Attention Long Short Term Memory: Evaluation of Time Series Forecasting Performance, a Comparison With LSTM and SARIMA

Giorgio Acquati

Supervisor: Dr. J. W. Böhmer

27th June 2021

Abstract

Long Short Term Memory is a standard architecture for recurrent neural network in the field of time series analysis. It is composed of a chain of cells, each of which is able to pass information to the following cell in the chain. This structure clearly causes the ability of the model to retain memory of distant past values to be strictly bound to the computational requirements of the model, supposedly limiting its performance. This paper proposes a new architecture, which allows cells to directly access information from all previous cells in the chain, allowing for better information passing between time steps. In the present study, we describe the implementation details of the proposed model and focus on assessing its performance regarding time series forecasting, by performing a comparison with LSTM and SARIMA models.

1 Introduction

Long Short Term Memory (LSTM) layers are a particular type of *Recurrent Neural Network* (RNN) that have become standard in the field of time series forecasting, classification and regression. These neural networks are designed to retain relevant information about past values of the input data, and are trained to interpret such information to predict the value of a time series one time step ahead [1]. If we take the output of one cell and use it as input for the next cell repeatedly, we can allow this architecture to forecast multiple time steps ahead, by using each time step forecast to predict the next one. However, the architecture of LSTM requires information that needs to be passed from one time step to another to also go through the intermediate time steps between the origin and the destination. This poses a significant limitation to the ability of LSTM's to retain information about distant past values and to learn patterns in the time series that span across several time steps, supposedly resulting in a loss of performance.

In recent years, LSTM's and other RNN's have been surpassed by *Multi-headed Self Attention Layers*, also known as *Transformers*, in the domain of natural language processing [2]. Unlike LSTM's, this new architecture does not require the time series data to be processed in order, but allows for parallel processing of the input data, significantly reducing the required computational needs. This is done through a machine learning mechanism known as *Attention*, a technique that loosely resembles cognitive attention. Attention aims at building a model which is capable of recognizing more important parts of the input, allowing to concentrate the effort of the computation on more relevant information [2].

An alternative is to allow LSTM layers to retrieve information directly from each of the preceding time steps. This approach would drastically improve transmission of information between distant time steps, potentially allowing this new architecture to recognize patterns in the time series that span across significantly more time steps, while preserving the relationship of causality between time steps. This architecture, which takes the name of *Attention Long Short Term Memory* (ALSTM), would hopefully overcome some of the issues faced by traditional LSTM layers and outperform it in some of the traditional tasks from the domain of time series analysis.

The aim of this research is to describe how the proposed ALSTM architecture can be implemented, in order to provide a proof of concept and to assess its performance in the task of time series forecasting. The performance of the model is compared to that of the *Seasonal Autoregressive Integrated Moving Average* model (SARIMA) [8], which is an extension of the *Autoregressive Moving Average* model (ARMA) [3], a more classical approach to time series forecasting. The ARIMA extension of ARMA allows to forecast non-stationary time series by working on a differentiation of the time series, while the SARIMA extension is particularly suited to forecast time-series which show seasonal trends. An additional comparison is then made to traditional LSTM layers, which aims at showing whether the proposed architecture can overcome LSTM's limitations. Moreover, in the context described in Section 2, the performance of ALSTM is also compared to that of forecasting models used in in the electric grid operator industry. Such comparisons aims at giving context to the experimental results obtained by the proposed model to find whether ALSTM can provide a valid tool for time series forecasting, potentially opening up for further research on the matter.

2 Methodology

2.1 Implementation

The proposed ALSTM model was implemented in Python 3.8, using the *PyTorch* machine learning framework. The reference LSTM implementation used for the comparison is the default one provided with PyTorch [1]. Moreover, the SARIMA model implementation used comes from the *pmdarima*[2] Python library.

