

# Day-ahead electricity price prediction applying hybrid models of LSTM-based deep learning methods and feature selection algorithms under consideration of market coupling

Wei Li<sup>a,\*</sup>, Denis Becker<sup>a</sup>

<sup>a</sup>NTNU Business School, Norwegian University of Science and Technology, 7491 Trondheim, Norway

## Abstract

The availability of accurate day-ahead electricity price forecasts is pivotal for electricity market participants. In the context of trade liberalisation and market harmonisation in the European markets, accurate price forecasting becomes difficult for electricity market participants to obtain because electricity forecasting requires the consideration of features from ever-growing coupling markets. This study provides a method of exploring the influence of market coupling on electricity price prediction. We apply state-of-the-art long short-term memory (LSTM) deep neural networks combined with feature selection algorithms for electricity price prediction under the consideration of market coupling. LSTM models have a good performance in handling nonlinear and complex problems and processing time series data. In our empirical study of the Nordic market, the proposed models obtain considerably accurate results. The results show that feature selection is essential to achieving accurate prediction, and features from integrated markets have an impact on prediction. The feature importance analysis implies that the German market has a salient role in the price generation of Nord Pool.

**Keywords:** Deep learning, Electricity price forecasting (EPF), Electricity market coupling, Feature selection, Long short-term memory (LSTM), The Nord Pool system price

## 1. Introduction

Over the last two decades, worldwide energy markets have experienced a transition towards deregulation and harmonisation [1]. Under trade liberalisation, the traditional vertically integrated power utilities are replaced with decentralised business entities whose targets are to maximise their profits. Consequently, a growing number of market participants are exposed to intense competition, and their need for suitable decision support models to increase margins and reduce risk has significantly increased [2]. Thus, the availability of accurate day-ahead electricity price forecasts is vital for market participants to adjust production plans and to perform effective bidding strategies to make an economic profit. However, due to the productive structure and characteristics of electricity prices, highly accurate forecasting is quite challenging [3, 4]. With the increasing integration of electricity markets, making accurate forecasts becomes even more difficult in the complex and integrated system. This is because the forecasting of electricity prices needs to consider a large number of factors from an ever-growing number of interconnected, neighbouring power markets. These factors include electricity prices, production, consumption, and other important features that influence cross-border electricity markets.

Numerous research efforts have contributed to the exploitation and development of advanced technologies for day-ahead electricity price forecasting (EPF), aimed at highly accurate forecasting results [5, 6]. A considerable amount of literature has been devoted to EPF models, which can be classified into

the following five categories [5]: multi-agent [7, 8], fundamental [9, 5], reduced-form [10, 11], statistical [12, 13, 14], and computational intelligence (CI) models [15, 16, 17]. Compared with the other four traditional models, CI models are regarded as state-of-the-art techniques, and their superior performance contributes to their prevalence in EPF in recent years. In particular, deep neural networks (DNNs) have gradually become the most avant-garde CI approach in other disciplines [18, 19, 20] and entered the scientific research field related to EPF.

DNNs are often categorised into three main classes: feed-forward neural networks (FNNs), recurrent neural networks (RNNs), and convolutional neural networks (CNNs). Different types of DNNs are used to solve different problems. For time series prediction, RNNs have achieved superior performance by building extra mappings to hold relevant information from past inputs. The long short-term memory (LSTM) and gated recurrent units (GRUs) are important variants of this kind of network, which overcome the vanishing gradient problem of RNNs [21]. Compared with GRUs, LSTM is more accurate on the dataset using long sequences. Due to the superiority of LSTM in time series forecasting, researchers have gradually paid attention to its application in EPF [22, 23, 24]. However, as with other DNNs, when LSTM models are applied to high-dimensional data, a critical issue occurs, known as the curse of dimensionality [25]. This means that, with a large amount of features,<sup>1</sup> the performance of LSTM will degrade because of overfitting [26]. Thus, LSTM cannot be employed directly for electricity price prediction with a large number of features as input under consideration of market coupling. While some researchers have

\*Corresponding author

Email address: wei.n.li@ntnu.no (Wei Li)

<sup>1</sup>In machine learning, features are individual independent variables as input in a model.

## Nomenclature

$\beta$	Coefficients of independent variables	$i$	Input/update gate's activation vector
$\hat{y}$	Predicted output	$o$	Output gate's activation vector
$\lambda$	Tuning parameter	$r_1, r_2$	Random number from the interval [0, 1]
$\mu$	Expected electricity exchange capacity	$W$	Weight matrix
$\omega$	Constant that determines the significance of $\vec{v}$	$w^1$	Weight vector between the input and the hidden layer
$\otimes$	Element-wise product	$w^2$	Weight vector between the hidden layer and the output
$\vec{g}$	Global current best solution	$X, Y$	Variable
$\vec{p}$	Particle's past best solution	$x$	Input variable
$\vec{v}$	Velocity of the particle	$y$	Output variable
$\Phi$	Kernel function	<b>Abbreviations</b>	
$\phi$	Activation function	CNNs	Convolutional neural networks
$\rho$	PC coefficient	ConvLSTM	Convolutional LSTM
$\sigma_X$	Standard deviation of $X$	DM	Diebold-Mariano
$\sigma_Y$	Standard deviation of $Y$	DNNs	Deep neural networks
$\sigma_{FD}$	Cross-border flow deviation	ELM	Extreme learning machine
$\mathbf{w}$	Parameter vector	EPF	Electricity price forecasting
Tanh	Hyperbolic tangent function	FNNs	Forward neural networks
$\varepsilon$	Error variable	GA	Genetic algorithm
$\xi, \xi_i^*$	Slack variable	GRUs	Gated recurrent units
$\zeta$	Sigmoid function	LSTM	Long short-term memory
$b$	Bias vector	MAE	Mean absolute error
$C$	Regularisation constant	MAPE	Mean absolute percentage error
$c$	Cell state vector	MSE	Mean squared error
$c_1$	Constant that determines the significance of $\vec{p}$	NARMAX	Moving Average with eXogenous Input
$c_2$	Constant that determines the significance of $\vec{g}$	PC	Pearson's Correlation
$cov$	Covariance	PSO	Particle swarm optimisation
$d^{F1,F2}$	Loss differential	RMSE	Root mean squared error
$e$	Forecast error	RNNs	Recurrent neural networks
$f$	Forget gate's activation vector	SHAP	SHapley Additive exPlanations
$g(*)$	Loss function	SMAPE	Symmetric mean absolute percentage error
$h$	Hidden state vector		

attempted to involve explanatory variables from integrated markets to make a prediction of electricity price by neural networks [27, 28, 29], no existing research has investigated the state-of-the-art LSTM-based deep neural networks for this purpose. Besides, some research starts to pay attention to the influence of the market integration on Nord Pool [30, 31, 32]. However, efficient ways to utilise the ever-growing information from the electricity market integration for the Nordic EPF have yet to be explored.

Typically, feature selection is an efficient way to avoid the curse of dimensionality. It is the process of selecting a subset of relevant attributes in the dataset when developing a predictive model. It can reduce the computation time, improve model prediction performance, and help to get a better understanding of the dataset [33]. The ideal feature selection is to search the space of all variable subsets with an algorithm, which is impractical except for quite small sized feature spaces. However, as the space of variables subset grows exponentially with the number of variables, heuristic search methods are commonly used to

search for an optimal subset [34]. The current research on feature selection algorithms can be categorised as filter, wrapper, and embedded methods [33]. In particular, the filter methods use a proxy measure to estimate a feature subset before training a prediction model. Pearson's Correlation (PC) is a typical indirect assessing measure for the regression problem [35]. In contrast, the wrapper methods evaluate selected feature subsets by employing a predictive model directly. Each subset is used to train a new forecasting model, and the optimisation method is used to search for the best performing model in the process of feature selection. The embedded methods can implement an automatic feature selection in the process of estimating the parameters of predictive models. This means this catch-all group of techniques performs the process of feature selection during the training of the model. The particle swarm optimisation combined with the extreme learning machine method (PSO-ELM) and genetic algorithm combined with the extreme learning machine method (GA-ELM) are two typical wrapper-based methods. They have been widely used for various feature selection

problems [36, 37, 38, 39, 40, 41]. Guyon et al. [42] proposed another popular wrapper approach, known as recursive feature elimination combined with support vector machine for regression (RFE-SVR). The Lasso regression method is one of the most popular embedded feature selection methods proposed by Tibshirani [43].

### 1.1. Contributions

To the best of our knowledge, no existing study considers how to apply LSTM models in an integrated market EPF and detect the impact of the features from cross-border markets on EPF. To fill this scientific gap, we propose three hybrid architectures of LSTM-based deep learning predictive models combined with advanced feature selection algorithms: the two-step hybrid architecture, the autoencoder hybrid architecture, and the two-stage hybrid architecture. Different feature selection methods have different selection mechanisms, which will lead to different sets of selected features. To explore the influence of different feature selections on LSTM-based EPF, we employed five feature selection algorithms, PC, PSO-ELM, GA-ELM, RFE-SVR, and the Lasso regression method, in the case study of Nord Pool and its neighbouring, interconnected countries. The main contributions of this study are as follows:

1. We compare and analyse the forecasting performance of the proposed models in the case study of the Nord Pool system price forecasting, considering six integrated markets (sixty-two features). The results indicate that the cross-border markets influence the Nordic electricity price formation. As the rapid market coupling development in Europe, we show that it is necessary to consider cross-border information for EPF in future studies.
2. We introduce three architectures of hybrid LSTM-based deep neural networks for EPF and conclude that different feature selection algorithms yield divergent subsets of features, which, in turn, affect the prediction accuracy of the proposed LSTM models. In addition, the results show that hybrid models are an efficient way to deal with the ever-growing information and obtain accurate prediction results in cross-border markets.
3. We employ a game theoretical approach (SHapley Ad-ditive exPlanations) to explore the relevance of various cross-border features in EPF. The analysis of Shapley values increases the transparency of the prediction and provides advice for policy makers and market participants.

The remainder of this paper is organised as follows. Section 2 describes the dataset used in this research. In Section 3, we present the methodology. Section 4 describes the model training and introduces evaluation criteria applied in the empirical study. Section 5 reports the forecasting results of the implemented models. Finally, Section 6 concludes the paper and proposes future research developments.

## 2. Data description

The Nordic system price is the central reference price in the Nordic electricity market. It is used as a settlement price for the derivatives market. Each hourly system price is calculated by Nord Pool based on all bids and offers posted in Nordic bidding zones, which is referred to as a market-clearing price, without taking into account any congestion restrictions. The daily system price represents the arithmetic average of the 24 hourly prices. This paper discusses and evaluates several hybrid LSTM-based approaches for the prediction of the Nordic hourly and daily system prices.

Previous empirical research on the prediction of electricity prices has considered information from both price and supply/demand sides. To find out what matters when predicting the day-ahead Nordic system price in coupling markets, we also included the electricity exchange between Nord Pool and its integrated countries. The Russian electricity market is excluded because it differs significantly from European models. To consider the influence of the correspondence between electricity flow and capacity, we introduced a new daily feature, namely the cross-border flow deviation. It can be calculated as  $\sigma_{FD} = \sqrt{\sum_{i=1}^N (X_i - \mu_i)^2 / N}$ , where  $X_i$  is the hourly electricity flow,  $\mu_i$  is the hourly expected exchange capacity, and  $N$  stands for 24 hours.

In summary, we consider eight categories of input features: day-ahead price, production, production prognosis, consumption, consumption prognosis, currency exchange rate, cross-border electricity flow, and flow deviation. The first five are the basic features from local markets for predicting electricity price. Some historical/predictive information, such as weather and human social activities, does not directly impact electricity price but influences the supply/demand for electricity, incorporated in those five fundamental variables. The last three are the features spawned from cross-border trade.

