ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/318329563

Stock market's price movement prediction with LSTM neural networks

Conference Paper · May 2017

DOI: 10.1109/IJCNN.2017.7966019

CITATIONS
27

READS
5,116

3 authors:

David M. Q. Nelson
Federal University of Minas Gerais
1 PUBLICATION **27** CITATIONS

SEE PROFILE

Adriano C. M. Pereira
Federal University of Minas Gerais
112 PUBLICATIONS **374** CITATIONS

SEE PROFILE

Renato Arantes de Oliveira
Federal University of Minas Gerais
1 PUBLICATION **27** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project A Reinforcement Learning Trading System for Stock Market View project

Project Credibility of Web Applications and Services View project

All content following this page was uploaded by Adriano C. M. Pereira on 03 January 2018.

The user has requested enhancement of the downloaded file.

Stock Market's Price Movement Prediction With LSTM Neural Networks

David M. Q. Nelson, Adriano C. M. Pereira, Renato A. de Oliveira
Departamento de Ciência da Computação (DCC)
Universidade Federal de Minas Gerais (UFMG)
Belo Horizonte – MG – Brazil
Email: davidmn@ufmg.br, adrianoc@dcc.ufmg.br, ra.oliveira@dcc.ufmg.br

***Abstract*—Predictions on stock market prices are a great challenge due to the fact that it is an immensely complex, chaotic and dynamic environment. There are many studies from various areas aiming to take on that challenge and Machine Learning approaches have been the focus of many of them. There are many examples of Machine Learning algorithms been able to reach satisfactory results when doing that type of prediction. This article studies the usage of LSTM networks on that scenario, to predict future trends of stock prices based on the price history, alongside with technical analysis indicators. For that goal, a prediction model was built, and a series of experiments were executed and theirs results analyzed against a number of metrics to assess if this type of algorithm presents and improvements when compared to other Machine Learning methods and investment strategies. The results that were obtained are promising, getting up to an average of 55.9% of accuracy when predicting if the price of a particular stock is going to go up or not in the near future.**

I. Introduction

Predictions on stock markets have been object of studies for many decades, but given it's innate complexity, dynamism and chaoticness, it has proven to be a very difficult task. The number of variables and sources of information considered are immense and the signal-to-noise ratio insignificant. That makes the task of predicting stock market prices behavior in the future a very hard one. For many decades, there's been discussions in Science regarding the possibility of such a feat and it's notable in the related literature that most prediction models fail to provide precise prediction in a general sense.

Nonetheless, there is huge amount of studies from various disciplines seeking to take on that challenge, presenting a large variety of approaches to reach that goal.

One common approach is to use Machine Learning algorithms to learn from price historic data to predict future prices. This article goes in that direction but studying an specific method using recurrent neural networks. Such networks have a short term memory capability and the hypothesis to explore here is that this feature can present gains in terms of results when compared to others more traditional approaches in Machine Learning field.

The algorithm of choice here is the LSTM (Long-Short term memory) network. It's a type of recurrent network that has proved very successful on a number of problems given its capability to distinguish between recent and early examples by giving different weights for each while forgetting memory it considers irrelevant to predict the next output. In that way, it is more capable to handle long sequences of input when compared to other recurrent neural networks that are only able to memorize short sequences.

Historic price data (candlesticks) from different stocks from the Brazilian stock exchange (Bovespa) will be used as source of information for the network. Along with this data, a large amount of technical indicators will also be generated to feed the network as features. Upon this dataset the model will be trained, evaluated and will attempt to predict whether the price of a particular stock will go up or not in the next 15 minutes.

The objective of this project is to study the applicability of recurrent neural networks, in particular the LSTM networks, on the problem of stocks market prices movements prediction. Assess their performance in terms of accuracy and other metrics through experiments on top of real life data and analyze if they present any sort of gain in comparison to more traditional machine learning algorithms.

In addition to that, the model is also validated in regards to its financial performance, by comparing it with some simple while valid investments strategies in terms of overall returns and net result per trading operation.

The main contributions of this work are the following: (1) a new price movement prediction model for stock markets using deep learning based technique; (2) the validation of the model using real data from Brazilian stock exchange; (3) evaluation of the model by comparing and analyzing it against some typical baselines.

