Algorithmic Trading Using Long Short-Term Memory Network and Portfolio Optimization

Riccardo Lucato, Edgar Jimenez, Eduardo Salvador Rocha, Yang Qi, Marina Gavrilyuk and Rafael Rêgo Drumond

Abstract Investors typically rely on a mix of experience, intuition, knowledge of economic fundamentals and real-time information to make informed choices and try to get as high a rate of return as possible. Their decisions are customarily more instinct-driven than methodical. Propelled by the need for numerically inspired judgments, ever stronger within the financial community, in recent years the usage of computational and mathematical tools has been taking root. In this work we used a Long Short-Term Memory (LSTM) Network trained on historical prices to predict future daily closing prices of several stocks listed on the Standard & Poor 500 (S&P500) index. We compared the predictions of our LSTM network with those produced by another state-of-the-art approach, the Hidden Markov Model (HMM), in order to validate our findings. We then fed our forecasts into a Markowitz Portfolio Optimization (PO) procedure to identify

---

Riccardo Lucato · Edgar Jimenez · Eduardo Salvador Rocha · Yang Qi · Marina Gavrilyuk · Rafael Rêgo Drumond
Stiftung Universität Hildesheim, Universitätspl. 1, 31141 Hildesheim, Germany
✉ lucato@uni-hildesheim.de,
✉ jimenez@uni-hildesheim.de,
✉ salvador@uni-hildesheim.de,
✉ qiyang@uni-hildesheim.de,
✉ gavrilyu@uni-hildesheim.de,
✉ radrumond@ismll.uni-hildesheim.de

ARCHIVES OF DATA SCIENCE, SERIES A
(ONLINE FIRST) DOI: 10.5445/KSP/1000098012/13
KIT SCIENTIFIC PUBLISHING ISSN 2363-9881
Vol. 6, No. 2, 2020

the best trading strategy. The purpose of PO, which allows for simultaneous and optimal trading of multiple stocks, is to compute a set of daily weights representing the portion of initial capital to be invested in each company. Our empirical results highlight two facts: Firstly, our LSTM model achieves higher accuracy than the standard HMM approach. Secondly, by trading various stocks at the same time we can obtain a higher rate of return than is possible by using the single stock strategy, while also greatly enhancing the real-world applicability of our model.

1 Introduction

Over the past 40 years the size of the global stock market has been following a steep upwards trend. In 1980 the market capitalization of listed domestic companies was approximately 2.5 trillion USD, in 2017 the same indicator was approaching the value of 80 trillion USD (The World Bank Group, 2019). As is often the case, the growth in size came at the price of increased complexity. The present-day stock market is characterized by swifter transactions, a deepened entanglement between economic variables and more frequent and severe downturns (Reid et al., 2017). In such a scenario, the traditional financial tools have long proved insufficient. Floor traders, although still present in many markets, are progressively being replaced by automated trading systems, capable of handling larger operations at a faster pace than their human counterparts (Lin, 2014). Particularly, a sizeable amount of interest was raised in both academy and industry by systems which do not merely execute given trading orders, but automatically select the strategy by predicting future stock prices and adjusting the orders accordingly. Foreseeing stock prices is a typical non-stationary and noisy time series forecasting problem (Wang, 2003) which finds extensive investigation in the state of the art, and to which we here propose an innovative solution. This consists in the preliminary usage of an LSTM network for the forecast of future stock prices and the subsequent employment of a Portfolio Optimization procedure, applied not only on observed prices as is settled financial practice, but rather on the values predicted by the LSTM. The present work is organized as follows: Section 2 characterizes the state of the art in the field of stock market prediction and algorithmic trading. Section 5 details the dataset used throughout our research. Sections 3 and 4 respectively outline the functioning of an LSTM network and of a Portfolio Optimization procedure and Section 6 illustrates all experiments which have been carried out to corroborate our findings. The results are then discussed in Section 7. Finally, Section 8 summarizes the paper and puts forward possible future directions of investigation.

