Forecasting Currency Exchange Rate Variations with News Headlines using LSTM Models

Nianmin Guo
Wanhui Han
{guo.ni,han.wa}@husky.neu.edu
Northeastern University

1 Introduction

Using Natural Language Processing (NLP) methods to predict financial market movements has been an interesting topic for the past decade. Many researchers have proposed sentiment analysis models for time series predictions. In this paper, we want to use sentiment analysis to predict the movement of currency. Foreign exchange markets, unlike other markets, have trading pairs available without limitations of timezone. Thus, variations of market due to external factors can be tracked seamlessly. In addition, variations on foreign exchange markets would affect the entire nation of a trading pair, instead of a specific company. In this paper, we want to evaluate a model based on historical news and currency exchanges data to show the correlation between sentiments factor and currency exchanges.

2 Related Works

Various researches has been done on analyzing sentiments of natural languages, with or without the context of a market. One group utilizes sentiment analysis and neural networks to predict the variations of the market, while the sentiment used is of general moods (*happy, alert*, etc.) Bollen et al. in 2008 [2] analyzed the moods of tweets using mood tracking modules for sentiment analysis and attempted to enhance the accuracy a of the Dow Jones Industrial Average (DJIA) predictor by feeding the sentiment of the twetterverse on the day of. They opted to use a Self-organizing Fuzzy Neural Network [4] and achieved 86% accuracy, compared to a 73% accuracy achieved without the sentiment information.

In 2014, Nassirtoussi et al. [3] implemented a SVM-based algorithm that predicts the foreign exchange market based on news headlines. This algorithm extracted and grouped features of the news headlines in 2-hour windows with time-shifted USD-EUR exchange rate information. The semantic and sentiment information of each headline is represented by SentiWordNet [1] scores of WordNet hypernyms [7]. The authors has also employed feature reduction within time groups to decrease the dimensions required for training. The algorithm achieved 83.33% accuracy for predicting future exchange rate variations. However, words not in WordNet or SentiWordNet were dropped, creating the possibility that potentially key information (e.g., Entity Name) is lost. This approach also suffers from ambiguity raised from entity names (e.g., *Caterpillar* the machinery company being compiled as *caterpillar* the insect), causing inaccuracies in sentiment ratings.

In recent years, Long Short Term Memory (LSTM) based recurrent neural networks has been popular in Natural Language Processing fields for its ability to hold long-term dependency information. In addition, attention-based LSTMs can enhance a market prediction network by valuing inputs causing more significant changes more. In 2018, Liu [5] proposed a prediction algorithm that takes in financial news headlines and predicts the stock prices of specific companies. Self-trained word embeddings were used instead of pre-trained models so that more business- and finance-related terms could be captured. The proposed algorithm employed several attention-based LSTM layers to select the “most variable" news and dates in terms of stock prices. This approach reached 66.5% accuracy, approaching state-of-the-art levels. However, the preprocessing steps in this research had categorized news articles by companies and had conducted the training separately. This approach might not be applicable in real-life or foreign exchange market settings since news irrelevant of a nation can still affect their rates against other currencies.

3 Data

3.1 Data Sources

For news headlines, *All-The-News* dataset from Kaggle[8] is used. It has 146,150 records including title, content, published dates from various data sources, such as New York Times, Reuters, and Washington Post, etc.

For foreign exchange rate data, we used Open Exchange Rates API and gathered data from from 2016-01-01 to 2017-07-07 (552 records in total). Trading pairs of major international currencies with the United States Dollar is used (Euro `[EUR]`, Great Britain Pound `[GBP]`, Chinese Yuan `CNY`, and Japanese Yen `[JPY]`). With the development of cryptocurrency markets we have also obtained the USD to Bitcoin `[BTC]` rates for prediction. Given its highly volatile nature, it might be interesting to observe the prediction differences produced by the neural network.