2.2 Data Set and Context

The data set used to assess the performance of the proposed model consists of 4 years of hourly readings of the Spanish electric grid load in megawatt published on the *ENTSO-E* portal [3]. The data set covers the entire period from January 1, 2015 to December 31, 2018. Additionally, the data set contains hourly weather measurements for Spain's five most populated cities, namely Barcelona, Madrid, Valencia, Seville and Bilbao from the

[1]`https://pytorch.org/docs/stable/generated/torch.nn.LSTM.html`
[2]`https://pypi.org/project/pmdarima/`
[3]`https://transparency.entsoe.eu/`

OpenWeather portal [4]. A copy of the entire data set is publicly available on the *Kaggle* website [5].
The weather data set consists of 9 features for each of the cities, for a total of 45 features. Out of 9 features only the following 2 were used in practice, due to computational limitations:

- Temperature, measured in Kelvin
- Rain in the last hour, measured in millimeters

The following 7 features are present in the data set, but were not used during the conducted experiments:

- Atmospheric pressure, measured in hectopascal
- Relative humidity, expressed in percentage
- Wind speed, measured in meters per second
- Wind direction, measured in degrees relative to the North
- Snow in the last three hours, measured in millimeters
- Clouds coverage, expressed in percentage
- Weather description identifier

Moreover, the data set contains hourly one-day ahead forecasts for the total grid load made by the transmission system operator, based on unspecified weather forecasts and socioeconomic factors.

2.3 Prediction Task

The main task proposed consists of forecasting energy loads based on previous measurements of the grid load. Moreover, for ALSTM and LSTM weather measurements are included in the input features, which should allow for more accurate predictions of energy demand for both models. The forecasting task will take place over daily frequency data, due to computational limitations. In fact, switching to a daily data set reduces the number of rows by a factor of 24, which made it feasible to run the experiments on the limited hardware equipment available. To do so, the data is aggragated in the following way:

- Temperature data: the hourly temperature data is aggregated into daily minimum, maximum and average temperature measurements for each of the 5 cities, resulting in a total of 15 features, measured in Kelvin.
- Rain data: the hourly rain data is aggregated into a sum of the rain measurements for the day, measured in millimeters.
- Total grid load and total grid load forecasts: the grid load measurements are aggregated into a daily sum, which respectively represent the daily actual and predicted load for the day in mWh.

[4] `https://openweathermap.org/`
[5] `https://www.kaggle.com/nicholasjhana/energy-consumption-generation-prices-and-weather`

The SARIMA model used is a one-dimensional model, which does not support multi-features inputs and outputs. Therefore, the only input feature used by this model is the total grid load measurement.
The presence of forecasts made by the *transmission system operator* (TSO) allows to compare the considered models to a real world prediction model, opening up for conclusions regarding the utility of the considered models in a real world scenario. However, it is important to consider that the day ahead forecasts reported by the TSO are based on weather forecasts rather than actual weather measurements, which makes the comparison not entirely fair. Moreover, the source of the data set does not specify exactly what parameters are taken into consideration by the TSO to perform these forecasts. These circumstances pose a clear limitation to the conclusions that can be drawn by these comparison results, but are still valuable since they enrich the context of the assessment by providing an example of what results are currently achieved by models that are used in the real world.

2.4 Performance Metrics

The performance comparison between the models makes use of both qualitative and quantitative metrics, which aim at assessing the validity of the forecasts from multiple points of view. The proposed quantitative metrics are the following:

- *Root mean squared error* (RMSE): this basic metric portrays a generic picture of the prediction quality and provides a good starting point for a comparison between models.
- Root mean squared error over first order differentiation: when performing long forecasts, rather than simpler one time step ahead predictions, early errors in the prediction can cause the forecast to gain momentum towards an incorrect direction, propagating the error throughout the entire duration of the forecast. This effect can pollute the basic RMSE metric result, making it not suitable for long forecasts. To account for this, RMSE is also applied to first order differentiation of the forecast, which cancels out the long term effect of early predictions error and provides a more suitable quantitative metric for long term predictions.

Additionally, the performance assessment will contain a study of distribution of the prediction error. In many real world scenarios, the cost function associated to an error in prediction might not be linear. This means that outliers in the prediction error could generate large costs regardless of how well the model performs on average. To account for these scenarios, a description of the distribution of the prediction error is provided, which might allow to identify desirable or undesirable effects produced by each model.

3 Attention Long Short Term Memory Specification

3.1 Formal Description

In the context of multi-headed self-attention, the ALSTM architecture is specified by the following equations. First, the similarity matrix S is computed as follows:

$$S^k := exp\left(\frac{1}{\sqrt{d}} X Q^k K^k X^T\right) \odot L \tag{1}$$

where L is the lower triangular matrix. Multiplying by L in equation 1 allows message passing to happen only forward in time, which respects the relation of causality between time steps, typical of time series forecasting tasks. Then, the weight matrix W and hidden message H are computed as follows:

$$W^k := S^k \oslash \left(S^k 1 1^T\right) \tag{2}$$

$$H := \sum_{k=1}^{m'} \sigma_C \left(X \Theta^k\right) \tag{3}$$

These equations describe how a single *Attention head* can be implemented in this context. If we stack several of these heads in parallel and use the sum of their results as output, we obtain a multi-headed ALSTM.