2 Related Work

The first investigations in the field mainly involved the use of simple neural networks. Pradeep et al. (2013) proposed a backpropagation neural network with a generalized delta rule to learn the interrelations between underlying financial variables, while Trippi and Desieno (1992) incorporated several neural networks into a single boolean decision rule system which generates a composite recommendation for the current day's position on the S&P 500 index. Following the research of Miao et al. (2007), which pointed out that simple backpropagation-based techniques are excessively prone to returning sub-optimal solutions due to the optimization algorithms frequently wallowing in local optima, in recent times more sophisticated strategies have been introduced. Zhang and Wu (2009) proposed a backpropagation neural network optimized via Bacterial Chemotaxis to develop a stock market forecasting model which showed significant learning ability and generalization. Wang et al. (2011) devised an efficient and doubtlessly innovatory wavelet de-noising-based backpropagation neural network which filters out the noise inherently present in the data and carries out predictions based on the noise-free data.

A second, equally popular category of strategies features the use of hidden Markov models (HMM). Kritzman et al. (2012) applied a two-states HMM to predict economic regimes in the presence of market turbulence and high inflation, Gupta and Dhingra (2012) presented a maximum a posteriori HMM approach to forecast the next day's stock values, while Nguyen (2014) pursued the same objective by making use of a HMM with both single and multiple observations. Most recently, Nguyen (2017) incorporated a HMM-based forecasting strategy into a larger framework for the efficient trade of stocks, which was shown to achieve a greater profit than the one obtained with a Naïve forecasting approach.

In this paper, we decided to follow a partially different path. First, we made use of a LSTM Network to iteratively forecast one trading year's worth of closing prices for different sets of stocks, then we fed the predictions into a portfolio optimization process to estimate the most efficient funds' allocation among the different target stocks. We tested our strategy with two experiments. The first one was a direct comparison between the performance of our model and that of the approach proposed in Nguyen (2017), which predicted the daily closing price of three well-known stocks, namely Apple Inc. (AAPL), Alphabet Inc. (GOOGL) and Facebook Inc. (FB) over an entire trading year (15.08.2016 - 11.08.2017), then used the predictions in combination with a simple buy-sell trading strategy to try and get a return. In addition, we extended the comparison to a different time window, specifically on the last trading year up until the day in which the experiment was performed (14.09.2018 - 13.09.2019), by re-implementing the baseline of Nguyen (2017) and matching its results to our model's. In the second experiment, carried out to assess our method's robustness and to further prove its generalization capability, we predicted the daily closing price of the 19 most-influential stocks listed on the S&P 500 index on the same, initial trading year (15.08.2016 - 11.08.2017), then used those predictions to trade the stocks, both independently from one another without PO and jointly with PO. All results were, again, set side to side with those obtained by our re-implementation of the model in Nguyen (2017).

The experiment results highlight two facts: First, our LSTM model almost invariably achieves higher accuracy than the standard HMM approach. Second, by trading various stocks at the same time with our Portfolio Optimization approach we can obtain a higher rate of return than is possible by using the single stock strategy, while also greatly enhancing the real-world applicability of our model.

3 Long Short-Term Memory Networks (LSTM)

LSTM (Hochreiter and Schmidhuber, 1997) are a category of recurrent neural networks (RNN) specifically designed to learn and exploit long-term dependencies in the input data. Classic RNN architectures can also, in theory, take account of durable relationships, but in practice were shown to suffer from a number of problems related to their backpropagation-based optimization algorithm, such as the well-known issue of the gradient slowly vanishing or exploding over time (Bengio et al., 1994). A LSTM Network features the same chain-like structure of repeating modules typical of a RNN, but consists of more sophisticated modules.

LSTM networks have successfully been applied to a variety of tasks including image captioning (Vinyals et al., 2014), vocabulary speech recognition (Xiangang and Xihong, 2014) and handwriting recognition (Graves et al., 2009), while in the context of time series forecasting they have been shown to greatly outperform simpler approaches such as ARIMA (Siami-Namini and Namin, 2018). In the present work we built upon the observation that the very nature of stock prices, determined by a combination of long-standing factors and continuous, pseudo-random fluctuations, renders them especially suitable for prediction via an LSTM Network. The choice of this model, capable of taking into account both kinds of relationships and to fully grasp the dynamic structure of the data, appears indeed more advantageous than a HMM-based approach, characterized by absence of memory and thus lack of temporal perspective.

