analyzing volatility, but this does not mean that it works for all other stocks, nor does it mean that it will guide stock investment.

Statement of Work

The whole project is really a team effort. At the beginning, all of us were involved in data collection and exploration to choose the most suitable news data source. As the computation of sentiment analysis is expensive, Chihshen is mainly responsible for this part, collecting required headlines of news data and transforming them to emotion value. Because of the sophisticated working experience, Qi Zhang provides the domain knowledge and market index explanation to our team. Both Chihshen and Qi Zhang participate in part of the supervised learning model exploration. Zhipeng is responsible for driving the project discussion and monitoring the project progress, aligning the internal meeting time and the meeting with our project instructor. Zhipeng mainly focuses on supervised learning. Along the project progress, all of us work close to each other in brainstorming, discussion, model evaluation and report writing.

Appendix

Financial Index Introduction

- **WMA**(Weighted Moving Average): The Weighted Moving Average calculates a weight for each value in the series. The more recent values are assigned greater weights.
- **EMA**(Exponential Moving Average): The Exponential Moving Average is a cumulative calculation, including all data. Past values have a diminishing contribution to the average, while more recent values have a greater contribution. This method allows the moving average to be more responsive to changes in the data.
- **CCI**(Commodity Channel Index): The Weighted Moving Average calculates a weight for each value in the series. The more recent values are assigned greater weights.
- **ADX**(Average Directional Movement Index): ADX calculations are based on a moving average of price range expansion over a given period of time.
- **KAMA**(Kaufman Adaptive Moving Average): Developed by Perry Kaufman, Kaufman's Adaptive Moving Average (KAMA) is a moving average designed to account for market noise or volatility.
- **MOM**(Momentum): The Momentum is a measurement of the acceleration and deceleration of prices. It indicates if prices are increasing at an increasing rate or decreasing at a decreasing rate.
- **WILLER**(Williams' %R): The Williams Percent Range, is a type of momentum indicator that moves between 0 and -100 and measures overbought and oversold levels. The Williams %R may be used to find entry and exit points in the market.
- **ROC**(Rate of change):The Price Rate of Change (ROC) measures the percentage change in price between the current price and the price a certain number of periods ago.
- **RSI**(Relative Strength Index) : The relative strength index (RSI) is a momentum indicator used in technical analysis that measures the magnitude of recent price changes to evaluate overbought or oversold conditions in the price of a stock or other asset.
- **ATR**(Average True Range): The ATR is a measure of volatility. High ATR values indicate high volatility, and low values indicate low volatility, often seen when the price is flat.

Reference

Araci, D. (2019). Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*.

Baumeister, R. F., Bratslavsky, E., Finkenauer, C., & Vohs, K. D. (2001). Bad is stronger than good. *Review of general psychology*, *5*(4), 323-370.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., & Askell, A. (2020). Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Burton Gordon Malkiel(1973): A Random Walk Down Wall Street: The Time-Tested Strategy for Successful Investing

Cho, K., Van Merriënboer, B., Bahdanau, D., & Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Davidson, R., Scherer, K., & Goldsmith, H. (2003). The role of affect in decision making. *Handbook of Affective Sciences*, 619-642.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

Hutto, C., & Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. Proceedings of the International AAAI Conference on Web and Social Media.

Kostolany, A. (1961). *Das ist die Börse: Bekenntnisse eines Spekulanten* (Vol. 1390). Goverts.

Li, K. Y. (2022). *Predicting Stock Prices Using Machine Learning*. https://neptune.ai/blog/predicting-stock-prices-using-machine-learning#:~:text=The%20stock%20market%20is%20known,financial%20performance%2C%20and%20so%20on.

Makridakis, S., & Hibon, M. (1997). ARMA models and the Box–Jenkins methodology. *Journal of forecasting*, *16*(3), 147-163.

Page, S. E. (2018). *The model thinker: What you need to know to make data work for you*. Basic Books.

POTTERS, C. (2021). *Volatility*. https://www.investopedia.com/terms/v/volatility.asp

Rae, J. W., Borgeaud, S., Cai, T., Millican, K., Hoffmann, J., Song, F., Aslanides, J., Henderson, S., Ring, R., & Young, S. (2021). Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*.

Shaw, P., Uszkoreit, J., & Vaswani, A. (2018). Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*.

Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2019). The performance of LSTM and BiLSTM in forecasting time series. 2019 IEEE International Conference on Big Data (Big Data),

Wikipedia. *Efficient market hypothesis*. https://en.wikipedia.org/wiki/Efficient-market_hypothesis

Wikipedia. *Random Walk Hypothesis*. https://en.wikipedia.org/wiki/Random_walk_hypothesis