

# Modelling Energy Demand Response Using Long-Short Term Memory Neural Networks

J. J. Mesa-Jiménez<sup>1,2,3</sup>, L. Stokes<sup>3</sup>, Q. Yang<sup>1</sup>, and V. N. Livina<sup>2</sup>

<sup>1</sup>Brunel University, Kingston Lane, Uxbridge, UK

<sup>2</sup>National Physical Laboratory, Hampton Road, Teddington, UK

<sup>3</sup>Mitie, The Shard, 32 London Bridge Street, UK

## Abstract

We propose a method for detecting and forecasting events of high energy demand, which are managed at national level in Demand Side Response programmes, such as the UK Triads. The methodology consists of two stages: load forecasting with Long-Short-Term-Memory neural network and dynamic filtering of the potential highest electricity demand peaks by using the exponential moving average. The methodology is validated on real data of a UK building management system case study. We demonstrate successful forecasts of Triad events with  $RRMSE \approx 2.2\%$  and  $MAPE \approx 1.6\%$  and general applicability of the methodology for Demand Side Response programme management, with reduction of energy consumption and indirect carbon emissions.

**Keywords**— Load Forecasting, Demand Side Response, Machine Learning, Long-Short Term Memory, Triad forecasting, Electricity Demand, Neural Networks

## Nomenclature

	MAE	Mean Absolute Error	
ANFIS	Adaptive Neural Fuzzy Inference System	MARS	Multivariate Adaptive Regression Spline
ANN	Artificial Neural Network	MLR	Multiple Linear Regression
ARIMA	Autoregressive Integrated Moving Average	NDF	National Demand Forecast
CNN	Convolutional Neural Network	R1	First Reconciliation
CPP	Critical Peak Pricing	R2	Second Reconciliation
DNN	Deep Neural Network	RBF	Radial Basis Function
DSR	Demand Side Response	RMSE	Root Mean Square Error
ELM	Extreme Learning Machine	RNN	Recurrent Neural Network
EMA	Exponential Moving Average	SARIMA	Seasonal Autoregressive Integrated Moving Average
FE	Forecasted Error	SD	Settled Demand
GRU	Gated Recurrent Unit	SF	Settlement Final
INDO	Initial Demand Out-Turn	SVM	Support Vector Machines
KNN	K-Nearest Neighbors	TNUoS	Transmission Network Use of System
LSSVM	Least Square Support Vector Machine	ToU	Time of Use
LSTM	Long-Short-Term Memory	TSDf	Transmission Demand Forecast
MA	Moving Average		

# 1 Introduction

The changes in energy policy aim to substantially increase renewable energy generation and reduce carbon emissions. Addressing the growing energy demand, ageing infrastructure and intermittency of renewable energy requires an efficient forecasting methodology to predict periods of peak energy demand. Long-term power load forecasting at national level is an important basis for Demand Side Response planning, which aims to reduce the need for last-minute energy generation from non-renewable sources.

In the energy sector, Demand Side Response (DSR) is meant to substantially reduce the need for investment in peak generation. This is done by minimising consumption at times of high demand. With the goal of adding stability to the system, demand response lowers the need for coal and gas-fired spinning reserves. This reduces carbon emissions because most power plants burn fuel/coal continuously in order to supply power at short notice and thus, reduces climate change impact and decreases the need for local network investments. Demand response refers to “voluntary changes by consumers of their electricity use pattern”, either in response to changes in the price of electricity over time or through incentive payments.

Reducing electricity demand peaks is a key issue for DSR programs and it is the next step for reduction of carbon emissions, and therefore less power will have to be generated by coal and gas. In [1], 15 recent empirical assessments of residential dynamic pricing programs are surveyed, most conducted in the US after the year 2000. According to their survey, Time-of-Use (ToU) tariffs induce a reduction in peak consumption that ranges from 3% to 6% and Critical Peak Pricing (CPP) has the effect of decreasing peak usage by between 13% and 20%. The variety of DR programs has been increasing in Europe over the past years, the number of systems specifically oriented to national DSR programs in the scientific literature is lacking.

There is a rich variety in the literature about methodologies for peak load forecasting, and there have been significant improvements in time series forecasting due to the increase of the computer capacity which has led to new computational methods such as Machine Learning and other AI approaches. In [2], in the comparison of machine learning models for short-term load forecasting in the Greek electric grid, six machine learning methods are compared: Support Vector Machines (SVM), K-Nearest Neighbors (KNN), random forests, Neural Networks, xgboost and model trees. This is very relevant due to its load forecasting methods used at a national level. Four experiments were performed in order to minimize the error of prediction accuracy. The results of these experiments show that, overall, model trees performed better in terms of prediction error, followed by xgboost and SVM. In another comparative study, [3], three methodologies are compared for electricity demand forecasting: Multivariate Adaptive Regression Spline (MARS), Autoregressive Integrated Moving Average (ARIMA) and SVM. The results of this study show that, in terms of statistical metrics, MARS model yielded the most accurate results for 0.5 h and 1.0 h forecasts, whereas the SVR models were better for a 24 h horizon and the ARIMA model’s performance was lower for all forecasting horizons as it generated very high forecast errors.

Another commonly used approach for load forecast are Artificial Neural Networks (ANNs). ANNs are composed of a network of processing nodes (or neurons), which perform numerical transformations and are interconnected in a specific order so different weights are assigned to give importance to different factors through training the network. According to [4], ANNs are well-known for being able to forecast the outputs of nonlinear datasets, to efficiently perform different simultaneous tasks. There are several studies for load forecasting using ANNs, such as the comparative study, [5], in which three methodologies, ARIMA, ANN and Multiple Linear Regression (MLR) were deployed to forecast the electricity demand in Thailand. The results showed that based on the historical data and on the error measurement the ANN model was superior to the other two. In [6], mathematical models and neural networks to forecast the long-term electricity demand in Turkey are compared. Some short-term load forecasting studies combine ANNs with other methods, such as [7] which is an ANN based peak load forecasting using Levenberg-Marquardt and quasi-Newton methods. Also, study[8] focused on the periodic behavior of consumption for forecasting the Spanish monthly electricity demand, in which the trend of electricity demand was predicted using an ANN combined with Fourier series. There are novel alternative methods that have been compared to more traditional ANN approaches, such as [9], whose study integrates an evolutionary approach with ANNs for short-term load forecast, based on “follow the leader” behavior of sheep. This hybrid approach is compared with four other variations of

ANNs, showing that they are outperformed by the "follow the leader" hybrid approach. The emerging class of ANN, extreme learning machine (ELM) plays an important role for this purpose in [10], because it is invoked to predict the hourly load of the next day and it improves its performance significantly.

Deep Neural Networks (DNNs) are also used for the purpose of load forecasting. DNNs are ANNs with several hidden layers, adding complexity to its structure. [11] studies one day ahead forecasting of hourly loads based on deep networks. The study of [12] has a more specific goal, the aim is to use DNNs to predict the monthly electricity demand in Australia based on time series of consumption rates as well as socio-economic and environmental factors.

Other methodologies such as Radial Basis Function (RBF) have been used to address the problem of load forecasting, [13, 14, 15]. The study of [13] combines the RBF neural network with the Adaptive Neural Fuzzy Inference System (ANFIS) to adjust the prediction by taking into account the real-time electricity price. [15] compared three different versions of RBF to predict electricity load. In the area of short-term load forecasting, [16] addressed this problem by using adopted ARIMA model and similar day method for intraday load forecasting. For very short-term load forecasting, [17], also proposed an ANN based predictor and takes the load values of the current and previous time steps as the input to predict the load value at the coming step.

SVMs are also very relevant in the literature for load forecasting from earlier years. This is shown in [18] and [19], as well as more recent modified SVM versions, which are combined with other methods in order to achieve a better accuracy. This is the case of [20] for load forecasting method using a combined Least Square SVM (LSSVM) and modified artificial bee colony (ABCclo-LSSVM), which proved to have a better performance than the standard ABC-LSSVM and LSSVM. Another example of modified SVM for load forecasting is [14], which uses the sperm whale algorithm and wavelet least square support vector machine with DWT-IR for feature selection.

Recurrent Neural Networks (RNNs) are also very popular in the scientific literature as they can work on sequences of arbitrary length. More particularly in [21], a comparative study of short-term load forecast is performed by using different classes of RNNs, and although there is not a specific RNN model that outperforms the others in every prediction problem, it shows that LSTM and Gated Recurrent Units (GRUs) achieve outstanding results in many sequence learning problems. As a peculiarity, LSTM, together with GRUs, present no vanishing/exploding gradient problem. This has been proven in [22] which shows that LSTM outperforms traditional forecasting methods in the short-term electric load forecasting. They compare its performance with other methods such as Seasonal Autoregressive Integrated Moving Average model (SARIMA), a nonlinear autoregressive neural network model with exogenous inputs (NARX), SVM and NNETAR, a feed-forward neural network model for univariate time series forecasting with a single hidden layer and lagged inputs. Some other studies combine these methodologies, such as Tian et al. [23], which uses a deep neural network model for short-term load forecast based on LSTM and Convolutional Neural Network (CNN), achieving the lowest error in comparison to the other algorithms tested. [24] performs short-term load residential load forecasting using an LSTM Recurrent Neural Network showing a Mean Absolute Percentage Error (MAPE) between 1.5% to 35%, depending on the household. There are other several publications about LSTM for speech recognition, sentiment analysis and autonomous driving systems. [25, 26, 27].