4 Portfolio Optimization

Diversification is a crucial financial factor that consists in allocating funds in such a way as to limit the exposure to a single potential source of peril. Portfolios composed of a single stock entail no diversification as their performance is entirely determined by the fluctuations in price of one asset. Diversified portfolios, which are composed of multiple almost uncorrelated assets, have the advantageous property of being characterized by lower variance (thus volatility) than the weighted average of the variances of the single assets they comprise (O'Sullivan and Sheffrin, 2003). A mathematical tool commonly employed to identify the portfolio featuring the best combination of diversification and performance is portfolio optimization (Markowitz, 1952). Given a number of assets of interest, portfolio optimization is the process of selecting their most desirable distribution, generally computed in the form of a set Θ of weights θ_s, one weight for each target stock s, summing to 1. Following the classical portfolio optimization theory (Engels, 2004), we computed Θ in two ways: The minimum variance portfolio approach (equation 1), which picks the distribution of weights that minimizes the variance of the portfolio regardless of its expected return, and the tangency portfolio approach (equation 2), which selects the one leading to the highest expected return per unit of risk:

$$\Theta_{MV} = \frac{\Sigma^{-1}\bar{1}C_0}{\bar{1}^T\Sigma^{-1}\bar{1}} \quad , \tag{1}$$

$$\Theta_{TG} = \frac{\Sigma^{-1}\mu C_0}{\bar{1}^T \Sigma^{-1}\mu} \quad , \tag{2}$$

where μ denotes the portfolio's expected return, C_0 the initial capital and Σ the covariance matrix of the returns of all stocks in the portfolio. In either case, the optimization of the portfolio does not need normal distribution assumptions regarding the return distribution to be optimal. We also evaluated the equal weight portfolio approach, consisting in attributing a stable $\frac{1}{N}$ fraction of the capital to each of the N stocks in the portfolio.

5 Dataset

As is established practice in the field of financial modeling (Nguyen, 2014), (Nguyen, 2017), in our analysis we make exclusive use of stocks listed on the S&P 500 Index (Standard & Poor's Global, 2019), a stock market index gathering 500 large market capitalization companies mainly listed either on the NYSE (NYSE, 2019) or the NASDAQ (NASDAQ, 2019). Given the size of the companies it comprises, the S&P 500 is often regarded to as a bellwether for the United States' stock market and, by extension, the whole world's. The performance of every stock listed on the S&P 500 is constantly monitored and most S&P 500-based financial datasets generally contain information on the daily open, close, high and low prices of each company. These are, in order, the price of the stock at the opening of the market, the price at its closing and the highest and lowest price reached throughout the same day. Among those, for our predictions we solely consider the closing prices, which are known to be the best indicators of the market's actual conditions due to large-scale investors often dragging their positioning choices into each day's last few trading hours. The dataset we used, retrieved using the Yahoo Finance Python API (Yahoo Finance, 2019), holds information on the daily open, close, high and low prices of all stocks used throughout our work (MSFT, AAPL, AMZN, FB, JPM, GOOG, GOOGL, JNJ, XOM, V, PG, T, BAC, HD, VZ, MA, CVX, DIS, INTC) on the interval 02.01.2008 through 13.09.2019. The data concerning the two stocks Visa (V) and Facebook (FB), which were first traded on the S&P 500 on 20.03.2008 and 18.05.2012 respectively, is clearly missing for earlier points in time.

6 Experimental Design

6.1 LSTM Design

In our experiment we designed a LSTM network whose hyperparameters were selected after an extensive grid search. The aim of the grid search was to identify a single set of hyperparameters that could lead to satisfying results on all considered stocks. The problem is non-trivial: Stocks can exhibit exceedingly different behaviors with regard to the degree of volatility and magnitude of the starting price, which renders the conception of a single, overarching model a rather challenging task. On account of this matter, singling out a satisfactory set of hyperparameters requires that the architecture of our LSTM be trained on several years' worth of data. The set of parameters was chosen that returned the most accurate forecasts on the test data across all stocks. This is presented in Table 1. As part of the implementation, we normalized the data and iteratively adjusted the learning rate via the Adam Optimizer (Kingma and Ba, 2014) in order to accelerate training and improve generalization.

Table 1: LSTM structure and parameters of the model.