4 Methods

4.1 Data Preprocessing

4.1.1 News Headlines

For the news headlines, the following preprocessing is performed:

1. Removed irrelevant fields, such as *Publisher*, *Title*, and *URL*.
2. Clean up article content, including line breaks `\n`, `\r`, punctuation, and stopwords. Numbers are replaced with token `<NUM>`.
3. Remaining words are then converted to lower case and then tokenized

The raw news dataset is noisy as news titles are not limited to only the political and financial categories. Here we design a greedy algorithm in combination with trading pair-specific knowledge and SentiWordNet [1] absolute scores to select top 10 news titles for each day, in order to unify the feature dimension that goes into the LSTM time series model and reduce noise for news titles. The news selection algorithm for USD-GBP is shown in Figure 1.

4.1.2 Exchange Rates

For the currency exchange data obtained, the daily and weekly Return based on last days' data is calculated. The calculated rate of change is then translated into a 4-element class. There were cases where the daily return was 0%, and those dates are consequently removed.

4.2 Models

We have determined to use LSTM-based models that takes daily news headlines selected by the mentioned greedy algorithm and date as input and variation classes as output. For the numerical representation of headlines, we have used both integer vocabulary approach as well as the `word2vec` word embeddings [6].

Because we are dealing with time series data, we were not able to implement the k-fold algorithm directly as random sampling will disturb the sequence order. But we used a sliding window technique to avoid overfitting. We split the dataset into $2k$ windows, we use the first k windows as train dataset and use $k, k+1$ as test dataset. Figure 3 shows an example window split for $k = 5$.

4.2.1 Name to Entity with Attention-Based LSTM

The name to entity feature vector maps words to integers. The input layer takes in the dense representation, which will be converted to one-hot representation in the next embedding layer. Dropout layers are added before and after attention layer to avoid overfitting. A fully connected layer with `relu` activation function comes before the final output layer with linear activation function.

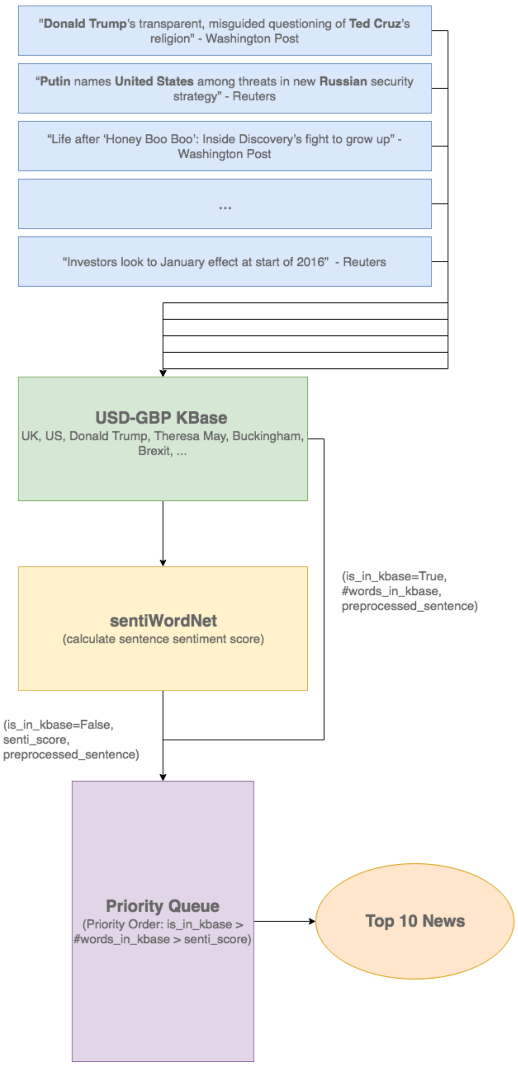


Figure 1. News Selection Process for USD-GBP, with SentiWordNet Scores and Knowledge base for each currency pair.

Change	Tag
<= -2.5%	NL
(-2.5%, 0%)	NS
(0%, 2.5%)	PS
>= 2.5%	PL

Figure 2. Exchange Rate Variation Class Table

4.2.2 Name to Entity with LSTM

Similar to the architecture in 1.1, but the attention layer is removed.

