

Applying Recurrent Neural Networks for Multivariate Time Series Forecasting of Volatile Financial Data

DOUGLAS GARCIA TORRES

HONGLIANG QIU

dgt@kth.se | hqiu@kth.se

January 16, 2018

Abstract

Financial time series are one of the most difficult types of data to forecast due to its high volatility. Moreover, in recent years there are additional factors incrementing this volatility, like the low latency of rumor and information spread through all kind of communication networks available, e.g. Twitter and Facebook. On the other side, thanks to the improvements of hardware and software capabilities, we are able to use more data and advanced techniques like Deep Learning to generate more accurate predictions. In this research, we analyze the motivations behind applying a particular kind of Artificial Neural Networks, namely Recurrent Neural Networks (RNNs), to generate time series predictions of volatile financial variables making use of multivariate related data. In particular, we focus on two specific types of networks; Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU). While the first type is commonly used for time series forecasting, the second is more novel and presents interesting differences with the former. Our experimental results show that GRU can be more suitable to use due to their training time performance and can yield similar accuracy to LSTM. However, our results also show no significant performance difference with simpler traditional forecasting models.

Contents

1	Introduction	3
2	Theoretical Framework	3
2.1	Long Short Term Memory (LSTM)	3
2.2	Gated Recurrent Unit (GRU)	4
3	Related Work	4
4	Research Questions	4
5	Research Methodology	4
6	Results and Analysis	6
6.1	Long Short Term Memory (LSTM)	6
6.2	Gated Recurrent Unit (GRU)	6
6.3	ARIMA with Dynamic Regression	7
6.4	ARIMA (without Dynamic Regression)	7
7	Discussion	7

1 Introduction

Recent research suggest promising results using Deep Learning Neural Networks models for financial time series predictions in contrast with traditional forecasting models (e.g. ARIMA)[1]. Between the different types of Neural Networks, related work proposes that Recurrent Neural Networks are the most suitable models for predictions and time series forecasting [2]. More specifically, Long Short Term Memory networks are the type of RNN that is often being tested. Nevertheless, the architecture of LSTM models can be complex and their results can be often disappointing. Therefore, considering that there are other types of RNNs, like Gated Recurrent Unit networks, this project investigates how these models perform in comparison with LSTM and a widely used traditional forecasting ARIMA (Auto-regressive Integrated Moving Average) models. In this work, we analyze and compare the use of LSTM and GRU models to predict financial time series data. To test such differences, we implement these neural networks to predict one of the most volatile variables nowadays in financial markets: the price of the Bitcoin crypto-currency (BTC). We perform a comparison between the predictions generated by these models, and also, against the results we get from forecasting with ARIMA models.

2 Theoretical Framework

A Recurrent Neural Network is a class of Artificial Neural Network where connections between neurons form a directed cycle. They can be used in many fields where the data can be presented as a sequence, which in theory makes them perfectly suitable for predicting daily prices of financial assets. The basic difference with the simplest type of Artificial Neural Network (Feed Forward Network) is that instead of having only neurons with connections from the input to the output, it also has neurons with connections from the output, again to the input. This additional connection can be used to store information over time providing the network with dynamic memory [3]. In RNNs, the current state of the network is a function of its previous steps. Therefore, unlike feed-forward networks, they can use the information from the past to handle and predict sequential data. RNNs also have the property of being extremely deep, which is a drawback when it comes to train a neural network as they are more often susceptible to the vanishing gradient problem[4].