Window Size	Nodes per layer	Epochs	Training Data	Test Data
100	100; 50	130	01.2008-08.2016 and 01.2008-09.2019	08.2016-08.2017 and 09.2018-09.2019

Following Nguyen (2017), both model training and predictions were performed using a sliding window approach: The individual close prices from each company were divided into sequences of length w, defined as the window size. Training starts on the data in the first time window w, with the datapoint at $t = 1$ being the first training value and the datapoint at $t = w + 1$ the target value for the first window. After training on the first time window is complete, the following time window, starting with datapoint 2, is used, with $t = w + 2$ being its target value. This scheme is repeated, shifting the entire window by one datapoint at a time, until the very last training datapoint is a target value. At this point, one epoch is complete. In our case, as shown in Table 1, a total of 130 epochs were performed. Forecasting is done in the same way: Each datapoint to be predicted requires the trained architecture to be fed with the previous w datapoints. This approach was maintained in order to fully exploit the LSTM network's capacity to model both short- and long-term dependencies within the data. Accuracy of the forecasts was measured as a mean absolute percentage exam (MAPE) score.

6.2 Trading Component Design

As already mentioned above, in our experiments we made use of 2 different trading strategies: The simple sell-buy rule used in Nguyen (2017), which always takes into account the predictions output by the forecasting model, and a more complex portfolio optimization-based joint trading strategy, which can either consider the predictions or disregard them.

1. **Individual stocks trading via simple sell-buy rule**: If the forecasting model being used predicts the price of stock i on the following day p_{t+1} to be higher than the current price p_t, buy 100 units of stock i and sell all of them tomorrow; if the price is predicted to be lower, do not take any actions. In this configuration, the initial capital is the price of 100 stocks of i bought on the first day any trading was done.
2. **Portfolio optimization-based joint trading**: If the LSTM's predictions are neglected, simply compute the matrix of historical returns up to the day before the current trading day, based on the matrix of historical closing prices. If the LSTM's predictions are considered, stack the vector of forecast closing prices for the current trading day on the matrix of historical closing prices. Based on this, compute the matrix of historical returns. Either way, use the matrix of historical returns to compute, on a daily basis over the entire period, the minimum variance and tangency portfolio. Also compute the equal weight portfolio by attributing the same weight $\frac{1}{N}$ to each one of the stocks. For each one of the three approaches, compute the profit percentage at the end of the year. This involves, for each day t_{test} and each stock i, the purchase at price $p_{i,t_{\text{test}}}$ of the amount of stocks indicated by its daily weight θ_i and its sale the day after at price $p_{i,t+1}$. In this scenario, negative weights correspond to short sell orders,

consisting in the sale of non-owned stocks and their subsequent buyout the following day, and, therefore, involve the opposite procedure: Sell today, buy tomorrow.

6.3 Experiment 1

In the first experiment we employed our LSTM model to forecast the closing prices of FB, GOOG and AAPL over one trading year, then used the predictions as indicator for trading following the trading strategy 1 mentioned above. Closing prices from the period 02.01.2008 - 14.08 .2016 were used as training data and the daily closing prices from the period 15.08.2016 - 11.08.2017 (251 trading days) as test data. The LSTM's MAPE values obtained by our network were compared to the MAPE values from the HMM model and the naïve model in Nguyen (2017). Further, the HMM and naïve predictions were also used as indications for trading based on trading strategy 1, and the results thus achieved were compared to the ones attained by our LSTM.

Table 2: Results of experiment 1. Columns 6 to 8 refer to the individual trading based on the forecasts of the respective models, while column 9 refers to the best Portfolio Optimization-based trading.

		MAPE (%)			**Profit (%)**			
Trading Period	Stocks	Naïve	HMM	LSTM	Naïve's	HMM	LSTM	$LSTM_{PO}$
2016-2017	AAPL	1.33*	1.13*	1.04	32.47*	31.91*	31.98	**32.32**
	FB	2.13*	1.16*	1.12	20.54*	23.53*	31.53	
	GOOG	1.37*	1.07*	1.04	3.4*	24.86*	13.74	
	Avg	**1.61**	**1.12**	**1.06**	**18.8**	**26.76**	**25.75**	
2018-2019	AAPL	2.14	1.65	1.25	-4.23	8.22	13.31	**3.59**
	FB	2.27	1.83	1.26	-2.81	19.76	24.55	
	GOOG	1.90	1.64	1.16	-2.56	2.60	2.78	
	Avg	**2.10**	**1.70**	**1.22**	**-3.20**	**10.19**	**13.55**	

* Result taken directly from Nguyen (2017).