This paper proposes a system for detecting events of high energy demand at national level in the context of DSR programmes. The system is designed in two stages: electricity demand forecasting with LSTM model and dynamic filtering of the potential highest electricity demand peaks with an Exponential Moving Average (EMA). The system is validated in a specific case study of UK Triads, which are the three highest electricity demand peaks of the UK energy system from November to February. This application is of high importance for the UK energy market and the EU countries that already use DSR programmes.

The rest of the paper is organised as follows: In section 2, the system for peak load forecast is first presented. Beginning with a description of the forecasting method based on a LSTM approach, followed

by the peaks extraction method, Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and the relative error (%) based on MAE and RMSE (MAPE and RRMSE) as the performance measurement indicators. Next in section 3, the system presented in section 2 is used and adapted for a specific case study: UK Triad forecast. Here the input data for the algorithm is analysed and the parameters are adjusted through all the forecasting horizon to obtain the results. Last in the results section, the outputs of the prior system are analysed by comparing the highest daily load peak obtained from the forecast with the actual demand peak of the day after all demand units have been submitted. Then for this specific case study, another analysis based on the number of successfully forecasted Triads versus the number of warning signals is performed. After this section the conclusions are presented.

## 2 Methodology

Here we describe the LSTM, along with the filters performed by the exponential moving averages.

### 2.1 LSTM Description

Long-Short Term Memory [28] has proven to be a useful method for time series analysis of records with several factors correlated with the output. This method can provide a good working system for the purpose of UK national electricity demand forecast, and an effective way to ensure that the system addresses correlations in that data.

LSTM cells manage two state vectors, and for performance reasons they are separate [28]. The scheme of a single cell is illustrated below in Figure 1:

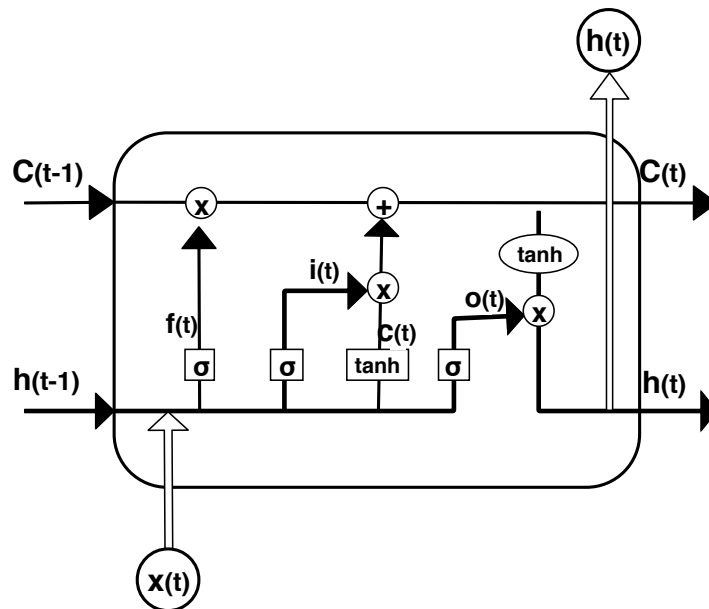


Figure 1: LSTM scheme (Following [28]).

The state of the cell is split into two vectors:  $\mathbf{h}_t$  and  $\mathbf{c}_t$ . Vector  $\mathbf{h}_t$  can be interpreted as the short-term state and  $\mathbf{c}_t$  as the long-term state.

The current input vector  $\mathbf{x}_t$  and the previous short-term state  $\mathbf{h}_{t-1}$  are fed to four different fully connected gates. They serve different purposes:

- The main gate is the one that outputs  $\mathbf{c}_t$ . It has the usual role of analysing the current inputs  $\mathbf{x}_t$  and the previous short-term state  $\mathbf{h}_{t-1}$ .

- The *forget gate* (controlled by  $\mathbf{f}_{(t)}$ ) controls which part of the long-term state should be erased.
- The *input gate* (controlled by  $\mathbf{i}_{(t)}$ ) controls which parts of  $\mathbf{c}_{(t)}$  should be added to the long-term state.
- The *output gate* (controlled by  $\mathbf{o}_{(t)}$ ) controls which part of the long-term state should be read and output at this time step (both to  $\mathbf{h}_{(t)}$  and  $\mathbf{c}_{(t)}$ ).

$\sigma$  represents the logistic function transformation after a fully connected NN set. The key idea is that the network can learn what to store in the long-term state, what to throw away, and what to read from it. The long-term state transverses the network from left to right, it goes through a *forget gate*, dropping some memories, and it adds new memories through the addition operation. After that it is copied and processed through the *tanh* function, whose result is filtered by the *output gate*. This produces the short-term state  $\mathbf{h}_{(t)}$ .

Equations 1 - 6 summarise how to compute the cell long-term state, and its output at each time step for a single instance:

$$\mathbf{i}_{(t)} = \sigma(\mathbf{W}_{xi} \cdot \mathbf{x}_t + \mathbf{W}_{hi} \cdot \mathbf{h}_{t-1} + \mathbf{b}_i), \quad (1)$$

$$\mathbf{f}_{(t)} = \sigma(\mathbf{W}_{xf} \cdot \mathbf{x}_t + \mathbf{W}_{hf} \cdot \mathbf{h}_{t-1} + \mathbf{b}_f), \quad (2)$$

$$\mathbf{o}_{(t)} = \sigma(\mathbf{W}_{xo} \cdot \mathbf{x}_t + \mathbf{W}_{ho} \cdot \mathbf{h}_{t-1} + \mathbf{b}_o), \quad (3)$$

$$\mathbf{c}_{(t)} = \mathbf{g}_{(t)} = \tanh(\mathbf{W}_{xg} \cdot \mathbf{x}_t + \mathbf{W}_{hg} \cdot \mathbf{h}_{t-1} + \mathbf{b}_g), \quad (4)$$

$$\mathbf{c}_{(t)} = \mathbf{f}_{(t)} \otimes \mathbf{c}_{(t-1)} + \mathbf{i}_{(t)} \otimes \mathbf{g}_{(t)}, \quad (5)$$

$$\mathbf{y}_{(t)} = \mathbf{h}_{(t)} = \mathbf{o}_{(t)} \otimes \tanh(\mathbf{c}_{(t)}), \quad (6)$$

where:

- $\mathbf{W}_{xi}$ ,  $\mathbf{W}_{xf}$ ,  $\mathbf{W}_{xo}$ ,  $\mathbf{W}_{xg}$  are the weight matrices of each of the four gates for their connection to the input vector  $\mathbf{x}_t$ .
- $\mathbf{W}_{hi}$ ,  $\mathbf{W}_{hf}$ ,  $\mathbf{W}_{ho}$  and  $\mathbf{W}_{hg}$  are the weight matrices of each of the four gates for their connection to the previous shot-term state  $\mathbf{h}_{t-1}$ .
- $\mathbf{b}_i$ ,  $\mathbf{b}_f$ ,  $\mathbf{b}_o$  and  $\mathbf{b}_g$  are the bias terms for each of the four gates.
- $\otimes$  represents element-wise vector multiplication.

In order to achieve a more accurate result, we perform parameter tuning later in section 4. These parameters are the number of years of data used for training, the number of cells and the number of epochs. This will also be discussed later on in section 4.

## 2.2 EMA Description

Exponential Moving Average (EMA) is a modified version of the simple Moving Average (MA), i.e. a type of moving average with more weight given to the latest data. The EMA works as a classifier in this case, generating binary signals, 1 when the peak is over the EMA, 0 when is below it. The formula represents the EMA as follows:

$$S_t = \alpha y_{t-1} + (1 - \alpha)S_{t-1}, \quad (7)$$

where:

- $S_t$ : value of the EMA for  $t = now$ .

- $\alpha$ : smoothing constant. When  $\alpha$  is close to 1, dampening is quick and when  $\alpha$  is close to 0, dampening is slow.
- $y_{t-1}$ : actual observation for  $t - 1$ .
- $S_{t-1}$ : value of the EMA for  $t - 1$ .