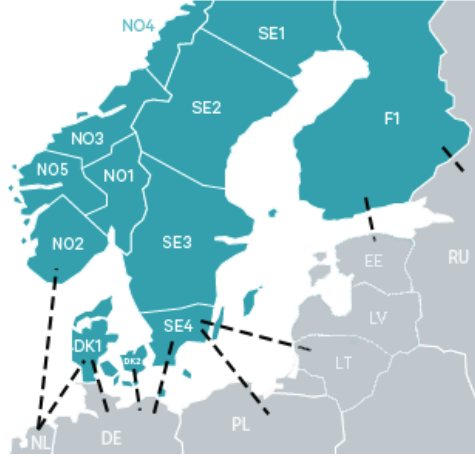
### 2.1. Data

We collected data from the Nord Pool,<sup>2</sup> Thomson Reuters Eikon,<sup>3</sup> and Entsoe.<sup>4</sup> The available time series ranges from 01/01/2015 to 31/12/2019. Nord Pool provides cross-border transmissions with Germany (DE), the Netherlands (NL), Lithuania (LT), Estonia (EE), Poland (PL), and Russia (RU). The map in Figure 1 shows both the Nord Pool markets as well as transmissions (black dashed lines) between the Nord Pool and its coupling bidding areas. There are five bidding zones in Norway (NO1, NO2, NO3, NO4, and NO5), four in Sweden (SE1, SE2, SE3, and SE4), and two in Denmark (DK1 and DK2), and one in Finland (FI). Since the transmissions between DK1 and NL started at 01/09/2019, the data series is not sufficient for the application of deep learning models. Besides, the electricity exchange between SE4 and LT started at 09/12/2015. Therefore, the entire available dataset employed in this study

<sup>2</sup>Nord Pool: <https://www.nordpoolgroup.com/>

<sup>3</sup>Thomson Reuters Eikon: <https://eikon.thomsonreuters.com/>

<sup>4</sup>Entsoe: <https://transparency.entsoe.eu/>



**Figure 1** Overview of the Nord Pool market coupling.

ranges from 09/12/2015 to 31/12/2019. The features included in the dataset are shown in Table 1. The hourly data is converted into the daily data by the arithmetic average (e.g., price) or the aggregate (e.g., flow).

## 2.2. Cross-border electricity transmission

Figure 2 shows the electricity exports from Germany, the Netherlands, Lithuania, Poland, and Russia in 2019.<sup>5</sup> The exports to the Nord Pool comprised 16.03% of the whole exports from these coupling countries. In Figure 3, we can see that the electricity exports of the Nord Pool comprised 4.82% of its total production in 2019. The EU aims to achieve 15% interconnection capacity in 2030 for each EU country [44].

**Table 1** The features included in the dataset.

Feature	Description (Units)	Data Source
F1	System Day-ahead price 1-Lag (EUR/MWh)	Nord Pool
F2	SE1 Day-ahead price (EUR/MWh)	Nord Pool
F3	SE2 Day-ahead price (EUR/MWh)	Nord Pool
F4	SE3 Day-ahead price (EUR/MWh)	Nord Pool
F5	SE4 Day-ahead price (EUR/MWh)	Nord Pool
F6	FI Day-ahead price (EUR/MWh)	Nord Pool
F7	DK1 Day-ahead price (EUR/MWh)	Nord Pool
F8	DK2 Day-ahead price (EUR/MWh)	Nord Pool
F9	NO1 Day-ahead price (EUR/MWh)	Nord Pool
F10	NO2 Day-ahead price (EUR/MWh)	Nord Pool
F11	NO3 Day-ahead price (EUR/MWh)	Nord Pool
F12	NO4 Day-ahead price (EUR/MWh)	Nord Pool
F13	NO5 Day-ahead price (EUR/MWh)	Nord Pool
F14	EE Day-ahead price (EUR/MWh)	Nord Pool
F15	LT Day-ahead price (EUR/MWh)	Nord Pool
F16	PL Day-ahead price (PLN/MWh)	Thomson Reuters Eikon
F17	DE Day-ahead price (EUR/MWh)	Thomson Reuters Eikon
F18	NL Day-ahead price (EUR/MWh)	Thomson Reuters Eikon
F19	Nordic production (MWh)	Nord Pool
F20	EE production (MWh)	Nord Pool
F21	LT production (MWh)	Nord Pool

Continued on next column

Continued from previous column

Feature	Description (Units)	Data Source
F22	PL production (MWh)	Entsoe
F23	DE production (MWh)	Entsoe
F24	NL production (MWh)	Entsoe
F25	Nordic production prognosis (MWh)	Nord Pool
F26	EE production prognosis (MWh)	Nord Pool
F27	LT production prognosis (MWh)	Nord Pool
F28	PL production prognosis (MWh)	Entsoe
F29	DE production prognosis (MWh)	Entsoe
F30	NL production prognosis (MWh)	Entsoe
F31	Nordic consumption (MWh)	Nord Pool
F32	EE consumption (MWh)	Nord Pool
F33	LT consumption (MWh)	Nord Pool
F34	PL consumption (MWh)	Entsoe
F35	DE consumption (MWh)	Entsoe
F36	NL consumption (MWh)	Entsoe
F37	Nordic consumption prognosis (MWh)	Nord Pool
F38	EE consumption prognosis (MWh)	Nord Pool
F39	LT consumption prognosis (MWh)	Nord Pool
F40	PL consumption prognosis (MWh)	Entsoe
F41	DE consumption prognosis (MWh)	Entsoe
F42	NL consumption prognosis (MWh)	Entsoe
F43	EUR/NOK	Nord Pool
F44	EUR/SEK	Nord Pool
F45	EUR/DKK	Nord Pool
F46	EUR/PLN	Thomson Reuters Eikon
F47	NO2 ↔ NL flow (MWh)	Nord Pool
F48	DK1 ↔ DE flow (MWh)	Nord Pool
F49	DK2 ↔ DE flow (MWh)	Nord Pool
F50	SE4 ↔ DE flow (MWh)	Nord Pool
F51	SE4 ↔ PL flow (MWh)	Nord Pool
F52	SE4 ↔ LT flow (MWh)	Nord Pool
F53	FI ↔ EE flow (MWh)	Nord Pool
F54	FI ↔ Russia flow (MWh)	Nord Pool
F55	NO2 ↔ NL flow deviation	Calculation
F56	DK1 ↔ DE flow deviation	Calculation
F57	DK2 ↔ DE flow deviation	Calculation
F58	SE4 ↔ DE flow deviation	Calculation
F59	SE4 ↔ PL flow deviation	Calculation
F60	SE4 ↔ LT flow deviation	Calculation
F61	FI ↔ EE flow deviation	Calculation
F62	FI ↔ Russia flow deviation	Calculation

## 3. Methodology

### 3.1. LSTM

The LSTM architecture was initially introduced by [45] and has since been enhanced by other researchers to achieve better performance [46, 47, 48]. An LSTM network is a special kind of recurrent neural network that is capable of learning long-term dependencies. Unlike simple RNNs, an LSTM network has built-in mechanisms that control how information is memorised or abandoned throughout time. The architecture of the LSTM network is shown in Figure 4 and is defined by the following system of equations [49]:

<sup>5</sup>Fraunhofer ISE provides the electricity exchange data of Germany/Europe: <https://www.energy-charts.de/>

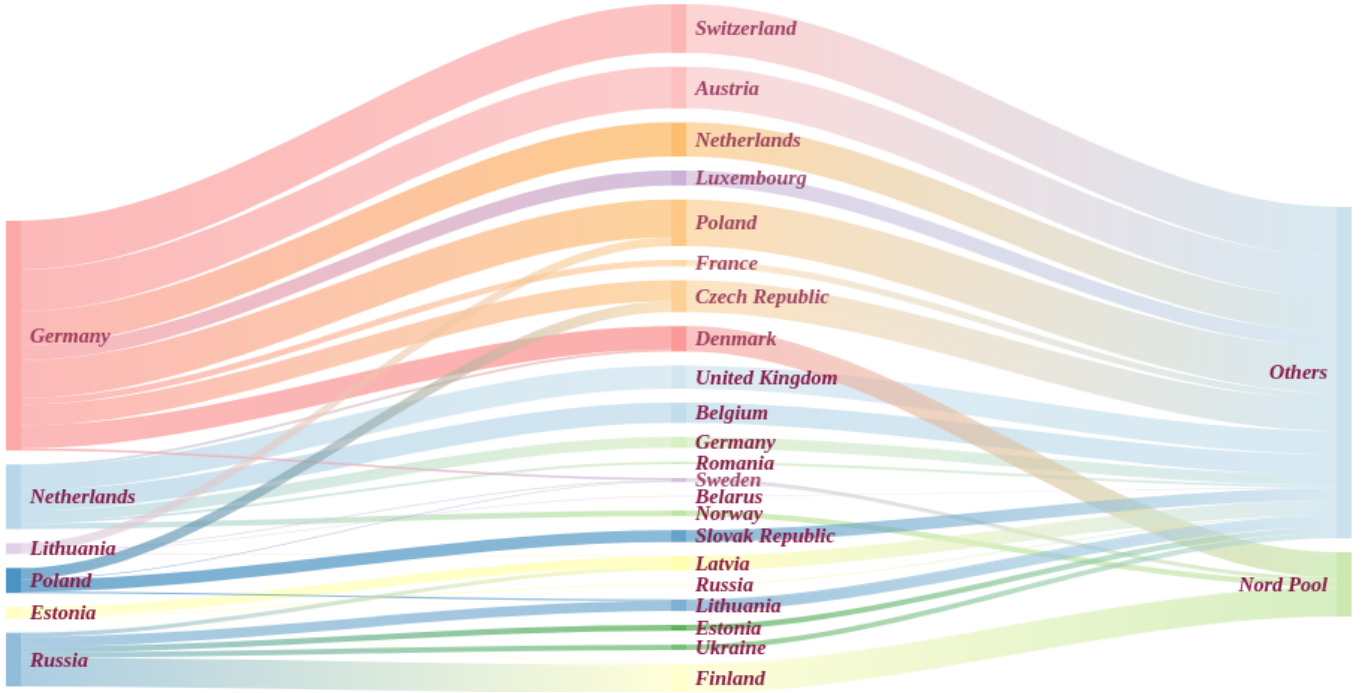


Figure 2 The electricity cross-border transmission from the coupling countries to Nord Pool.

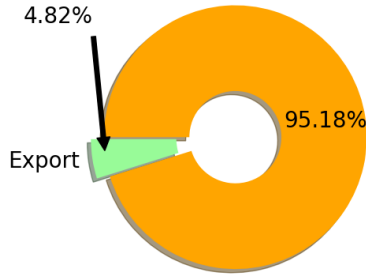


Figure 3 The percentage of the Nord Pool production for exporting.

$$f_t = \zeta(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \quad (1)$$

$$i_t = \zeta(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (2)$$

$$o_t = \zeta(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \quad (3)$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes \text{Tanh}(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (4)$$

$$h_t = o_t \otimes (c_t) \quad (5)$$

where  $f_t$ ,  $i_t$ ,  $o_t$ ,  $c_t$ , and  $h_t$  indicate the values of the forget gate state, input gate state, output gate state, memory cell, and hidden state at time  $t$  in the sequence, respectively.  $\zeta$  and  $\text{Tanh}$  are the sigmoid function and hyperbolic tangent function,  $W$  and  $b$  are the weight matrix and bias vector, and  $\otimes$  denotes the element-wise product.

### 3.2. Architectures of hybrid models

Typically, there are three hybrid architectures for EPF when working with LSTM. The first architecture consists of two steps, as shown in Figure 5. The first step includes data processing and feature selection, and the second step contains training

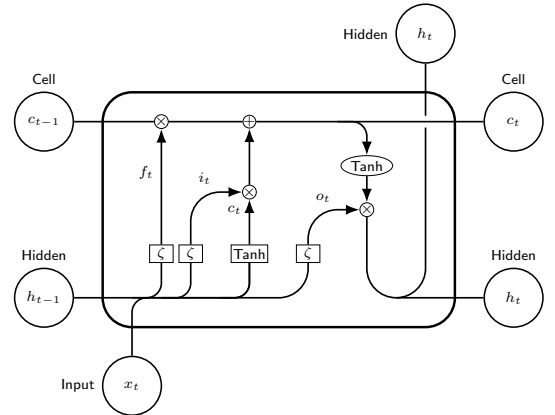


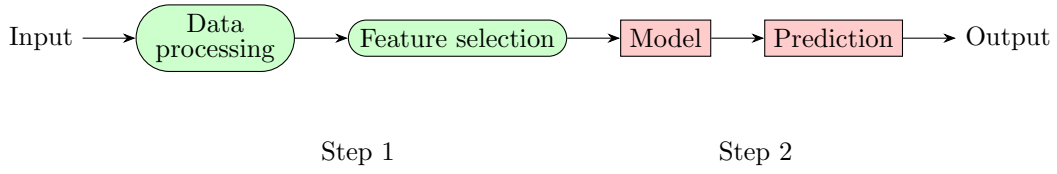
Figure 4 LSTM cell.

of the predictive models and making predictions. The second architecture can be referred to as an autoencoder model. Here, the input data will be turned into a compressed representation rather than specifically showing which features are selected, as shown in Figure 6. The third combines the two aforementioned architectures, and it is referred to as two-stage feature selection. In this architecture, the explanatory variables will be selected by some feature selection method in the first stage. The selected features will then become the input for the autoencoder models in the second stage. Figure 7 shows this architecture.