We thus generalized these results by applying our model to a different interval of time, specifically the latest trading year up until the day in which the experiment was carried out, meaning 14.09.2018 to 13.09.2019 (again 251 trading days), for the same three stocks. We implemented the Naïve and HMM models from Nguyen (2017) and assessed their performance on the same time window. All results are summarized in Table 2.

6.4 Experiment 2

In our second experiment we aimed at further generalizing our findings to a larger number of stocks. To this end, the 19 stocks bearing the greatest influence on the S&P 500, namely MSFT, AAPL, AMZN, FB, JPM, GOOG, GOOGL, JNJ, XOM, V, PG, T, BAC, HD, VZ, MA, CVX, DIS and INTC were considered. This specific set of stocks was chosen in the light of its great degree of representativeness over the entire index: Since the 10 largest companies listed on the S&P500 account for as much as 21.8% of the performance of the index (Indices, 2014), they are very much indicative of the entire group of 500. The 9 extra stocks were used for the sake of thoroughness. Except for the stock selection, experiment 2 was carried out in exactly the same fashion as experiment 1. We implemented the naïve and HMM forecast models from Nguyen (2017) and compared their performance, both on forecasting and trading, with that of our model. The results are shown in Table 4.

Table 3: Rate of return obtained with all three PO strategies (minimum varianve (MV), tangency (TG), equal weight (EQ). Subscript F indicates the cases in which the LSTM forecasts were taken into account. Rows 1 and 3 pertain to experiment 1, while row 2 is part of experiment 2.

Trading Dates (No. of Stocks)	MV (%)	MV_F (%)	TG (%)	TG_F (%)	EQ (%)
2016-2017 (3)	29.6	29.6	23.3	23.3	**32.3**
2016-2017 (19)	0.0	0.0	28.2	**29.7**	16.9
2018-2019 (3)	3.3	3.4	3.6	4.2	**7.1**

Table 3 shows the profit attained by every PO strategy on every considered time interval and every experiment. It can be seen that the use of the forecasts improves, although slightly by reason of the much larger weight carried by the historical prices (several hundred days' worth of historical prices over a single day of prediction), the profit associated with the PO-only trading strategy.

Table 4: Rules of experiment 2 results. Also in this case column 8 refers to the result of the best PO strategy out of the three considered. The results obtained by all PO strategies are shown in Table 3.

	MAPE (%)			**Profit (%)**			
Stocks	Naïve	HMM	LSTM	Naïve's	HMM	LSTM	LSTM_{PO}
MSFT	1.02	0.83	0.89	3.61	31.07	23.44	
AAPL	1.11	1.52	0.92	32.45	30.78	29.96	
AMZN	1.30	1.21	0.98	22.49	26.30	16.06	
FB	1.16	1.47	1.07	20.44	14.01	30.72	
JPM	1.06	0.90	0.94	25.87	44.23	7.38	
GOOG	1.05	0.94	0.89	3.46	6.11	12.78	
GOOGL	1.03	1.71	0.87	14.32	2.77	-1.66	
JNJ	0.76	0.74	0.76	9.18	7.53	-4.04	
XOM	1.01	1.17	0.81	-14.05	-14.85	4.99	**29.7**
V	0.96	1.05	0.82	-1.89	16.15	26.39	
PG	0.85	0.75	0.76	-7.36	0.02	16.77	
T	1.01	1.09	0.92	-5.13	-7.56	-9.24	
BAC	1.52	1.20	1.05	34.27	33.47	20.39	
HD	0.98	1.31	1.08	7.39	8.49	13.15	
VZ	1.09	0.96	0.91	7.70	4.01	-21.14	
MA	1.01	0.76	0.83	6.51	15.86	42.20	
CVX	1.08	1.37	0.87	-0.97	12.89	7.26	
DIS	0.89	1.34	0.81	-8.81	1.03	11.65	
INTC	1.11	1.02	0.87	-7.61	7.52	14.58	
Avg	**1.05**	**1.12**	**0.90**	**7.46**	**12.62**	**12.72**	