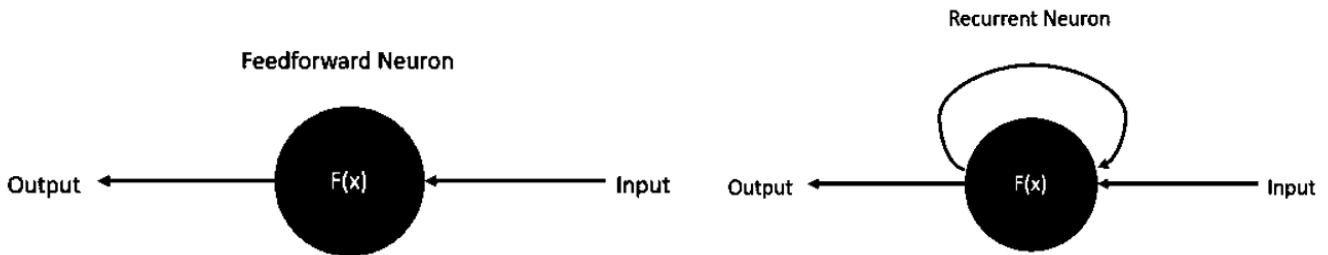


Figure 1: As illustrated in [5] the difference between feed-forward and recurrent neurons

2.1 Long Short Term Memory (LSTM)

In 1997, a new variant of RNN was proposed[6] to address the vanishing gradient problem. Long Short Term Memory solves this issue by controlling the way information flows in RNNs by introducing the concept of gates: one forget gate decides what information is going to be remembered/thrown away for the new state, an input gate determines which portion of a specific point in time is to be added, and an output gate that controls what is going to be the next output. These gates along with a memory cell, let the RNN to flexibly forget, memorize and expose data. This way, LSTM are designed to remember information of long sequences (long periods of time) and are widely used for a large variety of problems. However, its complex architecture may suggest that they may not be the most efficient network structure[7].

2.2 Gated Recurrent Unit (GRU)

Recently, simplified variations of LSTM have been studied and proposed. One of them is called Gated Recurrent Unit and was introduced in 2014[8] as a simpler model originally proposed for statistical machine translation. It introduces an “update gate” as a combination of LSTM’s input and forget gates:

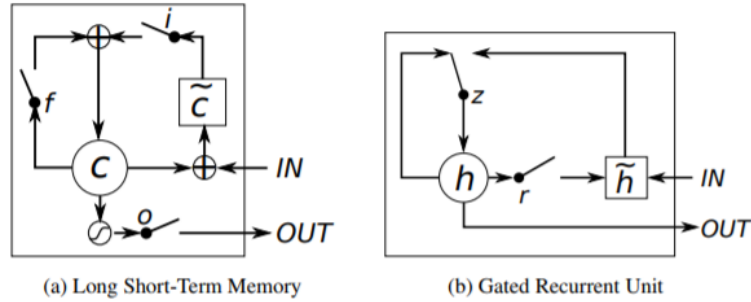


Figure 2: As illustrated in [6, Fig. 1] LSTM and Gated Recurrent Units. In (a), i , f and o , are the input, forget and output gates respectively. C and \tilde{C} denote the memory cell and the new memory cell content. In (b), r and z are the reset and update gates respectively. h and \tilde{h} are the activation and the candidate activation

Various studies[7] [9] [10] [11] [12] comparing LSTM and GRU networks have reported mixed results. However, none of the presented experiments have dealt with volatile financial data.

3 Related Work

In 2015, G. Batres-Estrada[13] tested a Deep Belief network coupled with a multi-layer perceptron on financial data. After his experiment with 134.500 records of only closing prices from S&P 500, Batres-Estrada showed satisfactory results with not more than 1000 training iterations and 47.11% of test classification error. In the discussion, Batres-Estrada mentioned as the main points of future work for improvement, augmenting the dataset with related data, testing different networks, and solving regression instead of classification. In 2016, S. McNally[14] tested a Bayesian optimized RNN and an LSTM network on a classification task to predict the direction in the price of Bitcoin. His results showed a better accuracy of 52% and a lower regression error of 8% when using the LSTM network. In addition, he showed that these deep learning models outperformed the forecasts of a simple ARIMA model without regressors. In 2017, X. Qian [1] performed a comparative analysis between ARIMA without regressors, a Multi-layer Perceptron (MLP), and Support Vector Machines (SVM) to predict financial stock market data. In this study, Qian shows a slightly higher accuracy of SVM models over the ARIMA and the MLP models, and encourages the research and experimentation with LSTM models.