As a filter, this EMA acts as a physical boundary, choosing as peak load any value above it. This filter value will be a certain percentage multiplied to the EMA, depending on the level of risk that the user can afford to take, and that will be applied according to each DSR intervention. The system scheme, as well as the final values chosen after the calibration stage, are shown in Figure 7 for the specific case study of Triad peaks forecasting. One epoch makes reference to the number of times the model has seen a number of instances equals to the number of instances in the training data. The batch size is the number of values taken by the algorithm at every step, in this case 6 days worth of data (48 half-hours each day).

### 2.3 Model evaluation

This study adopted a range of statistical error criteria in the testing period based on statistical indicators. As accuracy evaluator for this model, the Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and the relative error (%) based on MAE and RMSE (MAPE and RRMSE) have been chosen [29]. The formula can be seen in eqs. 8 to 11.

$$RMSE = \sqrt{\frac{1}{N} \sum_{j=1}^N (y_j - \hat{y}_j)^2}, \quad (8)$$

$$RRMSE = 100 * \frac{\sqrt{\frac{1}{N} \sum_{j=1}^N (y_j - \hat{y}_j)^2}}{\bar{y}_j}, \quad (9)$$

$$MAE = \frac{1}{N} \sum_{j=1}^N |y_j - \hat{y}_j| \quad (10)$$

$$MAPE = 100 * \frac{1}{N} \sum_{j=1}^N \left| \frac{y_j - \hat{y}_j}{y_j} \right|, \quad (11)$$

- $N$ : Total number of values. In this case the number of output values, which for a single day with HH data would be  $N = 48$ .
- $y_j$ : Actual (observed) value to compare the forecast with.
- $\hat{y}_j$ : Forecasted value, output of the LSTM.
- $\bar{y}_j$ : Average of the array of observed values.

## 3 Case study: UK Triads

Triads are the three half hour periods of peak power demand across the National Grid in a year (from November to February). These three points are used to calibrate the system costs, which are passed on to industry. The aim of the Triad system is to incentivise industry and users to help smooth out peaks in energy demand during the winter, especially in cold snaps [30].

According to [31], the Triad charging system encourages demand reduction at these peak hours and hence signals the need for less generation and transmission (which will be sized to predicted peak loads), as this creates the need for avoiding these peak hours. ELEXON provides a forecast for the UK electricity demand and energy managers, along with businesses rely on this information (publicly available) to know

when a Triad is going to happen, but this information is incomplete and inaccurate as the demand values that ELEXON seeks to forecast are not the ones which Triad is calculated against. The model proposed in this paper creates a better decision making framework because calling Triads implicates switching off equipment. Some companies can not handle the disruption internally, so they need to run fuel generators and this implicates a considerable cost.

### 3.1 Triad background

Triad forecasting is a matter of great interest for businesses, as this is an event that costs a significant amount of money, specially to those with higher number of infrastructural objects (banking, retail, telecommunications). TNuoS charges, which cover the costs of operating transmission networks, may represent around 5% of the bill. These fees are revised annually and forecasted for 5 years ahead. The 2017 forecast published by National grid [32], shows the value of Triad growing from an average of £44 ( $\approx 57.36\$$ ) per kW to £59 ( $\approx 76.90\$$ ) per kW used during peak times. This forecast can be seen below in Table 1.

The charge varies across 14 zones and is set based on user’s average half-hourly demand over three Triad periods taking place every winter season [32]. Because of economic interests for companies, most of the current Triad forecast systems are not publicly available.

In [33] the electricity demand of each building on an actual Triad peak date and time was predicted successfully, and an overall forecasting accuracy of 97.6% was demonstrated for the considered buildings. Marmaras’ model uses data from three different sources at various stages to predict the most probable half-hour of the day when the Triad could occur. These are data from National Grid, weather data and historical consumption; and its training set consist of historical data from 1990. This work, however, only validates the effectiveness of Triad forecasting using one year of data, not ensuring that the same model will work after periods and therefore, not offering a flexible framework when any changes (such as new policies) occur.

Region \ Season	18/19	19/20	20/21	21/22	22/23
Nothern Scotland	18.35	21.69	27.94	27.19	28.81
Southern Scotland	25.13	29.17	33.99	35.13	37.92
Nothern	36.92	41.50	44.31	48.63	51.20
Noth West	43.87	48.40	51.01	55.79	58.73
Yorkshire	43.83	48.47	51.15	56.25	59.31
Noth Wales & Mersey	45.43	49.97	52.66	57.27	60.70
East Midlands	47.39	52.26	55.59	60.83	64.07
Midlands	48.85	53.50	56.26	61.71	65.19
Eastern	49.37	54.33	57.91	63.17	66.51
South Wales	46.78	50.89	54.18	59.64	64.01
South East	52.52	57.11	60.34	65.51	68.72
London	54.84	60.21	63.92	69.30	72.89
Southern	53.80	58.55	61.67	66.74	69.94
South Western	53.86	57.30	60.08	63.71	67.73

Table 1: Triad fees (£/kW) forecast from 2018/2019 to 2022/2023

As changes happen very often in this field, ideally some parameters should be regulated and a single standalone system that works for every Triad season without having to do any modifications is difficult. Algorithm validation is also not easy to because of two reasons: data availability and constant changes in the patterns of the training data. This is why we propose to offer a certain degree of flexibility that the user can tune according to the degree of risk that can be afforded.

In this work, we use a period of 4 years, as there are policies that change in a relatively short period of time, and it will be validated for four different periods, all of them from November to February, when the Triad season occurs. We apply and train LSTM for this period.

## 3.2 Design of experiment

The design of the experiment begins with the analysis of the inputs to the model. We address this by comparing how the aggregated wind and solar generation are related to the load variation from first to last settlement release. Once we prove these inputs are relevant, we proceed to designing the LSTM model using the previously analysed inputs. The output will be the last settled load forecast, which will be dynamically filtered in order to obtain the peak demands of energy consumption through an exponential moving average. Those values above the filter are considered as potential Triads (for this particular case study) and are finally compared with the total number of signals provided.

## 3.3 Triad data analysis

In this section, we analyse the historic data and find the relationship between the settled demand data and the generation with some of the renewable sources. Next, we discuss a plot with the historical Triad, and last, we look for correlations in the data that is going to be used as input for the LSTM, as well as identify seasonality in the training data.

### 3.3.1 Settled data and renewables generation

The data used for selecting Triad days is not the Initial Demand Out-turn (INDO), but the Settlement Final/ 1st Reconciliation/ 2nd Reconciliation (SF/R1/R2), which is the actual load on the grid once the BM units have submitted all the sub-meter data. This data is settled at around 9, 20 and 90 days post the event and is the data that Triad is calculated against. The main difference between INDO and settled is the removal of the station load (the load the power station uses to power itself). This is why, the output forecast first, as previously explained in section 2, is the SF/R1/R2.

There are limitations in the models depending on the amount, type and quality of data available. There are forecasts provided for the INDO, however, as there are no forecasts for the SD available, a model needs to be defined based on the available data. The idea is to find parameters that keep a relationship with the difference between the INDO and the SD. Generation renewables (wind and solar generation), and the mentioned difference between actual INDO and the SD keep such relationship. As shown in Figure 2, plotting  $INDO - SD$  against the sum of solar and wind generation, the points obtained can be approximated by a linear regression, which makes solar and wind generation possible predictors for the SD forecast system.



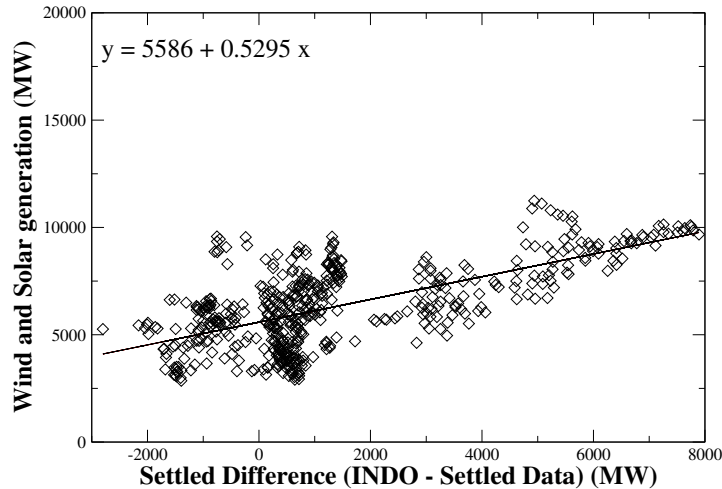


Figure 2: Correlation between Renewables Generation and Settled Variance

The fact that this data is correlated means that it can be used for a predictive model that forecasts the SD as a first step for Triad forecasting.

### 3.3.2 Historical Triads

In order to analyse Triad, it is useful to know when Triad historically happened, so statistical insights can be gained for future decisions.

