Using Data Science to Predict Stock Market Movements

Steven David Ong, John Kaspers, Chi Huen Fong, Pu Zeng

School of Information, University of Michigan

Problem Statement

Machine Learning ("ML") is playing an increasingly significant role in stock trading, and the core of a successful ML investment strategy is the application of sophisticated ML algorithms to not only predict market trends but also to delve deeply into patterns and dynamics of stock prices.

In this project, we plan to combine Large Language Models (LLM) and Reinforcement Learning (RL) algorithms for the machine to learn the techniques of earning money in the stock market with less risk tolerance.

Data Source and Data Cleaning

- Data Scrapped from Jan 2022 to Sep 2023 We extracted the comments data from StockTwits website JSON files, leveraging JSON files to access a wealth of user-generated insights and sentiments. The emojis and unintelligible symbols in the tweets were removed, and the stock symbols and dates were converted into a standard format that would align well with price data and FinBERT.
- Data from Previous Research Mukul et al.[2] used the StockTwits data from 2009 to 2020 to train a variety of NLP models. As we need to train and validate FinBERT and various RL models, consecutive data over a longer period is needed. Therefore, we use the cleaned version of this dataset for our LLM and RL training and validation.
- Yahoo Finance Yahoo Finance provides price data of the stocks and indices on a daily basis and was used as the source of the market data in our project.
- US Funds Dataset from Yahoo Finance[3] A dataset that collects different years' fund returns, which helped to compare RL model results with human experts.

Methodology

We use two methods to label the tweet data and train the corresponding FinBERT model. The sentiment scores of the FinBERT model were extracted for the RL algorithms.

FinBERT training

- FinBERT Trained with VADER Labels As FinBERT was primarily trained with formal corpus such as news, we enhanced its sentiment analysis ability on informal text such as tweets by training it with the rule-based Valence Aware Dictionary for Sentiment Reasoning ("VADER") generated labels. Therefore, the trained FinBERT can perform well on complex jargon and context-specific language prevalent in financial texts. The labels were categorized into 5 distinct levels ('bearish', 'nay', 'neutral', 'bullish', 'to the moon').
- FinBERT Trained with The Price Percentage Changes of The Next Day Different from purely sentiment analysis, we also labeled the tweets with their corresponding price change of the next day. The labels for the tweets were generated according to the price percentage changes and their standard deviation. The tweet will be labeled as "Positive" if the price percentage change is larger than one standard deviation and "Negative" if smaller than -1 $\times$ standard deviation. Others would be labeled as "Neutral".

Reinforcement Learning

- Discrete Action Space We used Deep Q-Network ("DQN"), Double Deep Q-Network ("DDQN"), Distributional Reinforcement Learning ("C51"), and Distributional Reinforcement Learning with Quantile Regression ("QRDQN") to output the target position of a single stock as the action.
- Continous Action Space Proximal Policy Optimization ("PPO") was used to output the target positions for each stock and cash.

FinBERT Training Results

We trained both FinBERT models with the tweets from 2009 to 2014 and used the tweets from 2015 to 2016 as the validation set to find the best model for the next step.

1. FinBERT Trained with VADER Labels. The FinBERT model trained with VADER labels achieved a 96.6% accuracy, which hints that the FinBERT model learned VADAR's ability on informal corpus quite well.
2. FinBERT Trained with Price Change. Our FinBert model achieved 40.6% precision for the positive labels, which is better than a random guess. Considering the chaotic nature of the stock market and tweets, it may not be easy to expect high precision like tasks in other scenarios.

(i) Classification report

```
              precision    recall  f1-score   support

           0      0.966     0.981     0.974     49099
           1      0.950     0.951     0.950     47164
           2      0.990     0.966     0.978     48896
           3      0.963     0.954     0.959     47330
           4      0.969     0.986     0.977     47721

    accuracy                          0.968    240210
   macro avg      0.968     0.968     0.968    240210
weighted avg      0.968     0.968     0.968    240210
```

(iii) Testing of model

```python
new_sentence = "Great share. to buy APPL. Tops!"
tokenized_sentence = tokenizer.encode(new_sentence, add_special_tokens=True)
input_ids = torch.tensor([tokenized_sentence]).to(device)
with torch.no_grad():
    outputs = model(input_ids)

predictions = torch.nn.functional.softmax(outputs[0], dim=-1)
predicted_class = torch.argmax(predictions).item()

print(predicted_class)
```