4 Research Questions

This study aims to answer three questions: 1) Is there one model that outperforms the others in terms of accuracy when predicting volatile financial stock prices using multivariate data? 2) Is there any performance difference in terms of resource consumption (i.e. training time) when training these models? 3) Are these deep learning models more accurate than the traditionally used forecasting ARIMA models?

5 Research Methodology

As our research questions can be answered based on quantitative measures, this research is designed as quantitative experiment. Nevertheless, existing knowledge and literature was studied to understand the differences, motivations and advantages behind the design and structure of Recurrent Neural Networks. In addition, related work experiments to specifically predict financial data using Deep Learning approaches

were analyzed. Subsequently, the LSTM and GRU variants of RNNs were implemented, and after running several experiments for hyper-parameter tuning, comparisons were made in terms of accuracy (RMSE) and in terms of training time. Finally, a simple ARIMA model, and an extended ARIMA model with Dynamic Regression were implemented to obtain computationally optimized automatic forecasts to compare with the previous results. The following tasks were carried out:

Work environment setup: the work environment consist on TensorFlow and Keras Machine Learning frameworks running on an Intel 4-core i7 CPU processor at 3.5GHz and an NVIDIA QUADRO M1000 GPU with 2 GB of memory.

Data collection: we considered the collection of various sets of related financial data like crypto-currency market variables, fiat currencies exchange data, stock markets data, and commodities prices. In addition, as sentiment analysis brings powerful indicators nowadays, we considered the suggestions of the work of Matta et al.[15], and included the Google trends[16] indicator of interest over time of the keyword “Bitcoin”. More details on the data sources are mentioned in the next section.

Data pre-processing: this task involved cleaning, formatting, completing missing values and joining the datasets on the time dimension (date).

Model implementation: LSTM and GRU models were be studied, analyzed and implemented. In addition we used the built-in ARIMA model function in the R programming language.

Time series forecasting: with each model, we generated predictions for the month of September, 2017.

Model tuning: a simple Grid search optimization algorithm was performed to find the most suitable parameters that yield the most accurate results from each implemented model. For each RNN variant, 320 different combinations of hyper-parameters were tested. The considered parameters to test were: the number of iterations, number of features, batch size, number of recurrent layers, number of neurons per recurrent layer, and the drop-out rate. The 10 most accurate models were selected considering the RMSE in 31 days, in 7 days and in 1 day of predictions. Then, the most accurate model was selected for both LSTM and GRU variants.

Model comparison: the main goal of this phase was, for each model, make a quantitative comparison based on the accuracy of the predictions against real data using the Root Mean Square Error (RMSE) metric. In addition, we measured the time required for training each model.

The collected dataset consist of 1615 records of daily data from the beginning of May 2013 until the end of September 2017 with 293 features composed as follows:

Dates: 4 features corresponding to year, the month (i.e. a number from 1 to 12), the day of the week (i.e. from 1 to 7) and the day of the month.

Trends: the Google trends interest over time indicator for the keyword “Bitcoin”. In a scale from 1 to 100, it represents the worldwide search interest relative to the highest point of interest in a given period. A value of 100 is the peak popularity for the term. A value of 50 means that the term is half as popular. Likewise a score of 0 means the term was less than 1% as popular as the peak.

Crypto-currencies: 108 features of daily data composed by opening, closing, high, and low prices, volume of transactions (OCHLV) and market capitalization of the 18 most traded crypto-currencies in the market (including Bitcoin). Collected from Bitcoin Charts Application Programming Interface[17].

Fiat currencies: data from the European Central Bank with daily exchange closing rates of 30 of the main (most traded) fiat currencies against the European Union currency (EUR). Collected from Quandl[18].

Stock markets: 138 features of daily data with open, close, adjusted close, high, and low prices, and volume of transactions of 23 of the main stock market indexes. Collected from Yahoo Finance[19].

