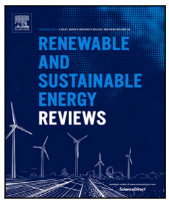
Renewable and Sustainable Energy Reviews 209 (2025) 115097

Contents lists available at ScienceDirect

Renewable and Sustainable Energy Reviews

journal homepage: www.elsevier.com/locate/rser



Original research article

Electricity demand uncertainty modeling with Temporal Convolution Neural Network models

Sujan Ghimire [a], Ravinesh C. Deo [a], David Casillas-Pérez [b,*], Sancho Salcedo-Sanz [c,a], Rajendra Acharya [a], Toan Dinh [d]

[a] School of Mathematics, Physics, and Computing, University of Southern Queensland, Springfield, QLD, 4300, Australia
[b] Department of Signal Processing and Communications, Universidad Rey Juan Carlos, Fuenlabrada, 28942, Madrid, Spain
[c] Department of Signal Processing and Communications, Universidad de Alcalá, Alcalá de Henares, 28805, Madrid, Spain
[d] Centre for Future Materials, University of Southern Queensland, Springfield, QLD, 4300, Australia

ARTICLE INFO

Dataset link: https://www.energex.com.a

Keywords:
Deep learning
Temporal Convolution Network
Uncertainty analysis
Hybrid models
Long Short-term Memory network

ABSTRACT

This work presents a Temporal Convolution Network (TCN) model for half-hourly, three-hourly and daily-time step to predict electricity demand (G) with associated uncertainties for sites in Southeast Queensland Australia. In addition to multi-step predictions, the TCN model is applied for probabilistic predictions of G where the aleatoric and epistemic uncertainties are quantified using maximum likelihood and Monte Carlo Dropout methodologies. The benchmarks of TCN model include an attention-based, bi-directional, gated recurrent unit, seq2seq, encoder–decoder, recurrent neural networks and natural gradient boosting models. The testing results show that the proposed TCN model attains the lowest relative root mean square error of 5.336-7.547% compared with significantly larger errors for all benchmark models. In respect to the 95% confidence interval using the Diebold–Mariano test statistic and key performance metrics, the proposed TCN model is better than benchmark models, capturing a lower value of total uncertainty, as well as the aleatoric and epistemic uncertainty. The root mean square error and total uncertainty registered for all of the forecast horizons shows that the benchmark models registered relatively larger errors arising from the epistemic uncertainty in predicted electricity demand. The results obtained for TCN, measured by the quality of prediction intervals representing an interval with upper and lower bound errors, registered a greater reliability factor as this model was likely to produce prediction interval that were higher than benchmark models at all prediction intervals. These results demonstrate the effectiveness of TCN approach in electricity demand modelling, and therefore advocates its usefulness in now-casting and forecasting systems.

1. Introduction

Sustainable economic development requires robust methods for electricity demand side planning and management systems. For decision-makers, electricity demand models offer unique perspectives to manage the future usage of energy and including the best options for increasing the mix of green and cleaner energy resources into day-to-day electricity consumption [1]. In times of an imbalanced demand or supply of electricity, fossil fuel generation can be prioritized to resolve an energy crisis. However, fossil fuel generation produces CO_2 emissions at an un-affordable rate, and may therefore affect the global power markets. As the Sustainable Development Goal (SDG-7) requires every person has access to affordable energy resources [2], countries must plan well-designed electricity demand monitoring systems to ensure a reliable supply of energy.

Increasing rates of electricity consumption can produce huge increase in electricity use in comparison with the other energy sources [3]. Existing storage technologies do not currently provide efficient storage of electricity at relatively large scales. Therefore, an electricity demand (G) prediction model are useful in the economic growth and sustainability of energy supply and management [4]. Climate change, on the other hand, based on future climate projections and increasing temperatures can place unprecedented demand on electricity use [5]. Underestimating or overestimating the future G may therefore result in numerous undesirable effects including financial burden, increased electricity price or shortage of electricity [6]. Even subtle changes in the accuracy of G prediction may significantly impact the operating costs and affect the already competitive electricity market. Despite

* Corresponding author.
E-mail addresses: sujan.ghimire@usq.edu.au (S. Ghimire), ravinesh.deo@usq.edu.au (R.C. Deo), david.casillas@urjc.es (D. Casillas-Pérez), sancho.salcedo@uah.es (S. Salcedo-Sanz), rajendra.acharya@usq.edu.au (R. Acharya), toan.dinh@usq.edu.au (T. Dinh).

https://doi.org/10.1016/j.rser.2024.115097
Received 9 May 2024; Received in revised form 4 September 2024; Accepted 4 November 2024
Available online 20 November 2024

Abbreviations

ACE	Average Coverage Error
AI	Artificial Intelligence
ANFIS	Adaptive Neuro Fuzzy Inference System.
ANN	Artificial Neural Network.
APB	Absolute Percentage Bias.
ARIL	Average Relative Interval Length.
ARIMA	Autoregressive Integrated Moving Average.
ARMA	Auto-Regressive Moving Average.
ATTLSTM	Attention LSTM.
AU	Aleatoric Uncertainty.
BHPO	Bayesian Hyperparameter Optimization Algorithm.
BILSTM	Bidirectional LSTM.
BM	Bayesian Models.
CLSTMED	Encoder–Decoder LSTM.
CNN	Convolutional Neural Network.
CNNGRU	CNN Combined with GRU Method.
DBM	Deep Boltzmann Machine.
DBN	Deep Belief Networks.
DL	Deep Learning.
DLSTM	Deep LSTM.
DM Test	Diebold–Mariano.
DNN	Deep Neural Network.
DRNN	Deep Recurrent Neural Network.
DT	Decision Trees.
ELM	Extreme Learning Machine.
EM	Ensemble Methods.
ES	Exponential Smoothing.
EU	Epistemic Uncertainty.
FLI	Fuzzy Logic Inference.
G	Electricity Demand.
GBM	Gradient Boosting.
GM	Grey Models.
GPI	Global Performance Indicator.
GRU	Gated Recurrent Unit.
KGE	Kling–Gupta Efficiency.
LM	Legates and McCabe's Index.
LSTM	Long Short-Term Memory.
LSTMCNN	LSTM Combined with CNN.
MAPE	Mean Absolute Percentage Error.
MCDO	Monte-Carlo Dropout.
MLP	Multi-Layer Perceptron Model.
MNLL	Mean Negative Log-Likelihood.
MSE	Mean Squared Error.
NGBOOST	Natural Gradient Boosting.
NLL	Negative Log-Likelihood.
NN	Neural Networks.
NS	Nash–Sutcliffe Index.
PACF	Partial Auto-Correlation Function.
PI	Prediction Interval.
PICP	Prediction Interval Coverage Probability.
PINAW	Normalized Mean Prediction Interval Width.
PINC	PI Nominal Confidence.
RBM	Restricted Boltzmann Machine.
RMAE	Relative MAE.
RMSE	Root Mean Squared Error.
RNN	Recurrent Neural Networks.
RRMSE	Relative RMSE.
SDG	Sustainable Development Goal.
seq2seq	Sequence to Sequence.
SVM	Support Vector Machine.
TCN	Temporal Convolutional Network.
TPU	Total Predictive Uncertainty = EU + AU.
TS	Time-Series.
WI	Willmott's Index.
WS	Winkler Score.
XGB	Xtreme Gradient Boosting.