3.2 Practical Implementation Details

The described ALSTM can not be executed in a sequential way like LSTM. In fact, with LSTM at each time step only the current input and the outputs from the previous time steps are needed to compute the output. Conversely, ALSTM needs input from all previous time steps.

In practical applications of LSTM and ALSTM, it might be desirable to have hidden messages that have larger dimensions than the output, to allow better communication between time steps. To overcome this, a simple linear layer can be applied on top of the base model, which allows to cast the hidden message to any desired output dimension.

4 Experimental Setup and Results

The two main experiments we will consider to assess the forecasting performance of the ALSTM model are the following:

- Following day prediction: this experiment evaluates the ability of the model to predict the total grid load for the day, based only on data from all previous time steps and weather measurements for the predicted day.
- Following 30 days forecast: this task consists in forecasting the total grid load for 12 distinct periods of 30 days each. The input will contain all measurements until the beginning of the 30 days period and only the weather measurements for the 30 days period.

The input features included for the prediction tasks are the following:

- Rain and temperature (minimum, maximum and average) data, for each of the five cities, amounting to 20 features.
- Total grid load for the previous period: in the case of one day ahead prediction, this feature contains the actual total grid load for the previous day. For the 30 days forecasting task this feature contains the total grid load forecast made by the model for the previous day. For the first of the 30 days, this feature contains the actual total grid load for the previous day.

The only target feature is the total grid load for the current day. This setup simulates a scenario in which we have extremely accurate weather predictions and we aim at forecasting energy consumption. At a given time step t, the model is fed data from all of the previous t - 1 time steps and at each time step outputs a prediction. The output at time step t is the prediction of the total grid load for the time step t, and is what we aim at optimizing. All the results reported in the following section are the average achieved over 7 runs of the tests, which should mitigate the variance caused by the randomness of the initialization of the model parameters. Sample standard deviations of the achieved results across the 7 runs are also reported along the average, in order to assess how consistently the models converge to the same solution.

4.1 Data Set Split

For ALSTM and LSTM, the 4 years daily data set is split in three parts. The first two years are used as training input and the results achieved over this period are used to optimize the model with gradient descent. Year 3 is used as a validation set. At each epoch, the model performs gradient descent on loss over years 1 and 2 data, and is tested over year 3. The instance at the epoch at which the year 3 loss is minimal is the one which is selected as the fully trained model for each run. Finally, the trained model is tested over year 4 data.
For SARIMA, in order to perform forecasts on year 4, the model must be trained on data up until the end of year 3. Therefore, the model was trained over the first three years of data. This could prove to be an advantage for SARIMA, since the model is trained on a significantly larger portion of the data set than LSTM and ALSTM. However, we have to consider that LSTM and ALSTM make use of the third year of data as well during the validation phase.

4.2 Data Set Normalization

The entire data set has been normalized column-wise to obtain values between 0 and 1 relative to maximums and minimums of the training portion (year 1 and year 2). As a consequence, all values from the training data set are within the range from 0 to 1, while occasionally values from the validation and test data set have values outside of these boundaries. This is done to mimic a real world forecasting task in which one would not know what the future maximums and minimums would be. All the reported losses and plots will show losses relative to the normalized data set, unless stated otherwise.

4.3 Hyperparameters

The following hyperparameters were used throughout these experiments:

- LSTM learning rate: 2×10^{-5}
- ALSTM learning rate: 4×10^{-6}
- LSTM and ALSTM hidden message size: 64
- ALSTM number of heads: 64
- SARIMA seasonality: 7, due to the daily nature of the data set

4.4 Following Day Prediction: Training

Figure 1: Following day prediction: RMSE loss over epochs for training data (year 1 and 2) of one sample run.

Figure 1 shows that both ALSTM and LSTM are able to correctly learn the training input series and to perform a stable gradient descent.