From the 2008/09 to the 2015/16 winter season, 45.8% of Triad occurred on Mondays and 29% on Thursday, with other weekdays only accounting for one in four Triad. Out of the total 24 cases, 22 occurred between 17:00 and 17:30 and 2 occurred between 17:30 and 18:00. A reason why a Triad may happen in a latter hour around February may be explained by the number of hours of sunlight, which grows longer after the January period and thus, moving forward the second peak of electricity demand, meaning that users switch on lighting a bit later than usual, generating possible peaks later than in the rest of the Triad season.

### 3.3.3 LSTM inputs analysis

First of all, it is necessary to study the influence factors of the SD. From the model point of view, it is also interesting to plot the temperature and observe the close correlation to the INDO.

- NDF (National Demand Forecast)
- WIND (Wind Generation Forecasting)
- TSDF (Transmission Demand Forecast)
- SOLAR (Solar Generation Forecasting)

The four input data variables are forecasts for the next 48 half-hour periods predicted by ELEXON, obtained 24 hours before the event to be forecasted. Solar and wind data is based on historical outturn data and detailed local wind and solar forecasts, used by National Grid forecasts likely levels of and solar wind generation. The system operator NDF is based on historically metered generation output for Great Britain. The values shown here take into account transmission losses and includes station transformer load, pump storage demand and interconnector demand [30]. Given the National Demand Forecast (NDF) and Transmission Demand Forecast (TSDF) data of several years as shown in Figure 3, it can be seen that the overall trend of both of them is decreasing over the years. This means that the actual demand does decrease and that, for further filtering, this fact needs to be taken into consideration. Historic data for NDF and TSDF can be seen in Figure 3.

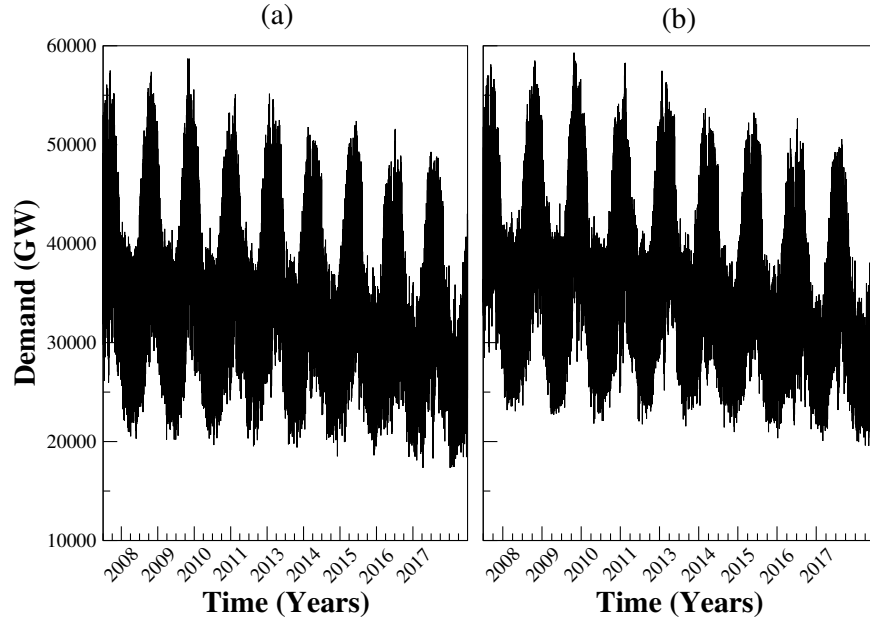


Figure 3: National Demand Forecast (a) and Transmission Demand Forecast (b) from January 2008 to October 2017

Also, it is useful to display every quarter of the year for the actual INDO to see the differences in terms of patterns of behaviour between seasons. So, taking the year 2017 as an example, each quarter of the year is plotted in Figure 4.

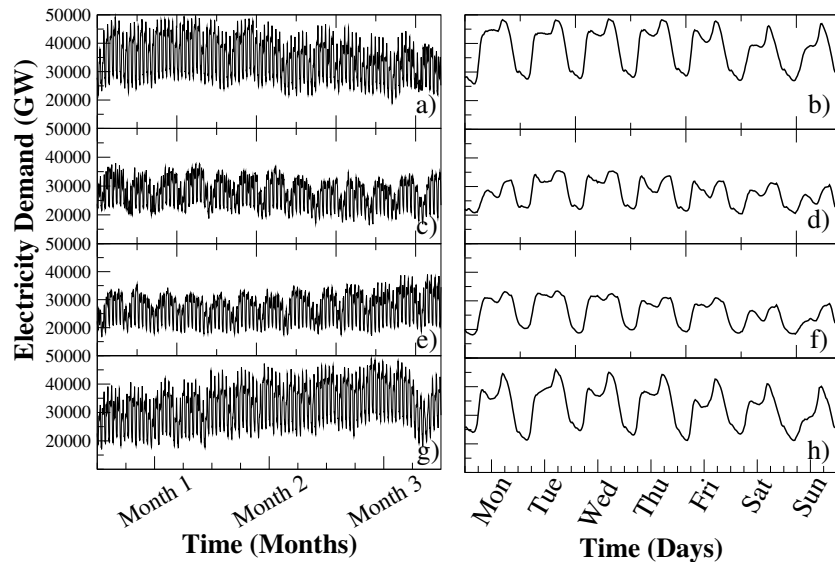


Figure 4: UK Electricity Demand for the 4 quarters of the year 2017 on the left: Panel (a) January - March (months 1-3), panel (c) April - June (months 1-3), panel (e) July - September (months 1-3) and panel (g) October - December (months 1-3). Random week taken from each corresponding quarter, Monday to Sunday, on the right: 16/01/2017 to 22/01/2017 (b), 01/05/2017 to 07/05/2017 (d), 07/08/2017 to 13/08/2017 (f) and 06/11/2017 - 12/11/2017 (h).

As can be seen in Figure 4, the patterns of behaviour are different depending on the season and/or day of the week, where the consumer energy use can be visible.

As for wind and solar energy, they depend on weather conditions. For the Triad season 2016/17, these values are displayed in Figures 5 and 6.

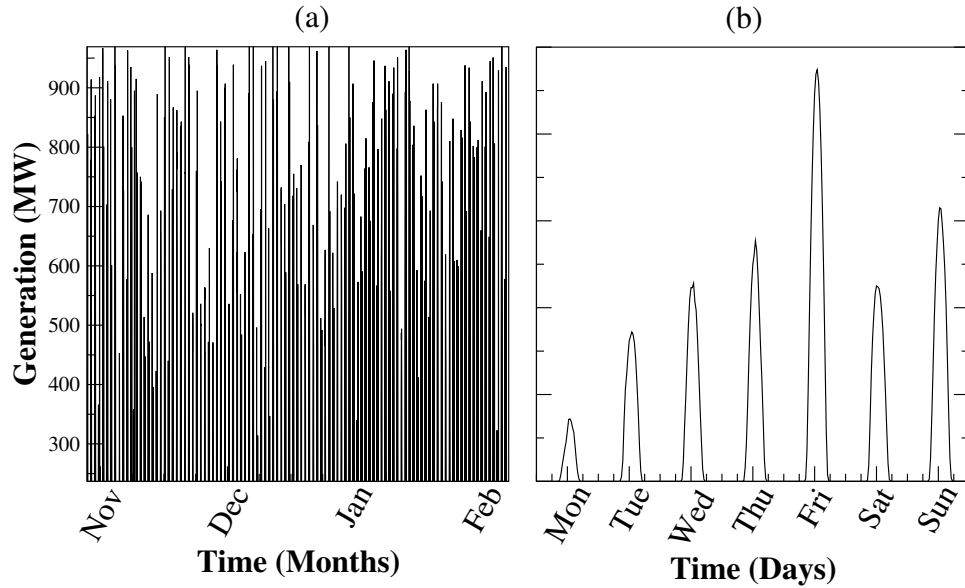


Figure 5: National total solar generation from November 2016 to February 2017 (a) and from 16<sup>th</sup> January 2017 to 22<sup>nd</sup> January 2017 (b). Data source [30].