the fact that there are no definitive criteria for classification of range of G predictions, prediction methods are grouped roughly into three categories: short term (from approximately 30 min to six hour ahead), mid-term (lead up to daily prediction horizons), and long-term (weekly, monthly or annual). This paper therefore aims to develop novel electricity demand forecasting models for the short-term (half-hourly, or 3-hourly) and the long-term (daily) time-steps.

For the purpose of analysis, characterization and prediction of G, statistical and computational intelligence models have been employed [7]. Conventional methods of predicting G are adopting stochastic time series and regression approaches. A common time-series (TS) model using Autoregressive Integrated Moving Average (ARIMA) method is capable of producing good results for linear prediction problems [8–11]. The TS-based models use trend analysis to predict the future electricity demand as shown by Amjady et al. [12] where an ARIMA model for forecasting loads based on different times of the year was proposed. There have been numerous studies that use historical demand, weather or other variables as exploratory variables, including Juberias et al.'s study of [13] that proposed an hourly load model using ARIMA and regression analyses or multivariate regressions by [14,15] to predict the G. The use Exponential Smoothing (ES) method [16,17] was also used successfully for G prediction. In particular, the ES methodology aims to consider the electricity demand time series as being locally stationary with a slowly moving average. Therefore, authors in [18] build an ES-based model to simulate the electricity demand time series where the prediction results indicates that their model did exceptionally well measured by a Root Mean Squared Error ($RMSE$) and Mean Absolute Percentage Error ($MAPE$) for shorter and medium prediction horizons. Similarly, Grey Models (GM) [19] have been widely adopted for G prediction as they are able to capture the characteristics of uncertain systems even when the data are relatively sparse [20–24].

A number of statistical models, such as ARIMA, are limited to linear distribution assumptions. As a result, the model's efficacy depends heavily on data required to calibrate model parameters and the likelihood of avoiding model over fitting [1]. In fact, an ARIMA model requires stationary characteristics in order to void poor predictive accuracy. In spite of GM models not requiring statistical assumptions, the accuracy could be dependent on the degree of dispersion of raw TS datasets [25]. In addition, there are often different strengths and weaknesses to each predictive models so no single approach is superior for all types of G prediction problems.

To address challenges of traditional models, artificial intelligence (AI) methods, inspired by human brain, capable of solving nonlinear problems, have been found to be more efficient. Some of the key AI methods include Neural Networks (NN) [26], Recurrent Neural Networks (RNN) [27], Ensemble methods (EM) [28], Support Vector Machine (SVM) [29], Gradient Boosting (GBM) [30], Bayesian Models(BM) [31], Decision Trees (DT) [32], Fuzzy Logic Inference (FLI) [33,

34], and Multi-layer Perceptron [35,36] (MLP) implemented for large scale *G* prediction problems. However, in their primitive forms, ANN, RNN, SVM, GBM, EM, BM, DT and MLP models are shallow models with a single hidden layer or simplistic architecture [37], which leads to a lesser capability to predict complex datasets such as *G*.

The study of Ekonomou et al. [38] developed a Multi-Layer Perceptron (MLP) model to predict *G* incorporating four factors: climate, Gross Domestic Product, historical electricity demand and power capacities, showing that the MLP model can produce promising results relative to a linear regression or SVM model. To predict the United States-based industrial energy demand, Kialashaki et al. [39] developed ANN and Multi-Linear Regression (MLR) models to show an ANN as more accurate. Gajowniczek et al. [40] implemented an SVM, ANN, random forest, ARIMA and step-wise regression for short-term *G* prediction at 24 time-steps. The prediction capability was higher for ANN compared with other models. Abraham et al. [41] implemented Fuzzy Neural Network for *G* prediction in Victoria Australia; showing its superior capability relative to ANN and ARIMA models. The authors in [42] used Adaptive Neuro Fuzzy Inference System (ANFIS) for *G* prediction in a province in Canada to show ANFIS as more accurate.

Despite most AI-based methods showing good ability to incorporate the best predictive features from *G* time series, their limitations include: (a) not being able to handle multiple time series [43], (b) the inability to find reliable time dependence information between short-term and long-term scenarios *G* [43], (c) prone to over-fitting [44], (d) lack of generalizability [45], (e) complexities in handling big data, and (f) slow convergence of model parameters [46]. These issues continue to inspire researchers to shift to the more promising Deep Learning (DL) methods for *G* prediction problems.

Comparatively to simplistic AI models, deep learning models can extract features automatically without having much knowledge of the data's background details [47,48]. Unlike shallow models, that transform input data only a few times, DL transforms inputs multiple times before producing an output and therefore are capable of learning extremely complicated patterns without manual input. Wan et al. [49] built a DeepNet method for short-term electricity load modelling whereas a Convolutional Neural Network (CNN) model was used by [50] to extract features from historical patterns used later to predict *G*. A Deep RNN (i.e., DRNN) model was implemented by Shi et al. [51] to predict short-term electricity load at regional aggregated/household level concluding that the DRNN model outperformed the shallow neural networks.