7 Results and Discussion

Experiment 1 aimed at comparing the performance of our strategy with that of another state-of-the-art model, namely the HMM proposed in Nguyen (2017), on the same three stocks that were considered on this baseline paper. The first portion of the experiment, that is to say the one carried out on trading year 2016-2017, is a direct comparison with the baseline, while the second time interval (the most recent one available at the time of the experiment) was evaluated for the sake of generalization. As shown by Table 2, our LSTM-based strategy attains a lower MAPE (thus higher accuracy) on all three stocks and in both time intervals than both the naïve and HMM-based approaches. On the side of the profit, our trading component achieves higher returns than the baseline's on two out of three stocks in trading year 2016-2017, and on three out of three stocks in trading year 2018-2019. When Portfolio Optimization is added to the picture, the profit increases further in the first trading year, while it decreases in the second one. In both cases, however, Portfolio Optimization permits to increase the diversification of the portfolio, which is a key financial feature that greatly enhances the real-world applicability of our model: A trading strategy based on the deal of a single stock could hardly find any use at all in reality as it would entail the outright lack of diversification, therefore being subject to an excessive amount of risk. By adding the portfolio optimization component, we instead allow for the simultaneous trade of any number of stocks, which permits to impact diversification, hence risk.

This is exemplified by the results obtained in experiment 2 and presented in Table 4: The individual profit gathered on several stocks is negative, which would be a disastrous outcome for an hypothetical lender investing solely on this specific share. The rate of return collected on the portfolio optimization-based strategy, besides being more than twice as high as the average of the 19 individual rate of returns (which is not necessarily always the case), is instead evidently less collapse-prone. An objective of experiment 2 was to generalize the findings of experiment 1 by considering a broader set of stocks, and particularly the 19 stocks bearing the greatest influence on the S&P500 index. In this case, as again shown in Table 4, our LSTM returns forecasts on average almost 20% more accurate than the HMM's. In the individual trading section, although the rate of return appears to depend very much on the stock being considered, our strategy is on average 0.8% more profitable than the HMM one and as much as 70% more profitable than the naïve one. Lastly, adding portfolio optimization stunningly leads to an average increase in profit of 133% over the LSTM-based strategy without PO, 135% over the HMM-based one and 298% over the naïve-based one. In addition to this, all considerations on Portfolio diversification mentioned above, of course, hold true in this case.

Figure 1: Experiment 2. Visualization of the growth of the initial capital through an entire trading year for all three PO strategies (tangency (TG), minimum (MV), equal weight (EQ)).

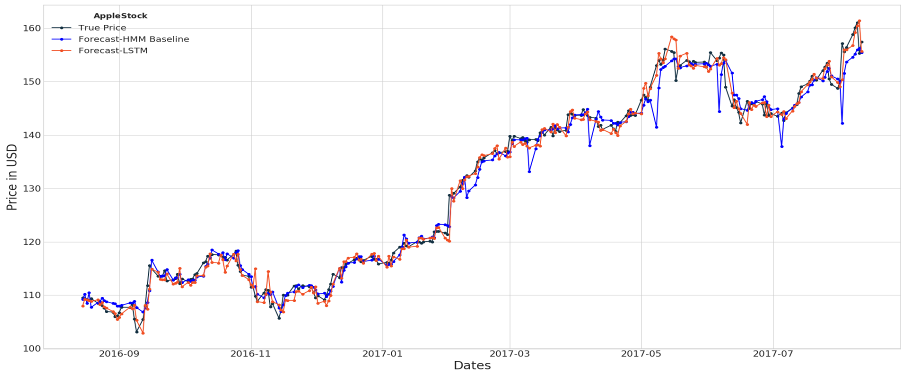

Figure 2: Forecast comparison between the HMM Baseline and the implemented LSTM method on 251 Trading Days (15.08.2016 through 11.08.2017) for stock Apple (AAPL).