The remainder of this paper is organized as follows: Section II presents an overview of the theoretical concepts that are base of this article, both in the context of stock markets and of machine learning. It also mentions some of the related work to this subject that can be found in the literature and highlights the novelty this project aims to bring; Section III details the model proposed in this article, in Section IV the experimental results are presented and discussed, and finally Section V concludes the article.

978-1-5090-6182-2/17/$31.00 ©2017 IEEE

II. Background and Related work

When it comes to stock markets, in addition to its inherent complexity and dynamism, there has been a constant debate on the predictability of stock returns. [1] introduced the Efficient-Market hypothesis that defines that the current price of an asset always reflects all previous information available for it instantly. There is also the Random-walk hypothesis [2] which claims that a stock price changes independently of its history, in other words, tomorrow's price will only depend on tomorrow information regardless of today's price. Those two hypothesis determine that there are no means to accurately predict a stock price. On top of that, [3] has performed a series of experiments showing that a random strategy can outperform some of the most classic methods of technical trading, like Moving Average Convergence Divergence (MACD) and Relative Strength Index (RSI).

On another hand, there are other authors who claim that, in fact, stock prices can be predicted at least to some degree [4]. And a variety of methods for predicting and modeling stock behavior have been object of study of many different disciplines, such as economics, statistics, physics and computer science. It's worth mentioning that in 2012, it was estimated that approximately 85% of trades within the United States' stock markets were performed by algorithms. [5].

A popular method of modeling and predicting the stock market is technical analysis, which is a method based on historic data from the market, mainly price and volume. It follows some assumptions: (1) prices are defined exclusively by the supply-demand relation; (2) prices change following tendencies; (3) changes on supply and demand cause tendencies to reverse; (4) changes on supply and demand can be identified on charts; And (5) patterns on charts tend to repeat [6]. In other words, technical analysis do not take into account any external factors like political, social or macro-economical.

In regards to computational intelligence there are plenty of studies assessing different methods in order to accomplish accurate predictions on the stock market. They go from evolutionary computation through genetic algorithms as exemplified in [7], statistical learning by using algorithms like Support Vector Machines (SVM) [8] and a variety of others including neural networks, component modeling, textual analysis based on news data, that are discussed by [9] which also proposed a new approach based on collective intelligence.

Taking a closer look into works related to deep learning in stock markets there are some examples like [10] where a study is made on the usage of a Deep Belief Network (DBN), which is composed of stacked Restricted Boltzmann Machines, coupled to a Multi-level Perceptron (MLP) and using long range log returns from stock prices to predict above-median returns for each day. [11] also use of DBN, but this time using price history in addition to technical indicators as input, in a similar approach to this project. Both of those works present improved results compared to their baselines, as well as in [12] where a survey in deep learning methods applied to finance is made and their improvements discussed.

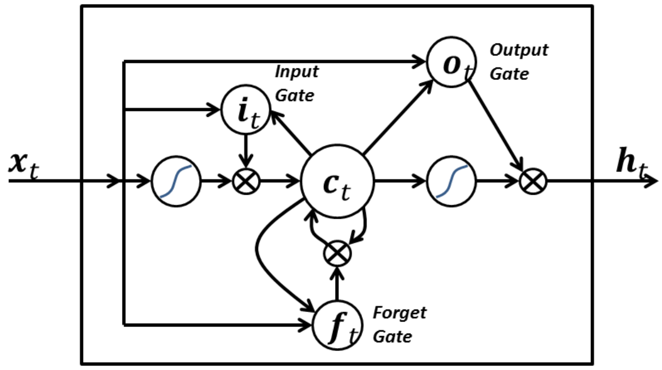


Fig. 1. Long short term memory neural network [13]

Long Short Term Memory (LSTM) networks (Figure 1), which are used in this project are a deep and recurrent model of neural networks. Recurrent networks differ from the traditional feed-forward networks in the sense that they don't only have neural connections on a single direction, in other words, neurons can pass data to a previous or the same layer. In which case, data doesn't flow on a single way, and the practical effects for that is the existence of short term memory, in addition to long term memory that neural networks already have in consequence of training. LSTM were introduced by [14] and it aimed for a better performance by tackling the vanishing gradient issue that recurrent networks would suffer when dealing with long data sequences. It does so by keeping the error flow constant through special units called ”gates” which allows for weights adjustments as well as truncation of the gradient when its information is not necessary.

