Comparison of deep learning algorithms for forecasting stock returns and portfolio optimization

Extended abstract

Sarah Witzman
switzman@princeton.edu

Mathieu-Savvas Dimitriades
md0101@princeton.edu

Keywords. Deep Learning, Neural Networks, Stock Returns Predictability, Time Series Forecasting, Performance Evaluation Criteria

Abstract. In our paper, we explore several deep learning algorithms applied to the prediction of stock returns of stocks in the CAC40, the French stock index. The algorithms we consider are: **Multi-layer perceptron, Convolutional, Recurrent (LSTM, GRU) and Transformer neural network architectures**. Since the main objective of market practitioners is as much to predict the effective return of an asset than to predict its sign [2], we equally focus on two tasks: prediction of stock returns and binary classification of the sign of those returns (+ / -) and compare the performance of the two approaches. For comparison, we will also consider predictions obtained using linear regression and logistic regression. An important aspect we analyze is the performance criteria used to assess these algorithms [2]. Once the forecasts of our models obtained, we propose the use of Enhanced Portfolio Optimization [16] to combine these forecasts into a trade strategy. This enables us to evaluate the **performances of the investment strategies considered against those of the CAC40 index**.

1 Introduction

Our paper explores the application of deep-learning techniques to predicting stock market returns. More specifically, we will try to forecast 1-day ahead returns of stocks in the CAC40 index using several deep-learning algorithms trained on three types of features: fundamental, macroeconomic and technical [3]. **Can deep learning techniques effectively forecast future returns from past information? How can we build, based on those forecast, a tradable portfolio? Can this portfolio outperform the market? How can we evaluate the quality of a given model?** Those are the questions we want to tackle in our study.

2 Related work

To the best of our knowledge, our paper is the first paper to specifically benchmark various deep learning algorithms and compare their performance based on the profit and loss of a strategy which combines the forecasts given by those models using portfolio optimization [16].

[3] propose a deep learning methodology based on long-short term memory (LSTM) to forecast next day returns of the S&P500 index. [21] propose a deep learning framework that combines stacked auto-encoders and LSTM models to predict stock returns. [22] used a deep learning model called WaveNet to predict stock price trends based on daily stock price data. [23] used a deep learning model with both LSTM and GRU layers to predict stock prices based on both numerical and textual information from financial news articles. [26] use return data and news sentiment analysis with various combinations of network architectures including MLP and LSTM. [27] propose an alternate combined LSTM/GRU architecture with both news sentiment analysis and a deep auto-encoder. [24] evaluate the performance of MLP, simple RNN, and LSTM models in predicting the returns of 4 stock groups within the Tehran stock exchange, finding that LSTM performed the best but also had the longest runtime. [25] compared MLP and CNN frameworks in forecasting the overall performance of the Indian National Stock Exchange.

However, the combination of the forecasts into a trade strategy was either non-existent ([3], [22], [24], [25], [26], [27]) or based on the simplistic criteria of going long the stock if the predicted returns are positive and going short the stock if the predicted returns are negative ([21], [23]). We want to further study the relevance of combining the forecasts using portfolio optimization in order to be able to compare the models considered in terms of both profitability and predictive capabilities. In addition to analyzing porfolio construction, we also aim to more thoroughly explore the performance of different models by directly comparing MLP, CNN, LSTM, GRU, and TFT architectures for the same dataset and optimization task.

3 Data

Our signal construction methodology is based on the work of [3] and we adapt their signals to the French stock market. We consider **fundamental, macroeconomic and technical data**. For each stock, the target variable is the one-day ahead forecast of the return of the stock in the case of prediction and the one-day ahead forecast of the sign of the return in the case of classification. The fundamental data we use is the lagged returns series. The macroeconomic data consists of the CBOE VIX index, the EUR/USD exchange rate, French unemployment rate and consumer sentiment index. The technical indicators consist of MACD, ATR and RSI.

