DeepFX

Raman Vilkhu
Department of Electrical Engineering
Stanford University
vilkhu@stanford.edu

Sasi Madugula
Department of Neuroscience
Stanford University
sasidhar@stanford.edu

Abstract

Accurate prediction of movements in financial markets is a well-known problem of interest. When dealing with complex financial markets, there are naturally numerous variables influencing market rates, thereby making an exact, complete modeling unlikely. However, it is still evident that there exist tangible inputs that correlate heavily with market behaviours. For the purpose of this paper we split these inputs up into two categories shown by previous studies to correlate with market movements: technical indicators and market sentiment. We discuss the use of Recurrent Neural Networks (RNNs) trained with both sentiment and technical data in order to perform predictions on foreign exchange markets, specifically targeting the movement of GBP/USD. The resulting deep neural network yielded a validation set accuracy of 55%, matching the performance of similar state-of-the-art networks.

1 Introduction

The economy provides societies with access to goods and services; markets are networks that respond to prices to squash inefficiency. With the advent of computers, the internet, and interconnected global trade, modern economies are more complex than ever. What started as simple speculation on the prices of various goods in ancient Eastern trade routes is now a highly intricate stack of meta-markets, culminating in the technological feats required for high-frequency traders to conduct transactions in milliseconds.

One benefit of this added technological complexity is that data are more publicly available than ever before. News, price movements, and many other indicators are available to download at moment's notice. As we've learned in class, major improvements in computing have led to a surge in market prediction.

One rule of the market is that any inefficiency is self-correcting. The only way to make profit off of it is to exploit a feature people have not observed before. We aim to do accomplish this by combining technical analysis of stock prices (prediction based just on numerical price data) with sentiment analysis, something few people have tried before. We present in this preliminary work two different nets: one for sentiment analysis and one for technical analysis.

The input to our sentiment analysis network is a concatenated string of the top 25 Reddit World News stories for each day preprocessed to remove all special characters and converted into a numerical vector according to the GloVe embeddings. This is then fed into a recurrent neural network that predicts whether the price for the USD relative to GBP will increase or decrease the next day.

The input to our technical analysis network is 30 days worth of closing prices from 9 global stock exchanges and indices, as well as daily open, close, high, and low values from GBP-USD. We want to predict whether the stock will go up or down on day 31.

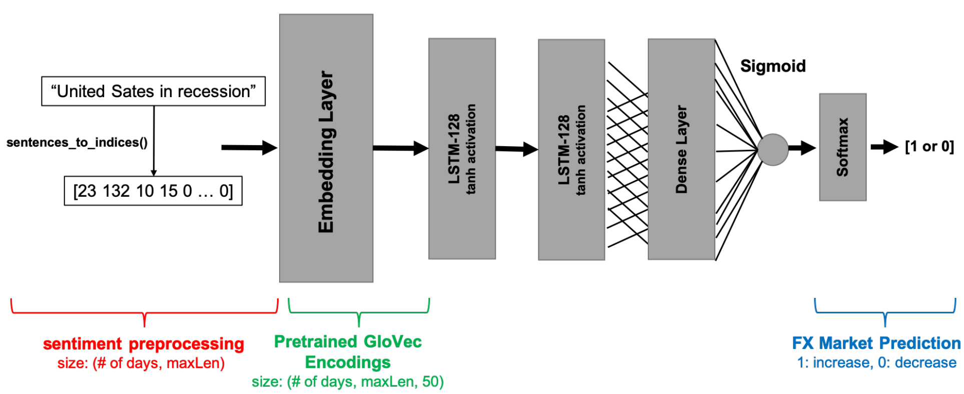
CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

2 Dataset and Architecture Design

2.1 Sentiment Analysis Network

The data set consisted of 1989 days of news data (top 25 Reddit news headlines stored as strings) and corresponding data indicating whether the USD/GBP foreign exchange price increased or decreased the following day. This data was put through a 80-20 split for training and validation, respectively. The FX data had to be pre-processed to correctly account for the one-day prediction window as well as subtracting the opening and closing price to determine the overall upward/downward movement of the currency. Additionally, the news story headlines were pre-processed to remove all special characters that did not fit the GloVec encoding scheme, and all 25 news stories were concatenated into one large string. Finally, a total input character size of 500 was taken after plotting the distribution of the news data set, and it was chosen so all inputs would be of consistent size – a requirement of the neural network. These characters were then encoded using the GloVec encoding scheme in the embedding layer of the Keras neural network. This common method of embedding words as numerical vectors to increase the accuracy of the network is one commonly used, and it was implemented because it was shown to add to accuracy in some of the discussed prior work.