Commodities: daily data with closing prices in USD of a list of the 13 most traded commodities. Collected from Quandl[18].

6 Results and Analysis

Given a feature space of size F , a drop-out rate R , and the integer values N , M corresponding to the number of hidden layers and the number of neurons for each hidden layer, the implemented RNN models are sequential networks with the Nesterov Adam optimizer[20] and a default Keras 0.002 learning rate. Both networks are composed as follows:

- 1 dense layer with a linear activation function and an output channel of F units
- N recurrent layers with M neurons, an hyperbolic tangent activation function and a drop-out rate R ,
- 1 dense layer with a linear activation function and an output channel of 1 unit

6.1 Long Short Term Memory (LSTM)

The most accurate LSTM model obtained from the experiments, consists of 1 recurrent layer with 100 neurons, a drop-out rate of 0.3 and a feature space of 11 variables. The RMSE obtained (for the 31 days of the test set) was of 272.96 BTC/USD with a training time of 61 seconds in 100 iterations.

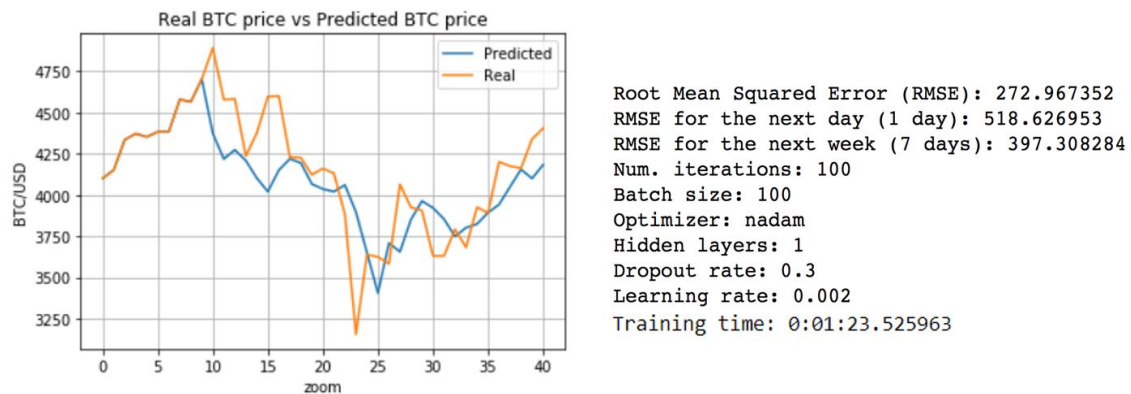


Figure 3: LSTM model predictions compared with the real values of the BTC/USD price in September 2017

6.2 Gated Recurrent Unit (GRU)

The most accurate GRU model obtained from the experiments, consists of 1 recurrent layer with 500 neurons, a drop-out rate of 0.1 and a feature space of 11 variables. The RMSE obtained (for the 31 days of the test set) was of 274.02 BTC/USD with a training time of 42 seconds in 100 iterations.