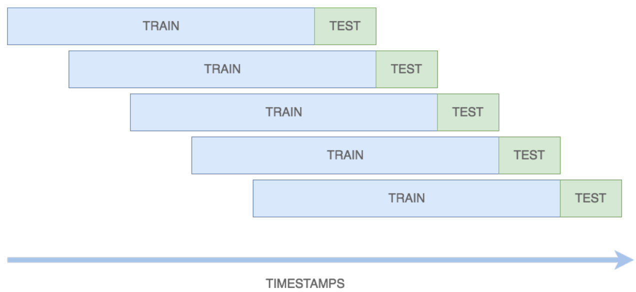


Figure 3. Implemented k-fold Split, $k = 5$

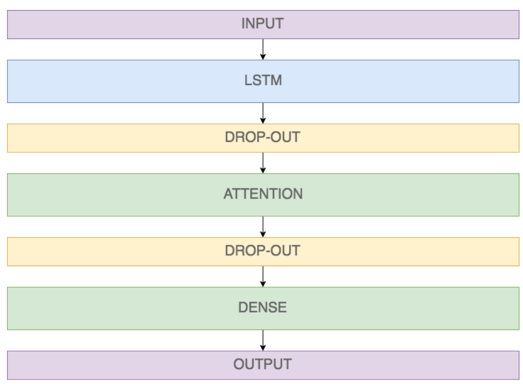


Figure 4. General Network Structures for the 4 Models

4.2.3 Word2Vec with Attention-Based LSTM

Similar to the architecture in 1.1, but the embedding layer is removed. Tensors from the input layer will be sent to the LSTM layer directly.

4.2.4 Word2Vec with LSTM

Similar to the architecture in 1.3, but the attention layer is removed.

4.3 Training

To ensure consistency, training of all the LSTM networks are done using the same hyperparameters. Under each k-fold, each network went through 100 epochs of training with a learning rate of 0.0025. Mean squared error is used for loss calculation, and the class-accuracy is used for accuracy calculation. The trained model is then evaluated using the testing set of the fold. We have chosen $k = 7$ in all training sessions, and with 5 trading pairs the training and evaluation of the models took approximately 3 hours on a Nvidia Tesla P40 Cloud instance.

5 Results

The resulting 7-fold mean-accuracy measures on each model are shown below. All baseline measures are generated by randomly assigning classes to inputs.

Pair	Feature	Model	Acc	Baseline Acc
EUR	INT	ATTENTION	0.5617	0.2444
EUR	INT	LSTM	0.5617	0.2477
EUR	VEC	ATTENTION	0.5464	0.2442
EUR	VEC	LSTM	0.5466	0.2363
GBP	INT	ATTENTION	0.471	0.2741
GBP	INT	LSTM	0.471	0.2625
GBP	VEC	ATTENTION	0.4671	0.2741
GBP	VEC	LSTM	0.5057	0.1776
CNY	INT	ATTENTION	0.575	0.282
CNY	INT	LSTM	0.5751	0.2893
CNY	VEC	ATTENTION	0.5531	0.249
CNY	VEC	LSTM	0.5604	0.2967
JPY	INT	ATTENTION	0.4982	0.271
JPY	INT	LSTM	0.4981	0.238
JPY	VEC	ATTENTION	0.4872	0.2161
JPY	VEC	LSTM	0.4871	0.2673
BTC	INT	ATTENTION	0.2286	0.25
BTC	INT	LSTM	0.2321	0.2428
BTC	VEC	ATTENTION	0.2893	0.2464
BTC	VEC	LSTM	0.2857	0.2785