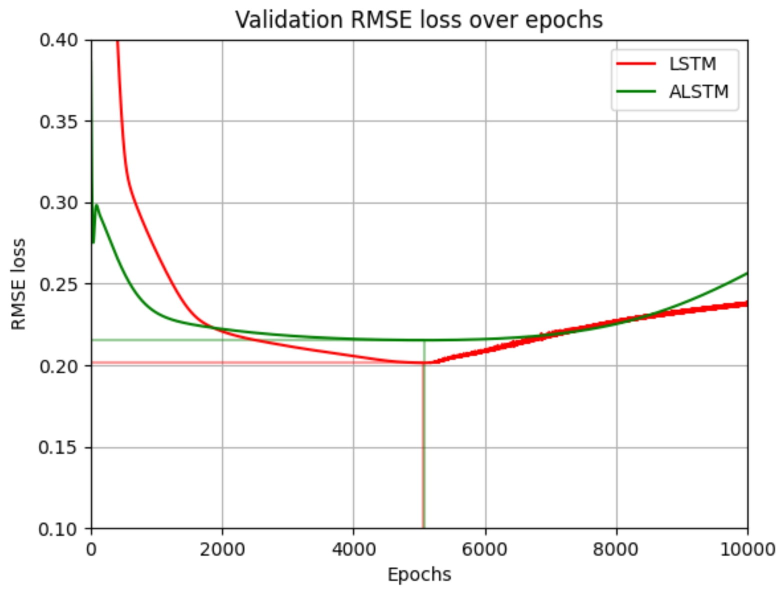

Figure 2: Following day prediction: RMSE loss over epochs for validation data (year 3) of the sample runs that obtained minimal loss over the validation set. The vertical lines indicate the epochs at which each of the models achieves the lowest loss respectively, while the lowest achieved losses are indicated by the horizontal lines.

Figure 2 shows that both ALSTM and LSTM are able to achieve similar losses over the validation data set. However, they both show a clear tendency to overfit, justifying the early stopping strategy adopted.

Model	Validation RMSE
ALSTM	$0,2219 \pm 0,0087$
LSTM	$0,2028 \pm 0,0016$

Table 1: Mean and sample standard deviations of RMSE losses over validation data of the 7 runs

Table 1 reports the average results achieved by the models in the 7 runs, up to two significant digits of the sample standard deviation. As we can see, the two models achieve very similar losses on the validation set, with LSTM coming slightly on top of ALSTM. However, while both samples standard deviations appear to be quite low in absolute terms, LSTM results show significantly lower standard deviation.

4.5 Following Day Prediction: Test Results

Model	Test RMSE
ALSTM	$0,2106 \pm 0,0059$
LSTM	$0,2110 \pm 0,0013$
SARIMA	$0,1802$
TSO Forecasts	$0,0193$

Table 2: RMSE over test data. LSTM and ALSTM results are reported as mean ± sample standard deviation over the 7 runs.

Table 2 reports the results achieved by the different models over the test data set. Analogously to validation, LSTM and ALSTM show very similar results to each other, and such results are consistent with the losses achieved over the validation data, indicating that the models correctly generalize to previously unseen inputs. However, both models achieve a slightly lower loss than the SARIMA model, which might be justified by the fact that SARIMA was trained on both the train and validation data (year 1 to 3). All the proposed models are outclassed by the forecasts made by the Transmission System Operator, which achieves RMSE of approximately one lower order of magnitude.

4.6 Following Day Prediction: Error Distribution

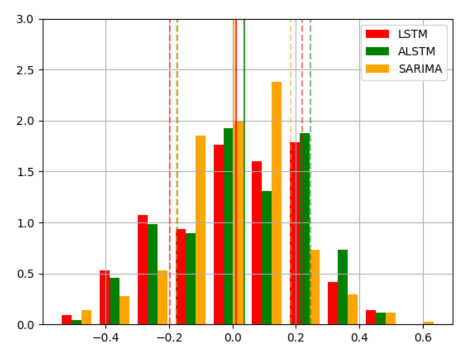

Figure 3: Histogram of the prediction error distribution achieved over the test data set. For LSTM and ALSTM, the best performing models out of the 7 runs over the validation set where used. The solid vertical lines indicate the mean, while the dashed lines indicate the standard deviation for each of the models. TSO forecasts were omitted due to their losses being extremely low compared to the other models. The histogram including the TSO forecasts is included in Appendix A.