These networks have been widely used and been able to accomplish some of the best results when compared to other methods [15], especially on Natural Language Processing (NLP), and in particular for handwriting recognition it is considered the state-of-the-art [16]. And since its inception it has been branched into a number of variations which were assessed against their original version by [13] but do not seem to present any considerable improvements so far.

For stock prices prediction, given the notorious performance LSTM networks have shown in NLP, it has been mostly used taking news text data as input to predict price trends. But there is also some work using price data to foresee price movement, [17] employ historic price data in addition to stock indexes to predict if a stock price will go up, down or stay the same on the day. [18] compares the performance of LSTM and MLP to their own proposed method based on a combination of wavelets and Convolutional Neural Networks, which outperforms both but has very close results to the LSTM network.

We propose a model based on LSTM to predict price movements using an input that is not based on text, which as mentioned before, is not something that has been widely explored. We plan to use a wide range of technical indicators to do so, and the intention is to assess the usage of such method that is something commonly used on investment strategies. Additionally, we want to test the hypothesis that the short term memory capability can present better results compared to traditional feed forward networks.

III. Methodology & Development

A classification model (Figure 2) was designed based on LSTM networks so as to perform predictions of price movements for a number of stocks. In other words, it attempts to determine if the price of a particular stock will be higher or not than the current price in 15 minutes in the future.

That model is regenerated and trained each trading day on top of historic price data and it is used for performing predictions each 15 minutes using the same model and weights until the end of the day.

A. Data processing

Historic price data for particular stocks are gathered in the format of a time series of candles (open, close, high, low and volume) in a granularity of 15 minutes. For this article, data from 2008 to 2015 was collected for stocks that part of the IBovespa index from the BM&F Bovespa stock exchange.

With the data in hand, a log-return transformation is performed as means of normalization as well as to stabilize the mean and variance along the time series. The log-return transformation can be expressed as (1):

$$log(p_i) - log(p_{i-1}) \tag{1}$$

On top of the historic price data, in order to reduce random variation and noise on the pricing series, exponential smoothing was performed through exponentially weighted moving averages (2) as indicated by [19] to cause improvements on the prediction capability.

Also on top of the price data, a set of technical indicators is generated using the TA-Lib[1] library. Such indicators are mathematical calculations intended to determine or predict characteristics from stocks based on their historic data. A total of 175 values are generated for each period, and they are intended to represent or predict a very diverse set of characteristics of the stock, like the future price, volume to be traded, the intensity of the current movement tendency, visual graphical patterns, among others.

$$z_i = \lambda \bar{x}_i + (1 - \lambda) z_{i-1} \tag{2}$$

A binary class y is assigned to each entry of the dataset, "1" will indicate that the price will go up on the following time step, and "0" that it won't. Therefore, given that i is the current moment and j is the following, then $j = i + timestep$, and for this project $timestep$ is equivalent to 15 minutes. In case the output is "1", a "buy" operation will be triggered at i.

For determining the class, the policy to set the value will be based on whether or not the closing price of the next period will be higher than the current one:

$$y = \begin{cases} 1 & \text{if } close_j > close_i \\ 0 & \text{if } close_j \leq close_i \end{cases} \tag{3}$$

The neural network will take k instances of X as input $(X_{i-k}, ..., X_{i-2}, X_{i-1}, X_i)$, where X consists in tuples of price data along with technical indicators, and that will be used to predict y_j.

IV. Evaluation

This model was designed to work on a rolling window fashion. A new neural network is generated at the end on each trading day, meaning that a new set of weights is defined using a new set of training and validation data. For training it is used the last 10 months of trading prior to the current day, and the model performance is validated by using the data of the past week. On the following day, all the predictions will be done using the most recent model.

Given that we are working with a time series, the supervised learning algorithm that was chosen was the LSTM neural network (Long short term memory), which is a recurrent neural network capable o classifying input data taking into account the previous instances.