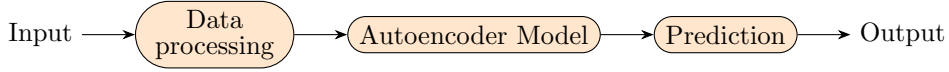
### 3.3. Feature selection methods

#### 3.3.1. PC

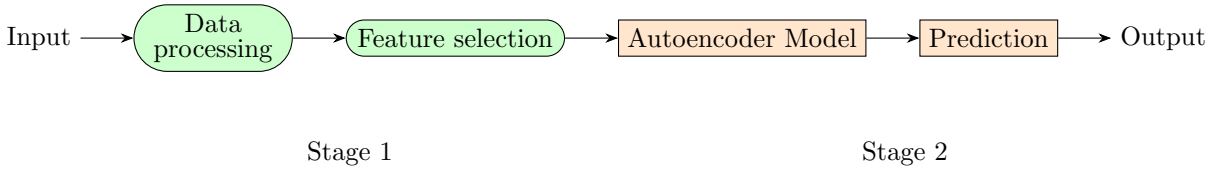
The PC coefficient is a statistic used to measure the linear relationship between two data samples. Given two variables



**Figure 5** The flowchart of a two-step hybrid model. The green nodes stand for the first step, and the red nodes stand for the second step.



**Figure 6** The flowchart of an autoencoder hybrid model. The orange nodes stand for the autoencoder process.



**Figure 7** The flowchart of a two-stage hybrid model. The green nodes stand for the first stage, and the orange nodes stand for the second stage.

$(X, Y)$ , the formula of the PC coefficient  $\rho$  is given by the following:

$$\rho(X, Y) = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} \quad (6)$$

where  $\text{cov}$  is the covariance,  $\sigma_X$  is the standard deviation of  $X$ , and  $\sigma_Y$  is the standard deviation of  $Y$ .

### 3.3.2. PSO-ELM and GA-ELM

PSO-ELM and GA-ELM are wrapper-based hybrid methods. ELM is a single hidden layer feedforward neural network. Its fast training [50] contributes to the popularity of its employment as a predictive model in wrapper-based feature selection [51, 52, 53]. The output of ELM is calculated as follows:

$$F_L(x) = \sum_{i=1}^L w_i^2 \phi(w_i^1 x_j + b_i), j = 1, \dots, N \quad (7)$$

where  $L$  is the number of hidden units,  $N$  is the number of training samples,  $w^2$  is the weight vector between the hidden layer and the output,  $w^1$  is the weight vector between the input and the hidden layer,  $\phi(\ast)$  denotes an activation function,  $b$  is a bias vector, and  $x$  is the input vector.

PSO and GA are different types of optimisation algorithms, which provide the subsets of features as the input to the ELM to detect the optimal feature selection. The basic idea of PSO is that a swarm of particles moves through the search space. The movement of each particle is guided by its own known best position and the entire swarm's known best position. PSO performs the search for the optimum by iteratively updating the velocities of the particles in the swarm [54]. The GA is a search metaheuristic that was inspired by Darwin's theory of natural

selection. In general, GAs will search for the optimal solution from a set of possible solutions, called a population. A solution is referred to as a chromosome or an individual. These chromosomes evolve over a number of generations by recombination (cross-over) and mutation [55]. The detailed introduction of the methods can be found in Appendix A.1 and A.2.

### 3.3.3. RFE-SVR

RFE-SVR is another wrapper-based feature selection method. The core idea of this algorithm is to search for the best subset of features by starting with all features and discarding the less important features. In particular, the RFE algorithm operates with SVR to perform feature selection and regression simultaneously. SVR performs well in high dimensionality space [56]. The detailed explanation of SVR is in Appendix A.3.

### 3.3.4. Lasso regression

The Lasso regression aims to increase the prediction accuracy of regression models by adding a penalty  $\lambda \sum_{j=1}^n |\beta_j|$  to the loss function. This means that instead of minimising a loss function,  $\sum_{i=1}^m (y_i - \sum_{j=1}^n x_{ij} \beta_j)^2$ , the loss function becomes  $\sum_{i=1}^m (y_i - \sum_{j=1}^n x_{ij} \beta_j)^2 + \lambda \sum_{j=1}^n |\beta_j|$ , where  $y$  is the vector of the dependent variable,  $x$  denotes independent variables, the  $\beta$  are the corresponding coefficients. The algorithm has the advantage that it shrinks some of the less critical coefficients of features to zero. Therefore, it removes less relevant features.

### 3.4. Autoencoder Model

An autoencoder is typically a neural network that aims to filter and compress the representation of its input, which consists of two components: an encoder and a decoder, shown in Figure 8. The encoder typically accepts a set of input data and compresses the information into an intermediate vector. The

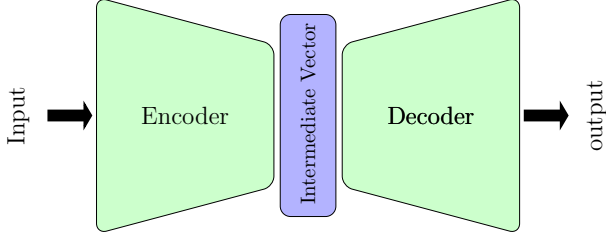


Figure 8 Structure of the Encoder-Decoder model.

decoder is typically a predictive model. In our case, a decoder is an LSTM network, and the encoders are LSTM, CNN, and convolutional layers, as described in the following.

#### 3.4.1. LSTM-LSTM Encoder-Decoder model

In an LSTM-LSTM Encoder-Decoder model, an LSTM model is used as the encoder to process the raw input time series and to transform it into an intermediate vector. LSTM is capable of extracting the complex dynamic information within the temporal input series and filtering useful information from long input series via internal memory.

#### 3.4.2. CNN-LSTM Encoder-Decoder model

In a CNN-LSTM Encoder-Decoder model, a CNN is the encoder to filter the input data. CNNs were originally and successfully used to process the image input data in image recognition tasks [57] or the sequence of input data in natural language processing problems [58]. The convolutional layers are usually followed by a pooling layer, which extracts information from the convolved features and produces a lower dimensional output. Then, the output values are flattened into a long intermediate vector representation.

#### 3.4.3. Convolutional LSTM Encoder-Decoder model

The computational mechanism of the convolutional LSTM (ConvLSTM) is similar to that of CNN-LSTM [59]. Unlike the CNN-LSTM, where the CNN model generates the input for the LSTM model, in the ConvLSTM model, the LSTM neural network processes the extracted information directly from preceding convolutional layers.

## 4. Experimental details

In this section, we introduce the concepts and methods employed in the process of training and evaluation of the constructed models. This section includes five parts. First, statistical performance measures are specified for evaluation and comparison. Second, theory and preliminary work for training models are explained. Third, configuration parameters are set for feature selection algorithms and LSTM-based models. Fourth and fifth, the benchmark model and feature explanation method used in the empirical study are introduced. They are necessary preparations for conducting the experiments.

## 4.1. Statistical performance measures

### 4.1.1. Evaluation metrics

In this paper, we employ several indicators to evaluate the accuracy of predictions: the mean absolute error (MAE), the root mean squared error (RMSE), the mean absolute percentage error (MAPE), and the symmetric mean absolute percentage error (SMAPE) as the model estimator. They are commonly adopted in EPF research [1]. Given a predicted output vector,  $\hat{y}_k = [\hat{y}_1, \dots, \hat{y}_N]$ , and a real output vector,  $y_k = [y_1, \dots, y_N]$ , the MAE, RMSE, MAPE, and SMAPE can be calculated as follows:

$$\text{MAE} = \frac{1}{N} \sum_{k=1}^N |y_k - \hat{y}_k| \quad (8)$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^N (y_k - \hat{y}_k)^2} \quad (9)$$

$$\text{MAPE} = \frac{100}{N} \sum_{k=1}^N \left| \frac{y_k - \hat{y}_k}{y_k} \right| \quad (10)$$

$$\text{SMAPE} = \frac{100}{N} \sum_{k=1}^N \frac{|y_k - \hat{y}_k|}{(|y_k| + |\hat{y}_k|)/2} \quad (11)$$

### 4.1.2. Diebold-Mariano test

The metrics for assessing the forecasting accuracy mentioned above cannot guarantee that the observed difference from two predictive models is statistically significant. In this context, the Diebold-Mariano (DM) test is typically used for evaluating the performance of two models [60, 61]. Given the actual values of a time series  $[y_t; t = 1, \dots, T]$ , two forecasts from two models,  $[\hat{y}_{1t}; t = 1, \dots, T]$  and  $[\hat{y}_{2t}; t = 1, \dots, T]$ , and the associated forecast errors,  $e_{1t} = \hat{y}_{1t} - y_t$  and  $e_{2t} = \hat{y}_{2t} - y_t$ , the DM test defines the loss differential between the two forecasts by the following:

$$d_t^{F1, F2} = g(e_{1t}) - g(e_{2t}) \quad (12)$$

where  $g(*)$  stands for loss function. In a one-sided DM test, the hypotheses is the following:

$$\begin{aligned} H_0 &: \mathbb{E}(d_t^{F1, F2}) \geq 0, \\ H_1 &: \mathbb{E}(d_t^{F1, F2}) < 0. \end{aligned} \quad (13)$$

A one-sided DM test is used to detect whether F2 is better than F1. If  $H_0$  is rejected, the test suggests that the accuracy of F1 is, statistically, significantly better than F2. The complementary one-sided DM test can be expressed as follows:

$$\begin{aligned} H_0 &: \mathbb{E}(d_t^{F1, F2}) \leq 0, \\ H_1 &: \mathbb{E}(d_t^{F1, F2}) > 0. \end{aligned} \quad (14)$$

If  $H_0$  is rejected, the test suggests that the accuracy of F2 is, statistically, significantly better than F1. In this study, we employ a one-sided DM test to assess the forecasting performance of the proposed models. We chose  $d_t^{F1, F2} = |e_{1t}| - |e_{2t}|$  as the loss differential.

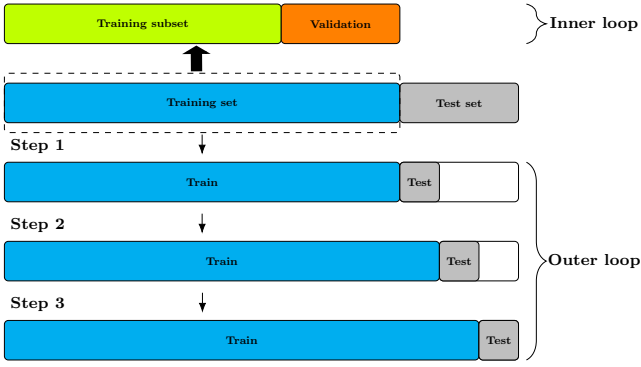


Figure 9 Walk forward nested cross-validation.

## 4.2. Model training

### 4.2.1. Walk forward nested cross-validation

To avoid over-fitting, it is common to include a validation set to evaluate the generalisation ability of the training model. The cross-validation is referred to as a method for tuning the hyperparameters and producing robust measurements of model performance. In [62], a nested cross-validation procedure was introduced, which considerably reduced the bias and provided an almost unbiased estimate of the true error. Because new observations become available over time, in time series modelling, we implemented a walk forward nested cross-validation in which the forecast rolls forward in time. More specifically, we successively considered each day as the test set and assigned all previous data to the training set (Outer loop). The training set is split into a training subset and a validation set. The validation set data comes chronologically after the training subset (Inner loop). Walk forward validation involves moving along the time series one time step at a time. The process requires multiple models to be trained and evaluated, but the additional computational cost will provide a more robust estimate of the expected performance of the predictive model on unseen data. It is shown in Figure 9.

### 4.2.2. Data division

We divided the whole database into two subsets: a training set and a test set. The training set includes a training subset and a validation subset, as shown in the dashed box in Figure 9. We initially apportioned the data set into training, validation, and test sets, with an 80-10-10 split. The magnitude of the test and validation set is anchored during the walk-forward test.

### 4.2.3. Data processing

For neural network model training, the input data is usually normalised to the intervals [0,1]. This is not only done because the normalised data will require less time to train, but the prediction performance will also increase. In addition, we linearly interpolate the missing data and eliminate duplicates due to daylight saving.

### 4.2.4. Ten experiments

Training algorithms for deep learning models have usually required the initialisation of the weights of neural networks

from which to begin the iterative training [63]. The random initial conditions for an LSTM network can result in different performances each time a given configuration is trained. Thus, we employed ten experiments for each model to reduce the impact of the variability on performance evaluation. Models were evaluated after taking the average of the experiments.