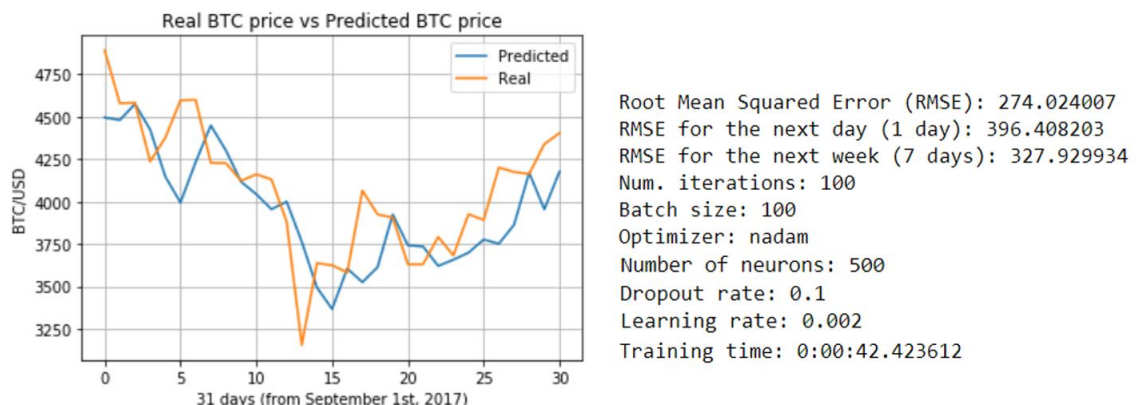


Figure 4: GRU model predictions compared with the real values of the BTC/USD price in September 2017

6.3 ARIMA with Dynamic Regression

The automatic ARIMA function of the FPP R package [21] extended with dynamic regression, generated a forecast for 31 days with a RMSE of 255.9 BTC/USD using 11 features from the dataset.

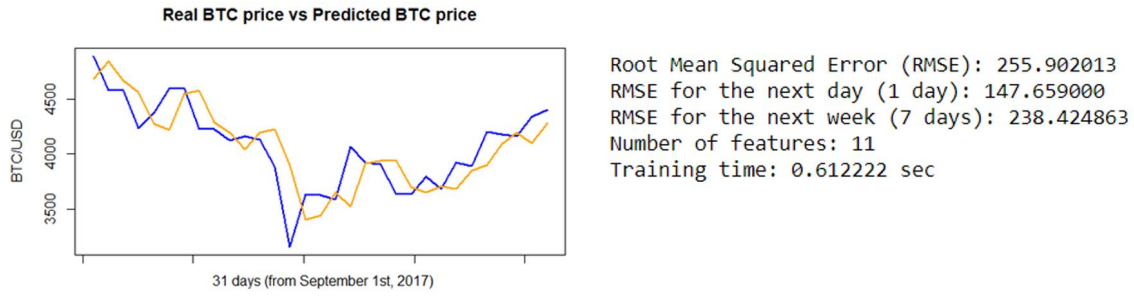


Figure 5: ARIMA with Dynamic Regression model predictions compared with the real values of the BTC/USD closing prices of September 2017

6.4 ARIMA (without Dynamic Regression)

It is relevant to point out that the results of the automatic ARIMA function without regressors, i.e. with a feature space of size 1 for the previous BTC/USD closing prices, generated a forecast considerably less accurate with a RMSE of 1196.19 BTC/USD.

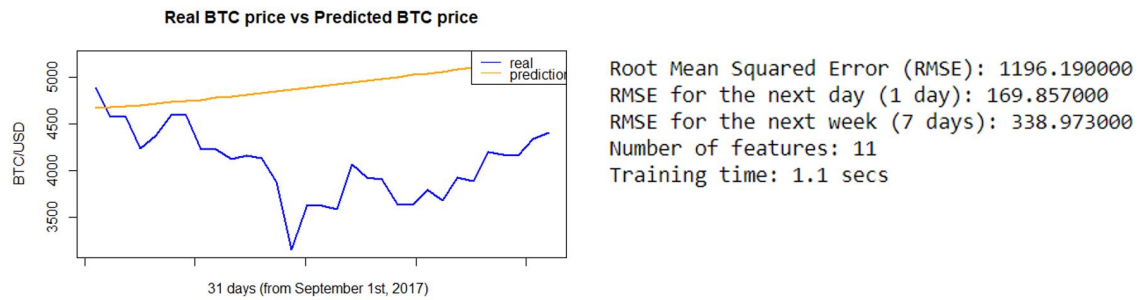


Figure 6: ARIMA model predictions compared with the real values of the BTC/USD closing prices of September 2017

7 Discussion

The results presented in table 1 show that from our experiments, we have seen no considerable difference in terms of prediction accuracy between LSTM models and GRU models. However, in terms of training time, GRU networks were 12% faster than LSTM networks. Figure 7 shows that in 5 experiments, using the combinations of hyper-parameters that yielded the most accurate results, GRU models always outperformed LSTM models.