Google's TensorFlow [2] was used to build the model, which consists of a LSTM input layer, that will take both technical indicators and pricing data as input and will feed an output layer using sigmoid activation (4).

$$S(t) = \frac{1}{1 + e^{-t}} \tag{4}$$

The input layers has a dimensionality of 180 features, that consists of the set of technical indicators (all of those generated by TA-Lib) plus the price return data (open, close, minimum, maximum) and volume. It will have an output using the $tanh$ function and that will be connected to the network's output layer through 20 connections.

A series of experiments using the proposed model were executed and average of the results were gathered using pricing data for a selection of few different stocks (BOVA11, BBDC4, CIEL3, ITUB4 and PETR4) from the Brazilian stock exchange for the year of 2014.

For evaluating the network performance, metrics around the algorithm performance and the financial results were collected and compared with the baselines selected for this project. The metrics were accuracy (5), precision (6), recall (7) and F-measure, a harmonic mean between precision and recall (8).

Those metrics are calculated based on the predictions correctly made for positive classes (true positives - tp), negative

[1] http://ta-lib.org

[2] https://www.tensorflow.org

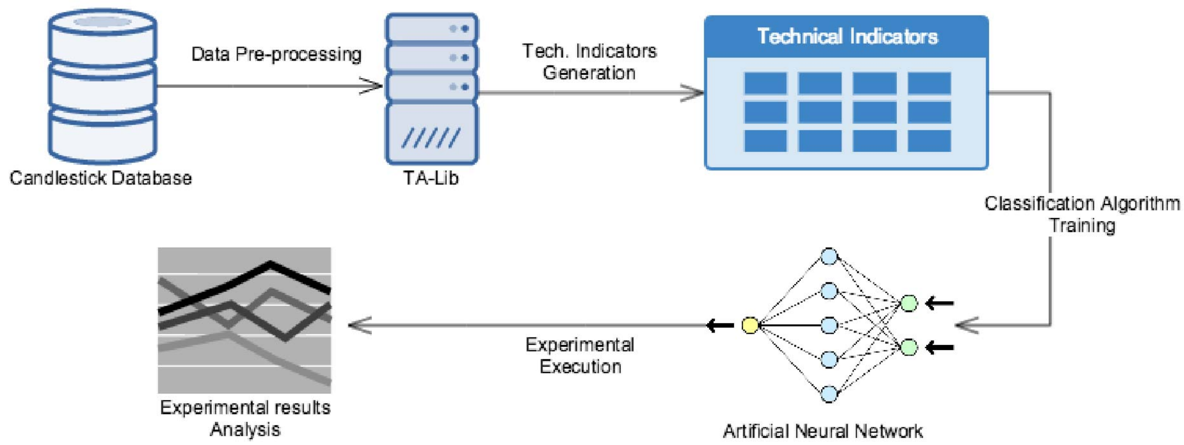


Fig. 2. Methodology

classes (true negatives - tn) and those made inaccurately for both classes (false positive - fp, false negative - fn).

$$A = \frac{tp + tn}{tp + fp + tn + fn} \tag{5}$$

$$P = \frac{tp}{tp + fp} \tag{6}$$

$$R = \frac{tp}{tp + fn} \tag{7}$$

$$F1 = 2\frac{P * R}{P + R} \tag{8}$$

A. Trading operations

When the predicted class is ”1”, in other words, in the case that the network predicts that the stock price will go up, then the strategy is to open a ”buy” position on the current moment (i) and close it on j. In that case, profit was defined as $close_j - close_i$.

The financial results were calculated based on hypothetical trades of sets of a hundred stocks per operation disregarding costs and taxes.

B. Baselines

For comparison, the baselines chosen are based on other classical machine learning algorithms in addition to other simplistic investment methods.

The machine learning methods consist of approaches that are traditional and widely used but less complex than the one this project is based on. Using the exact same input, trying to do the same predictions, the models chosen were Multi-Layer Perceptron, Random Forest, and a pseudo-random model that outputs a class based on probabilities following the class distribution.

The other baselines are the following investment strategies:

- Buy and hold: Buy at the first time step and sell at the latest ($profit = close_n - close_1$)
- Optimistic: If prices went up on the previous time step ($close_i > close_{i-timestep}$), then perform a buying operation and sell it on the following step.
- Pseudo-random (following class distribution): Decides whether or not to perform a trading operation based on probabilities according to the class distribution.