(ii) Confusion Matrix

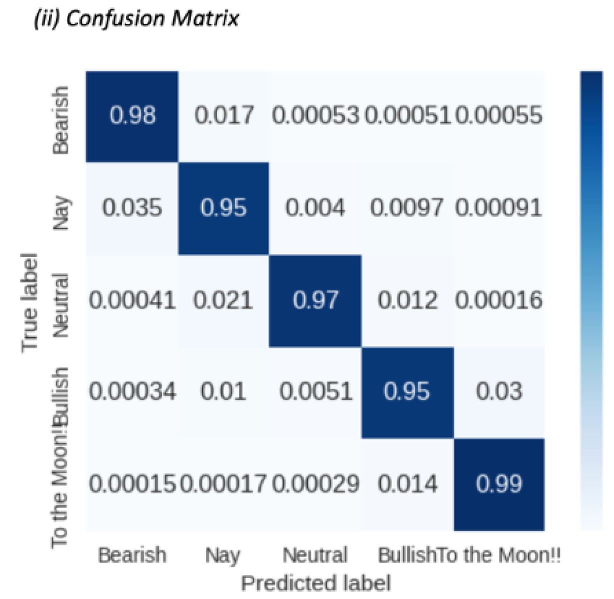

Figure 1. Evaluation Metric for The FinBERT Trained with VADER Labels

Reinforcement Learning for Single Stock Results

We use the features extracted by FinBERT and price data from 2017-2018 to train the RL algorithms and use the data of 2019 as the test set. Considering the volatile nature of RL models, we use the highest average returns and Sharpe ratio of 100 rounds as the result.

Returns(Sharpe Ratio)	DQN	DDQN	C51	QRDQN
AAPL	27.1%(1.98)	27.8%(2.01)	27.5%(1.97)	**28.0%(2.05)**
NFLX	**-1.0%(-0.05)**	-2.4%(-0.12)	-1.8%(-0.10)	-1.7%(-0.09)
AMZN	-0.1%(-0.01)	**0.4%(0.03)**	-0.3%(-0.02)	-0.4%(-0.02)
META	15.3%(1.03)	**15.5%(1.06)**	15.2%(1.03)	15.1%(1.01)
GOOG	**6.4%(0.51)**	6.2%(0.50)	6.3%(0.50)	5.3%(0.43)

Table 1. The Returns and Sharpe Ratio for RL Model with FinBERT Trained with VADAR Labels

Returns(Sharpe Ratio)	DQN	DDQN	C51	QRDQN
AAPL	27.9%(**2.05**)	27.2%(1.96)	27.4%(1.99)	**28.3%**(2.04)
NFLX	-1.9%(-0.09)	**-0.9%(-0.04)**	-1.2%(-0.06)	-1.2%(-0.06)
AMZN	-0.8%(-0.06)	0.2%(**0.10**)	0.3%(0.02)	**0.4%**(0.04)
META	16.5%(**1.10**)	15.5%(1.05)	**16.7%(1.10)**	15.7%(1.06)
GOOG	5.6%(0.43)	5.7%(0.47)	5.5%(0.44)	**6.0%(0.49)**

Table 2. The Returns and Sharpe Ratio for RL Model with FinBERT Trained with Price Change

In the discrete action space, the features trained with price change showed an advanced performance with the metric of both return and Sharpe ratio. Interestingly, DQN and DDQN perform better with the VADAR-trained features, while QRDQN performs better with the price-trained features.

Reinforcement Learning for Portfolio Management

PPO is good for continuous action space, and we believe its probabilistic nature is more appropriate for stock investment than other RL models for continuous action space, such as DDPG and TD3.

Like before, we use the data from 2017 to 2018 to train the PPO model and data from 2019 to test it. We also add a parameter "b," which controls the trading frequency for the continuous action space agent. Otherwise, it would spend most of its returns as broker fees. At each time, the agent would trade if a random number it generates is smaller than "b." In the continuous action experiment, we set b to be 0.1, which can be considered as the agent trade once in 10 days on average.