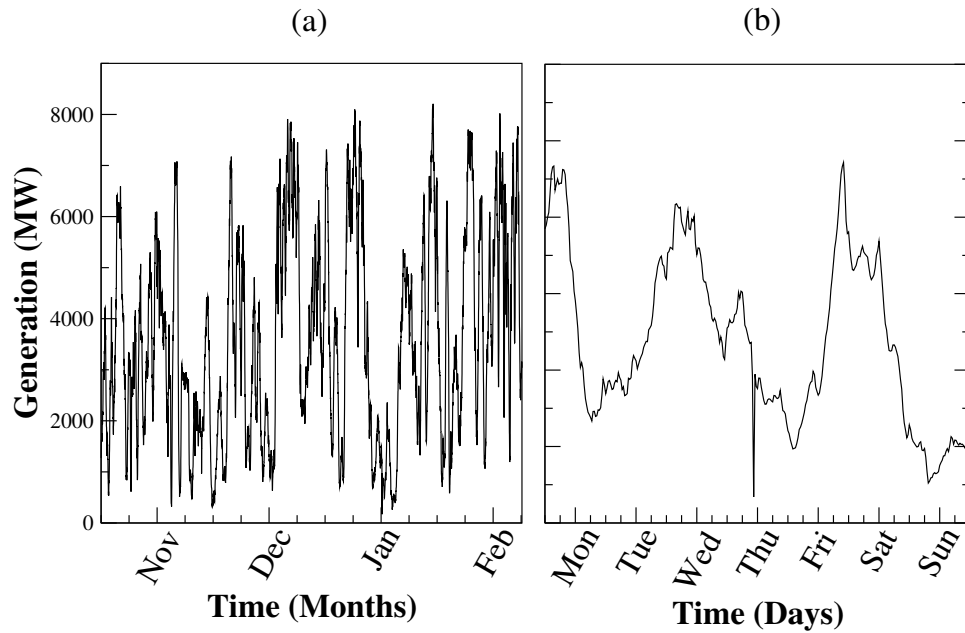


Figure 6: National total wind generation from November 2016 to February 2017 (a) and from 16<sup>th</sup> January 2017 to 22<sup>nd</sup> January 2017 (b). Data source [30].

To determine data correlations, the standard correlation coefficient (also called *Pearson's r*) can be computed. The result of the 4 inputs that we are using as inputs can be seen in Table 2.

Variables	Pearson's r with SD
NDF	0.99
TSDF	0.99
SOLAR	0.18
WIND	-0.15

Table 2: Pearson's r coefficients calculated between Settled Data and the other variables

As expected, a strong positive correlation can be found Between SD and NDF and TSDF values, as these are going to define the shape of the curve. Also there are correlation between the wind and solar generation, and SD data. Now that the relationships between variables and the data has been discussed, the model will be built and tested.

### 3.4 System configuration

The goal of the system proposed in section 2 for this specific case study is to produce Triad signals (as few as possible) to determine a DSR intervention.

First of all the data is rescaled between 0 and 1, this is a beneficial machine learning practice because when feeding the algorithm, the weights are assigned during the training stage of the system, and having different scales of values may lead to a bad fitting and not reaching a global maximum solution. The rest of the system, which is also described in Figure 7 is divided as follows:

- **LSTM forecasting:** Provides with the SD forecasted values for the next 48 half-hour settlement periods. This architecture consists of 40 concatenated cells, with 250 epochs, and a batch size worth of 6 days of data. The output of this system are the next 48 half-hour settlement periods of SD forecasting.
- **Peak extraction:** Next, the maximum demand peak of the day is extracted and added to a vector with the previous forecast peaks. For filtering purposes, weekends and Christmas period (23<sup>rd</sup> of December to 2<sup>nd</sup> of February) are excluded from the dataset so the Triad signals will be filtered by using the rest of the days, when Triad happens.
- **Filters:** Last, after the demand of the next 24 hours have been forecast, two different filters have been used based on a simple approach. The idea is to use two Exponential Moving Averages (EMAs) multiplied by a factor. As an example of what the filter values may be, for this paper we set the percentages to 3.5% and 4% for the soft and hard filters, lower and higher risk respectively.

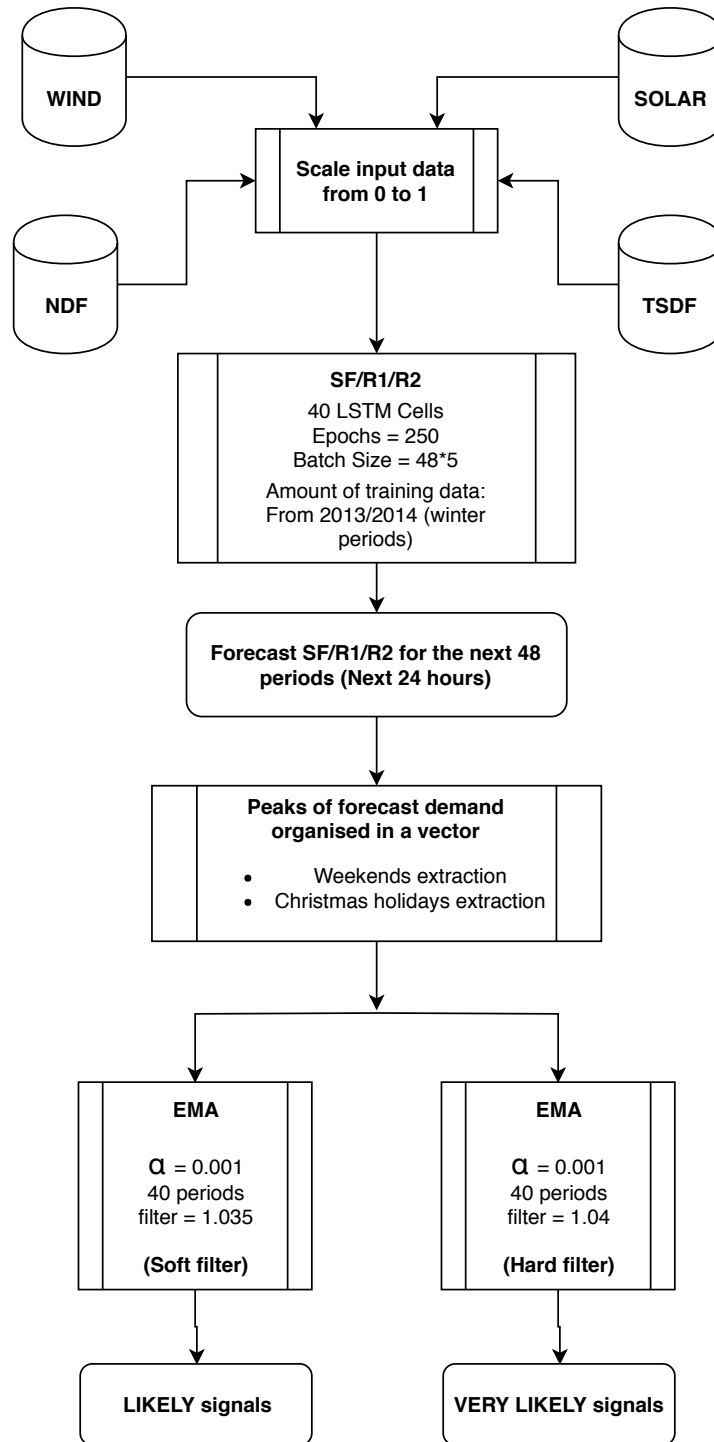


Figure 7: Triad system architecture

## 4 Results

The implementation of the system has been carried using Python, more specifically in the Keras library. The dataset has been divided in the following parts for calibrating the model. The training set includes the data from the four different inputs (WIND, SOLAR, NDF and TSDF) and the output (test set) refers to

the final settled data (SF/R1/R2), therefore all the time ranges mentioned below correspond to these parts of the dataset.

- Training Set: 1<sup>st</sup> November to 28<sup>th</sup> of February (29<sup>th</sup> when it is Leap year in 2016) years 2015-2018. November to February for each year is the period for which the SF/R1/R1 data is available. This training set is divided and consecutively added behind the test set, so the optimum number of years behind testing can be determined.
- Test set: next 14 days after the last day of the training set (15<sup>th</sup> to 28<sup>th</sup> of February 2018). The purpose of this testing set is to measure the variance of the performance when changing the hyperparameters in order to choose an optimum combination of these. This test data must not have been used for training so, as it is the consecutive data after the training set, the trained algorithm follows this sample sequence after training and the performance is measured through the different metrics.

A summary can be found in Table 3.

	Training	Testing
Number of points	11472	672
(%)	94.46%	5.54%
Date range	01.11.2015 to 28.02.2018	15.02.2018 to 28.02.2018

Table 3: Training and testing data information

We perform long-term forecast, thus the model is trained from the previous years and serves as outputs for a Triad season after training. The scope of the forecasting corresponds to four months worth of data.

As our model has four inputs, based on the description in the data analysis section, our LSTM architecture will contain a determined number of these cells concatenated. This number is to be determined through experimentation.

The scheme of the model can be seen in Figure 8.