C. Empirical Results

Experiments were carried out to predict price movement during December 2014 for the aforementioned stocks and following the methodology described on this article. The results presented below are an average of all executions per each stock.

Table I presents a characterization of the data used for the experiments, for each stock it has the price that it has on the first day of the period, and what it was on the last day. It also shows the percent of times that the price goes up which is basically the class distribution.

Stock	Start	End	Diff.	% times goes up
BOVA11	53.54	48.54	-5.00	0.471
BBDC4	35.68	30.79	-4.89	0.435
CIEL3	32.34	33.98	1.64	0.453
ITUB4	29.65	30.73	1.08	0.442
PETR4	18.39	10.03	-8.36	0.479

TABLE I
EXPERIMENTAL DATA

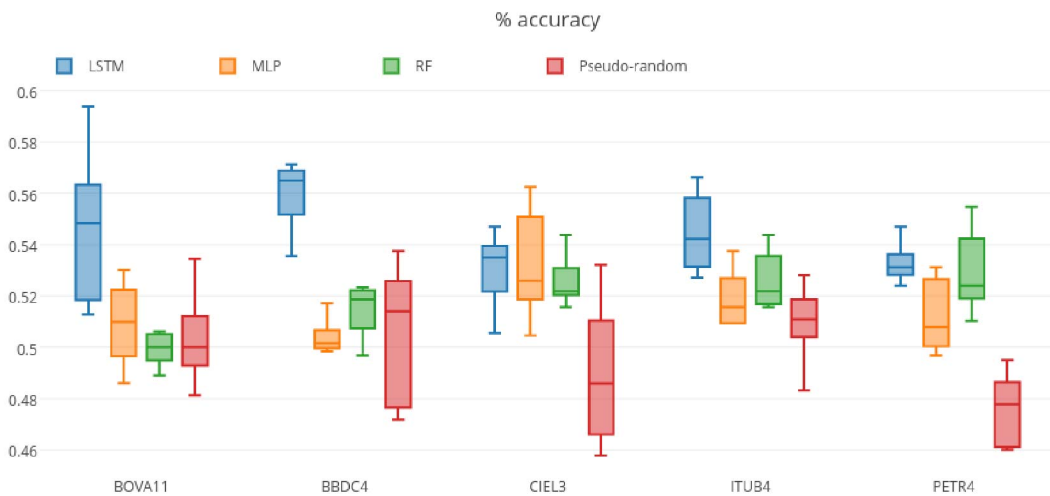


Fig. 3. Accuracy compared to baselines

Stock	Accuracy	Precision	Recall	F1
BOVA11	0.546	0.560	0.350	0.431
BBDC4	0.559	0.553	0.129	0.209
ITUB4	0.545	0.475	0.134	0.209
CIEL3	0.530	0.476	0.137	0.213
PETR4	0.533	0.563	0.231	0.327

TABLE II

RESULTS - METRICS

On Table II the metrics of the algorithm prediction performance are shown for each of the stocks, that tells how well it does as a prediction model. Figure 3 compares it to MLP and Random Forest baselines in terms of accuracy, showing that it outperforms both on average. And Figure 4 presents the standard deviation for a number of experiments on each of the models to give an understanding of the variance and consistency of each model.

In order to determine if there is significant improvement in terms of accuracy when comparing the proposed model to the baselines we executed a statistical test on top of the results.

The method chosen for that task is Kruskal-Wallis, which is a non-parametric test to compare multiple populations that do not assume that the values are normally distributed.

This test returns a value (p-value) that tells the probability of the observed results being equal of more extreme in relation to each other. If that value is below the specified significance level (typically 0.05), then it is understood that there is significant statistical difference between the populations.

In Figure 5 are shown the results of the statistical tests on the results of the LSTM based model compared to the baselines. The model has shown significant superiority for most occurrences, with a p-value smaller than 0.05.