A Long Short-term Memory (LSTM) model was developed by Torres et al. [52] and Kong et al. [53] to predict *G* over short-term and medium-term time horizons. Muralitharan et al. [54] implemented a NN-based genetic algorithm and neural NN-based particle swarm optimization to produce better results than a CNN model. Other approaches such as Autoencoders (AE) [55,56], LSTM [57–60], Gated Recurrent unit (GRU) [61], Restricted Boltzmann Machine (RBM) [62], Deep Belief Network (DBN) [63], BiDirectional-LSTM (Bi-LSTM) [64–66] and Deep Boltzmann Machine (DBM) [67] have all been recognized as promising DL methods that are frequently used in *G* prediction problems [68].

In prior studies, researchers proposed point-based models for *G* estimation. In order to capture uncertainties in future load fluctuations, point-based prediction is insufficient since it has a unique value at a given time step. Therefore, as a probabilistic extension to DL methods, this study adopts TCN used previously in many other prediction problems e.g., [69–71]. The TCN method, as a variation of CNN for sequence-to-sequence modelling, combines the merits of an RNN and a CNN model. Therefore, the two distinguishing characteristics of a TCN model leading to its implementation in this research paper are [69–71]:

- The convolutions in proposed model architecture are of the causal form so this can avoid "leakage" of information from past to future *G*.
- In the TCN architecture, mapping of any sequence can be done to an equivalent length's output sequence. It therefore possesses a long effective history size with capability to search for predictive information relatively far in history of data prior to making *G* predictions.

As the proposed TCN model in this study incorporates very deep networks along with residual layers and dilated convolutions, it can capture far-sight information, which has several advantages over other DL algorithms.

- **Parallel Processing:** In conventional methods, predictions for later time-step wait for their predecessor to be fully complete but in the proposed TCN model, convolutions are done in parallel as identical filters are used in all layers. Hence, in training and evaluation phases, a long input sequence found in *G* data is analysed as a whole rather than in a sequence to sequence manner.
- **Flexible Receptive Field Size:** TCN models have better control over memory size, since they can adjust the size of their receptive field by using various methods such as stacking more dilated causal convolutional layers or increasing the filter size.
- **Stable Gradients:** A TCN architecture, on the other hand, back-propagates relatively distinctly from the sequence's temporal direction, thereby avoiding the exploding and vanishing gradient problems.
- **Low Memory Requirement:** Real-time monitoring systems rely on *G* data that are relatively long or continuous. Due to this, traditional methods (such as LSTMs and GRUs) can consume a significant amount of memory when storing partial results in multiple cell gates. In a TCN-based model, there are shared filters across layers, so the path of backpropagation depends on the path of the network. Consequently, a TCN model offers a major advantage over gated methods (e.g. RNNs), which require a memory several fold larger.
- **Variable Length Inputs:** The proposed TCN model takes in inputs of arbitrary lengths by sliding 1-D convolutional kernels, providing the ability to incorporate multiple length inputs for *G* forecasting at multiple time-steps.

Although TCN provides significant motivation over other DL models, it does come with some constraints such as: *(i)* potential issue of data storage in evaluation phase where it must take in raw sequences up to the effective history length that could require more memory in model evaluation phase, or *(ii)* a potential for model's parameter changes in case of its transfer to a different application domain where little memory is needed compared to a domain requiring much longer memory.

This paper makes important contributions by capitalizing on earlier works [69–71] to develop a robust predictive model for electricity demand and uncertainty estimation:

1. A TCN-based model is developed and compared against a number of DL algorithms applied for the prediction of *G* based on statistically significant lag combination historical data.
2. The proposed TCN model is evaluated is using actual operational electricity demand data for three different locations in Southeast Queensland, Australia.
3. The proposed TCN model is improved for its capability to predict electricity demand by incorporating Monte Carlo dropouts and maximum likelihood techniques that are adopted to quantify the aleatoric and epistemic uncertainties.
4. An assessment of the *G* predictive model is made against state-of-the-art DL models.

2. Conceptual frameworks

Here, the TCN model (as well as variant DL models based on CNN, RNN, LSTM etc used as benchmark approaches) are discussed to justify

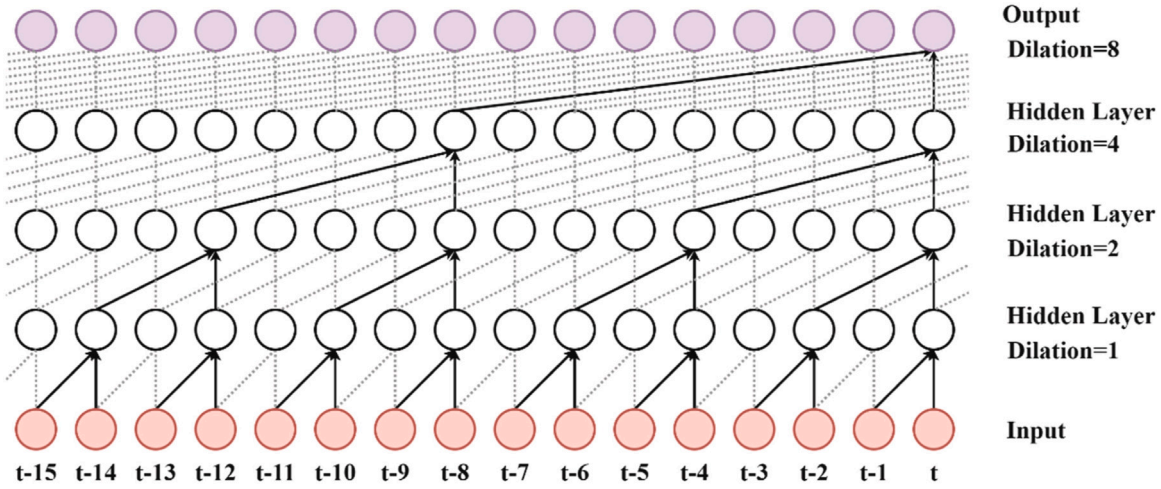


Fig. 1. Dilated causal convolution example; filter size $k = 2$; dilation rate $d = [1, 2, 4, 8]$.

the adoption of this method for multi-step G prediction. Details of other models e.g., DNN and Natural Gradient Boosting (NGBOOST) probabilistic regression are available in [72–74].