## 4.3. Model configuration parameters

### 4.3.1. Parameters of feature selection

The feature selection stopping criterion varies by algorithm, which is controlled by the parameters of models. The applied configuration of PSO was [ $c_1$ : 0.5,  $c_2$ : 0.3,  $\omega$ : 0.7], and the stop condition is satisfied after 10,000 iterations. For GA, the crossover possibility and mutation possibility were set to 0.5 and 0.2, respectively. The population size was 100, and the maximum number of generations was 10,000. On the basis of the predictive ELM, the amount of the selected features by PSO-ELM and GA-ELM was automatically set to 30. For the sake of input consistency, the magnitude of the selected features of the rest of the models was set to 30 as well. For the PC method, we ranked all features attributable to the correlation coefficients and selected the first 30 features. In terms of RFE-SVR, we ranked features by importance, discarded the least important features, and refit the model until 30 features remained. The regularisation parameter,  $\lambda$ , in Lasso regression was 0.02.

### 4.3.2. Network hyperparameters

Our study aimed to investigate the applications and impacts of different types of feature selection methods in a predictive LSTM architecture. We used a coherent configuration of a specific LSTM model for comparison and did not perform an extensive hyperparameter optimisation to search for the optimal configuration. After an inexhaustive grid search, we constructed our prediction model from an LSTM model with a single hidden layer of 300 units, followed by a fully connected dense layer with 100 neurons that preceded the output layer. The LSTM encoder has a hidden layer with 300 units. In the CNN-LSTM Encoder-Decoder model, the CNN encoder has two convolutional layers, with 96 units to amplify any salient features, followed by a max-pooling layer. In the ConvLSTM Encoder-Decoder model, the encoder is a convolutional layer with 64 units. The input sequence length is 14 days (2 weeks, commonly used in EPF). The optimiser is the Adam algorithm, and the loss function is Mean Squared Error (MSE).

## 4.4. Benchmark model

Among the traditional methods, the statistical models perform best for EPF. Thus, we select the Nonlinear AutoRegressive Moving Average with eXogenous Input (NARMAX) model as the benchmark (trained with the optimal structure) for our case study. This statistical model is widely used in energy price forecasting to handle multiple nonlinear inputs [32, 64]. The equation is represented as the following:

$$y(t) = F^{\ell} [y(t-1), \dots, y(t-N_y), x(t), \dots, x(t-N_x), \varepsilon(t-1), \dots, \varepsilon(t-N_{\varepsilon})] + \varepsilon(t) \quad (15)$$



**Table 2**  
The proposed models.

Mode	Category	Model Explanation
M0	Benchmark	NARMAX model
M1	Filter method	PC-LSTM model
M2	Wrapper method	PSO-ELM-LSTM model
M3	Wrapper method	GA-ELM-LSTM model
M4	Wrapper method	RFE-SVR-LSTM model
M5	Embedded method	LASSO-LSTM model
M6	Autoencoder method	LSTM-LSTM Encoder-Decoder model
M7	Autoencoder method	CNN-LSTM Encoder-Decoder model
M8	Autoencoder method	CovLSTM Encoder-Decoder model
M9	Two-stage method	PC-LSTM-LSTM Encoder-Decoder model
M10	Two-stage method	PSO-ELM-LSTM-LSTM Encoder-Decoder model
M11	Two-stage method	GA-ELM-LSTM-LSTM Encoder-Decoder model
M12	Two-stage method	RFE-SVR-LSTM-LSTM Encoder-Decoder model
M13	Two-stage method	LASSO-LSTM-LSTM Encoder-Decoder model

where  $x(t)$  is the input and  $y(t)$  is the output time-series;  $\varepsilon(t)$  is the uncertainties and possible noise;  $N_u$ ,  $N_y$ , and  $N_e$  are the input, output, and prediction error lags, respectively; and  $F^\ell$  is a nonlinear function.

#### 4.5. Feature explanation method

In this study, we used SHapley Additive exPlanations (SHAP) values to interpret the impact of certain values of a given feature on the expected price prediction. SHAP<sup>6</sup> is a theoretic game method to explain the output of machine learning models [65, 66, 67]. The Shapley value is used to assess the feature relevance relative to the expectation of the output [68]. In particular, a Kernel SHAP is used for explaining an optimal SVR model obtained by grid-search on the dataset.

## 5. Results

In this section, we report the empirical results obtained by the application of the introduced models. For similarity of presentation, the list of models and their acronyms are shown in Table 2.

### 5.1. Analysis of empirical results

The results of the feature selection are shown in Table 3. Overall, it can be observed that different selection mechanisms lead to different selections. From Table 3, we can see that M1 selects all the day-ahead prices. It is not surprising that the day-ahead prices from different bidding areas are more relevant to the Nord Pool system price than other feature variables. However, the over-selection results in information redundancies. Some researchers have recognised that the redundancy among features decreases the model’s performance [69, 70, 71]. Compared to M1, the wrapper-based methods, M2 and M3, eliminate several price variables rather than the other categories of variables. It is worth noticing the two methods does not select the lag system price (F1), which is commonly used in

time series, given their short-term autoregressive nature. As introduced in section 3.3.2, PSO-ELM and GA-ELM are widely used in research. However, the optimisation methods, such as PSO and GA, have the problem of trapping in local optima. Although re-setting and experimenting can increase the chance of avoiding traps, when dealing with high dimensional data sets, the optimisation methods still cannot guarantee they will find a global optimum solution, and they are not suitable for all cases [72]. The straightforward concept and fast computation of ELM contributed to its widespread application in an exhaustive grid search, but it does not consider the sequential relationships in time series data, as with other traditional neural networks. These could be the reasons why the two methods eliminate the lag system price as an input. The other wrapper model M4, selects various types of features, and eliminates less price features compared to M2 and M3. It is interesting to note that M4 does not pick up any features from the cross-border flow deviation. For the Lasso regression method, M5, we can summarise that it diversely chooses features such as M4 but puts more emphasis on electricity transmission.

**Table 3** The results of feature selection.

Feature	Feature selection model				
	M1	M2	M3	M4	M5
F1	✓	✗	✗	✓	✓
F2	✓	✓	✓	✓	✓
F3	✓	✗	✗	✓	✓
F4	✓	✓	✗	✓	✗
F5	✓	✗	✗	✗	✗
F6	✓	✗	✗	✗	✗
F7	✓	✗	✗	✗	✗
F8	✓	✓	✗	✗	✗
F9	✓	✗	✗	✓	✗
F10	✓	✗	✗	✓	✓
F11	✓	✓	✗	✓	✓
F12	✓	✗	✓	✓	✓
F13	✓	✓	✓	✓	✓
F14	✓	✗	✓	✗	✗
F15	✓	✗	✗	✗	✗
F16	✓	✗	✗	✗	✗
F17	✓	✗	✓	✗	✗
F18	✓	✓	✗	✓	✓
F19	✗	✓	✗	✓	✗
F20	✗	✓	✓	✓	✗
F21	✗	✗	✓	✗	✗
F22	✗	✓	✓	✓	✗
F23	✗	✓	✗	✓	✗
F24	✓	✓	✓	✗	✗
F25	✗	✗	✗	✓	✓

Continued on next column

<sup>6</sup>The Python package SHAP is available at <https://github.com/slundberg/shap>

Continued from previous column

Feature	Feature selection model				
	M1	M2	M3	M4	M5
F26	X	X	✓	✓	✓
F27	✓	✓	✓	X	✓
F28	X	✓	✓	X	X
F29	X	X	X	✓	✓
F30	X	✓	X	X	✓
F31	X	✓	X	✓	X
F32	X	✓	✓	✓	✓
F33	✓	✓	X	X	X
F34	✓	X	✓	✓	✓
F35	X	X	X	✓	✓
F36	X	✓	✓	✓	✓
F37	X	X	X	✓	X
F38	✓	✓	✓	✓	X
F39	✓	✓	✓	✓	✓
F40	X	X	✓	X	X
F41	✓	✓	X	✓	✓
F42	X	X	X	X	X
F43	✓	X	✓	✓	X
F44	✓	✓	✓	✓	✓
F45	✓	X	✓	X	X
F46	X	✓	X	X	✓
F47	X	X	X	X	X
F48	X	X	X	✓	✓
F49	X	X	✓	X	✓
F50	X	✓	✓	X	✓
F51	X	✓	X	X	✓
F52	X	✓	✓	X	X
F53	X	X	X	✓	✓
F54	X	X	✓	X	✓
F55	X	X	X	X	X
F56	X	X	✓	X	X
F57	X	✓	✓	X	X
F58	X	X	✓	X	✓
F59	✓	✓	✓	X	✓
F60	✓	✓	✓	X	X
F61	X	✓	X	X	✓
F62	X	X	X	X	X

Note: ✓ denotes that the feature is selected;

X denotes that the feature is not selected.

To evaluate the statistical significance in the difference of predictive accuracy, one-side DM tests, as defined in section 4.1.2, were applied, and the results are shown in Table 4. Table 5 exhibits the performance comparison of all the models in terms of SMAPE. As expected, the proposed LSTM mod-

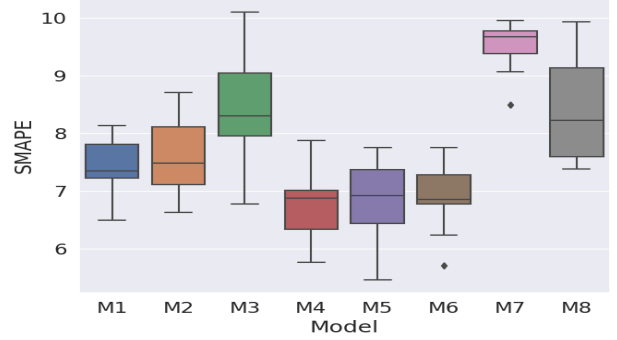


Figure 10 The SMAPEs of 10 experiments for M1, M2, M3, M4, M5, M6, M7, and M8.

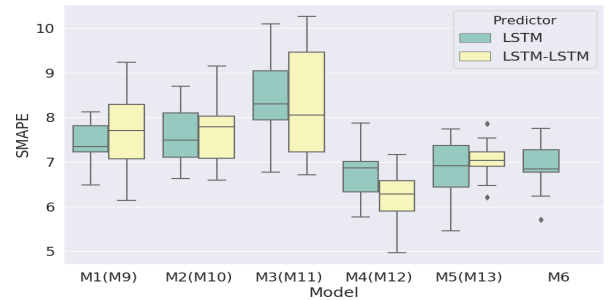


Figure 11 The comparison of SMAPEs between two-step LSTM models and two-stage LSTM-LSTM models.

els are overwhelmingly better than the benchmark statistical model, M0. The superior performance of deep learning models to statistical models has been recognised by numerous studies [22, 73]. Moreover, the performances of LSTM models in ten experiments are depicted in Figure 10, measured in terms of SMAPE. As seen in Figure 10, of all the models, M4, M5, and M6 perform better than the others. The statistical details of the model performance in ten experiments are listed in Appendix Tables A.7, A.8, A.9, and A.10 by means of MAD, RMSE, MAPE, and SMAPE. Based on the analysis of feature selection, we conclude that M4 and M5, i.e., the minimum redundancy maximum relevance algorithms, perform better than the others. The results are consistent with the observations from other literature [74, 75] that an elimination of the redundant and less relevant features increases the performance of models. In addition, some researchers have attempted to introduce the CNN-LSTM model and show its excellent performance in the energy field [76, 77]. However, we found that M6 performs better than M7 and M8. This means LSTM-LSTM is a better autoencoder structure than CNN-LSTM and ConvLSTM for EPF. The results are reasonable because convolutional neural networks (CNNs or ConvNets) were originally designed for image recognition and classification, while recurrent neural networks (LSTM) are for sequence and time series prediction.

We show the comparison of the SMAPE of the two-step and the two-stage models in Figure 11. From this figure, it can be seen that M4 has been improved by applying LSTM-LSTM as predictors. We used a one-sided DM test to detect whether the two-stage LSTM models were statistically better than two-step LSTM-LSTM models. The results are shown in Table 6. The

**Table 4**

The results of the one-sided DM test.