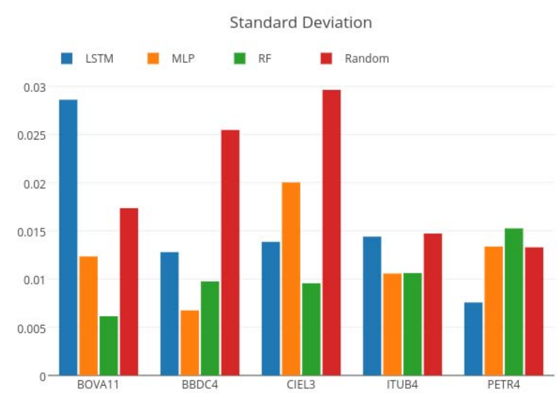
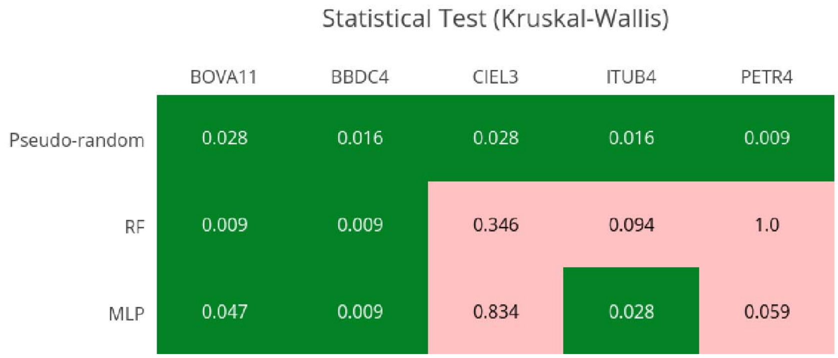


Fig. 4. Accuracy standard deviation

Figure 6 displays the average financial returns for the experiments per each stock, compared to each of the investment baselines that were used for this work, while Figure 7 shows the average earnings per individual buy-and-sell operation.

In Figure 8 it is shown the maximum drawdown for each of the strategies. It works as a downside risk metric, presenting the maximum losses from a peak that were faced by the algorithms during the execution. In that case, it indicates potential for losses of each method and stock combination, and for most of them the LSTM has provided better results.

Fig. 5. Statistical Test - Kruskal-Wallis



Statistical Test (Kruskal-Wallis)

	BOVA11	BBDC4	CIEL3	ITUB4	PETR4
Pseudo-random	0.028	0.016	0.028	0.016	0.009
RF	0.009	0.009	0.346	0.094	1.0
MLP	0.047	0.009	0.834	0.028	0.059

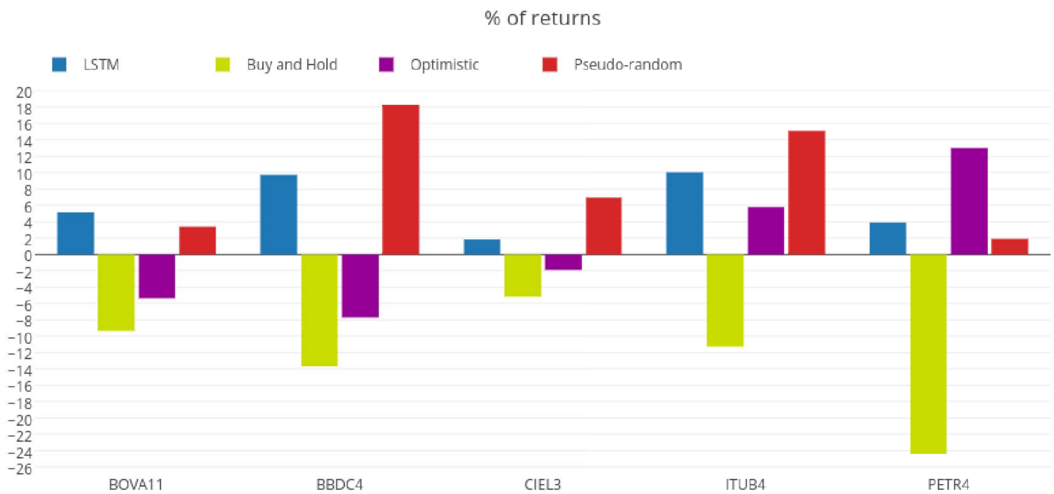


Fig. 6. Returns

V. Conclusions

We can observe that in general the model proposed on this article outperforms the baselines with few exceptions. The outcomes can be considered very promising as it has proven able to predict well compared to other approaches employed today in the literature.