Figure 1 depicts the evolution of the initial investment registered by several portfolio optimization strategies in the course of experiment 2. The same weight (EQ) strategy (blue line) and the minimum variance (MV) strategy with and without forecasts (orange and green line respectively, overlapped), all relatively safe strategies, show a stable increase in the value of the initial investment through the whole year, but not a very prominent one. The best-performing strategy is instead the tangency (TG) one: Even though the first few months see a downward trend, the return at the end of the trading year is the highest of all approaches.

Overall, our model is both quantitatively and qualitatively superior to the naïve and HMM-based competitors: On the quantitative side, the forecasts produced by our LSTM are on average 30% more accurate than the Naïve's and on average 17% better than the HMM's. A visual comparison of the predictions of the three approaches on one of the considered stocks is given in Figure 2. On the qualitative side, portfolio optimization overcomes the 1-stock limitation, increasing the practical relevance of the model. Our approach to portfolio optimization is itself an innotivation of the state of the art, as we apply it on forecast values rather than on observed ones as the standard rule would demand.

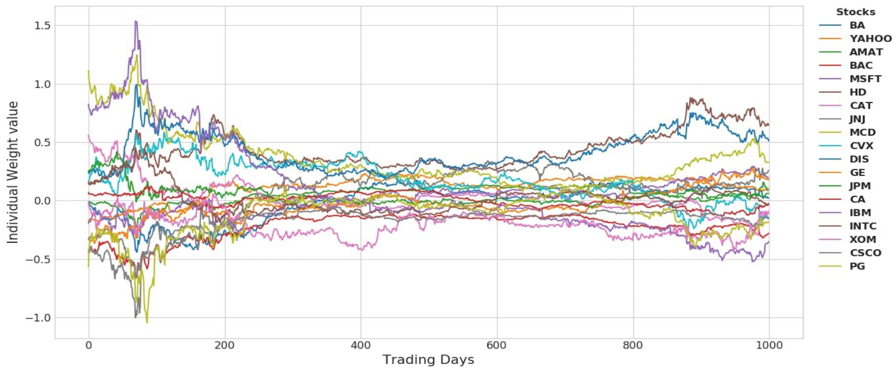

Figure 3: Daily evolution of the weights of a Tangency Portfolio over 1000 trading days as recorded during one of our experiments. Negative weights correspond to short sell orders.

8 Conclusions

In this paper we proposed a pipeline for the forecast of the closing price of stocks on the stock market and their efficient trade. In the forecasting component we used a LSTM network trained on eight years' worth of daily close prices to forecast future close prices of a number of stocks listed on the S&P 500 index.

In the trading component, the LSTM's forecasts were used both independently and in conjunction to three different tools of modern portfolio optimization theory, namely the minimum variance, tangency and equal weight portfolio, to perform informed trading. We compared all results to those achieved by the naïve and the HMM-based scheme described in Nguyen (2017). On the side of the accuracy of the forecasts, considering all tested stocks and time intervals our model outperforms the naïve- and HMM-based competitors by an average 30% and 17%, respectively. On the side of the profitability of the trading strategy, the individual trading following the predictions of our LSTM leads to similar results to those registered when following the predictions of the HMM. When PO is added, however, we manage to globally increase the profitability of the trading component and its applicability to a real-world context.

Future work might involve additionally boosting the accuracy of our forecasts by utilizing a broader range of training data rather than the closing prices alone, and incrementing the coordination between the forecasting and portfolio optimization component, conceivably through a reinforcement learning-based framework.

Acknowledgements Riccardo Lucato , Edgar Jimenez, Eduardo Salvador Rocha, Yang Qi and Marina Gavrilyuk have equally contributed to this paper.

References

Bengio Y, Simard P, Frasconi P (1994) Learning long-term dependencies with gradient descent is difficult. IEEE Transactions on Neural Networks 5(2):157–166. DOI: 10.1109/72.279181.

Engels M (2004) Portfolio optimization: Beyond markowitz. Master's thesis, Leiden University, Rapenburg 70, 2311 EZ Leiden, Netherlands.

Graves A, Liwicki M, Fernández S, Bertolami R, Bunke H, Schmidhuber J (2009) A novel connectionist system for unconstrained handwriting recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 31(5):855–868. ISSN: 0162-8828, DOI: 10.1109/TPAMI.2008.137.