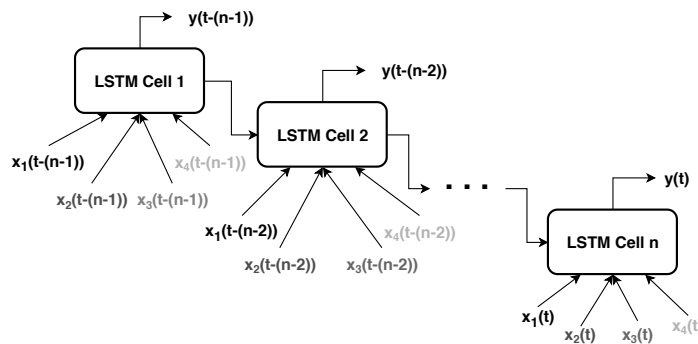


Figure 8: LSTM scheme with 4 inputs

#### 4.1 1<sup>st</sup> calibration stage: training data size

As previously mentioned in section 2, the LSTM parameters need to be tuned in order to offer a better performance. The hyperparameters of this network are calibrated according to the number of epochs, the number of years worth of data for the training stage and the number of cells constituting the network.

The experiment has been carried out by increasing the number of epochs, as well as the number of years behind the testing period, varying the batch size used for training the LSTM.

Table 4 shows the metrics with the best result obtained, that corresponds to 250 epochs and a  $48 \cdot 6$  batch size (6 days with 48 periods each) with the whole dataset. This means that for the earlier the year of

testing the shorter the amount of training data, therefore for further experiments, the whole dataset behind the testing period is going to be taken because less data is available.

Epochs	50	150	250	300	350
<b>Training accuracy</b>					
<b>RMSE (MW)</b>	740.85	694.44	693.11	699.76	698.65
<b>RRMSE (%)</b>	2.09	1.94	1.94	1.96	1.96
<b>MAE (MW)</b>	574.24	522.95	521.28	527.60	526.56
<b>MAPE (%)</b>	1.66	1.52	1.51	1.53	1.53
<b>Testing accuracy</b>					
<b>RMSE (MW)</b>	833.44	799.75	793.53	796.59	798.27
<b>RRMSE (%)</b>	2.35	2.24	2.22	2.23	2.24
<b>MAE (MW)</b>	625.15	578.4	578.15	577.67	578.43
<b>MAPE (%)</b>	1.75	1.60	1.60	1.60	1.60

Table 4: Training and testing accuracy metrics according to the numbers of epochs.

## 4.2 2<sup>nd</sup> calibration stage: Number of neurons

Next, the number of neurons of the LSTM needs to be determined by using the results obtained in the prior calibration stage. For the experiment in Table 4, 30 concatenated cells have been used by default and the average of five values have been taken for each metric. In Table 5 the results of this experiments can be seen together with a boxplot with the Forecasted Error (FE) [3], which is the difference between observed and predicted values, for each model represented on Figure 9.

Number of neurons	10	20	30	40	50	60
<b>RMSE (MW)</b>	801.82	798.85	795.54	795.66	794.33	797.97
<b>RRMSE (%)</b>	2.25	2.24	2.23	2.23	2.22	2.23
<b>MAE (MW)</b>	585.26	578.86	576.48	575.99	578.95	581.17
<b>MAPE (%)</b>	1.61	1.60	1.60	1.60	1.61	1.61

Table 5: Metrics for the number of neurons in the LSTM.

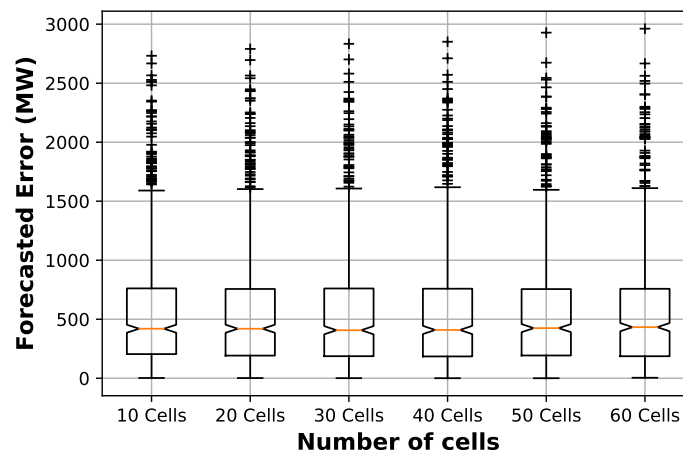


Figure 9:  $|FE| = |Observed\ value - Predicted\ value|$  boxplot for each model from number of cells = 10 to number of cells = 60.

From this experiment, the average of 10 values has been chosen for each experiment, concluding that, for this forecasting horizon, the number of concatenated cells chosen would be 40, as the metrics on table 5 indicates this one is showing one of the best possible results.

The training algorithm used is the Adam optimiser, with a learning rate = 0.001. The total number of parameters, weights and biases, is 7241 and the number of training instances is 11472.

The modelled demand obtained from the LSTM can be seen in Figure 10. For this testing, 14 days (672 points for 48 half-hours per day) have been taken in February 2018.

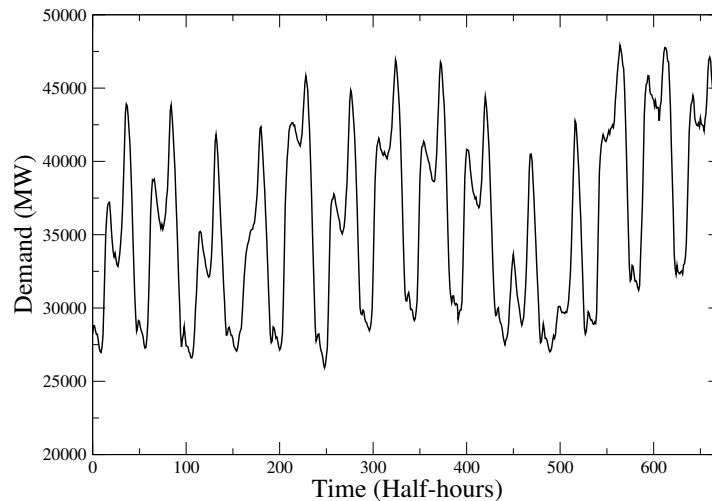


Figure 10: LSTM prediction points after training.

As can be seen in Figure 10, 14 days have been taken for demand forecast, from Wednesday to Tuesday. It also can be seen the patterns of behaviour for the weekends, being lower for these days than for weekdays. The next section will consider the filters and signals for Triad as the last stage of the system.

### 4.3 Results comparison with other models

In this section, we compare the performance of LSTM with several other popular methodologies mentioned in the background section. First, we compare the LSTM with the mean-only model and then with a simpler version of ANN with the same characteristics in terms of the number of cells and learning rate provided for the LSTM. We also compare compare LSTM with SVM regression, random forests and Bayesian regression. The results of the comparison are shown in Table 6 and illustrated in Figure 11.

Model	LSTM	Mean-only	ANN	SVM	Random Forests	Bayessian reg.
RMSE (MW)	795.66	6013.81	850.82	2321.85	1868.25	806.05
RRMSE (%)	2.23	16.91	2.37	6.50	5.27	2.27
MAE (MW)	575.99	5295.46	619.42	2009.95	1588.54	587.68
MAPE (%)	1.60	15.24	1.74	5.78	4.55	1.62

Table 6: Comparison of test accuracy of other models.



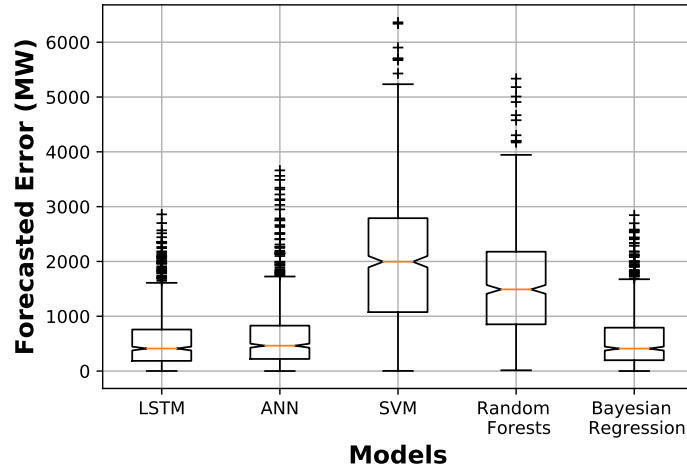


Figure 11: Forecast error of the five models.

This shows that LSTM model outperforms other models. It is important to mention that Bayesian regression results are following closely the performance of the LSTM in the second place, which makes this methodology also a good option and worth testing in similar DSR scenarios.

#### 4.4 Cross-validation

Generalised performance of a learning method and its prediction capability rely on independent test data [34]. Therefore, cross-validation is necessary to ensure that the results are reliable when new data is introduced in the future. As we are forecasting time-series data, we need a cross-validation that considers the serial correlation inherent to the problem [35], therefore we perform one step ahead cross validation [36] consisting of use  $1, \dots, k$  samples, to predict  $k + 1$  value (or alternatively  $k + 1, \dots, k + m$  values). We have perform this for the whole period of validation and, following this, we progressively added 24 hours of data to the model to obtain the following day's output.