F1 \ F2	M0	M1	M2	M3	M4	M5	M6	M7	M8
M0		5.83***	4.99***	3.90***	6.65***	7.09***	6.60***	4.61***	4.27***
M1	-5.83***		-1.15	-1.67*	2.31**	2.98***	1.59#	-0.16	0.88
M2	-4.99***	1.15		-0.91	2.70***	3.55***	3.84***	0.84	-0.20
M3	-3.90***	1.67*	0.91		3.19***	4.01***	4.31***	1.33	-0.97
M4	-6.65***	-2.31**	-2.70***	-3.19***		1.12	-0.23	-1.64#	2.57**
M5	-7.09***	-2.98***	-3.55***	-4.01***	-1.12		-0.97	-2.33**	3.19***
M6	-6.60***	-1.59#	-3.84***	-4.31***	0.23	0.97		-1.44	3.54***
M7	-4.61***	0.16	-0.84	-1.33	1.64#	2.33**	1.44		0.66
M8	-4.27***	-0.88	0.20	0.97	-2.57**	-3.19***	-3.54***	-0.66	

Note: \*\*\*, \*\*, \* and # denote 1%, 5%, 10%, and 15% significance levels, respectively. The positive sign of the DM value indicates that F2 is better F1. The negative sign of the DM value indicate that F1 is better F2.

**Table 5**

The SMAPE of M0, M1, M2, M3, M4, M5, M6, M7, and M8.

Model	M0	M1	M2	M3	M4	M5	M6	M7	M8
SMAPE	10.07	6.25	6.58	7.06	5.29	4.89	5.20	6.14	6.53

**Table 6**

The results of the one-sided DM test when comparing two-step models (F1) and two-stage models (F2).

F1	M1	M2	M3	M4	M5
	-0.6039	0.2222	2.4556 ***	2.4524 ***	-1.7053 ***

F2	M9	M10	M11	M12	M13

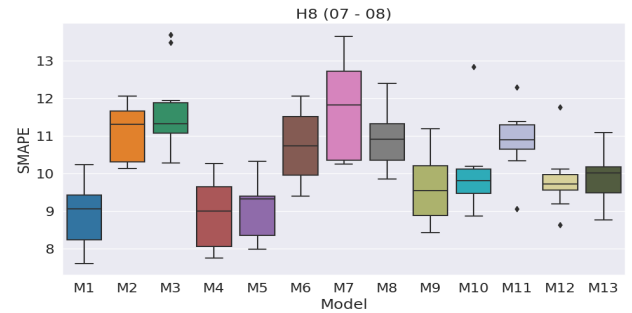
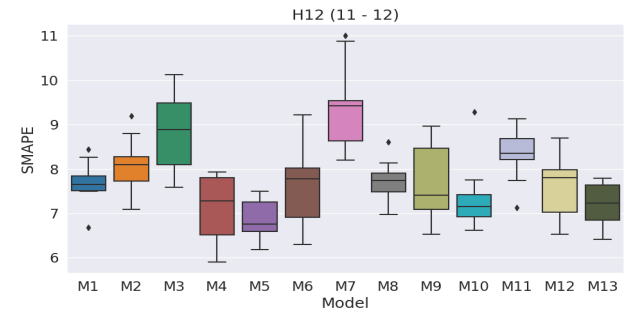
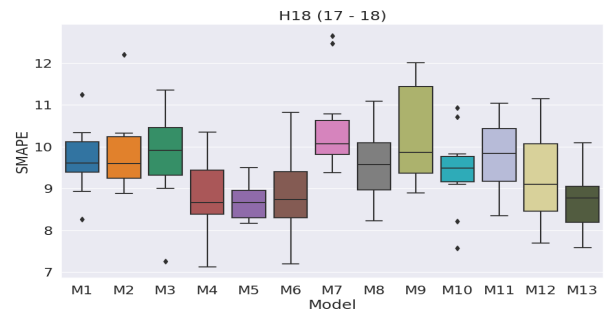
Note: \*\*\*, \*\*, \* and # denote 1%, 5%, 10%, and 15% significance levels, respectively. The positive sign: F2 is better F1. The negative sign: F1 is better F2.

superior features selection from M4 provides the possibility for autoencoder models to further process the selected features to obtain more meaningful information.

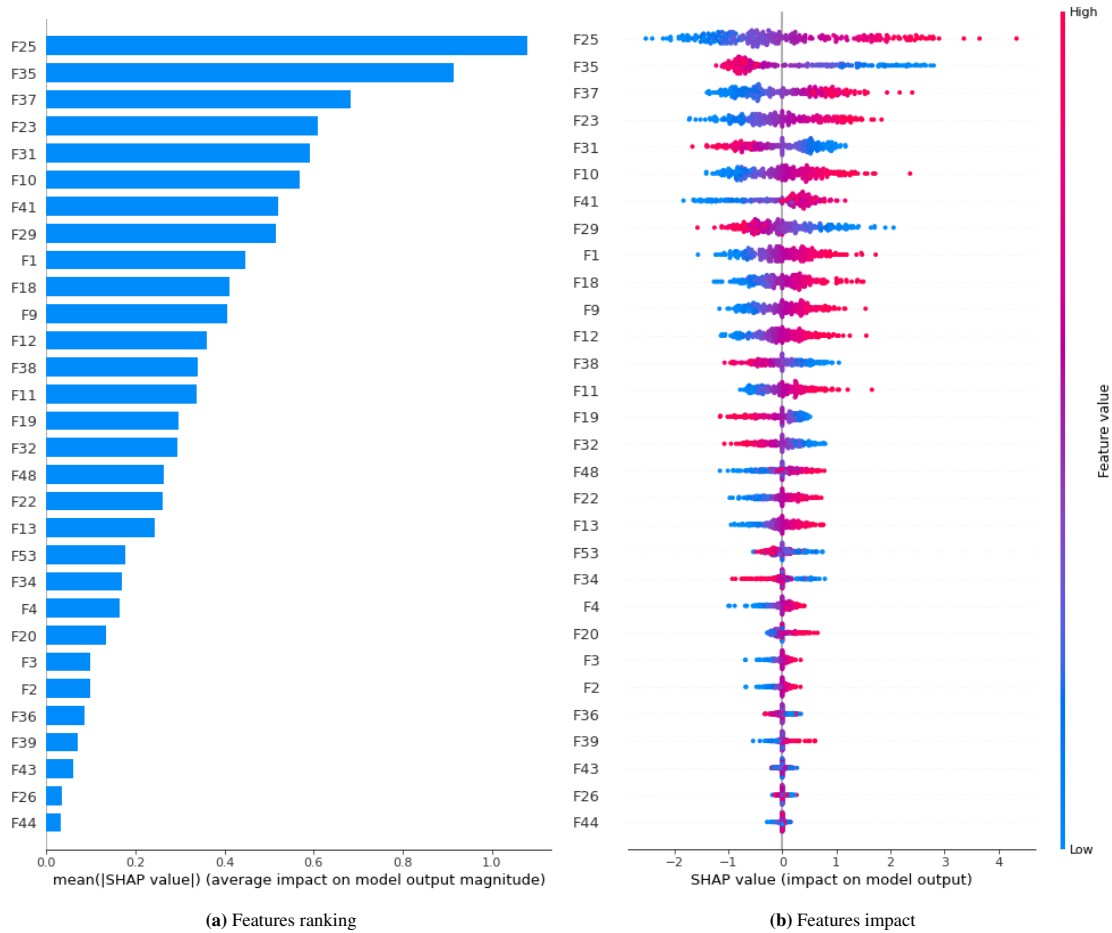
Additionally, we detected the forecasting performance for 24 hourly system prices. Figures 12, 13, and 14 show the results for the three peak hours: H8 (07 - 08), H12 (11 - 12) and H18 (17 - 18), respectively, measured in terms of SMAPE. We observed that the feature selections influence the forecasting accuracy and the models M4 and M5 are still relatively stable, performing better than other models. Indeed, the RFE-SVR and Lasso regression feature selection methods are applied to various areas in energy finance and achieve good performance for improving forecasting accuracy [78, 79, 80, 81].

## 5.2. Analysis of feature impact

Figure 15 shows the ranking of features and their impacts on the predicted price in terms of the selected features of the RFE-SVR model (M4), which is the model with the best performance. From Figure 15a, we can observe that the features from supply/demand sides are more important than the other

**Figure 12** The SMAPEs of 10 experiments for predicting H8.**Figure 13** The SMAPEs of 10 experiments for predicting H12.**Figure 14** The SMAPEs of 10 experiments for predicting H18.

features. In particular, production and consumption and their prognosis in the Nordic and German markets are prioritised



**Figure 15** The feature ranking and feature impact of the selected features of RFE-SVR. (a) Bar chart of the average SHAP value magnitude showing the importance of the features. (b) A set of beeswarm plots, where each dot corresponds to an individual day-ahead price. The dot's position on the x-axis shows the impact that feature has on the model's prediction for that price. Multiple dots landing at the same x position pile up to show density.

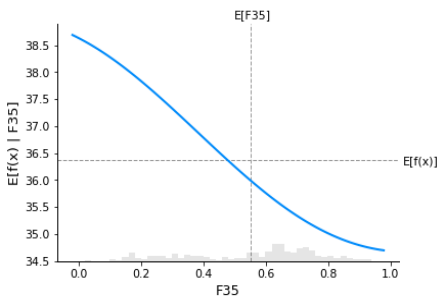
by the model. The significant impact from the German market can be explained by the fact that the German market has the most electricity cables and the highest electricity exports to the Nordic market, as shown in Figures 1 and 2. This indicates that it is critical to consider features from cross-border markets with increasing interconnections across Europe for EPF. Besides, electricity prices have more impact on EPF than the features from cross-border electricity trade. From Figure 15b, we can see that the features have asymmetric predictive influence on electricity price. For instance, the impact of DE consumption (F35) on EPF has a long-tail reaching to the right but not to the left. This indicates that German over-consumption can result in high Nordic electricity prices, but scarce consumption cannot significantly lower the price.

Moreover, we detected the relations between different types of features and the predicted price with their dependence plots. Figure 16a demonstrates the negative association between DE consumption and its conditional expectation of the predicted price. If the DE consumption is high, then its value tends to revert to its expectation. Thus, the downward expectation of DE consumption will lead to the expected decline of the import demand from the Nordic market, which further decreases the expectation of the Nordic price. Figure 16b represents the

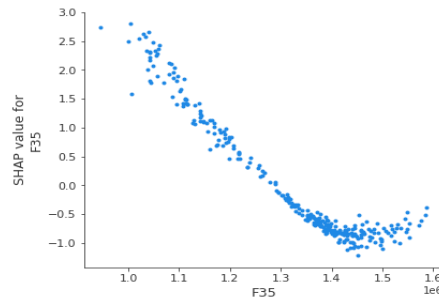
change in predicted price as DE consumption changes. Vertical dispersion at a single value of DE consumption represents the interaction effects with other features. For example, the interaction effect of DE consumption with the Nordic production (F19) is shown in Figure 16c. The dependence plot highlights that the impact of DE consumption differs with different levels of the Nordic production. The results reveal that the Nordic price is less sensitive to the German power consumption when the Nordic electricity is oversupplied. It indicates, in such a case, that the information from the Nordic market rather than cross-border countries drives the price prediction.

From Figure 17, we can see that the majority of the EUR/NOK exchange rates (F43) have no contribution to the prediction of the Nordic price (the y-axis value of the dots is zero). In addition, there is no obvious interaction effect of the Nordic productions and the exchange rates on the price. Thus, the predictive importance of exchange rate is extremely limited.