2.1. The temporal convolutional network model

The TCN model, a special case of CNN [75,76], has an efficient performance compared to its counterparts models due to (a) use of causal dilated convolutions, (b) simpler architecture that avoids gating mechanism, (c) an architecture with stable gradients, (d) computational efficiency given input sequence is processed once (not sequentially as with RNNs) and that current predictions do not wait for results from a previous time step [77].

For ensuring each subsequent layer to equal to its predecessor, the model employs a 1-D CNN architecture with no padding between hidden layers and input layers. By applying causal convolution only once to current and previous values of G, no information leaks from the past into the future value. Simple causal convolutions have linear history sizes compared to networks [78]. Real-time electricity use monitoring systems may require a large filter to process a long sequence of G inputs. Because of this constraint, dilated causal convolutions are used to skip values in order to enlarge the receptive field exponentially [79].

In Fig. 1, we depict a dilated causal convolution. For a sequenced $\mathrm{x} \in \mathbb{R}^n$, filter $f : \{0, 1, \ldots k-1\}$, we state the dilated convolution F on s sequence as:

$$F(s) = \left(\mathrm{x} *_d f\right)(s) = \sum_{i=0}^{k-1} f(i) \cdot x_{s-d \cdot i} \tag{1}$$

Here, k is the filter size, d is the dilation factor, $*$ is the convolution operator and the subscript $s - d, i$ refers the direction of the past.

With increasing depth of network, the dilation factor will exponentially increase following $d = 2^l$ where l is the network level. At the top layer ($d = 1$) (corresponding to regular convolutions), d increases exponentially and reaches 4 at the top layer ($d = 4$). As a result of pyramidal structure and aggregation, the TCN model's receptive field is effectively increased without affecting the input coverage as long memories can be achieved. As the TCN network deepens or is enlarged, the receptive field becomes increasingly dependent on network depth and dilation factor. The stabilization also becomes especially critical with this process so in general, the deep network layers lead to produce undesirable effects e.g., gradient disappearance and over-fitting [80]. To overcome this, residual learning frameworks [81] have shown to enhance and simplify the network training.

A generic residual module was used instead of convolutional layers with examples of residual TCN and connections, as shown in Fig. 2.(a) and .(b). There are two layers of dilated causal convolutions and ReLU in the residual block (Fig. 2.(a)). The convolutional filters are each subjected to the batch approach for this normalization [82]. Finally, a layer performing the dropout is added after every dilated convolutions following the strategy in earlier works [83].

A 1D CNN model is recommended for handling uni-dimensional data (e.g., G, wind speed, solar radiation) as they require simple calculations in forward and reverse propagation stages. [84]. Its performance, however, is influenced by several hyperparameters (hidden neurons, filter size, pooling methods) in the training phase. Here, the 1-dimensional input is fed in network and its output layer composed of desired neurons [85] as schematized in Fig. 3.

2.2. The benchmark models

To comprehensively evaluate the proposed TCN model, this study has also developed several variants of an RNN model which are used to overcome the issues of vanishing gradients in conventional models. Two noteworthy examples are an LSTM and a GRU model-based DL model [88]. The proposed TCN model for G forecasting is tested against the LSTM model as a benchmark model. The work of [89] introduced a conventional LSTM networks with memory cells or forget gates to capture long-term dependencies with distinct hidden states h_t and c_t that store both long-term and short-term information [90]. By changing the hidden layers, the key variants of LSTM model include: bidirectional LSTM (i.e., BILSTM) [91], Attention LSTM (i.e., ATTLSTM) [92] and Deep LSTM (i.e., DLSTM) proposed to address drawbacks of conventional LSTM models.

2.2.1. BILSTM

The proposed TCN model is also benchmarked using BILSTM, a variant of an LSTM model that uses two LSTM models for past and future information to be used in G prediction [93]. Fig. 2.(c) shows the two LSTM models with a hidden vector $\vec{h}_t$ in forward direction and hidden vector $\overleftarrow{h}_t$ in backward direction. By combining outputs of forward input and reverse input sequence, the y_t, denoted as the final output, takes the form:

$$\vec{h}_t = LSTM\left(x_t, \vec{h}_{t-1}\right) \tag{2}$$

$$\overleftarrow{h}_t = LSTM\left(x_t, \overleftarrow{h}_{t-1}\right) \tag{3}$$

$$y_t = f\left(W_{\vec{h}y}\vec{h}_t + W_{\overleftarrow{h}y}\overleftarrow{h}_t + b_y\right) \tag{4}$$

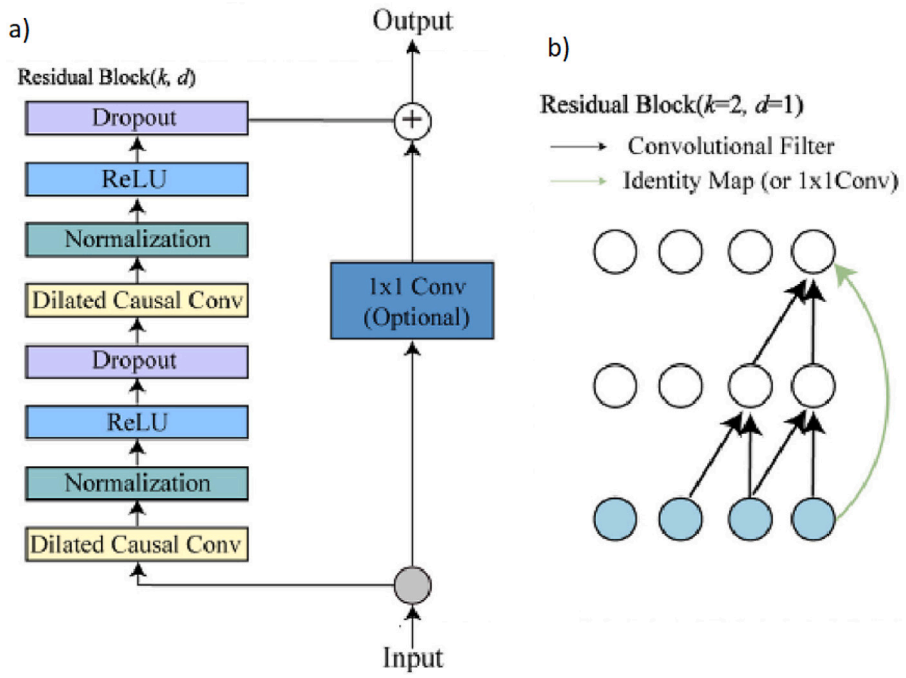


Fig. 2. (a) General schematic of a residual block of the TCN model. A convolution of 1 × 1 is applied for residual input and output that have unique dimensions, (b) Residual connection.