#### 4.5 Filters configuration and results

The data filters are the last stage of the data processing. They play the role of data points classification between Triad and no Triad.

For the filters, as mentioned in section 2, EMAs have been used. The idea, for this specific case study, is to call Triads as any value above the 3.5% and 4% of the 40 days EMA, so both EMAs will be multiplied by these factors. For the testing of the system, all the values taken will be the ones predicted by the LSTM.

For filter validity in terms of parameters selection it must satisfy the following:

- Be valid for all the scope of testing, which means to successfully predict the three Triads at least with one of the two filters.
- Call the minimum number of Triad possible so the energy disruptions in buildings are kept to minimum.
- Be able to successfully call at least two Triads among the signals for both filters.

The level of financial/energy risk that is to be taken into account depends on the user, this is why two filters are used as an example of how the risk may be managed. In Figures 12, 13, 14 and 15 we display the peaks for the Triad seasons 2014/15, 2015/16, 2016/17 and 2017/18 respectively, together with both filters:

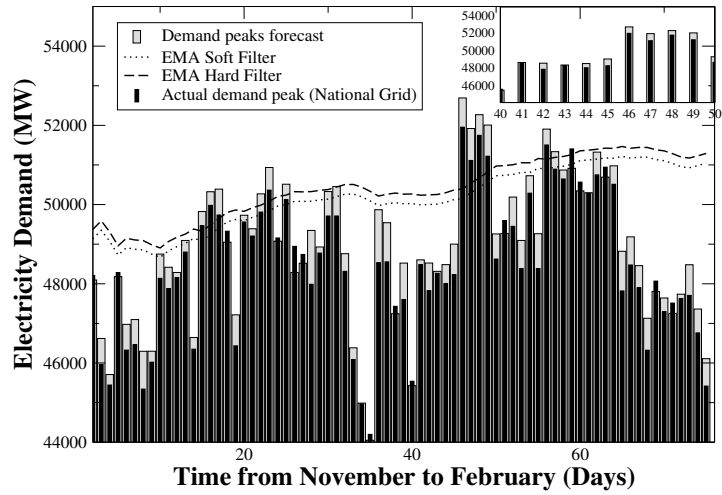


Figure 12: 2014/15 Forecast peaks for each day and EMA.

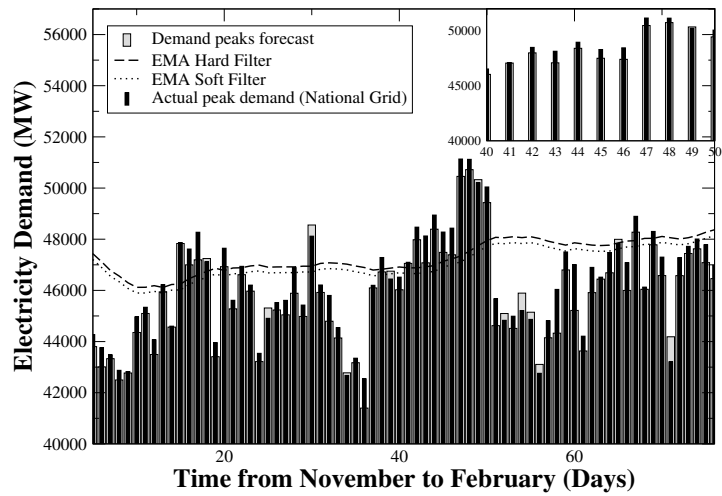


Figure 13: 2015/16 Forecast peaks for each day and EMA.

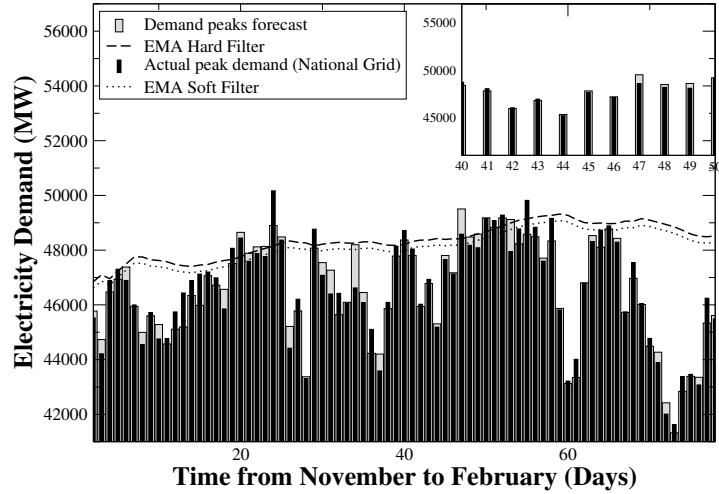


Figure 14: 2016/17 Forecast peaks for each day and EMA.

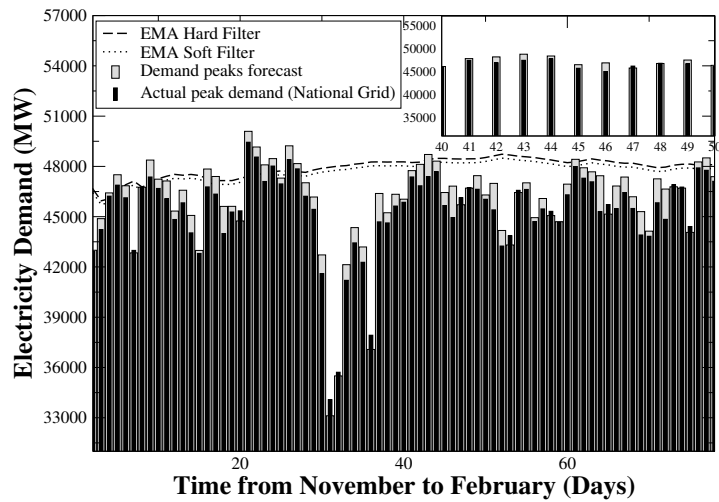


Figure 15: 2017/18 Forecast peaks for each day and EMA.

The performance of the filters can be measured by counting the number of signals that our model generates versus the number of Triad predicted in the hindcast. The summary of these results can be seen in Table 7, in which, for each year, the number of signals given by each filter, as well as the number of actual Triad days predicted. It shows the number of positive signals generated and the last two are the number of these signals that predicted the actual Triads.

In this case, the goals for at least one of the filters have been met by properly forecasting the three Triads the first three testing years (2014 to February 2017) and successfully predicting 2 out of 3 over the 2017/2018

Seasons	Soft Filter	Hard Filter	Triad Predicted (Soft)	Triad Predicted (Hard)
2014/15	19	14	3	3
2015/16	21	19	3	3
2016/17	19	14	3	3
2017/18	21	16	3	2

Table 7: Number of signals calling Triad for each filter and number of Triads predicted.

period. The idea has been to call the minimum number possible of signals, so EMA parameters in this case are valid for all three years of testing, meaning that the three conditions for filters calibration mentioned in Section 4.5 have been met.

This system failed to predict only one of the Triads with one of the filters but, for this example certain parameters have been left fixed for all the testing periods. The flexibility of this system permits to re-calculate the values for the filters in future scenarios.

## 5 Limitations/further work

This paper proposes a methodology for load forecasting by using several key variables of the energy market (NDF, TSDF, Solar and Wind). The model produces a satisfactory load forecast at the national level. Although this model captures the demand trend, it does not consider indoor physical factors, such as occupancy, internal system's efficiency, which may require more power from the grid in the case of older infrastructures. The future research may include electric vehicles in the consumption patterns, as well as consider varying electricity prices, as those have a definite impact on the electricity generation patterns. Also, due to the satisfactory results produced by Bayesian regression, further work may include this methodology for comparison with LSTM if similar data is used.

## 6 Conclusions

The goal of this paper was to design a system for load forecasting focusing on DSR events, either long or short term, depending on the DSR intervention performed. The system is composed of two steps: Load forecasting and highest peaks extraction with respect to the latest  $n$  days. In Section 2 we present the LSTM model for load forecasting, as well as the EMA for peaks extraction. We evaluate accuracy of the model by using different metrics: RMSE, RMSE, MAE and MAPE. Next, we apply this methodology for the specific case of UK Triads forecasting in Section 3, in which the performance of the model is measured in terms of the number of peaks forecast vs number of Triad signals. The goal is to forecast all the three highest peaks with the least possible number of Triads in order to reduce the number of DSR interventions. In Section 4 we calibrate the LSTM model and compare its performance with ANN, SVM, random forests, Bayesian regression and the mean-only model. This demonstrates that LSTM outperforms other models, and that its performance is closely followed by Bayesian regression. We show that over the 4 years of testing, 11 peaks are forecast in total, showing that the number of signals for the soft and hard filters are, respectively, 19 and 14 for the 14/15 period, 21 and 19 for the 15/16 period, 19 and 14 for the 16/17 period and 21 and 16 for the 17/18 period.