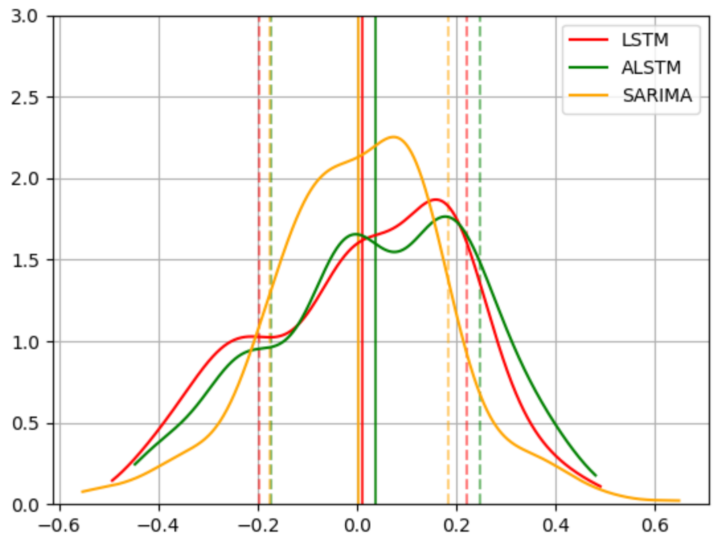

Figure 4: kernel density estimation of prediction error on the test data set. Scott's rule was used to choose the bandwidth parameters. For LSTM and ALSTM, the best performing models out of the 7 runs over the validation set where used.

Model	Mean	Sample Standard Deviation
ALSTM	-0,0233	0,210
LSTM	-0,0227	0,209
SARIMA	-0,00338	0,180
TSO Forecasts	0,000272	0,019

Table 3: Mean and sample standard deviation of the prediction error. For LSTM and ALSTM, the reported results are relative to all of the 7 runs.

Figure 4 shows the kernel density estimation of the distribution achieved by the considered models. ALSTM and LSTM show very similar distributions, with slightly different means. However, table 3 which takes into consideration all 7 runs, shows that the mean error obtained by both models are extremely similar. Qualitatively speaking, both distributions appear to be multimodal, with ALSTM showing a more pronounced trimodal tendency than LSTM. Conversely, SARIMA shows a much more clear normal distribution, with just a hint of multimodality close to the mean, which is consistent with SARIMA having lower RMSE over the test data set than ALSTM and LSTM.

4.7 Following 30 Day Forecast: Test Results

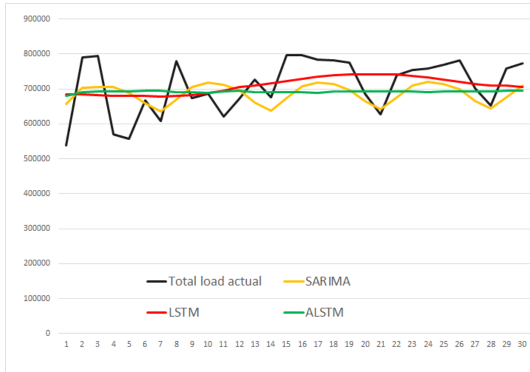

Figure 5: Forecasts for the first of the 30 days periods, unscaled. For LSTM and ALSTM, the best performing models out of the 7 runs over the validation set where used.

Model	RMSE
ALSTM	$0,218 \pm 0,028$
LSTM	$0,216 \pm 0,024$
SARIMA	$0,195 \pm 0,025$

Table 4: RMSE for the 30 day forecasting task achieved over the test data set, across all 12 periods. For ALSTM and LSTM, the reported results are relative to all of the 7 runs.

Model	RMSE
ALSTM	$0,226 \pm 0,028$
LSTM	$0,250 \pm 0,026$
SARIMA	$0,249 \pm 0,026$

Table 5: RMSE over first order differentiation for the 30 day forecasting task achieved over the test data set, across all 12 periods. For ALSTM and LSTM, the reported results are relative to all of the 7 runs.

Table 4 and Table 5 report the main results achieved by the models on the 30 days forecasting task. From these results we can see that the differences observed regarding following day forecasting performances carried over to the long term forecasting task. ALSTM and LSTM are within margin of error from each other, while SARIMA is at a slight advantage. However, this advantage is ironed out when considering the results relative to the first order differentiation of the time series, in which ALSTM comes slightly on top of the other two models, although always within margin of error.