Although the input dimension is very large, the algorithm has demonstrated acceptable capability to learn from it without the need of any dimension reduction technique like feature selection, for example.

When compared to the other machine learning models it has displayed considerable gains in terms of accuracy, but in another hand we believe that variance could be lower and that would contribute for a more reliable model.

When it comes to the financial results it's important to note that it was able to keep it positive for all stocks, even though it didn't necessarily had the best results when compared to the baselines.

Another positive aspect is that it had a high return ratio per operation, meaning that it had more success on detecting high variations which is extremely important when account transactions costs and taxes are taken into account.

Besides that, when observing the maximum losses it is possible to conclude that the LSTM based model offers less risks when compared to the other strategies.

A. Forthcoming Research

We intend to keep investigating ways to improve our model and its predictions by studying changes on the neural network architecture and different approaches for pre-processing the input data as well as adding new different features.

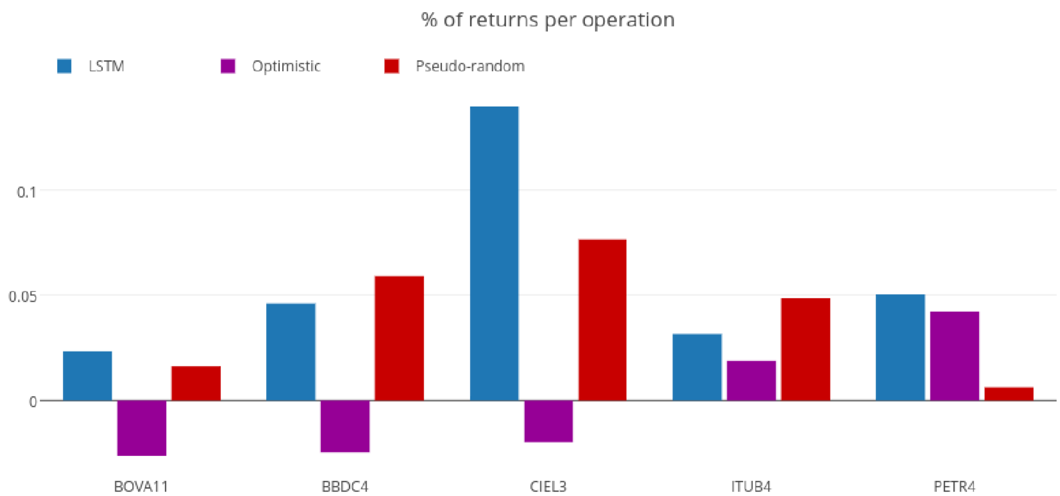


Fig. 7. Average return per operation

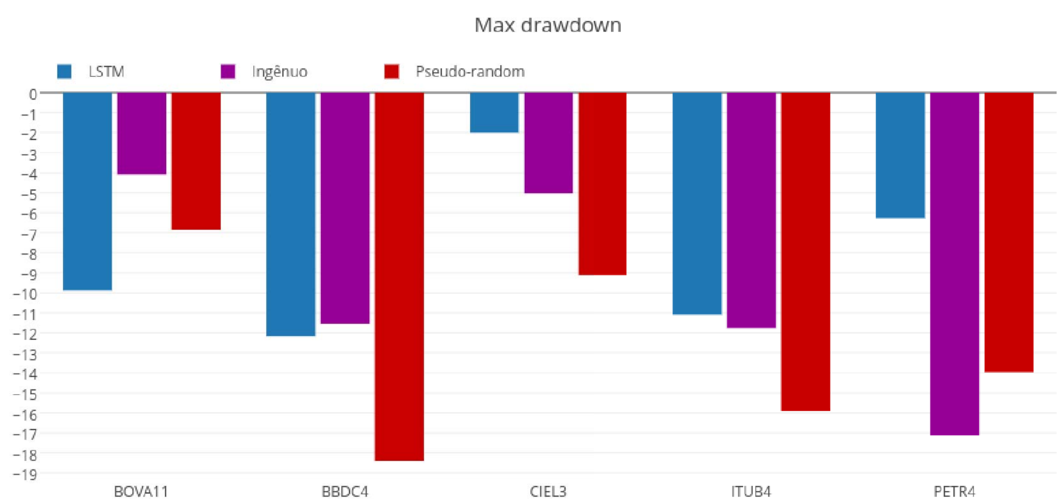


Fig. 8. Maximum drawdown

Given what we could observe so far, the key to get to better prediction results lies in improving the input like it is the exact same case for many other Machine Learning problems.