The sentimental analysis architecture is based on a 4-layer LSTM-based RNN neural network, shown in the figure below. The first layer is an Embedding layer in Keras that maps the input word indices to their respective GloVec encoding. This then feeds into two 128 node LSTM layers, then a fully connected 128 neuron output layer with sigmoid activation. We use binary crossentropy loss, Adam Optimization, learning rate of 1e-4, and a batch size of 32. This design was adopted primarily from a similar project that had success with sentiment analysis using reddit data, however significant data pre-processing had to be conducted to use this resource with foreign exchange data.[3] The hyperparameters yielding the highest impact results were the batch size and activation activation type, so these were sweeped in the code to determine the optimal values.

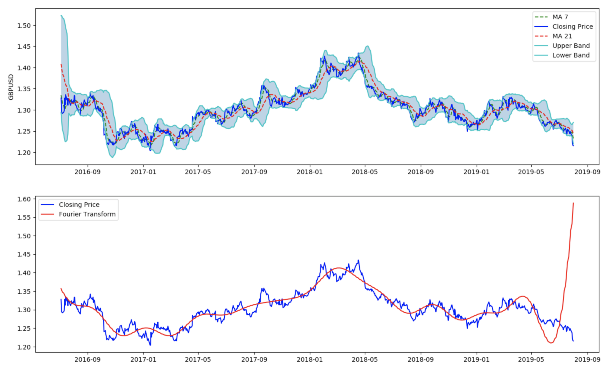


2.2 Technical Indicators Network

GBP/USD daily foreign exchange rates (High, Low, Open, and Close) were downloaded from Yahoo Finance over 8 years from 8-08-2008 to 7-01-2016 when market was open to yield 1800 data points. Additionally, we obtained the closing values for correlated assets such as currency exchanges EURO/GBP, EURO/USD, GBP/JPY, USD/JPY, GBP/CNY, and USD/CNY, as well as global market indicators such as the Dow Jones Index, NASDAQ, FTSE 100, Nikkei 225, Treasury Yield (a measure of interest rate), and VIX (a general market volatility measure).

GBP/USD closing prices were processed using conventional technical analysis techniques. For 2,000 days of closing price data we computed 7 and 21-day moving averages, Bollinger Bands, exponential moving averages, momentum, and 40-component Fourier transform. We combined these features with unaltered closing prices of the correlated assets mentioned previously.

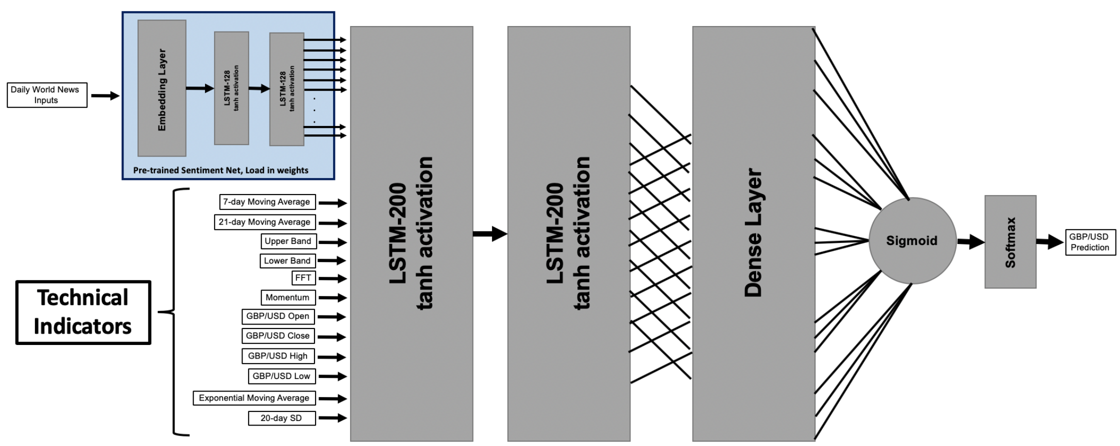
For technical analysis we used a simple architecture of a two-stack LSTM. We used defaults of learning rate=1e-4 with batch size=32, also default. This is because the features make a much bigger difference than LSTM architecture for FOREX prediction according to the literature. We search over