Gupta A, Dhingra B (2012) Stock market prediction using hidden markov models. In: 2012 Students Conference on Engineering and Systems, pp. 1–4. DOI: 10.1109/SCES.2012.6199099.

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Computation 9(8):1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

Indices SDJ (2014) S&P500 factsheet. Standard & Poor Global.

Kingma D, Ba J (2014) Adam: A Method for Stochastic Optimization. International Conference on Learning Representations.

Kritzman M, Page S, Turkington D (2012) Regime shifts: Implications for dynamic strategies. Financial Analysts Journal 68(3). URL: https://ssrn.com/abstract=2066848.

Lin TCW (2014) The new financial industry. Tech. Rep., Temple University, 1801 N. Broad Street Philadelphia, PA 19122 USA. URL: https://ssrn.com/abstract=2417988.

Markowitz H (1952) Portfolio selection. The Journal of Finance 7(1):77–91. DOI: 10.1111/j.1540-6261.1952.tb01525.x

Miao K, Chen F, Zhao ZG (2007) Stock price forecast based on bacterial colony RBF neural network. Journal of Qingdao University (Natural Science Edition) 2:50–54.

NASDAQ (2019) The NASDAQ Index. URL: https://www.nasdaq.com/.

Nguyen N (2017) An analysis and implementation of the hidden markov model to technology stock prediction. Risks 5(4). DOI: doi.org/10.3390/risks5040062.

Nguyen NT (2014) Probabilistic methods in estimation and prediction of financial models. PhD thesis, The Florida State University at Tallahassee. URL: https://diginole.lib.fsu.edu/islandora/object/fsu:254481/datastream/PDF/download/citation.pdf.

NYSE (2019) The NYSE Index. URL: https://www.nyse.com/index.

O'Sullivan A, Sheffrin SM (2003) Economics: Principles in Action. Pearson Prentice Hall, New Jersey. ISBN: 01-3063-459-X.

Pradeep A, Goyal S, Bloom JA, Cox IJ, Miller M (2013) A detection of statistical arbitrage using machine learning techniques in Indian stock market. Tech. Rep., Indian Institute of Technology, Kanpur. URL: https://cse.iitk.ac.in/users/cs365/2013/submissions/~deepu/cs365/project/report.pdf.

Reid J, Nicol C, Burns N, Chanda S (2017) Long-term asset return study, the next financial crisis. Tech. Rep., Deutsche Bank Markets Research AG London, Deutsche Bank.

Siami-Namini S, Namin AS (2018) Forecasting economics and financial time series: ARIMA vs. LSTM. CoRR abs/1803.06386. URL: http://arxiv.org/abs/1803.06386.

Standard & Poor's Global (2019) S&P500. Published via: https://us.spindices.com/indices/equity/sp-500.

The World Bank Group (2019) Market capitalization of listed domestic companies. Data retrieved from World Federation of Exchanges database.

Trippi RR, Desieno D (1992) Trading equity index futures with a neural network. Journal of Portfolio Management 19:27–33. DOI: 10.3905/jpm.1992.409432.

Vinyals O, Toshev A, Bengio S, Erhan D (2014) Show and tell: A neural image caption generator. CoRR abs/1411.4555. URL: http://arxiv.org/abs/1411.4555, 1411.4555.

Wang JZ, Wang JJ, Zhang ZG, Guo SP (2011) Forecasting stock indices with back propagation neural network. Expert Systems with Applications 38(11):14346–14355. DOI: https://doi.org/10.1016/j.eswa.2011.04.222.

Wang YF (2003) Mining stock price using fuzzy rough set system. Expert Systems with Applications 24(1):13–23. DOI: https://doi.org/10.1016/S0957-4174(02)00079-9.

Xiangang L, Xihong W (2014) Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition. CoRR abs/1410.4281. URL: `http://arxiv.org/abs/1410.4281`, 1410.4281.

Yahoo Finance (2019) The S&P500 index historical prices. URL: `https://finance.yahoo.com/quote/%5EGSPC/history?p=%5EGSPC`.

Zhang Y, Wu L (2009) Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network. Expert Systems with Applications 36(5):8849–8854. DOI: https://doi.org/10.1016/j.eswa.2008.11.028.