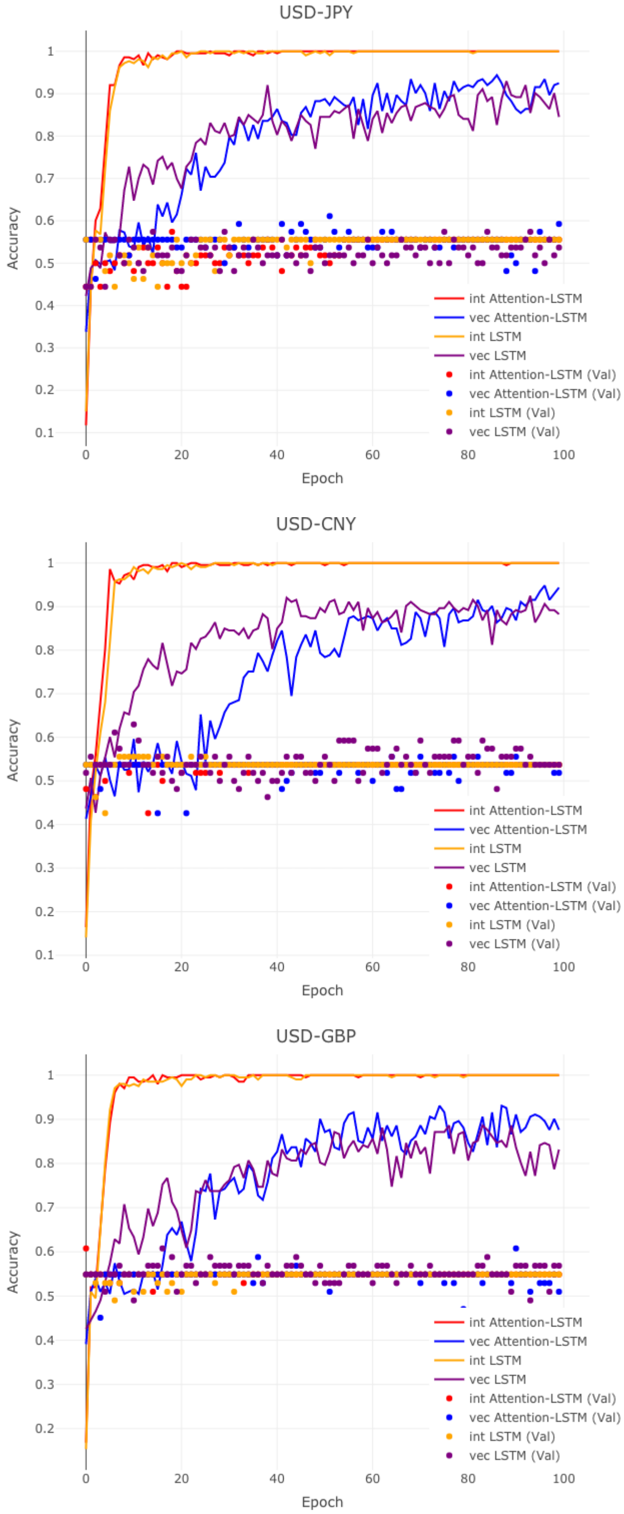
Figure 5. Results Table using mean accuracy of 7-fold validation.

The accuracy measures has performed better than baseline in all major currency trading pairs, while accuracy of `USD-BTC` pair is similar to the baseline measure. This might be caused by the high volatility of the cryptocurrency and a lack of resolution in out currency data. We were also able to observe that the training data is quickly overfitted in integer dictionary models compared to word2vec models.

6 Conclusion

We performed sentiment analysis with news titles and predicted Forex movements with LSTM models. We looked into 5 Forex trade-pairs and compared the co-dynamics among them. We invented a greedy algorithm using knowledge base and sentiWordNet to reduce input news titles noise and unify number of features for timesteps. We constructed feature vectors with both name to entity and word embedding. We started with basic LSTM model and later improved it by adding the Attention mechanism. We found that word embedding in general performs better than name to entity. The Attention mechanism does not always improve prediction accuracy, but was able to achieve higher accuracy for certain pairs.