Model	31 days	7 days	1 day
ARIMA with Dyn.Reg.	255.9	238.4	147.6
LSTM	272.96	397.3	518.6
GRU	274.02	327.9	396.4
ARIMA	1196.1	338.9	169.8

Table 1: Accuracy (RMSE) of four models for 1 month, 1 week and 1 day of predictions of the BTC/USD closing price.

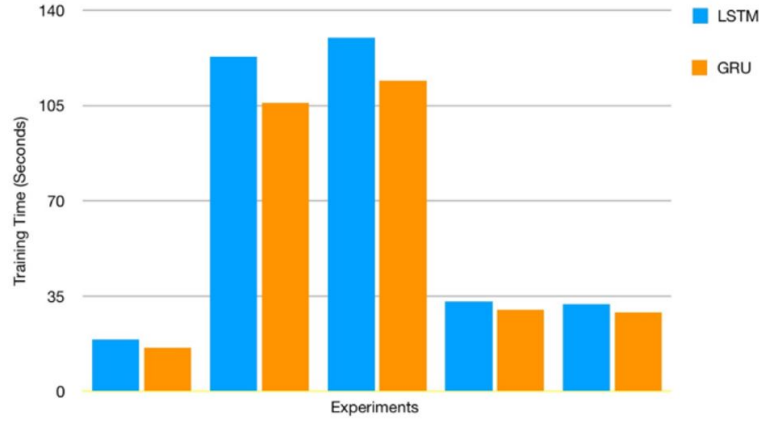


Figure 7: training time (in seconds) of LSTM networks and GRU networks in 5 experiments to accurately predict BTC/USD closing prices of September 2017

The need for better predictive models is latent in every industry sector. In particular, it is a requirement for the financial planning which goes in hand with the economic growth goals of our society[22]. We are convinced that there is a lot of room for improvements in the area of predictive analytics using deep neural networks. Therefore, these results are not meant to discourage the research for better forecasting models that can outperform the traditionally used methods. The future work related to this study could be focused on testing different insights obtained after studying the related work. For example, Greff et al.[9] suggests that one of the most critical components for the LSTM performance is the output activation function. Another point to research would be testing newer variations of RNNs like the Minimal Gated Recurrent Unit network proposed by Zhou et al. in March 2016[11]. In addition, it would be a good exercise to replicate this study using random search instead of grid search as the strategy for the parameter optimization. The results and conclusions presented in this study should not be taken as a financial or investment advice. We consider that using the predicted behavior of financial assets should not be the main factor for investment decisions. Ethics play a more important role in this aspect.