4 Deep-learning algorithms considered

4.1 Multi-layer perceptron

The simplest deep networks are called multilayer perceptrons (MLP) [4], and they consist of multiple layers of neurons each fully connected to those in the layer below (from which they receive input) and those above (which they, in turn, influence).

4.2 Convolutional neural network

Convolutional neural networks are a class of neural networks that use a mathematical operation called convolution in place of general matrix multiplication in at least one of their layers [6]. CNNs are regularized versions of MLP. The full connectivity of the latter networks make them prone to overfitting data. Typical ways of regularization, or preventing over-fitting include: penalizing parameters during training (weight decay) or trimming connectivity (skipped connections, dropout...). CNNs take advantage of the hierarchical pattern in data and assemble patterns of increasing complexity using simpler patterns implied by their filters.

4.3 Recurrent neural network

RNNs are latent variables models, where the latent variables store information up to the previous time-step. They build on the Back-Propogation Through Time algorithm [9], which requires us to expand (or unroll) the computational graph of an RNN. The unrolled RNN is essentially a FFN with the property that the same parameters are repeated throughout the unrolled network, appearing at each time step. The gradient with respect to each parameter must be summed across all places that the parameter occurs in the unrolled net. This leads to complications because sequences can be rather long, leading to vanishing and exploding gradient issues [10]. Two fixes to the vanishing / exploding gradient where proposed with the Long-Short Term Memory (LSTM) and Gated Reccurent Unit (GRU) architectures on which we will focus in our study.

4.4 Transformer

The last architecture we consider is the transformer architecture [8]. Transformers are able to capture long-range dependencies and interactions, a feature very attractive for time series modeling. The core idea behind the transformer model is the attention mechanism. In our paper, we consider a specific transformer called **Time-Fusion Transformer** [14].

5 Portfolio construction

As described in the introduction, each of our prediction algorithms will try to forecast the returns or the sign of those returns. Although we could use the same algorithm for both tasks, simply taking the sign of the forecasted returns as a forecast for the sign for the returns, our approach is to use regression models to forecast returns and classification models to forecast the sign of these returns. This is because we believe that the increased simplicity of the second task may lead to better prediction performance overall. For each of these tasks, we propose a portfolio construction methodology.

6 Results

For all the models considered, we apply time-series validation to evaluate their performances. We apply the extended window approach to validate our time-series models. We obtain, in this section, a benchmark of the performance of each of the models considered against the performance of the baseline linear models and the CAC40 market index. We hence give the performance of the best performing model for each model type (MLP, CNN, LSTM, GRU, TFT) as well as the performance of the market index (CAC40) during the same period.

7 Further directions

The main improvement areas of our paper are as follows:

- To avoid predicting noise in the sign of the returns, we could perform a three class classification, assigning a label of zero to the returns that are not significantly different from zero.
- Due to limited computational capabilities, the scope of the cross-validated parameters was relatively narrow. Increasing the range of cross-validated parameters can lead to better results overall.
- Our models could be explored in the context of different markets and financial data-sets to observe their performance truly out-of-sample.
- More advanced models which combine different architectures (CNN + LSTM, CNN + GRU, GRU + LSTM...) were not included in our study and could lead to better performance overall.
- Further research could be conducted regarding our covariance matrix estimation methodology.

References

[1] Huang, J., Chai, J., Cho, S. Deep learning in finance and banking: A literature review and classification, 2020

[2] Jean Dessain, Machine learning models predicting returns: Why most popular performance metrics are misleading and proposal for an efficient metric, Expert Systems with Applications, Volume 199, 2022

[3] Hum Nath Bhandari, Binod Rimal, Nawa Raj Pokhrel, Ramchandra Rimal, Keshab R. Dahal, Rajendra K.C. Khatri, Predicting stock market index using LSTM, Machine Learning with Applications, Volume 9, 2022

[4] Sarker, I.H. Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. SN COMPUT. SCI. 2, 420, 2021

[5] Cybenko, G., Approximation by superpositions of a sigmoidal function. Mathematics of control, signals and systems, 2(4), 303–314, 1989