In the future, we hope to achieve higher prediction accuracy by experimenting with more precise Forex data where timesteps are smaller. We also plan to investigate more into other time sequence models, such as oh-LSTM [**?**] and GRU



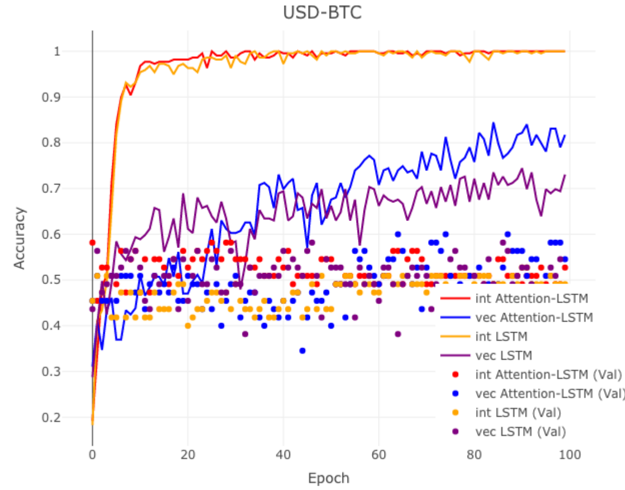
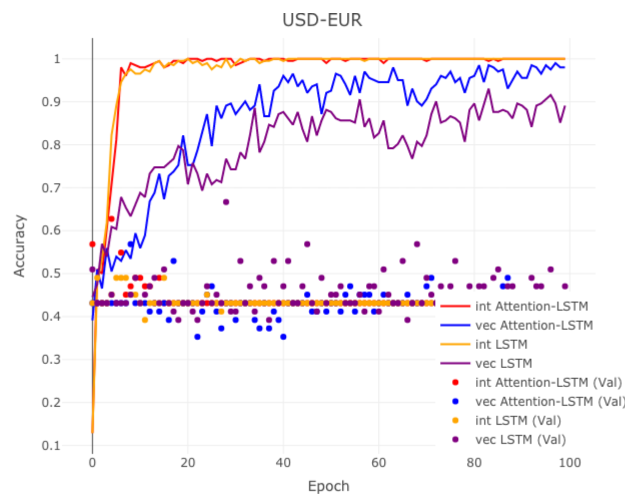


Figure 6. Training Results in Each Trading Pair. The lines represent training set results while the dots represent validation set results.



and expand our current financial knowledge base to support more trading pairs.

References

[1] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. [n. d.]. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining *(Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10))*. European Languages Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2010/pdf/769_Paper.pdf

[2] Johan Bollen, Huina Mao, and Xiaojun Zeng. 2011. Twitter mood predicts the stock market. *Journal of computational science* 2, 1 (2011), 1–8.

[3] Arman Khadjeh Nassirtoussi, Saeed Aghabozorgi, Teh Ying Wah, and David Chek Ling Ngo. 2015. Text mining of news-headlines for FOREX market prediction: A Multi-layer Dimension Reduction Algorithm with semantics and sentiment. *Expert Systems with Applications* 42, 1 (2015), 306–324. https://doi.org/10.1016/j.eswa.2014.08.004

[4] Gang Leng, Girijesh Prasad, and Thomas Martin McGinnity. 2004. An on-line algorithm for creating self-organizing fuzzy neural networks. *Neural Networks* 17, 10 (2004), 1477–1493. https://doi.org/10.1016/j.neunet.2004.07.009

[5] Huicheng Liu. 2018. Leveraging Financial News for Stock Trend Prediction with Attention-Based Recurrent Neural Network. (2018).

[6] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. [n. d.]. Distributed representations of words and phrases and

their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[7] George A. Miller. 1995. WordNet: a lexical database for English. *Commun. ACM* 38, 11 (1995), 39–41. https://doi.org/10.1145/219717.219748

[8] Andrew Thompson. 2017. All the news: 143,000 articles from 15 American publications. =https://www.kaggle.com/snapcrack/all-the-news.