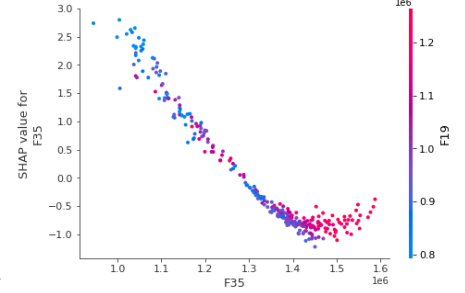
From Figure 18, we can find that the predicted price is expected to increase when observing a high DK1→DE cross-border electricity flow, indicating a relatively low current Nordic price. By contrast, a high flow from DE to DK1 implies that the Nordic price is relatively high and expected to decline. In addition, it can be seen via an interaction effect of the DK1



(a) SHAP partial dependence plot of DE consumption

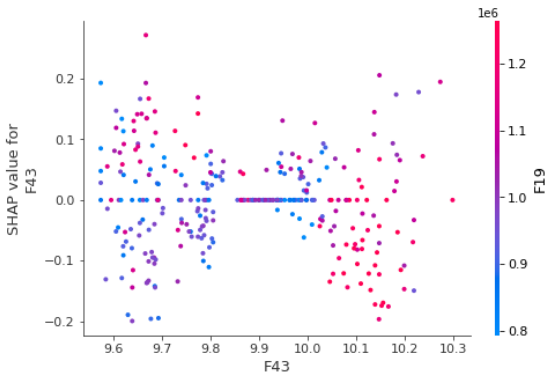


(b) SHAP dependence plot of DE consumption



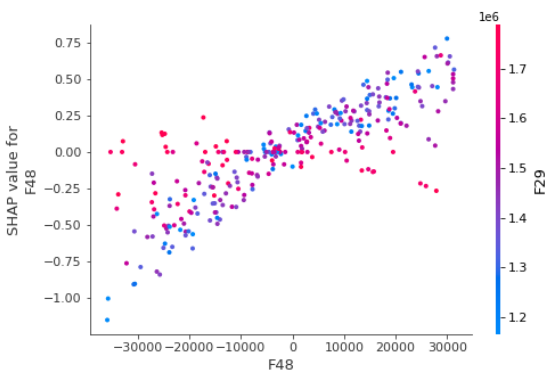
(c) SHAP dependence plot of the interaction effect of DE consumption with the Nordic production (F19)

**Figure 16** The SHAP partial dependence (a) and dependence (b, c) plot of DE consumption (F35).  $\mathbb{E}[F35]$  is the expectation of the DE consumption, and  $\mathbb{E}[f(x)]$  is the expectation of the Nordic price. The grey histogram in (a) shows the distribution of the feature in the test dataset. For (a), the x-axis is the normalised DE consumption. The x-axes in (b) and (c) are the real values of the consumption.



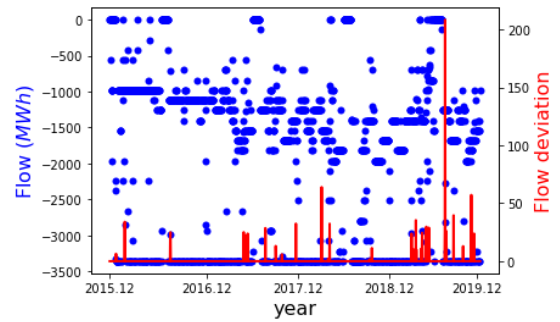
**Figure 17** The SHAP dependence plot of EUR/NOK (F43). The interaction effect of EUR/NOK with the Nordic production (F19).

↔ DE flow with the DE production prognosis that the flow has less impact on the predicted price, with high expected production in Germany. The high production prognosis from cross-border countries will lead to a sharp decline in the expected cross-border transmission. Thus, the impact of the cross-border flow on the Nordic price formation on the following day will decrease significantly.



**Figure 18** The SHAP dependence plot of DK1 ↔ DE flow (F48). The interaction effect of DK1 ↔ DE flow with DE production prognosis (F29).

Last but not least, not all of the cross-border electricity flows and flow deviations are helpful for forecasting. The reason for this is that, in many cases, the flow capacity is fully occupied.



**Figure 19** FI ↔ Russia flow (F54) (blue dots) versus FI ↔ Russia flow deviation (F62) (red line).

The lack of variability results in their inability to provide useful information for forecasting. An example of the flow and flow deviations between FI and Russia can be seen in Figure 19. From Figure 19, it is evident that the majority of flow deviations are zero. The findings indicate that the non-selection of flow deviations from M4 is essential and reasonable. However, the capacity utilisation indicates the potential for more electrical power transmission across the Europe-wide market which would increase the overall socio-economic benefits.

### 5.3. Discussion of practical importance

The obtained results in the empirical study show that the LSTM-based hybrid models with various features from cross-border markets have considerably accurate prediction results for electricity price. An accurate prediction can be highly beneficial for the electricity market participants in practice. A power market firm that is capable of forecasting the volatile electricity price with a reasonable level of accuracy can reduce trading risk and maximise profits in the day-ahead market by adjusting its bidding strategy and the schedule for production or consumption. More specifically, a 1% improvement in MAPE of forecast accuracy (within a 5% to 14% range) leads to about a 0.1 - 0.35% cost reduction [82]. On average, a 1% reduction in the MAPE of short-term price forecasts can result in savings of \$1.5 million per year for a typical medium-sized utility company with 5-GW peak load [83, 29]. Furthermore, electricity is economically non-storable, and the imbalance between production and consumption can result in power system instability

[84]. Accurate electricity forecasting allows energy firms to efficiently organise production or consumption, and this improves the stability of the power system.

In view of the findings from the analysis of feature impact, some implications are important for policy makers to improve cross-border trading in an integrated European power market:

1. The external trading capacities from the German market play a salient role in the generation of Nordic electricity price, and an increasing influence<sup>7</sup> is expected. Thus, all trading capacity between the Nordic and German markets allocated to Nord Pool for implicit auction in the day-ahead price formation could lead to a notable contribution to achieve better allocation of cross-border network capacity, such as the Nordic and Baltic bidding areas.
2. The German production prognosis has a significant predictive impact on the Nordic price. However, the progressive introduction of intermittent renewable energies in Germany<sup>8</sup> makes it difficult to yield accurate predictions. Thus, it is essential to establish formal obligations for cross-border markets to collaborate by sharing useful prospective information. It can better serve the effective demand needed of the electricity trade.
3. The presence of physical transmission constraints can cause a substantial disruption to the market integration and the network congestion implies the shrinkage of the commercial capacity. Thus, an optimal network at a European level should be constructed by a decisive plan.

## 6. Conclusion

In this paper, we present three LSTM-based hybrid architectures for the EPF. This study puts emphasis on the influence of feature selection methods in the proposed hybrid models. In particular, we compare the prediction performance of the two-step feature selection, the autoencoder, and two-stage feature selection models based on the empirical study on the Nord Pool day-ahead system price. In addition, we employ a SHAP method to evaluate the importance and impact of the features on predicting this price. The main findings are the following: (1) We conclude that the different feature selection methods will lead to different feature selections. As input, diverse features will have a comparably significant impact on the performance of LSTM-based predictive models. (2) Compared to CNN-LSTM and ConvLSTM, LSTM-LSTM is a better autoencoder structure for EPF. (3) The two-stage models can improve the forecasting accuracy of two-step models to some extent. The superior feature selection from the RFE-SVR model

allows the autoencoder model to detect more meaningful information for more accurate predictions. (4) The features from the German market (with the most power cables linking to Nord Pool) are more significant for EPF than others. This indicates that more interconnections will increase the cross-border influence on EPF. (5) Compared to other features, the exchange rates are relatively less important. (6) Flow deviation cannot significantly contribute to the price prediction because of its lack of variability. In many cases, the expected flow capacity is fully occupied. The network congestion implies that more interconnections are expected for an efficient Europe-wide electricity market.

For future studies, several extensions of the current study can be developed. Indeed, although the forecasting performance of the proposed models is considerable, we did not conduct an extensive grid search to optimise hyperparameters. It is reasonable to believe that the LSTM-based models with more comprehensive architecture will achieve better forecasting performance. The results will benefit spot electricity traders and policymakers, who make decisions based on accurate price predictions. Moreover, we envision that more testing on other feature selection models can obtain more and different feature selection subsets. They can provide more possibilities for researchers and industries to understand how different features affect prediction accuracy. Finally, the study was carried out using the data from the Nord Pool market, but the generality of the proposed models ensures a possible application to other integrated markets, such as EPEX and OMIE.

## Acknowledgement

This work acknowledges research support by COST Action “Fintech and Artificial Intelligence in Finance - Towards a transparent financial industry” (FinAI) CA19130, and has been performed within the +CityxChange<sup>9</sup> (Positive City ExChange) project under the Smart Cities and Communities topic that has received funding from the European Union’s Horizon 2020 research and innovation programme under Grant Agreement No. 824260. Critical comments and advice from Florentina Paraschiv, Rüdiger Kiesel, and Frode Kjørland are gratefully acknowledged. The computations were performed on resources provided by UNINETT Sigma2 - the National Infrastructure for High Performance Computing and Data Storage in Norway.

<sup>7</sup>The NordLink, the power cable being built between Norway and Germany, is expected to commence operation in 2021.

<sup>8</sup>Renewable power generation covered more than 46% of Germany’s power consumption in 2020. The forecasting information of intermittent renewable energies, wind and photovoltaic infeed forecasts, in Germany are essential supply-side variables for the adjustment positions of intraday trading and are updated every 15 minutes until the physical delivery of electricity [85].

<sup>9</sup><https://cityxchange.eu/>

## Appendix A.

### Appendix A.1. PSO

Each particle has knowledge about its current velocity, its own past best solution ( $\vec{p}(t)$ ), and the current global best solution ( $\vec{g}(t)$ ). Based on this information, each particle's velocity is updated such that it moves closer to the global best and its past best solution at the same time. The velocity update is performed according to the following equation:

$$\begin{aligned} \vec{v}(t+1) = & \omega \vec{v}(t) + c_1 r_1 (\vec{p}(t) - \vec{x}(t)) \\ & + c_2 r_2 (\vec{g}(t) - \vec{x}(t)) \end{aligned} \quad (\text{A.1})$$

where  $c_1$  and  $c_2$  are constants defined beforehand, which determine the significance of  $\vec{p}(t)$  and  $\vec{g}(t)$ .  $\vec{v}(t)$  is the velocity of the particle,  $\vec{x}(t)$  is the current particle position,  $r_1$  and  $r_2$  are random numbers from the interval  $[0,1]$ , and  $\omega$  is a constant ( $0 \leq \omega \leq 1$ ). The new position is calculated by summing the previous position and the new velocity as follows:

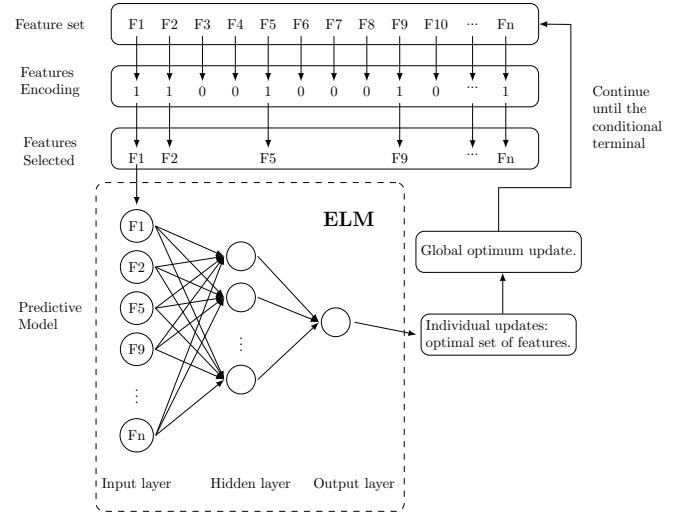
$$\vec{x}(t+1) = \vec{x}(t) + \vec{v}(t+1) \quad (\text{A.2})$$

This iterative process is repeated until a stopping criterion is satisfied.

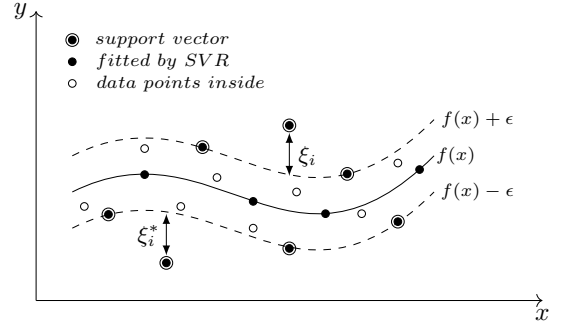
### Appendix A.2. GA-ELM workflow

Figure A.20 shows the workflow of PSO-ELM and GA-ELM models for feature selection. The process flow of GA-ELM can be described as follows:

- Step 1:** Initialise the population with a set of random individuals, each individual representing a particular subset of features. For a specific individual (feature set), the features are encoded as "1" or "0", as shown in Figure A.20. "1" means that the feature is selected, and "0" means that it is not selected.
- Step 2:** The selected features are the input for the ELM. The prediction results of the ELM are used to evaluate the fitness value of the individuals. The fitness value is calculated based on the MSE.
- Step 3:** Select the best individual with regard to the fitness value. If its fitness is higher than the lowest value in the existing mating pool, it will replace the individual with the worst fitness. Furthermore, the global optimum will be updated accordingly.
- Step 4:** The child individuals are generated by crossover and mutation. The new generation is composed of a set of new individuals that are encoded and prepared to be evaluated. The whole process continues until meeting the iteration terminal. The best feature subset in the mating pool is the optimal selection.



