Tesla to the Moon

Jeremy Chen, Steven Cheung, Jason Ho, Neal Mahajan

Prediction of Tesla Stock Price Using Twitter Sentiment

Introduction

The price of a stock is said to follow the expectations of company performance in the future. Being able to predict stock price changes is vital to companies in the financial industry because it controls their profits and risk.

Our goal is to use Twitter data and previous stock information to predict the next time step's Tesla stock (TSLA) price. Our hope is that the popularity and sentiment of Tesla related tweets can be an estimate of investor expectations of Tesla performance. Particularly, Tesla has a large twitter following and is quite volatile, meaning an accurate prediction model reduces investor risk.

Data Processing

We gather stock and tweet data from 2/6/20 to 3/6/20 and use the TextBlob package to calculate each tweet's sentiment. To fit the tweets to the stock data, we calculate an overall sentiment score for each trading hour, using equation **[1]** For non trading hours, such as 4PM to 9:30AM or weekends, we take the average of all hours' scores within the time period which becomes the next trading hour's aggregate sentiment score.

$$\text{Aggregated Sentiment} = \sum_{\text{tweet in hour}} \text{tweet sentiment} \times (\text{favorites} + \text{retweets} + \text{replies} + \text{quotes}) \quad \textbf{[1]}$$

Initial Analysis

Of the 71,600 tweets we gathered, TextBlob labeled 10,790 tweets as negative, 31,881 as neutral, and 28,929 as positive. When sentiment score was plotted against the close price for each trading hour *(see figure 1)*, a large number of points were grouped around 0 sentiment and there did not seem to be an apparent relationship between close price and sentiment.

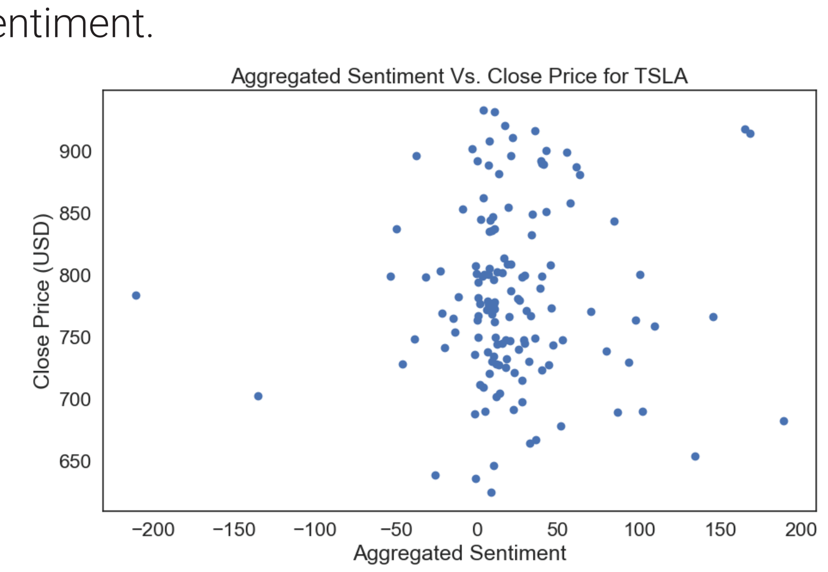

***Figure 1**: Aggregated sentiment vs. next step's close price*

We found five outliers in our aggregated sentiments (z > 3, z < -3). Each outlier had a corresponding news event that likely caused the drop or spike in sentiment. We chose to leave the outliers in our dataset since they still reflect feel-

Ethical Concerns

1. Twitter data is user data and must be respected
2. Companies should implement responsible risk management strategies to protect their investors and the overall market. Unchecked automated trading could lead to a "flash crash" similar to that in 2010.

P-Value/Coefficient Analysis of Linear Multiple Regression

Because a higher p-value suggests that a variable is less likely to be relevant to the multiple regression model, we decided to remove the high, low, and volume variables.