[6] Ian Goodfellow and Yoshua Bengio and Aaron Courville. Deep Learning. MIT Press. p. 326, 2016

[7] Serkan Kiranyaz and Onur Avci and Osama Abdeljaber and Turker Ince and Moncef Gabbouj and Daniel J. Inman, 1D Convolutional Neural Networks and Applications: A Survey, 2019

[8] Ashish Vaswani and Noam Shazeer and Niki Parmar and Jakob Uszkoreit and Llion Jones and Aidan N. Gomez and Lukasz Kaiser and Illia Polosukhin, Attention Is All You Need, 2017

[9] George Bird and Maxim E. Polivoda, Backpropagation Through Time For Networks With Long-Term Dependencies, 2021

[10] Bengio, Simard and Frasconi, Learning long-term dependencies with gradient descent is difficult. IEEE transactions on neural networks, 5(2), 157-166, 1994

[11] Hochreiter and Schmidhuber, Long short-term memory, Neural computation, 9(8), 1735–1780, 1997

[12] Cho, Van Merriënboer, Bahdanau and Bengio. On the properties of neural machine translation: encoder-decoder approaches, 2014

[13] Jimmy Lei Ba, Jamie Ryan Kiros and Geoffrey E. Hinton, Layer Normalization, 2016

[14] Bryan Lim, Sercan O. Arik, Nicolas Loeff and Tomas Pfister, Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting, 2020

[15] Harry Markowitz, Portfolio Selection, The Journal of Finance, Vol. 7, No. 1., pp. 77-91, 1952

[16] Lasse Heje Pedersen, Abhilash Babu, and Ari Levine, Enhanced Portfolio Optimization, Financial Analysts Journal, 77(2): 124-151, 2021

[17] Jerome Friedman, Trevor Hastie and Robert Tibshirani, Sparse inverse covariance estimation with the graphical lasso, 2007

[18] Hendrik Bessembinder, Trade Execution Costs and Market Quality after Decimalization, The Journal of Financial and Quantitative Analysis, Vol. 38, No. 4, pp. 747-777 (31 pages), 2003

[19] Bell, F., Smyl, S., 2018. Forecasting at Uber: An introduction. Accessed on 2023-04-23. `https://eng.uber.com/forecasting-introduction/`

[20] `https://optuna.org/`

[21] Wei Bao, Jun Yue and Yulei Rao, A deep learning framework for financial time series using stacked autoencoders and long-short term memory, 2017

[22] Jingyi Shen and M. Omair Shafiq, Short-term stock market price trend prediction using a comprehensive deep learning system, 2020

[23] Ryo Akita, Akira Yoshihara , Takashi Matsubara, Kuniaki Uehara, Deep learning for stock prediction using numerical and textual information, 2016

[24] M. Nabipour, P. Nayyeri, H. Jabani, A. Mosavi, E. Salwana, and S. S., "Deep Learning for Stock Market Prediction," Entropy (Basel), vol. 22, no. 8, p. 840, Jul. 2020, doi: 10.3390/e22080840.

[25] Mukherjee, S., et al.: Stock market prediction using deep learning algorithms. CAAI Trans. Intell. Technol. 8(1), 82–94 (2023). https://doi.org/10.1049/cit2.12059

[26] B L, S. and B R, S. (2023), "Combined deep learning classifiers for stock market prediction: integrating stock price and news sentiments", Kybernetes, Vol. 52 No. 3, pp. 748-773. https://doi-org.ezproxy.princeton.edu/10.1108/K-06-2021-0457

[27] K. Rekha and M. Sabu, "A cooperative deep learning model for stock market prediction using deep autoencoder and sentiment analysis," PeerJ Comput Sci, vol. 8, p. e1158, Nov. 2022, doi: 10.7717/peerj-cs.1158.

[28] Jimmy Lei Ba and Jamie Ryan Kiros and Geoffrey E. Hinton, Layer Normalization, 2016