**Figure A.20** The workflow of the two-step wrapper-based feature selection model.



**Figure A.21** Fitted SVR.

### Appendix A.3. SVR

To minimise the forecasting errors, SVR individualises the hyperplane by maximising the margin. To solve a nonlinear regression problem, the following linear estimation function is considered as follows [86]:

$$f(x) = (\mathbf{w} \times \Phi(x)) + b \quad (\text{A.3})$$

where  $\mathbf{w}$  is the parameter vector,  $\Phi(x)$  is a kernel function and  $b$  is a bias vector. The function formulation of the SVR model can be transformed into the following convex minimisation problem:

$$\min \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \quad (\text{A.4})$$

subject to the following constraints:

$$\begin{aligned} (\mathbf{w} \times \Phi(x_i) + b) - y_i & \leq \varepsilon + \xi_i \\ y_i - (\mathbf{w} \times \Phi(x_i) + b) & \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* & \geq 0; i = 1, 2, \dots, n \end{aligned}$$

where  $C$  is a regularisation constant and  $\xi_i$  and  $\xi_i^*$  are slack variables, which are used to handle the situation where no such

function  $f(x)$  exists to satisfy the constraint  $|y_i - (\mathbf{w} \times x_i + b)| \leq \varepsilon$  for all points. They are regarded as the soft margin to allow regression errors,  $\varepsilon$ , to exist up to  $\xi_i$  and  $\xi_i^*$  and still satisfy the constraint. Only the points outside the  $\varepsilon$ -radius contribute to the final cost. The error parameter,  $\varepsilon$ , represents the region of the tube located around the regression function,  $f(x)$ , as shown in Figure A.21.

#### Appendix A.4. The statistical details of the model performance

**Table A.7**

The MAD (%) results for M1, M2, M3, M4, M5, M6, M7 and M8.

	M1	M2	M3	M4	M5	M6	M7	M8
count	10	10	10	10	10	10	10	10
mean	2.78	2.99	3.31	2.54	2.61	2.67	3.67	3.28
std	0.17	0.29	0.40	0.25	0.30	0.25	0.15	0.38
min	2.42	2.58	2.64	2.16	2.04	2.20	3.30	2.84
25%	2.72	2.77	3.10	2.34	2.45	2.59	3.65	2.99
50%	2.77	2.95	3.25	2.59	2.65	2.65	3.69	3.18
75%	2.92	3.24	3.56	2.63	2.74	2.85	3.73	3.60
max	2.98	3.41	3.93	2.99	3.00	3.09	3.89	3.95

Note: 25%, 50%, and 75% denote 25%, 50%, and 75% percentiles.

**Table A.8**

The RMSE (%) results for M1, M2, M3, M4, M5, M6, M7 and M8.

	M1	M2	M3	M4	M5	M6	M7	M8
count	10	10	10	10	10	10	10	10
mean	3.60	3.79	4.22	3.25	3.46	3.33	4.74	3.99
std	0.24	0.42	0.55	0.33	0.43	0.32	0.29	0.46
min	3.17	3.15	3.35	2.61	2.79	2.85	4.13	3.50
25%	3.51	3.45	3.87	3.04	3.27	3.13	4.62	3.60
50%	3.64	3.78	4.16	3.30	3.50	3.30	4.90	3.92
75%	3.71	4.05	4.55	3.45	3.71	3.47	4.91	4.33
max	3.96	4.51	5.11	3.73	4.08	3.91	4.97	4.77

Note: 25%, 50%, and 75% denote 25%, 50%, and 75% percentiles.

**Table A.9**

The MAPE (%) results for M1, M2, M3, M4, M5, M6, M7 and M8.

	M1	M2	M3	M4	M5	M6	M7	M8
count	10	10	10	10	10	10	10	10
mean	7.24	7.83	8.75	6.66	6.79	7.01	9.31	8.73
std	0.46	0.73	1.05	0.65	0.78	0.68	0.47	1.06
min	6.28	6.82	6.89	5.72	5.32	5.67	8.56	7.50
25%	7.05	7.22	8.19	6.18	6.35	6.84	9.08	7.90
50%	7.24	7.71	8.62	6.73	6.85	6.95	9.16	8.44
75%	7.61	8.49	9.35	6.87	7.15	7.45	9.50	9.59
max	7.80	8.93	10.37	7.86	7.91	8.09	10.27	10.66

Note: 25%, 50%, and 75% denote 25%, 50%, and 75% percentiles.

**Table A.10**

The SMAPE (%) results for M1, M2, M3, M4, M5, M6, M7 and M8.

	M1	M2	M3	M4	M5	M6	M7	M8
count	10	10	10	10	10	10	10	10
mean	7.42	7.58	8.46	6.76	6.85	6.87	9.51	8.38
std	0.50	0.67	0.95	0.65	0.72	0.60	0.44	0.90
min	6.49	6.63	6.77	5.76	5.46	5.71	8.49	7.38
25%	7.22	7.10	7.95	6.33	6.43	6.77	9.38	7.59
50%	7.34	7.48	8.30	6.87	6.91	6.85	9.67	8.22
75%	7.81	8.10	9.05	7.01	7.37	7.27	9.77	9.13
max	8.13	8.70	10.09	7.87	7.74	7.75	9.95	9.93

Note: 25%, 50%, and 75% denote 25%, 50%, and 75% percentiles.

## References

- [1] Weron R. Modeling and forecasting electricity loads and prices: A statistical approach. Wiley; 2006.
- [2] Bunn D. Modelling prices in competitive electricity markets. Wiley; 2004.
- [3] Nogales FJ, Contreras J, Conejo AJ, Espinola R. Forecasting next-day electricity prices by time series models. IEEE Transactions on Power Systems 2002;17(2):342–348. <https://doi.org/10.1109/TPWRS.2002.1007902>.
- [4] Bunn DW. Forecasting loads and prices in competitive power markets. Proceedings of the IEEE 2000;88(2):163–169. <https://doi.org/10.1109/5.823996>.
- [5] Weron R. Electricity price forecasting: A review of the state-of-the-art with a look into the future. International Journal of Forecasting 2014;30(4):1030 – 1081. <https://doi.org/10.1016/j.ijforecast.2014.08.008>.
- [6] Nowotarski J, Weron R. Recent advances in electricity price forecasting: A review of probabilistic forecasting. Renewable and Sustainable Energy Reviews 2018;81:1548 – 1568. <https://doi.org/10.1016/j.rser.2017.05.234>.
- [7] Ventosa M, Baillo A, Ramos A, Rivie M. Electricity market modeling trends. Energy Policy 2005;33(7):897 – 913. <https://doi.org/10.1016/j.enpol.2003.10.013>.
- [8] Kiose D, Voudouris V. The acemem framework: An integrated agent-based and statistical modelling laboratory for repeated power auctions. Expert Systems with Applications 2015;42(5):2731 – 2748. <https://doi.org/10.1016/j.eswa.2014.11.024>.
- [9] Burger M, Schindlmayr G, Graeber B. Managing Energy Risk: An Integrated View on Power and Other Energy Markets. Wiley; 2007.
- [10] Islyayev S, Date P. Electricity futures price models: Calibration and forecasting. European Journal of Operational Research 2015;247(1):144 – 154. <https://doi.org/10.1016/j.ejor.2015.05.063>.
- [11] Weron R, Misiorek A. Forecasting spot electricity prices: A comparison of parametric and semiparametric time series models. International Journal of Forecasting 2008;24(4):744 – 763. <https://doi.org/10.1016/j.ijforecast.2008.08.004>.
- [12] Conejo AJ, Contreras J, Espinola R, Plazas MA. Forecasting electricity prices for a day-ahead pool-based electric energy market. International Journal of Forecasting 2005;21(3):435 – 462. <https://doi.org/10.1016/j.ijforecast.2004.12.005>.
- [13] Misiorek A, and Trueck S, Weron R. Point and interval forecasting of spot electricity prices: Linear vs non-linear time series models. Studies in Nonlinear Dynamics & Econometrics 2006;10(3). <https://10.2202/1558-3708.1362>.
- [14] Gonzalez JP, Roque AMSMS, Pérez EA. Forecasting functional time series with a new Hilbertian ARMAX model: Application to electricity price forecasting. IEEE Transactions on Power Systems 2018;33(1):545–556. <https://doi.org/10.1109/TPWRS.2017.2700287>.
- [15] Catalao J, Mariano S, Mendes V, Ferreira L. Short-term electricity prices forecasting in a competitive market: A neural network approach. Electric