Figure 3 shows the linear regression coefficients of each variable given that all other variables are held constant. Open, high, close and adjusted close had the largest magnitude coefficients, suggesting that they had more impact on the regression while volume had the smallest coefficient.

Interestingly, Twitter Score had a coefficient close to zero and the Twitter Score's p-value is extremely high, indicating it may not be relevant for prediction in linear multiple regression.

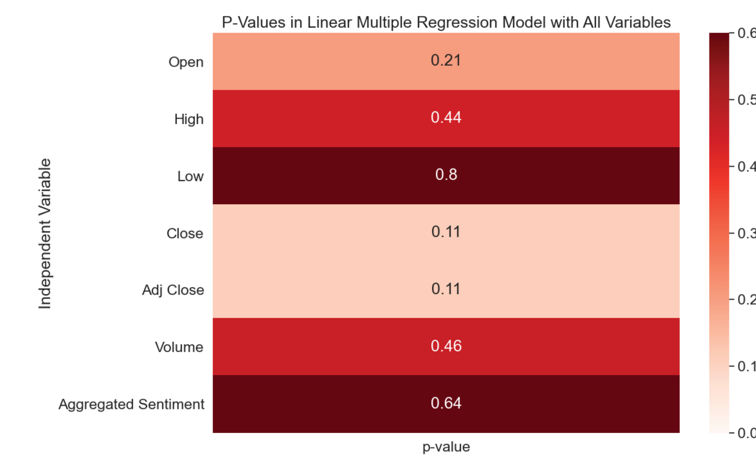

***Figure 2:** p-values of linear multiple regression model*

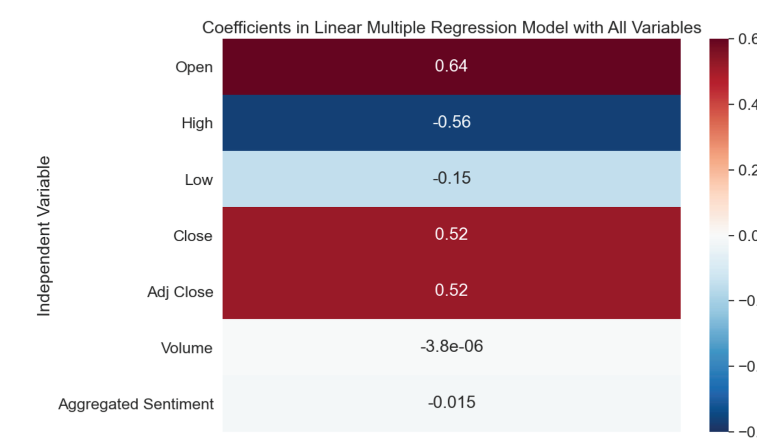

***Figure 3:** coefficients of linear multiple regression model*

Results

We found that the MSE of the baseline models were about the same ((except for polynomial degree 3), if not lower than their respective counterparts, implying that aggregated sentiment score may not have much of an effect on our prediction models. The CNN had the lowest MSE of the models regardless if aggregated sentiment was included.