the same hyperparameter range as for the sentiment network above and final values are found in the code.

2.3 Combined Network

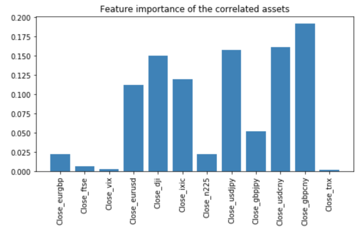
The final network architecture was a combination of the two independently designed networks, adding to the novelty of the approach. The pre-trained sentiment RNN was connected in series, with the output activations serving as inputs, into the technical indicators RNN, see figure below.

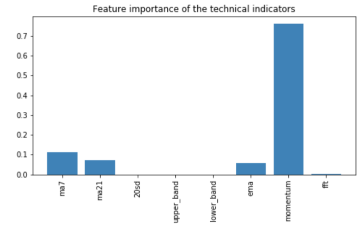


For the combined network, a similar approach was used to tweak the hyperparameters as that of the technical network, yielding the finalized hyperparameter values found in the code. Specifically, the batch size and number of neurons made the biggest impact on the overall accuracy of the network, so these were swept in the code to find the optimal values (batch size = 32 and number of neurons = 200).

2.4 Assessing Feature Performance

We implemented gradient-boosted decision trees using the XGBoost module in Python in order to assess importance of the correlated asset and technical features. Boosting works by stacking many models, usually decision trees, to fit data in order to reduce error. We found a wide disperity between the impact of features on the XGB prediction:

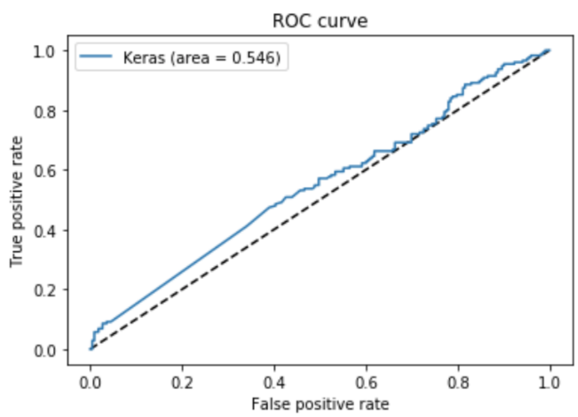




3 Results

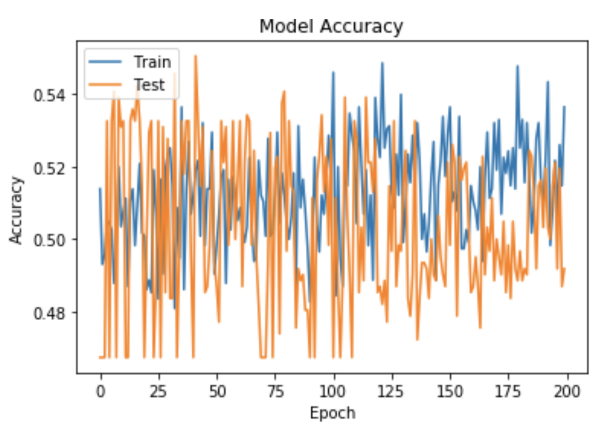
3.1 Sentiment Analysis Network

For the sentiment analysis, we achieve the following 66% and 56% training and test accuracy, respectively. Perhaps a more useful analysis is the Area Under Curve (AUC) analysis, shown below. Here, we see an average of 4.6% better performance than the expected 50%. This essentially shows that the performance of our binary classifier (whether the GBP/USD exchange rate went up or down) is better than chance. While it is not massively greater than 50%, ROC in the achieved range is considered good for stock market sentiment – a turbulent, unpredictable prediction space.