5 Responsible Research

5.1 Ethical implications

Time series forecasting is a widespread field that finds several applications in the real world. A clear example of a real world application is presented by the forecasting task used to evaluate the performance of our model. In the context of electric grid load forecasting, being able to make accurate predictions allows the Transmission System Operator to make informed decisions about the operation of electricity generator. Additionally, with the rapid rise of renewable energy generation technology and smart grids, the relation between weather conditions and electricity supply is becoming more relevant [5] [6] [7]. Therefore, being able to properly forecast the electricity demand and supply enables the grid operator to predict electricity storage requirements, allowing for more informed decision regarding electricity trading. Moreover, higher quality long term forecasts of the energy demands allows to properly plan the construction of energy infrastructure. Given the rising relevance of the ecological issue, research in this field will play an increasingly important role in building a more sustainable society.
However, forecasts of any type should always be taken with a grain of salt. In fact, forecasts can convey a false sense of certainty and the ability of a model to forecast the consequences of exceptional events is limited to the amount of exposure to such events that the given model was given during training. Often, the availability of data regarding these events is severely limited due to their exceptional nature, which prevents adequate training of forecasting models to deal with these scenarios. Moreover, the reader should consider that forecasting models aim at reducing reality to a mathematical problem, which is always significantly simpler than the reality it represents. As a result, forecasts can never be considered certain and should always be reported with an accompanying confidence interval for practical applications.

5.2 Reproducibility of results

The experiments conducted can easily be reproduced. In fact, the code implementation of the ALSTM model using the PyTorch framework is trivial once its formal description is properly understood as presented in Section 3. Moreover, Section 3 also describes how LSTM and ALSTM can be adapted to produce outputs of any desired size. Additionally, Subsection 2.2 clearly reports the sources of the used data set and provides links to repositories where they are publicly available, while 4.1 clearly state how the data set was manipulated for the proposed experiments. Finally, Subsection 4.3 clearly states what hyperparameters were used to initialize the models.

6 Discussion

This section discusses the results presented in Section 4 for both experiments, under the methodology described in Subsection 2.4.

6.1 Following Day Forecasting Task

- Prediction error: the results presented in Table 2 show that ALSTM can provide a viable tool to perform short term time series forecasting. In fact, it manages to achieve performance virtually identical to that of LSTM, although slightly less consistent. However, both models were slightly outperformed by the simpler SARIMA, which suggests that the training might not have been sufficient to fulfill their potential. Moreover, it is possible that the chosen data set is particularly hard to forecast with the given data, which makes a much more simple model like SARIMA perform slightly better by betting on the seasonality the time series.

- Prediction error distribution: As shown in Figure 4, ALSTM and LSTM show a similar error distribution. In fact it appears that both models produce forecasting errors multimodally distributed, which might be undesirable in real world scenarios. Instead, SARIMA manages a much more composed error distribution that more closely resembles normality, which might be more desirable for practical applications. Table 3 shows that SARIMA manages a mean prediction error roughly one order of magnitude closer to 0, which along with the tighter distribution might justify the model slightly outperforming ALSTM and LSTM in the following day forecasting task.

6.2 Following 30 Days Forecasting Task

:

- First order differentiation of prediction error: the results reported in Table 5 show that ALSTM performance is in line with that of LSTM and SARIMA for long term time series forecasting. In fact, the slight difference between the average RMSE of the forecasts achieved by the three models is roughly within one standard deviation, which makes such disparities negligible. This shows that ALSTM is a suitable model for longer term forecasts, but fails at proving any kind of architectural advantage over LSTM.

7 Conclusions and Future Work

The results presented in Section 6 show that the proposed model is a viable tool for time series forecasting, both for short and longer forecasts, fulfilling the main objective of this research to provide a proof of concept of the model. However, the results indicate a failure to prove the superiority of ALSTM compared to LSTM, but rather showed how the older model produces forecasts of very similar quality. As mentioned before, this could be due to shortcomings in selecting a favorable prediction task for ALSTM, which did not allow it to leverage its architectural advantages regarding the ability of non adjacent time steps to communicate directly.

