Replacing Financial Charting with Sequence Models for Trading Stocks

Hiu Wai Terry Chan (tchan1@stanford.edu) | Songlin Qing (qing0001@stanford.edu)

Department of Computer Science, Stanford University
CS 230 Deep Learning | Taught by Andrew Ng and Kian Katanforoosh

1. The Problem

- A common practice by Wall Street discretionary traders is to predict the future direction of a stock by reading the historical price chart of stock (aka. Technical Analysis)
- This method depends on a lot of subjective judgement and often times not a repeatable process
- This process can be automated and enhanced by framing the problem as a binary classification problem using deep learning
- Given the time-series nature of the problem, we will use recurrent neural network to build a model
- Because Technical Analysis or chart reading uses only price and volume data, we will use only price and volume data and its derivatives as input features for this project

2. The Data

- Using "WIKI Prices" dataset provided by Quandl.com, a startup providing professional grade historical financial data
- Historical Price on the S&P 500 companies
- 10 years of dividend- and split- adjusted historical daily data

3. Data Cleaning and Wrangling

- **Starting Data:** 10 years worth of daily data on the price and volume for the S&P 500 index constituents, which gave us ~1.8m rows x 2 features of data

	Procedure	Motivation / Significance
1	**Aggregating the daily data into weekly data on both price and volume**	• Reduce the noise in the data • Smooth out random fluctuations of the stock • Use closing price from last trading day as weekly closing price • Sum daily volume as weekly volume
2	**Grouping the data by ticker and sort by date**	• Allow calculation of sequential weekly changes of features for the same stock
3	**Adding volatility as an extra feature**	• A popular feature in the financial A popular feature in the financial community is volatility • We use a 12-week rolling annualized volatility as our feature • A 12-week window provides enough data for a meaningful volatility calculation (which is the standard deviation of returns)
4	**Taking the log on the changes in volume and price**	• Make values less extreme and more symmetrical • Asset prices are known to follow lognormal distribution
5	**Construct time-series**	• Add past 52 weeks of price/volume/volatility change as separate columns
6	**Drop NA's**	• Drop first 1.3 years of data as they are used to calculate features (such as return and volatility) and do not include numerical features themselves
7	**Centering and standard scaling the features**	• To allow the model to train faster by standardizing different features • Lowering the significance of outlier return data • Centering the return actually has a similar effect removing the average market component from the return of the stock
8	**Transform output**	• Convert price change to up or down

- **Log return formula:** $log\ (1 + weekly\ return)$
- **Annualized Volatility formula:** $\sigma(weekly\ return) * \sqrt{52}$
- **Resulting data:** ~10 years worth of weekly data, or ~200,000 rows x 156 features of data (52 weeks of weekly data for price change, volume change, and price volatility)

4. Training, Dev, Test Sets

- **Training set:** Apr 2008 – Aug 2017
- **Dev set:** Sep 2017 – Oct 2017
- **Test set:** Nov 2017 – Dec 2017
- Note: Given we dealing with time-series data, extra cares were taken to avoid look ahead bias (ie. peaking into future data when we train our model)

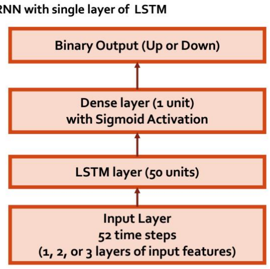
5. Input Feature Sets

- **Feature Set 1:** Price Change Only (1 Layer)
- **Feature Set 2:** Price and Volume Change (2 Layers)
- **Feature Set 3:** Price, Volume, and Volatility Change (3 Layers)

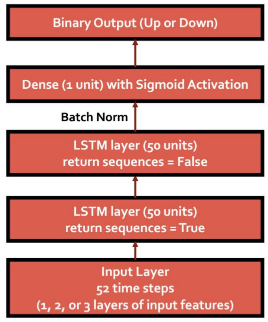
6. Sequence Models

- **Loss:** Binary cross-entropy
- **Optimizer:** Adam, beta_1=0.9, beta_2=0.999
- **Learning rate:** 0.0001 with decay 0.1
- **Initialization:** He-normal

A. **RNN with single layer of LSTM**



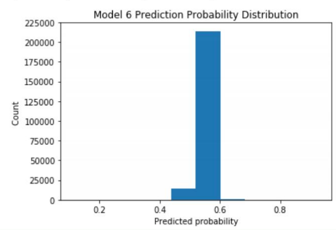
B. **RNN with two layers of LSTM**



7. Performances of Various Models

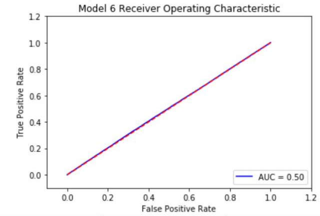
S/N	Model Description	Training AUC Score	Dev AUC Score	Test AUC Score
1	Feature Set 1 + Network A	0.5011	0.4921	
2	Feature Set 2 + Network A	0.5002	0.5119	
3	Feature Set 3 + Network A	0.5011	0.4915	
4	Feature Set 1 + Network B	0.5036	0.4984	
5	Feature Set 2 + Network B	0.5014	0.5080	
6	Feature Set 3 + Network B	0.5031	0.5154	0.4857

- Model performance on training data differs little among the models with different feature sets and network structures
- Model 1 , 3, and 4 performed worse than a random classification model on dev set
- Model 6 has the best result in dev set, thus chosen to be validated with test set. Model 6 predicted test set output with probability concentrating between 0.5 and 0.6



8. Result for the final model

- Model 6 produced an AUC score of 0.4857 in test set, worse than a random classifier
- The model predicts the outcome with probability very close to the maximum likelihood estimator (0.536 for test set) , despite differences in input values



	Predicted Down	Predicted Up
Actual Down	7	11746
Actual Up	14	14134

- Same behavior is observed on other models
- Symptom remains after ruling out code bug and adjusting learning rate, optimizer, normalization method, LSTM layers

9. Conclusion and Takeaways

- Longer term share price prediction does not have very high accuracy with only technical data (price, volume, and volatility); the potential remedy might be:
  1. adding more non-price/volume features and
  2. shorten the time frame significantly similar to the practice of high frequency trading firms (need accurate tick data)
  3. using alternative data such as social media feeds
- Our result might suggest that technical analysis used by Wall Street practitioners maybe more of an art than science (not rigorous statistically but may include other subjective "features" such as market experiences and judgement of market sentiments
- Eugene Fama's famous "Efficient Market Hypothesis" might have some merit: the weak form of EMH suggests that price information has already been taken into account by the market and hence investors cannot profit from this information

10. Further work

- Using fundamental company data (such as earnings, revenue, P/E ratio) to predict price; these data comes out in quarterly frequency and hence might not be enough for deep learning
- Incorporate alternative data (such as real time news, social media feeds); need to have accurate timestamps for these data

Selected References

1. Lo, Mamaysky, & Wang, Foundations of Technical Analysis: Computational Algorithms, Statistical Inference, and Empirical Implementation, THE JOURNAL OF FINANCE • VOL. LV, NO. 4 • AUGUST 2000, https://www.cis.upenn.edu/~mkearns/teaching/cis700/lo.pdf
2. Kai Chen, Yi Zhou, & Fangyan Dai, A LSTM-based method for stock returns prediction : A case study of China stock market, 2015 IEEE International Conference on Big Data, Oct, 2015, http://ieeexplore.ieee.org/document/7364089