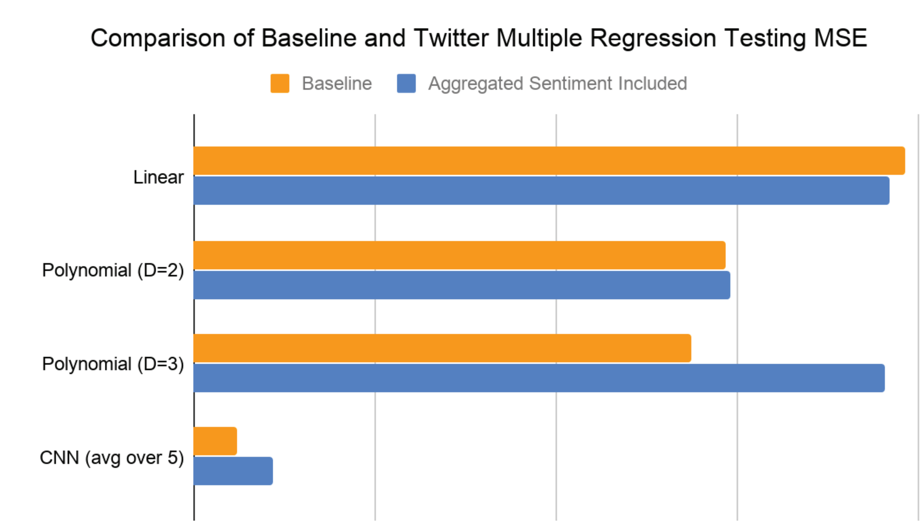

***Figure 4:** Test MSE of all models*

All models made a positive return and outperformed the random actor. The deep learning model made the most money and almost performed as well as the optimal model. We believe this is because the CNN takes into account multiple time steps worth of data in the past, as well as its increased ability to capture complexity. The CNN works well when there is sufficient enough data to train it as opposed to multiple regression that performs relatively well on less data.

Of the regression models, the linear model was able to make the greatest returns despite its higher test MSE. While the exact cause is unclear, we believe the linear model made more correct decisions during larger price jumps but missed smaller price differences. The polynomial degree 2 and polynomial degree 3, despite the discrepancy in MSE, make similar movements in stock price, resulting in similar returns.

Another interesting result is that all models actually performed better without Twitter data. Our initial analysis showed that there may not be a relationship between Twitter scores and TSLA price. Therefore, our models could be performing worse because they learn unnecessary or nonexistent relationships between sentiment and Tesla price.