References

- [1] Xin-Yao Qian. “Financial Series Prediction: Comparison Between Precision of Time Series Models and Machine Learning Methods”. In: *arXiv:1706.00948 [cs, q-fin]* (June 2017). arXiv: 1706.00948. URL: <http://arxiv.org/abs/1706.00948> (visited on 12/22/2017).
- [2] Zhengping Che et al. “Recurrent Neural Networks for Multivariate Time Series with Missing Values”. In: *arXiv:1606.01865 [cs, stat]* (June 2016). arXiv: 1606.01865. URL: <http://arxiv.org/abs/1606.01865> (visited on 12/22/2017).
- [3] Jeffrey Elman. “Finding Structure in Time”. In: *Cognitive Science* 14 (Mar. 1990). DOI: 10.1016/0364-0213(90)90002-E.
- [4] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-Term Memory”. In: *Neural Comput.* 9.8 (Nov. 1997), pp. 1735–1780. ISSN: 0899-7667. DOI: 10.1162/neco.1997.9.8.1735. URL: <http://dx.doi.org/10.1162/neco.1997.9.8.1735> (visited on 12/22/2017).
- [5] *Recurrent Neural Networks - Combination of RNN and CNN - Convolutional Neural Networks for Image and Video Processing - TUM Wiki*. URL: <https://wiki.tum.de/display/lfdv/Recurrent+Neural+Networks+-+Combination+of+RNN+and+CNN> (visited on 01/16/2018).
- [6] Junyoung Chung et al. “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling”. In: *arXiv:1412.3555 [cs]* (Dec. 2014). arXiv: 1412.3555. URL: <http://arxiv.org/abs/1412.3555> (visited on 12/22/2017).
- [7] K. Greff et al. “LSTM: A Search Space Odyssey”. In: *IEEE Transactions on Neural Networks and Learning Systems* 28.10 (Oct. 2017), pp. 2222–2232. ISSN: 2162-237X. DOI: 10.1109/TNNLS.2016.2582924. URL: <https://arxiv.org/abs/1503.04069>.
- [8] Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. “An Empirical Exploration of Recurrent Network Architectures”. In: *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*. ICML’15. Lille, France: JMLR.org, 2015, pp. 2342–2350. URL: <http://dl.acm.org/citation.cfm?id=3045118.3045367> (visited on 12/22/2017).
- [9] Kyunghyun Cho et al. “Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation”. In: *arXiv:1406.1078 [cs, stat]* (June 2014). arXiv: 1406.1078. URL: <http://arxiv.org/abs/1406.1078> (visited on 12/22/2017).
- [10] Guo-Bing Zhou et al. “Minimal Gated Unit for Recurrent Neural Networks”. In: *arXiv:1603.09420 [cs]* (Mar. 2016). arXiv: 1603.09420. URL: <http://arxiv.org/abs/1603.09420> (visited on 12/22/2017).
- [11] Junyoung Chung et al. “Gated Feedback Recurrent Neural Networks”. In: *arXiv:1502.02367 [cs, stat]* (Feb. 2015). arXiv: 1502.02367. URL: <http://arxiv.org/abs/1502.02367> (visited on 12/22/2017).
- [12] Bilberto Batres-Estrada. “Deep learning for multivariate financial time series”. eng. PhD thesis. KTH Royal Institute of Technology, 2015. URL: <http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-168751> (visited on 12/22/2017).
- [13] Y. Bengio, P. Simard, and P. Frasconi. “Learning long-term dependencies with gradient descent is difficult”. In: *IEEE Transactions on Neural Networks* 5.2 (Mar. 1994), pp. 157–166. ISSN: 1045-9227. DOI: 10.1109/72.279181.
- [14] Sean McNally. “Predicting the price of Bitcoin using Machine Learning”. en. masters. Dublin, National College of Ireland, Sept. 2016. URL: <http://trap.ncirl.ie/2496/> (visited on 12/22/2017).

- [15] Martina Matta, Ilaria Lunesu, and Michele Marchesi. “Bitcoin Spread Prediction Using Social And Web Search Media”. In: *Deep Content Analytics Techniques for Personalized & Intelligent Services* (June 2015).
- [16] *Google Trends*. URL: <https://trends.google.com> (visited on 12/22/2017).
- [17] *Bitcoincharts*. URL: <https://bitcoincharts.com/> (visited on 12/22/2017).
- [18] *Quandl*. URL: <https://www.quandl.com> (visited on 12/22/2017).
- [19] *Yahoo Finance - Business Finance, Stock Market, Quotes, News*. URL: <https://finance.yahoo.com/> (visited on 12/22/2017).
- [20] Timothy Dozat. “Incorporating Nesterov Momentum into Adam”. In: 2015.
- [21] Rob J. Hyndman. *fpp: Data for "Forecasting: principles and practice"*. Mar. 2013. URL: <https://CRAN.R-project.org/package=fpp> (visited on 12/22/2017).
- [22] *Sustainable Development Goals: 17 Goals to Transform Our World*. URL: <http://www.un.org/sustainabledevelopment/> (visited on 12/22/2017).