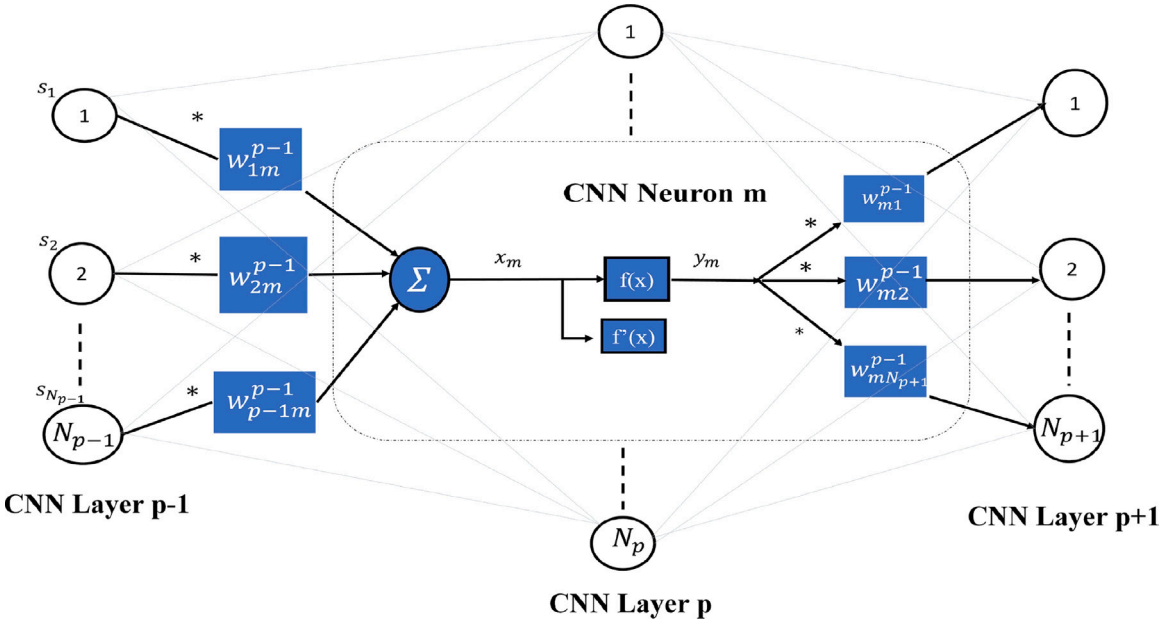


Fig. 3. Schematic of a three-layered 1-D CNN model [86] utilized in developing the TCN model for short-term and long-term electricity demand modelling.

2.2.2. ATTLSTM

To benchmark the efficacy of the TCN model we adopted ATTLSTM, a variant of the LSTM model. The ATTLSTM model employs attention mechanism similar to human selective visual attentions [94] that predominate key elements from input sequence to affect the output data. Therefore, the ATTLSTM model can learn critical information from input sequences [95]. By mimicking a human brain, it gains a deeper understanding of the inputs, implemented as follows:

$$Vat_q = \text{atten}\left[Vec_q\right] \tag{5}$$

$$\xi_q = \text{softmax}\left[Vat_q\right] = \frac{\exp\left(Vat_q\right)}{\sum_{q=1}^{Q}\exp\left(Vat_q\right)} \tag{6}$$

$$\chi = \xi^T Vec \tag{7}$$

As a first step, the $\text{atten}[\cdot]$ is a learning function to determine each element in vector Vec denoted by Vat, and q represents the index value. To obtain the attention weight vector ξ, Vat is normalized using $\text{softmax}[\cdot]$, where Q is the length of Vat. For attention vector χ, Vec is multiplied by ξ. For more information on an ATTLSTM model, readers can consult [96].

2.2.3. CLSTMED, CNNGRU and LSTMCNN hybrid models

Three competing models to evaluate TCN model for G and uncertainty predictions are: *(i)* CNN combined with seq2seq LSTM to

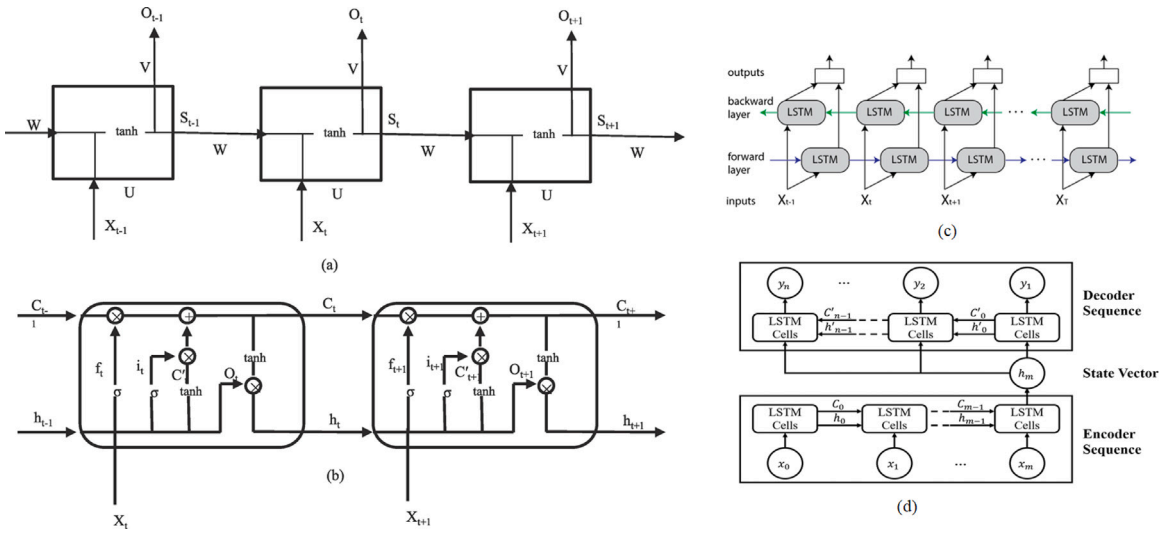


Fig. 4. (a) RNN, (b) LSTM, (c) seq2seq LSTM (i.e., LSTMED model), (d) Bidirectional LSTM (i.e., BILSTM) model. The symbols/terms are as per [87].

generate CLSTMED or encoder–decoder LSTM [97], *(ii)* CNN combined with GRU to construct CNNGRU [98], *(iii)* LSTM combined with CNN to generate LSTMCNN [99–101].