3.2 Technical Indicators Network

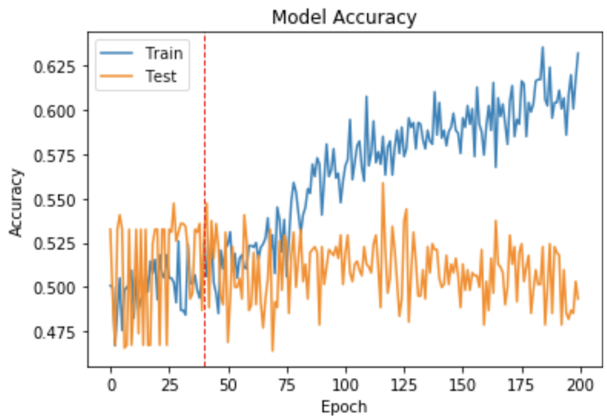
For technical analysis, since the literature achieves a max success rate of about 55 percent with LSTMs, we used that as a metric for sensible training. We do not achieve quite the same accuracy with our technical model, but with a little more fine tuning of features can get close to 54 percent accuracy on some runs. An example of a typical technical analysis training run is shown below:



3.3 Combined Network

For financial analysis, we have two outcome measures: the absolute accuracy of our network and accuracy required to make a profit. The precise value varies depending on the market and trading broker, so precise back-testing is required to validate the performance of the network for actual profit.

Our test accuracy peaks at around 55 percent for a small number of epochs (40) and decreases and training accuracy increases. Note the high variability, also a standard observation in FOREX prediction (7).



4 Analysis and Discussion

Since our network achieves the widely acknowledged barrier of 55 percent, others have already back-tested predictions at that rate. Fabrice Daniel (7) showed that for GBP/USD, an accuracy rate of 55 percent yields 10 percent annual profit a year over 3 years. This is of course highly dependent on the underlying regime of the market and might not generalize well to.

Therefore our model MAY be viable in a real marketplace. However there are many directions we could take to improve our model.

5 Next Steps

- Collect more headline data from sources other than Reddit.
- Use finance-specific embeddings.
- Find more assets correlated to national GDP and run NLP on
- financial reports
- Make half-day or quarter-day predictions (which would also help add more training data)
- Try adding GAN
- Make multi-class softmax predictions to discern magnitude of increase or decrease.
- Generalize results to other currency markets to diversify portfolio

6 Code

The code for this project can be accessed on github: `https://github.com/ramanvilkhu/deepFX`

7 Contributions

Raman worked on the sentiment analysis network design and code. Sasi worked on the technical indicators network design and code as well as the combined network code. Both team members split the work on the final report and poster design.

References

[1] V. S. Pagolu, K. N. Reddy, G. Panda and B. Majhi, "Sentiment analysis of Twitter data for predicting stock market movements," 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), Paralakhemundi, 2016, pp. 1345-1350.

[2] Alex Davies. A Word List for Sentiment Analysis of Twitter. http://alexdavies.net/2011/10/word-lists-for-sentiment-analysis-of-twitter/

[3] https://www.dlology.com/blog/simple-guide-on-how-to-generate-roc-plot-for-keras-classifier/

[4] Ni, Lina, et al. "Forecasting of Forex Time Series Data Based on Deep Learning." Procedia computer science 147 (2019): 647-652.

[5] Oncharoen, Pisut, and Peerapon Vateekul. "Deep learning for stock market prediction using event embedding and technical indicators." 2018 5th International Conference on Advanced Informatics: Concept Theory and Applications (ICAICTA). IEEE, 2018.

[6] Using the Latest Advancements in Deep Learning To Predict Stock Price Movements Boris B - https://towardsdatascience.com/aifortrading-2edd6fac689d

[7] Daniel, Fabrice. Deep Learning for Forex Trading. https://medium.com/lusis-ai/deep-learning-for-forex-trading-ba5d466e92cb.