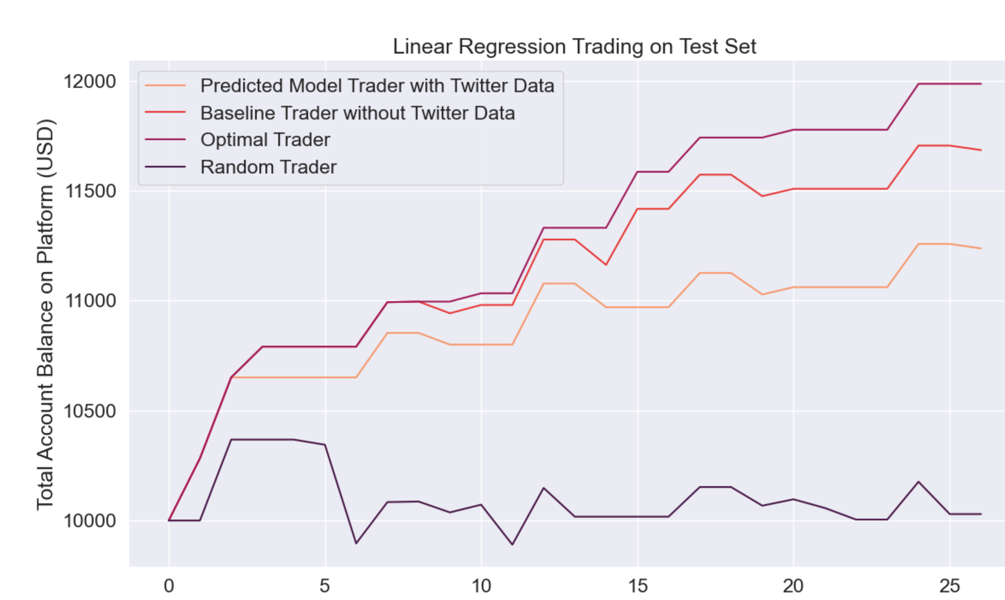

***Figure 5:** Baseline Returns: 17.58%, Twitter Model Returns: 12.71%*

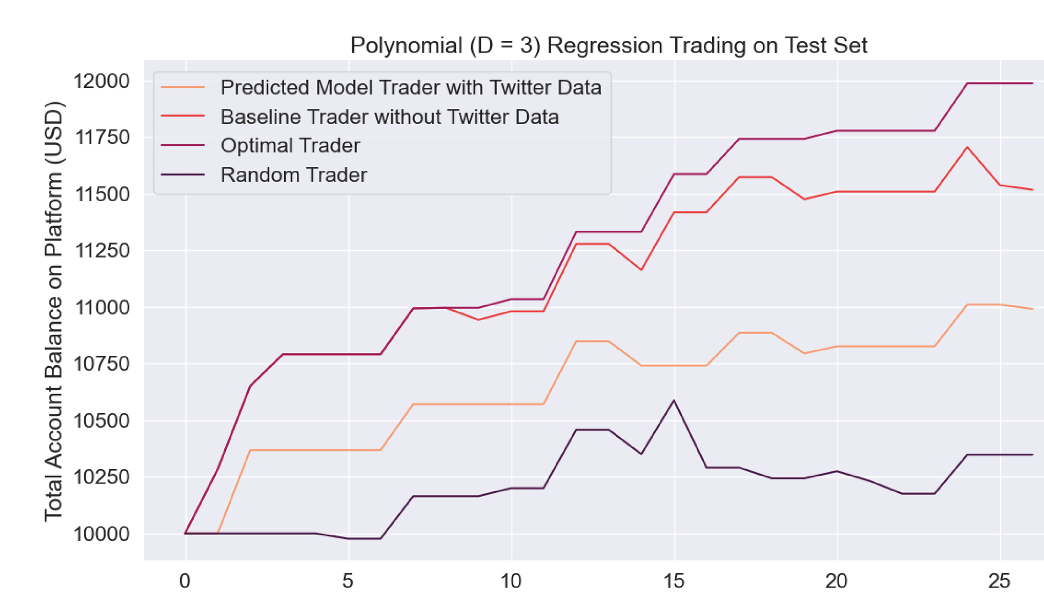

***Figure 6:** Baseline Returns: 15.12%, Twitter Model Returns: 11.06%*

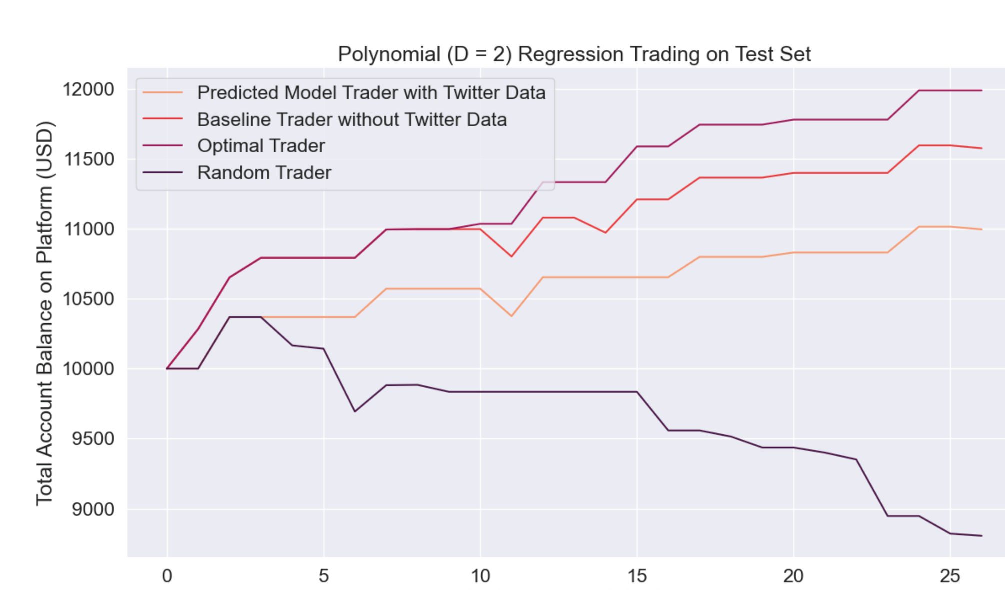

***Figure 7:** Baseline Returns: 15.78%, Twitter Model Returns: 11.18%*

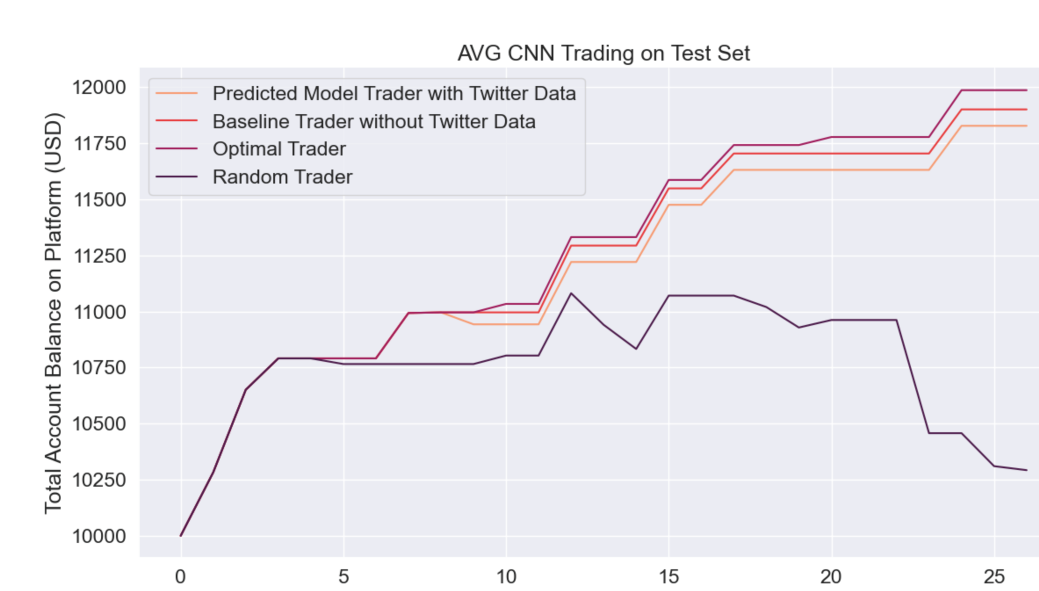

***Figure 8:** Average of 5 Train/Test results: Baseline Returns: 18.93%, Twitter Model Returns:*

Methodology

Because our goal is to predict future time step prices, we ordered our data chronologically and made the first 80% the training data and the last 20% testing data.

We use multiple regression to factor in several independent variables that could account for share price. We explored linear, polynomial degree 2, and polynomial degree 3 regression to account for varying degrees of complexity. We also tested a deep learning model (trained on 2000 epochs) because it may capture more complexity where regression cannot. While not entirely intuitive, multiple papers cite convolutional neural networks (CNN) as one of the models with the greatest accuracy for predicting stock price (*See Citations).*

To measure performance, we wrote a script that mimics trading on the testing set using the model's prediction of the next price simulating how it would be deployed and judged in real life. Our trading script provided all models 10,000 USD initially with no trading fee. We ran each model with and without Twitter data and ran them against a random actor and an optimal actor (that knows the next price).

Limitations and Future Work

1. Our trading algorithm bets everything on every prediction. This is not indicative of real world use as all financial companies will employ a degree of risk management. A future possibility would be to write a script that trades a certain amount based on the expected jump, instead of going "all in".
2. Not all investors tweet, meaning our aggregated sentiment may not be reflective of how investors feel at a given moment. We also only look at English tweets even though the market is global. In the future we might incorporate news sentiment and international tweets.
3. TextBlob's sentiment analysis may be inaccurate.. Future work may require us to create our own model that also takes into account stock terms.
4. We worked with a small time frame so our results may not generalize. In the future, we plan on increasing the time frame of our data so our results could be more conclusive.

Citations

1. Gudelek, Ugur & Boluk, Arda & Ozbayoglu, Murat. (2017). A deep learning based stock trading model with 2-D CNN trend detection. 1-8. 10.1109/SSCI.2017.8285188.