Fig. 4 shows a seq2seq network with encoder/decoder system. The encoder pulls out features from data [73]. As an input, variable-length sequenced data $x = (x_1, x_2, \ldots, x_n)$ is encoded into a fixed-length vector C. Through the combination of input data at a current time, the decoder calculates the next time sequence Y by decoding the state vector C. When input data are read, h is also updated. As vector C contains information from the sequence X, it can be viewed as a summary of all the information extracted from that sequence.

2.3. Uncertainty analysis of predicted electricity demand

As a major task in this study, we adopted improved the TCN model's capability to analyse the aleatoric uncertainty (AU) and the epistemic uncertainty (EU) that are required to construct a robust electricity demand monitoring and management system. Notably, the modelling of G so far focused on point-based predictions without much consideration to the quantification of predicted uncertainty.

Therefore, to explain these uncertainties in the TCN model, we consider $Y = f(x) + \epsilon$ where statistically, $f(\cdot)$ is the predicted point estimate of a learned model known as the EU part. Notably, the origin of EU is the model variance σ_m^2; ϵ or the irreducible noise known as the AU part of the uncertainty. Regardless of how the G data were generated and measured, the AU-based component in a predictive almost always exist as this type of uncertainty is intrinsically linked to the variance in the modelled data.

Fig. 5 shows briefly the nature of AU and EU in typical regression models. Note that in accordance with this, the total predictive uncertainty (TPU) is therefore comprised of *(i)* EU, *(ii)* AU expressed, and is expressed $TPU = EU + AU$.

In Fig. 5.(a), it is apparent that because of the presence of noisy data such as that indicated in yellow circle, there is non-deterministic relationship between the model inputs (or predictor variables) and the model output (or the target), and thus, the value of AU is considerably high. Additionally, from Fig. 5.(b) we also note that since there are no data indicated as red circles, it seems to be rather difficult to determine the model parameters and the structure, so a high value of EU is possible. Thus, quantifying the predictive uncertainties in G data can reveal the amount of noise in observed data and the effectiveness of the TCN model.

For uncertainty analysis, one popular technique is Monte-Carlo Dropout (MCDO) method to quantify EU [102,103]. More generally,

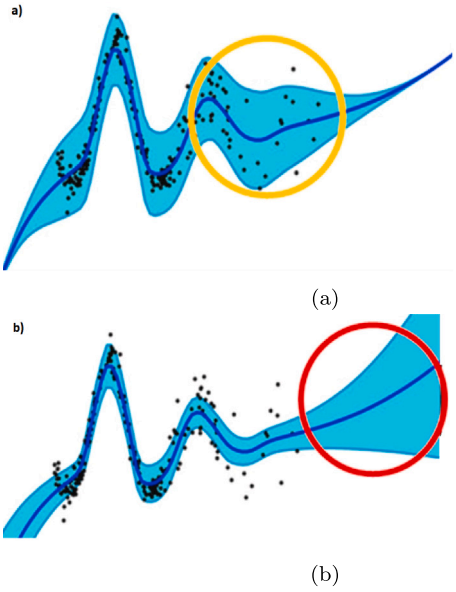


Fig. 5. (a) Noisy data in yellow circle causes aleatoric uncertainty, (b) Lack of data causes epistemic uncertainty in this regression problem. Uncertainty intervals are indicated by the black areas in the sky.

the dropout prevents over-fitting as it progressively disables (i.e., dropping out) the units while the training of model takes place. This can be considered as applying random noise [104,105].

In principle, MCDO refers to the use of random units during forward pass dropped from networks while prediction occurs with a trained network. Repeating this multiple times, many different TCN model outcomes are generated so that the prediction distribution can represent the degree of uncertainty in the underlying model and the MCDO can approximate to a deep Gaussian Process approach [106]. For full mathematical derivation of the MCDO used in modelling of uncertainty, authors may refer to [107].

Consider an example of a neural network model with L layers where W_l, b_l and K_l is the weight matrices, bias vectors, dimensions of lth layer. We indicate the output and target class of ith input $x_i (i = 1, \ldots, N)$ using $\hat{y}_i$ and y_i. The objective function L_2 regularization is:

$$L_{dropout} = \frac{1}{N}\sum_{i=1}^{N} E\left(y_i, \hat{y}_i\right) + \lambda \sum_{l=1}^{L} \left(\|W_i\|_2^2 + \|b_i\|_2^2\right) \quad (8)$$

For each input and its respective layer, except for the output layer, the binary variables are sampled with a probability of p_i. A unit i is dropped if it is 0 for an input data value. To update the parameters, the backward pass uses the same values.

As a primary goal for the G prediction problem, we quantified the AU by estimating $p(y|x)$ through a probability density function defined by the Gaussian curve. We aimed to approximate the conditional probability distribution $p(y|x,\theta)$ of the target y given an observed input value x [108]. Therefore, the parameters θ of a DL network f_θ was fitted in such a way that it produced a prediction $\hat{y} = f_\theta(x)$.

The mean squared error (MSE) was minimized between the true and predicted G with the network typically trained for a single point estimation. It is useful to ascertain that the optimization of MSE refers setting right parameters of the conditional distribution. In our case, this was estimated by a Gaussian function $p(y|x) = \mathcal{N}(\hat{y}(x), \sigma^2)$. Here, the mean $\hat{y}(x)$ varies with the input x and has a constant noise term σ^2 [109]. Our approach relaxes the constraint of constant noise by estimating it from the input. Instead of a point-estimate $\hat{y}$, we consider $[\hat{\mu}, \hat{\sigma}^2] = f_\theta(x)$ whereby true conditional distribution can be estimated. In doing so, we applied a Gaussian function $p(y|x) = \mathcal{N}(\hat{\mu}(x), \hat{\sigma}^2(x))$ by using mean/variance that formed the functions of x (or the input variable). We achieved this using a maximum likelihood (ML) approach (see [110]).