We also intend to evaluate the model using different and more realistic trading strategies, instead of simply buying and selling after a fixed amount of time. And also take into account intrinsicalities of stock markets, like timing, execution booking and associated transactions costs.

Acknowledgment

This research was supported by the Brazilian National Institute of Science and Technology for the Web (CNPq grant numbers 573871/2008-6 and 477709/2012-5), MASWeb (grant FAPEMIG/PRONEX APQ-01400-14), EUBra-BIGSEA (H2020-EU.2.1.1 690116, Brazil/MCTI/RNP GA-000650/04), CAPES, CNPq, and Fapemig.

REFERENCES

[1] E. F. Fama and B. G. Malkiel, "Efficient capital markets: A review of theory and empirical work," *The Journal of Finance*, vol. 25, no. 2, pp. 383–417, 1970. [Online]. Available: http://dx.doi.org/10.1111/j.1540-6261.1970.tb00518.x

[2] B. G. Malkiel, *A Random Walk Down Wall Street*. Norton, New York, 1973.

[3] A. E. Biondo, A. Pluchino, A. Rapisarda, and D. Helbing, "Are Random Trading Strategies More Successful than Technical Ones?" *PLoS ONE*, vol. 8, p. e68344, Jul. 2013.

[4] A. Lo and A. MacKinlay, *A non-random walk down Wall Street*. Princeton, NJ [u.a.]: Princeton Univ. Press, 1999. [Online]. Available: http://gso.gbv.de/DB=2.1/CMD?ACT=SRCHA&SRT=YOP&IKT=1016&TRM=ppn+249613484&sourceid=fbw_bibsonomy

[5] M. Glantz and R. Kissell, *Multi-asset Risk Modeling*, 1st ed. Academic Press, 2013.

[6] C. Kirkpatrick and J. Dahlquist, *Technical Analysis: The Complete Resource for Financial Market Technicians*, 1st ed. FT Press, 2006.

[7] F. Allen and R. Karjalainen, "Using genetic algorithms to find technical trading rules," *Journal of Financial Economics*, vol. 51, no. 2, pp. 245 – 271, 1999. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0304405X9800052X

[8] K. Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol. 55, no. 1–2, pp. 307 – 319, 2003, support Vector Machines. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231203003722

[9] B. Melo, "Considerações cognitivas nas técnicas de previsão no mercado financeiro," *Universidade Estadual de Campinas*, 2012.

[10] B. Batres-Estrada, "Deep learning for multivariate financial time series," 2015.

[11] A. Sharang and C. Rao, "Using machine learning for medium frequency derivative portfolio trading," *CoRR*, vol. abs/1512.06228, 2015. [Online]. Available: http://arxiv.org/abs/1512.06228

[12] J. B. Heaton, N. G. Polson, and J. H. Witte, "Deep learning in finance," *CoRR*, vol. abs/1602.06561, 2016. [Online]. Available: http://arxiv.org/abs/1602.06561

[13] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *arXiv preprint arXiv:1503.04069*, 2015.

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997. [Online]. Available: http://dx.doi.org/10.1162/neco.1997.9.8.1735

[15] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, ser. Studies in Computational Intelligence. Springer, 2012, vol. 385. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-24797-2

[16] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 855–868, May 2009.

[17] K. Chen, Y. Zhou, and F. Dai, "A lstm-based method for stock returns prediction: A case study of china stock market," in *Big Data (Big Data), 2015 IEEE International Conference on*, Oct 2015, pp. 2823–2824.

[18] O. H. Luca Di Persio, "Artificial neural networks approach to the forecast of stock market price movements," *nternational Journal of Economics and Management Systems*, vol. 1, pp. 158–162, 2016.

[19] L. Khaidem, S. Saha, and S. R. Dey, "Predicting the direction of stock market prices using random forest," *CoRR*, vol. abs/1605.00003, 2016. [Online]. Available: http://arxiv.org/abs/1605.00003

View publication stats