- PPO for FinBERT Trained with VADAR Labels The highest average returns and Sharpe ratios of 100 rounds achieved 30.3% and 1.46, which outperformed the S&P 500 Index and Dow Jones Index in 2019. In reinforcement learning, models are always compared with human experts, and this model beat 98.5% large cap mutual funds in the year 2019.
- PPO for FinBERT Trained with Price Change The highest average returns and Sharpe ratios of 100 rounds achieved 27.5% with a Sharpe ratio of 1.49, and beat 72% mutual fund in the year 2019.

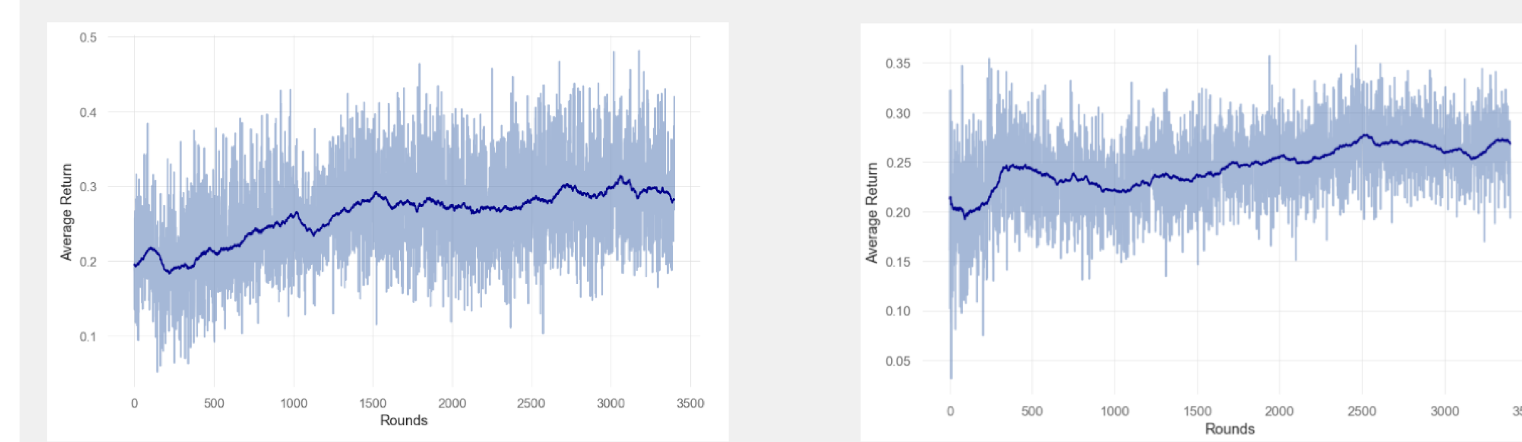

Figure 2. PPO Returns (VADER Left and Price Right)

Discussion

Though both of our features perform well with reinforcement learning models, one of the most interesting discoveries is that the sentiment scores generated by FinBert and Vader achieved a much higher return for portfolio management. Its advantage over the FinBert trained with the next-day price change in portfolio management may be caused by the fact that there is a universal standard for the sentiment scores for all the stocks with the Vader model. On the other hand, the other FinBert data is labeled with the comparison between the next day's price change and its corresponding standard deviation for each stock, so the bullish or bearish signal is not equivalent for different stocks.

Though the FinBERT trained with price change has a smaller return, it achieved the highest Sharpe ratio. We found that the FinBERT trained with price change is more likely to discover tweets followed by volatile stock prices rather than identify the bullish or bearish labels, which may help the RL models better avoid risks.

References

[1] Dogu Araci.
Finbert: Financial sentiment analysis with pre-trained language models.
arXiv preprint arXiv:1908.10063, 2019.

[2] Mukul Jaggi, Priyanka Mandal, Shreya Narang, Usman Naseem, and Matloob Khushi.
Text mining of stocktwits data for predicting stock prices.
Applied System Innovation, 4(1):13, 2021.

[3] Stefano Leone.
Us funds dataset from yahoo finance, 2021.

steveong@umich.edu, kaspersj@umich.edu, chfong@umich.edu, puzeng@umich.edu