- Power Systems Research 2007;77(10):1297 – 1304. <https://doi.org/10.1016/j.epsr.2006.09.022>.
- [16] Keles D, Scelle J, Paraschiv F, Fichtner W. Extended forecast methods for day-ahead electricity spot prices applying artificial neural networks. *Applied Energy* 2016;162:218 – 230. <https://doi.org/10.1016/j.apenergy.2015.09.087>.
- [17] Peter S, Raglend I. Sequential wavelet-ANN with embedded ANN-PSO hybrid electricity price forecasting model for indian energy exchange. *Neural Comput & Applic* 2017;28:2277–2292. <https://doi.org/10.1007/s00521-015-2141-3>.
- [18] Hinton G, Deng L, Yu D, Dahl GE, rahman Mohamed A, Jaitly N, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine* 2012;29:82 – 97. <https://doi.org/10.1109/MSP.2012.2205597>.
- [19] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. arXiv:1409.0473 [cs.CL] 2014; Available from: <http://arxiv.org/abs/1409.0473>.
- [20] Li L, Yuan Z, Gao Y. Maximization of energy absorption for a wave energy converter using the deep machine learning. *Energy* 2018;165:340 – 349. <https://doi.org/10.1016/j.energy.2018.09.093>.
- [21] Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks* 1994;5(2):157–166. <https://doi.org/10.1109/72.279181>.
- [22] Lago J, Ridder FD, Schutter BD. Forecasting spot electricity prices: Deep learning approaches and empirical comparison of traditional algorithms. *Applied Energy* 2018;221:386 – 405. <https://doi.org/10.1016/j.apenergy.2018.02.069>.
- [23] Chang Z, Zhang Y, Chen W. Electricity price prediction based on hybrid model of adam optimized LSTM neural network and wavelet transform. *Energy* 2019;187:115804. <https://doi.org/10.1016/j.energy.2019.07.134>.
- [24] Kuo PH, Huang CJ. An electricity price forecasting model by hybrid structured deep neural networks. *Sustainability* 2018;10(4). <https://doi.org/10.3390/su10041280>.
- [25] Hastie T, Tibshirani R, Friedman J. *The elements of statistical learning: data mining, inference and prediction*. 2 ed.; Springer; 2009.
- [26] Li J, Cheng K, Wang S, Morstatter F, Trevino RP, Tang J, et al. Feature selection: A data perspective. *ACM Computing Surveys* 2017;50(6). 10.1145/3136625.
- [27] Ziel F, Steinert R, Husmann S. Forecasting day ahead electricity spot prices: The impact of the exaa to other european electricity markets. *Energy Economics* 2015;51:430 – 444. <https://doi.org/10.1016/j.eneco.2015.08.005>.
- [28] Panapakidis IP, Dagoumas AS. Day-ahead electricity price forecasting via the application of artificial neural network based models. *Applied Energy* 2016;172:132 – 151. <https://doi.org/10.1016/j.apenergy.2016.03.089>.
- [29] Lago J, De Ridder F, Vrancx P, De Schutter B. Forecasting day-ahead electricity prices in europe: The importance of considering market integration. *Applied Energy* 2018;211:890 – 903. <https://doi.org/10.1016/j.apenergy.2017.11.098>.
- [30] Uribe JM, Mosquera-López S, Guillen M. Characterizing electricity market integration in nord pool. *Energy* 2020;208:118368. <https://doi.org/10.1016/j.energy.2020.118368>.
- [31] Marcjasz G, Lago J, Weron R. Neural networks in day-ahead electricity price forecasting: Single vs. multiple outputs. arXiv:2008.08006 [stat.AP] 2020; Available from: <https://arxiv.org/abs/2008.08006>.
- [32] Johannesen NJ, Kolhe M, Goodwin M. Deregulated electric energy price forecasting in nordpool market using regression techniques. In: 2019 IEEE Sustainable Power and Energy Conference (iSPEC). 2019, p. 1932–1938. <https://ieeexplore.ieee.org/abstract/document/8975173>.
- [33] Chandrashekar G, Sahin F. A survey on feature selection methods. *Computers & Electrical Engineering* 2014;40(1):16 – 28. <https://doi.org/10.1016/j.compeleceng.2013.11.024>; 40th-year commemorative issue.
- [34] Sanz H, Valim C, Vegas E, Oller JM, Reverter F. Svm-rfe: selection and visualization of the most relevant features through non-linear kernels. *BMC Bioinformatics* 2018;19(1):432. 10.1186/s12859-018-2451-4.
- [35] Guyon I, Elisseeff A. An introduction to variable and feature selection. *The Journal of Machine Learning Research* 2003;3(null):1157–1182.
- [36] Chen X, Zeng X, van Alphen D. Multi-class feature selection for texture classification. *Pattern Recognition Letters* 2006;27(14):1685 – 1691. <https://doi.org/10.1016/j.patrec.2006.03.013>.
- [37] Nguyen HB, Xue B, Liu I, Andreae P, Zhang M. New mechanism for archive maintenance in PSO-based multi-objective feature selection. *Soft Computing* 2016;20:3927 – 3946. <https://doi.org/10.1007/s00500-016-2128-8>.
- [38] Shang L, Zhou Z, Liu X. Particle swarm optimization-based feature selection in sentiment classification. *Soft Computing* 2016;20:3821 – 3834. <https://doi.org/10.1007/s00500-016-2128-8>.
- [39] Zhou Y, Zhou N, Gong L, Jiang M. Prediction of photovoltaic power output based on similar day analysis, genetic algorithm and extreme learning machine. *Energy* 2020;204:117894. <https://doi.org/10.1016/j.energy.2020.117894>.
- [40] S. Krishnan G, S. S. A novel GA-ELM model for patient-specific mortality prediction over large-scale lab event data. *Applied Soft Computing Journal* 2019;80:525–533. 10.1016/j.asoc.2019.04.019.
- [41] Luo P, Zhu S, Han L, Chen Q. Short-term photovoltaic generation forecasting based on similar day selection and extreme learning machine. *IEEE Power and Energy Society General Meeting* 2018;2018-January:1–5. 10.1109/PESGM.2017.8273776.
- [42] Guyon I, Weston J, Barnhill S, Vapnik V. Gene selection for cancer classification using support vector machines. *Machine Learning* 2002;46(1):389–422. 10.1023/A:1012487302797.
- [43] Tibshirani R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* 1996;58(1):267–288. <https://doi.org/10.1111/j.2517-6161.1996.tb02080.x>.
- [44] Greenfish. Shaping our electrical future: Moving towards an integrated european network. <https://www.greenfish.eu/shaping-our-electrical-future-moving-towards-an-integrated-european-network/>; 2019. [accessed 13 September 2020].
- [45] Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Computation* 1997;74(8):1735 – 1780. <https://doi.org/10.1162/neco.1997.9.8.1735>.
- [46] Gers FA, Schmidhuber J. Recurrent nets that time and count. In: *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*; vol. 3. 2000, p. 189–194. <https://doi.org/10.1109/IJCNN.2000.861302>.
- [47] Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks* 2005;18(5):602 – 610. <https://doi.org/10.1016/j.neunet.2005.06.042>; iJCNN 2005.
- [48] Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv:1406.1078 [cs.CL] 2014; Available from: <https://arxiv.org/abs/1406.1078>.
- [49] Graves A. Generating sequences with recurrent neural networks. arXiv:1308.0850 [cs.NE] 2013; Available from: <https://arxiv.org/abs/1308.0850>.
- [50] Huang GB, Zhu QY, Siew CK. Extreme learning machine: Theory and applications. *Neurocomputing* 2006;70(1):489 – 501. <https://doi.org/10.1016/j.neucom.2005.12.126>.
- [51] Saraswathi S, Sundaram S, Sundararajan N, Zimmermann M, Nilsen-Hamilton M. ICGA-PSO-ELM approach for accurate multiclass cancer classification resulting in reduced gene sets in which genes encoding secreted proteins are highly represented. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 2011;8(2):452–463. <https://doi.org/10.1109/TCBB.2010.13>.
- [52] Chyzyk D, Savio A, Graña M. Evolutionary ELM wrapper feature selection for Alzheimer’s disease CAD on anatomical brain MRI. *Neurocomputing* 2014;128:73 – 80. <https://doi.org/10.1016/j.neucom.2013.01.065>.
- [53] Ahila R, Sadasivam V, Manimala K. An integrated PSO for parameter determination and feature selection of ELM and its application in classification of power system disturbances. *Applied Soft Computing* 2015;32:23 – 37. <https://doi.org/10.1016/j.asoc.2015.03.036>.
- [54] Zhang Y, Wang S, Ji G. A comprehensive survey on particle swarm optimization algorithm and its applications. *Mathematical Problems in Engineering* 2015;2015:931256. 10.1155/2015/931256.

- [55] Whitley D. A genetic algorithm tutorial. *Statistics and Computing* 1994;4(2):65–85. [10.1007/BF00175354](https://doi.org/10.1007/BF00175354).
- [56] Drucker H, Burges CJC, Kaufman L, Smola A, Vapnik V. Support vector regression machines. In: Mozer MC, Jordan M, Petsche T, editors. *Advances in Neural Information Processing Systems*; vol. 9. MIT Press; 1997. <https://proceedings.neurips.cc/paper/1996/file/d38901788c533e8286cb6400b40b386d-Paper.pdf>.
- [57] Szegedy C, Wei Liu, Yangqing Jia, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2015, p. 1–9. <https://doi.org/10.1109/CVPR.2015.7298594>.
- [58] Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. In: Ghahramani Z, Welling M, Cortes C, Lawrence ND, Weinberger KQ, editors. *Advances in Neural Information Processing Systems 27*. Curran Associates, Inc.; 2014, p. 3104–3112. <https://arxiv.org/abs/1409.3215>.
- [59] Shi X, Chen Z, Wang H, Yeung DY, Wong Wk, Woo Wc. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In: Cortes C, Lawrence ND, Lee DD, Sugiyama M, Garnett R, editors. *Advances in Neural Information Processing Systems 28*. Curran Associates, Inc.; 2015, p. 802–810. <https://arxiv.org/pdf/1506.04214.pdf>.
- [60] Diebold FX, Mariano RS. Comparing predictive accuracy. *Journal of Business & Economic Statistics* 2002;20(1):134–144. <https://doi.org/10.1198/073500102753410444>.
- [61] Harvey D, Leybourne S, Newbold P. Testing the equality of prediction mean squared errors. *International Journal of Forecasting* 1997;13(2):281–291. [https://doi.org/10.1016/S0169-2070\(96\)00719-4](https://doi.org/10.1016/S0169-2070(96)00719-4).
- [62] Varma S, Simon R. Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics* 2006;7:91. <https://doi.org/10.1186/1471-2105-7-91>.
- [63] Goodfellow I, Bengio Y, Courville A. *Deep Learning*. The MIT Press; 2016.
- [64] McHugh C, Coleman S, Kerr D, McGlynn D. Daily energy price forecasting using a polynomial narxax model. In: Lotfi A, Bouchachia H, Gegov A, Langensiepen C, McGinnity M, editors. *Advances in Computational Intelligence Systems*. Cham: Springer International Publishing. ISBN 978-3-319-97982-3; 2019, p. 71–82.
- [65] Lundberg SM, Erion G, Chen H, DeGrave A, Prutkin JM, Nair B, et al. From local explanations to global understanding with explainable ai for trees. *Nature Machine Intelligence* 2020;2(1):56–67. [10.1038/s42256-019-0138-9](https://doi.org/10.1038/s42256-019-0138-9).
- [66] Janzing D, Minorics L, Blöbaum P. Feature relevance quantification in explainable ai: A causal problem. 2019. [arXiv:1910.13413](https://arxiv.org/abs/1910.13413); available from: <https://arxiv.org/abs/1910.13413>.
- [67] Sundararajan M, Najmi A. The many shapley values for model explanation. 2020. [arXiv:1908.08474](https://arxiv.org/abs/1908.08474); available from: <https://arxiv.org/abs/1908.08474>.
- [68] Lundberg S, Lee SI. A unified approach to interpreting model predictions. 2017. [arXiv:1705.07874](https://arxiv.org/abs/1705.07874); available from: <https://arxiv.org/abs/1705.07874>.
- [69] Hanchuan Peng, Fuhui Long, Ding C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2005;27(8):1226–1238. [10.1109/TPAMI.2005.159](https://doi.org/10.1109/TPAMI.2005.159).
- [70] Yu L, Liu H. Efficient feature selection via analysis of relevance and redundancy. *J. Mach. Learn. Res.* 2004;5:1205–1224.
- [71] Langley P. Selection of relevant features in machine learning. In: *Proceedings of the AAAI Fall symposium on relevance*. AAAI Press; 1994, p. 140–144.
- [72] Jamian JJ, Abdullah MN, Mokhlis H, Mustafa MW, Bakar AHA. Global particle swarm optimization for high dimension numerical functions analysis. *Journal of Applied Mathematics* 2014; <https://doi.org/10.1155/2014/329193>.
- [73] Somu N, Raman MR G, Ramamritham K. A deep learning framework for building energy consumption forecast. *Renewable and Sustainable Energy Reviews* 2021;137:110591. <https://www.sciencedirect.com/science/article/pii/S1364032120308753>. <https://doi.org/10.1016/j.rser.2020.110591>.
- [74] Shao Z, Yang S, Gao F, Zhou K, Lin P. A new electricity price prediction strategy using mutual information-based svm-rfe classification. *Renewable and Sustainable Energy Reviews* 2017;70:330–341. <https://www.sciencedirect.com/science/article/pii/S1364032116309297>. <https://doi.org/10.1016/j.rser.2016.11.155>.
- [75] Radovic M, Ghalwash M, Filipovic N, Obradovic Z. Minimum redundancy maximum relevance feature selection approach for temporal gene expression data. *BMC Bioinformatics* 2017;18(1):9. <https://doi.org/10.1186/s12859-016-1423-9>. [10.1186/s12859-016-1423-9](https://doi.org/10.1186/s12859-016-1423-9).
- [76] Kim TY, Cho SB. Predicting the household power consumption using cnn-lstm hybrid networks. In: Yin H, Camacho D, Novais P, Tallón-Ballesteros AJ, editors. *Intelligent Data Engineering and Automated Learning – IDEAL 2018*. Cham: Springer International Publishing; 2018, p. 481–490.
- [77] Kim TY, Cho SB. Predicting residential energy consumption using cnn-lstm neural networks. *Energy* 2019;182:72–81. <https://doi.org/10.1016/j.energy.2019.05.230>.
- [78] Sultana T, Khan ZA, Javaid N, Aimal S, Fatima A, Shabbir S. Data analytics for load and price forecasting via enhanced support vector regression. In: Barolli L, Khafa F, Khan ZA, Odhabi H, editors. *Advances in Internet, Data and Web Technologies*. Cham: Springer International Publishing; 2019, p. 259–270.
- [79] Nawaz M, Javaid N, Mangla FU, Munir M, Ihsan F, Javaid A, et al. An approximate forecasting of electricity load and price of a smart home using nearest neighbor. In: Barolli L, Hussain FK, Ikeda M, editors. *Complex, Intelligent, and Software Intensive Systems*. Cham: Springer International Publishing; 2020, p. 521–533.
- [80] Brusaferrri A, Fagiano L, Matteucci M, Vitali A. Day ahead electricity price forecast by NARX model with LASSO based features selection. *IEEE International Conference on Industrial Informatics (INDIN) 2019;2019-July:1051–1056*. <https://doi.org/10.1109/INDIN41052.2019.8972263>.
- [81] Leerbeck K, Bacher P, Junker R, Goranović G, Corradi O, Ebrahimi R, et al. Short-term forecasting of CO2 emission intensity in power grids by machine learning. *Applied Energy* 2020;277. <https://doi.org/10.1016/j.apenergy.2020.115527>.
- [82] Zareipour H, Canizares CA, Bhattacharya K. Economic impact of electricity market price forecasting errors: A demand-side analysis. *IEEE Transactions on Power Systems* 2010;25(1):254–262. <https://doi.org/10.1109/TPWRS.2009.2030380>.
- [83] Uniejewski B, Nowotarski J, Weron R. Automated variable selection and shrinkage for day-ahead electricity price forecasting. *Energies* 2016;9(8). <https://doi.org/10.3390/en9080621>.
- [84] Kaminski V. *Energy markets*. Risk Book; 2013.
- [85] Li W, Paraschiv F. Modelling the evolution of wind and solar power infeed forecasts. *Journal of Commodity Markets* 2021;100189. <https://doi.org/10.1016/j.jcomm.2021.100189>.
- [86] Herceg S, Željka Ujević Andrijić, Bolf N. Development of soft sensors for isomerization process based on support vector machine regression and dynamic polynomial models. *Chemical Engineering Research and Design* 2019;149:95 – 103. <https://doi.org/10.1016/j.cherd.2019.06.034>.