Moreover, the comparison with SARIMA suggests that the chosen time series might not have been particularly adequate to exploit the features of either ALSTM or LSTM. In fact, although the better performance achieved by SARIMA might be justified by the fact that the model was exposed to significantly more training data, we must consider that training an instance of SARIMA does not require any validation, making it fair to compare it to ALSTM and LSTM trained on a smaller data set. Moreover, the long term forecasts plot presented in Figure 5 show how SARIMA was far better at identifying the weekly seasonality of the time series, but still achieved similar results to ALSTM and LSTM when performing long term forecasts. This speaks to the difficulty of making accurate forecasts in the proposed context, suggesting that the considered models might not have been exposed to enough data during training.

As expected, the comparison with the TSO forecasts offers very little insights due to the limited available information regarding these forecasts, but suggests that the models used in the electric grid operators industry must have access to much more appropriate data than what was taken into consideration for this research.
The conducted experiments were significantly limited by the available computational power, which posed a considerable challenge in evaluating the model performance. In fact, training ALSTM over more than two years of data proved to be practically unfeasible with the available equipment. Moreover, these limitations prevented any kind of significant experiments to be conducted on the hourly data set, which would have enriched the context of the comparison.
Further work on the matter is needed to draw more definite conclusion about the performance of ALSTM, hopefully identifying scenarios in which the model is particularly adequate and outperforms its competitors either by reducing the required training effort or by achieving higher quality forecasts.

A Appendix A: Error Distribution Plots Including TSO Forecasts

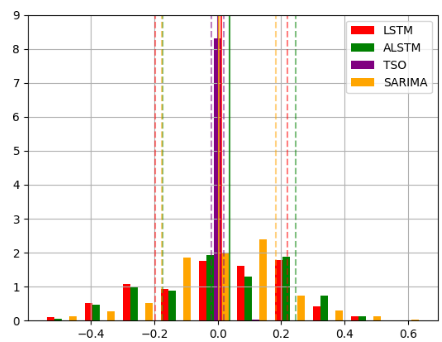

Figure 6: Histogram of the prediction error distribution achieved over the test data set, including TSO forecasts.

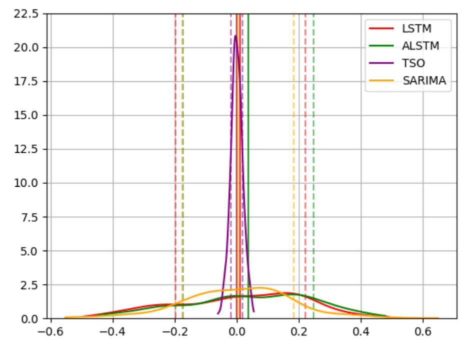

Figure 7: Kernel density estimation of the prediction error distribution achieved over the test data set, including TSO forecasts.

References

[1] Hochreiter, Sepp & Schmidhuber, Jürgen. (1997). Long Short-term Memory. Neural computation. 9. 1735-80. 10.1162/neco.1997.9.8.1735.

[2] Vaswani, Ashish & Shazeer, Noam & Parmar, Niki & Uszkoreit, Jakob & Jones, Llion & Gomez, Aidan & Kaiser, Lukasz & Polosukhin, Illia. (2017). Attention Is All You Need. Advances in neural information processing systems, pages 5998–6008.

[3] Whittle, P. (1951). Hypothesis testing in time series analysis. Uppsala: Almqvist & Wiksells boktr.

[4] Weglarczyk, Stanislaw. (2018). Kernel density estimation and its application. ITM Web of Conferences. 23. 00037. 10.1051/itmconf/20182300037.

[5] Perera K.S., Aung Z., Woon W.L. (2014). Machine Learning Techniques for Supporting Renewable Energy Generation and Integration: A Survey. In: Woon W., Aung Z., Madnick S. (eds) Data Analytics for Renewable Energy Integration. DARE 2014. Lecture Notes in Computer Science, vol 8817. Springer, Cham. https://doi.org/10.1007/978-3-319-13290-7_7

[6] Wang, Huaizhi & Lei, Zhenxing & Zhang, Xian & Zhou, Bin & Peng, Jianchun. (2019). A review of deep learning for renewable energy forecasting. Energy Conversion and Management. 198. 111799. 10.1016/j.enconman.2019.111799.

[7] Kaur, Devinder& Islam, Shama & Mahmud, Md. Apel & Dong, Z.Y.. (2020). Energy Forecasting in Smart Grid Systems: A Review of the State-of-the-art Techniques.

[8] Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: Principles and Practice. (2nd ed.) OTexts. https://otexts.org/fpp2/