Our next step is to use the negative log-likelihood (NLL) function, which is a normal distribution that acts as a loss function in our problem by ignoring the constants such as [111,112]:

$$\mathcal{L}_{NLL} = \frac{1}{N}\sum_{i=1}^{N}\frac{\left(\hat{\mu}(x_i) - y_i\right)^2}{2\hat{\sigma}^2(x_i)} + \frac{1}{2}\log\hat{\sigma}^2(x_i). \tag{9}$$

Note that if there are multiple samples, the mean negative log-likelihood is calculated by averaging over the log-likelihoods (MNLL).

$$\mathcal{L}_{MNLL} = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_{NLL}(x_i, y_i) \tag{10}$$

The numerator of NLL (Equation. (9)) encourages mean prediction $\hat{\mu}(x)$ to approach the observed data whereas denominator $\hat{\sigma^2(x)}$ is elevated when the deviation from the mean $(y - \hat{\mu}(x))^2$ is large. By counterbalancing the variance, the first term prevents it from growing indefinitely. As a result of this strategy, the aleatory uncertainty of the data generation process is captured.

3. Procedure

This section describes the pre-processing steps for gathering electricity demand data for Southeast Queensland, Australia, the development of the TCN model, and finally, the deterministic, statistical, and probabilistic metrics that were used to evaluate the model comprehensively.

3.1. Description of electricity demand datasets

Energex Limited, the largest government-owned electricity company in Australia, provided G data for this study. Energex operates under provisions of Government Owned Corporations Act 1993. Energex owns, operates, and maintains electricity distribution networks that serve Brisbane, Gold Coast, Sunshine Coast, Logan, Ipswich, Redland, and the Moreton Bay region in Southeast Queensland. The company also distributes electricity to 3.5 million residential and business customers (https://www.energex.com.au). Each day, G in megawatts is recorded half-hourly and by using this database, maintenance planners are able to estimate actual demand every half-hour.

In developing the TCN model, 30 min, three hours, and daily prediction intervals at Acacia Ridge, Alexandra Headland, and Zillmere were used from 01/07/2011 to 30/06/2021 (see Fig. 6). Energex half-hour intervals were converted to 3-hourly intervals by summing six values per interval, and 24-hour interval data were similarly converted.

The boxplots (Fig. 7.(a)-7.(c)) illustrate the distribution of electricity demand (G,MW) across three different locations – Acacia Ridge, Zillmere, and Alexandra Headland – over three time intervals: half-hourly, 3-hourly, and daily. At Acacia Ridge, G remains relatively low, with a narrow interquartile range (IQR), indicating limited variability. Outliers appear sporadically, especially in the half-hourly data, suggesting occasional spikes in G. As the time interval shifts to 3-hourly and daily, both the median and variability increase slightly, but the overall distribution remains relatively stable compared to the other locations. Zillmere shows higher G than Acacia Ridge, with a wider IQR and more pronounced variability, particularly in the 3-hourly and daily data. The frequency of outliers increases across all time intervals, reflecting more fluctuations in G, especially in longer time frames. Alexandra Headland consistently exhibits the highest levels of G and the greatest variability. In the half-hourly data, the median G is much higher than at the other locations, with a broader range and numerous outliers, suggesting frequent power surges. This trend continues and intensifies in the 3-hourly and daily intervals, where Alexandra Headland demonstrates significantly larger fluctuations. The daily data shows an especially wide range of values, with a large number of outliers, indicating highly variable and often extreme G levels. In summary, Alexandra Headland consistently experiences the highest and most variable G across all time intervals, followed by Zillmere, which also displays considerable variability. Acacia Ridge, in contrast, exhibits lower G and smaller fluctuations. This analysis suggests that Alexandra Headland is the most dynamic location in terms of power output, with larger and more frequent fluctuations compared to the other two sites.

Fig. 8 presents the distribution of G across different time frames for Acacia Ridge sub-station, with green boxplots showing the variability and the blue line representing the mean values. In the hourly distribution, G is lowest during the early hours of the day, gradually rising and peaking between 8 AM and 8 PM, with the highest demand observed around noon. Demand variability is greater during the day, with more outliers in the evening, while nighttime demand is lower and more stable. In the daily distribution, demand increases from Sunday through Wednesday, with Wednesday showing the highest median demand, before declining towards the weekend, where Saturday has the lowest demand. Finally, in the monthly distribution, electricity demand fluctuates throughout the year, with peaks during the winter months (June to August) and a slight decrease in spring (November). Variability remains consistently high across all months, though certain months show more pronounced outliers, indicating sporadic spikes in demand.

3.2. Development of the temporal convolutional network and benchmark models

This section describes the predictive model development, selection of electricity demand time series data, pre-processing, normalization and partitioning of training and validation sets and the TCN (and benchmark) model developments and hyperparameters tuning steps.

3.2.1. Input selection

To determine the optimal lagged combinations of G data required to build the predictive model, partial auto-correlation function ($PACF$) of G is used [113]. Fig. 9 shows $PACF$ for half-hourly, 3-hourly and daily prediction horizon. Evidently, the $PACF$ of training set collapses at 10, 10 and 9 for half-hourly, 3-hourly and daily data, respectively, for Acacia Ridge study site.