Once a Triad signal is positive, then as much equipment as possible is switched off and generators are run, consuming fuel for every action taken against the DSR signal. The factors chosen for the filters are to be defined by the user, in this case, 3.5% and 4% of the value has been chosen for the soft and hard filter respectively, but this defines the level of risk that the company, building manager or DSR manager wants to take. The risk assessment would determine the number of signals that the organisation can afford in terms of fuel/disruption, and the risk of missing the Triad, which is subject to a cost.

There is a possibility that circumstances of the energy system layout may change, such as re-distribution of transmission losses per region according to the P350 amendment approved on the 24th of March 2017 [32] which could affect the way the forecast behaves, and may lead to a correction factor for a better forecasting, as well as factors re-calibration. The most limiting factor in the system design has been data availability.

Due to recent changes in energy systems, it is necessary to focus on more generalised methodologies that offer a certain degree of flexibility in order to be adapted to DSR interventions. This work may lead to further developments in the area of more flexible forecast for different long/short term DSR scenarios.

The future energy systems will require either nonlinear growth of infrastructures, which is not sustainable, or wider-scale, smart interventions which are agile, low-cost and reduce carbon emissions. The UK

energy market presents a set of DSR interventions which are economically grounded and of high potential of implementation in other countries with similar demands for energy, without large investments into infrastructure. This makes modelling and forecasting of DSR programmes of high relevance to international energy markets. The modelling approach we introduce in this paper is concise, accurate, computationally light and flexible for further tuning, according to market and risk management requirements.

## 7 References

### References

- [1] A. Faruqui, S. Sergici, Household response to dynamic pricing of electricity: a survey of 15 experiments, *Journal of regulatory Economics* 38 (2) (2010) 193–225.
- [2] P.-N. Kouroupetroglou, G. Tsoumakas, Machine learning techniques for short-term electric load forecasting, Thesis.
- [3] M. S. Al-Musaylh, R. C. Deo, J. F. Adamowski, Y. Li, Short-term electricity demand forecasting with MARS, SVR and ARIMA models using aggregated demand data in Queensland, Australia, *Advanced Engineering Informatics* 35 (2018) 1–16.
- [4] S.-T. Chen, D. C. Yu, A. R. Moghaddamjo, Weather sensitive short-term load forecasting using nonfully connected artificial neural network, *IEEE Transactions on Power Systems* 7 (3) (1992) 1098–1105.
- [5] K. Kandananond, Forecasting electricity demand in Thailand with an artificial neural network approach, *Energies* 4 (8) (2011) 1246–1257.
- [6] Ü. B. Filik, Ö. N. Gerek, M. Kurban, A novel modeling approach for hourly forecasting of long-term electric energy demand, *Energy Conversion and Management* 52 (1) (2011) 199–211.
- [7] L. Saini, M. Soni, Artificial neural network based peak load forecasting using Levenberg–Marquardt and quasi-Newton methods, *IEEE Proceedings-Generation, Transmission and Distribution* 149 (5) (2002) 578–584.
- [8] E. González-Romera, M. Jaramillo-Morán, D. Carmona-Fernández, Monthly electric energy demand forecasting with neural networks and fourier series, *Energy Conversion and Management* 49 (11) (2008) 3135–3142.
- [9] P. Singh, P. Dwivedi, Integration of new evolutionary approach with artificial neural network for solving short term load forecast problem, *Applied energy* 217 (2018) 537–549.
- [10] S. Li, L. Goel, P. Wang, An ensemble approach for short-term load forecasting by extreme learning machine (Mar 2016).  
URL <https://www.sciencedirect.com/science/article/pii/S0306261916302707>
- [11] W. He, Load forecasting via deep neural networks, *Procedia Computer Science* 122 (2017) 308–314.
- [12] H. Hamedmoghadam, N. Joorabloo, M. Jalili, Australia’s long-term electricity demand forecasting using deep neural networks, arXiv preprint arXiv:1801.02148.
- [13] Z. Yun, Z. Quan, S. Caixin, L. Shaolan, L. Yuming, S. Yang, RBF neural network and ANFIS-based short-term load forecasting approach in real-time price environment, *IEEE Transactions on power systems* 23 (3) (2008) 853–858.
- [14] J.-p. Liu, C.-l. Li, The short-term power load forecasting based on sperm whale algorithm and wavelet least square support vector machine with DWT-IR for feature selection, *Sustainability* 9 (7) (2017) 1188.

- [15] A. Khwaja, X. Zhang, A. Anpalagan, B. Venkatesh, Boosted neural networks for improved short-term electric load forecasting, *Electric Power Systems Research* 143 (2017) 431–437.
- [16] X. Cao, S. Dong, Z. Wu, Y. Jing, A data-driven hybrid optimization model for short-term residential load forecasting, in: *Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM)*, 2015 IEEE International Conference on, IEEE, 2015, pp. 283–287.
- [17] P. Qingle, Z. Min, Very short-term load forecasting based on neural network and rough set, in: *Intelligent Computation Technology and Automation (ICICTA)*, 2010 International Conference on, Vol. 3, IEEE, 2010, pp. 1132–1135.
- [18] B.-J. Chen, M.-W. Chang, et al., Load forecasting using support vector machines: A study on EUNITE competition 2001, *IEEE transactions on power systems* 19 (4) (2004) 1821–1830.
- [19] W.-C. Hong, Electric load forecasting by support vector model, *Applied Mathematical Modelling* 33 (5) (2009) 2444–2454.
- [20] M. A. M. Daut, M. Y. Hassan, H. Abdullah, H. A. Rahman, M. P. Abdullah, F. Hussin, An improved building load forecasting method using a combined least square support vector machine and modified artificial bee colony, *ELEKTRIKA-Journal of Electrical Engineering* 16 (1) (2017) 1–5.
- [21] F. M. Bianchi, E. Maiorino, M. C. Kampffmeyer, A. Rizzi, R. Jenssen, An overview and comparative analysis of recurrent neural networks for short term load forecasting, *arXiv preprint arXiv:1705.04378*.
- [22] J. Zheng, C. Xu, Z. Zhang, X. Li, Electric load forecasting in smart grids using long-short-term-memory based recurrent neural network, in: *Information Sciences and Systems (CISS)*, 2017 51st Annual Conference on, IEEE, 2017, pp. 1–6.
- [23] C. Tian, J. Ma, C. Zhang, P. Zhan, A deep neural network model for short-term load forecast based on long short-term memory network and convolutional neural network, *Energies* 11 (12) (2018) 3493.
- [24] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, Y. Zhang, Short-term residential load forecasting based on LSTM recurrent neural network, *IEEE Transactions on Smart Grid*.
- [25] A. Graves, A.-r. Mohamed, G. Hinton, Speech recognition with deep recurrent neural networks, in: *Acoustics, speech and signal processing (icassp)*, 2013 IEEE international conference on, IEEE, 2013, pp. 6645–6649.
- [26] J. Wang, L.-C. Yu, K. R. Lai, X. Zhang, Dimensional sentiment analysis using a regional CNN-LSTM model, in: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Vol. 2, 2016, pp. 225–230.
- [27] H. Xu, Y. Gao, F. Yu, T. Darrell, End-to-end learning of driving models from large-scale video datasets, *arXiv preprint*.
- [28] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural computation* 9 (8) (1997) 1735–1780.
- [29] S. A.-M. Mohanad, C. D. Ravinesh, L. Yan, Particle swarm optimized–support vector regression hybrid model for daily horizon electricity demand forecasting using climate dataset, in: *E3S Web of Conferences*, Vol. 64, EDP Sciences, 2018, p. 08001.
- [30] ELEXON, Elexon web site, (<https://www.elexon.co.uk/>) (2018).  
URL <https://www.elexon.co.uk/>
- [31] D. Newbery, High level principles for guiding GB transmission charging and some of the practical problems of transition to an enduring regime, Report Prepared for and Commissioned by Project TransmiT, Great Britain Office of Gas & Electricity Markets.

- [32] National-Grid, National Grid UK (<https://www.nationalgrid.com/uk>) (2018).  
URL <https://www.nationalgrid.com/uk>
- [33] C. Marmaras, A. Javed, L. Cipcigan, O. Rana, Predicting the energy demand of buildings during triad peaks in GB, *Energy and Buildings* 141 (2017) 262–273.
- [34] J. Friedman, T. Hastie, R. Tibshirani, *The elements of statistical learning*, Vol. 1, Springer series in statistics New York, 2001.
- [35] S. Arlot, A. Celisse, et al., A survey of cross-validation procedures for model selection, *Statistics surveys* 4 (2010) 40–79.
- [36] R. J. Hyndman, G. Athanasopoulos, *Forecasting: principles and practice*, OTexts, 2018.