Therefore, for Acacia Ridge location the time lag of input (i.e., G) variable was chosen to be (t - 10) for the half-hourly and daily electricity demand predictions whereas for 3-hourly period, the time lag derived from historical electricity demand data was selected to be (t - 9). In other words, we should comfortably be able to choose the ten time-based historical G data points before the current time as the input variables for the TCN (and comparative) models to predict G at a current time period. To validate the PACF criteria, we also used an

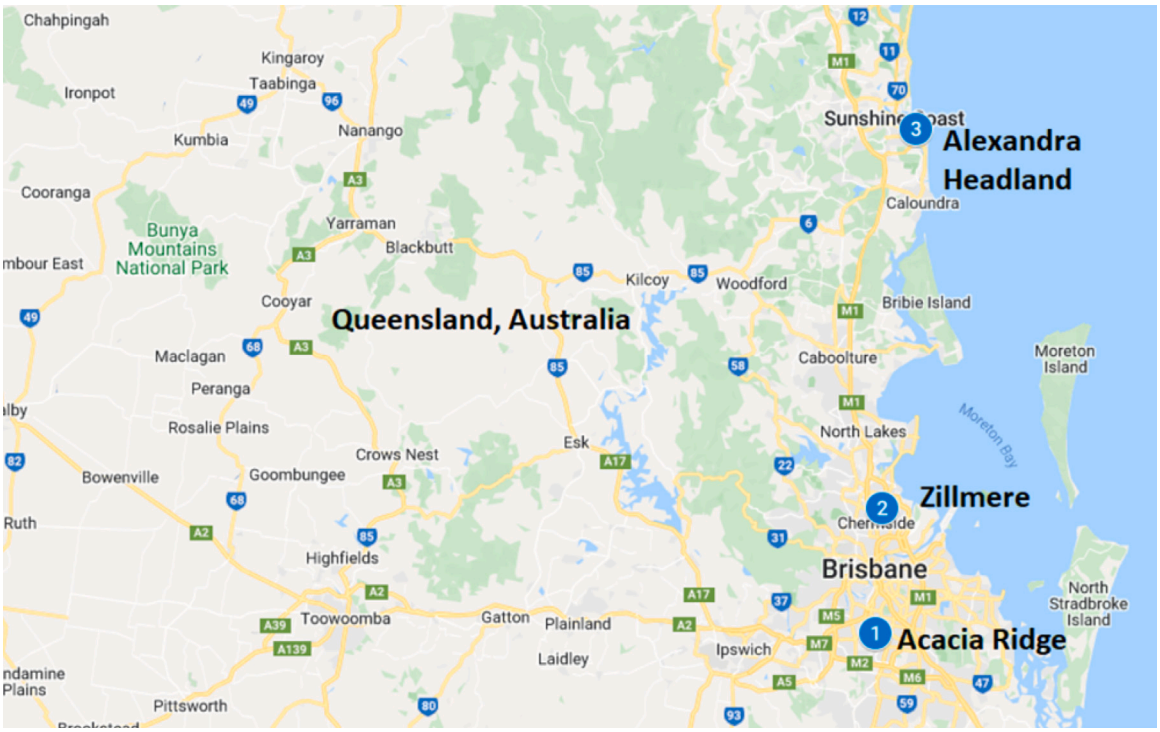


Fig. 6. Geographic map of study area in southeast Queensland, Australia where G prediction model was developed.

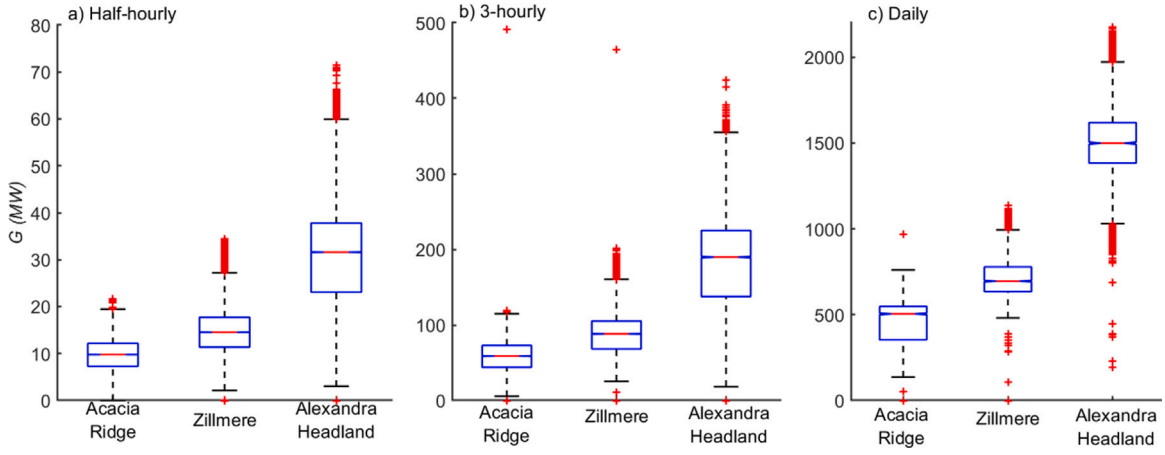


Fig. 7. Box-plot of G, MW distribution. (a) half-hourly, (b) 3-hourly, (c) daily at three study locations over 1/07/2011–30/06/2021.

ELM model based on electricity demand time series to select historically lagged combinations of input variables.

For the Acacia Ridge study site, Fig. 10 plots the performance of ELM model. Here, the transfer function was *Sigmoid* and *50* neurons were used with varying lagged data (*1-40*). Notably, the most suitable lagged combinations of input variables occur at 10, 9 and 12 for half-hourly, 3-hourly, and daily G data, respectively. Similarly, for other study site the optimal lag were *10-12-8* (Zillmere) and *12-10-10* (Alexandra Headland) for half-hourly, 3-hourly and daily G data, respectively. For TCN and benchmark models, the inputs and outputs (for 10 lag) are:

$$input = \left\{ \begin{array}{c} x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10} \\ x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10} \\ \vdots \\ x_{t-1}, x_{t-2}, \ldots, x_{t-9}, x_{t-10} \end{array} \right\}, \quad t \geq 10 \tag{11}$$

$$output = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_t \end{pmatrix}^T, \quad t \geq 10 \tag{12}$$

3.2.2. Preprocessing of electricity demand dataset

Data were segregated into train/validate/test sets to develop the model. For half-hourly and 3-hourly G prediction, six years (training) and two years (testing) were used whereas for the daily prediction, 8 years (training) and two years (testing) were considered, see Table 1. In respect to validation of model, we used one fifth of training data to search for best hyperparameters. Furthermore, to increase the optimization speed of model networks [114], the input features are normalized. In this normalization process the data were subtracted by its mean and then divided by its standard deviation, which can be represented as:

$$x_{scaled} = \frac{x - \hat{x}}{\sigma} \tag{13}$$

where x is the original value, $\hat{x}$ is the mean and σ is the standard deviation.

3.2.3. Model development and hyperparameter tuning

To develop the proposed TCN and benchmark models for G prediction at half-hourly, three-hourly and daily-step step horizons, Intel

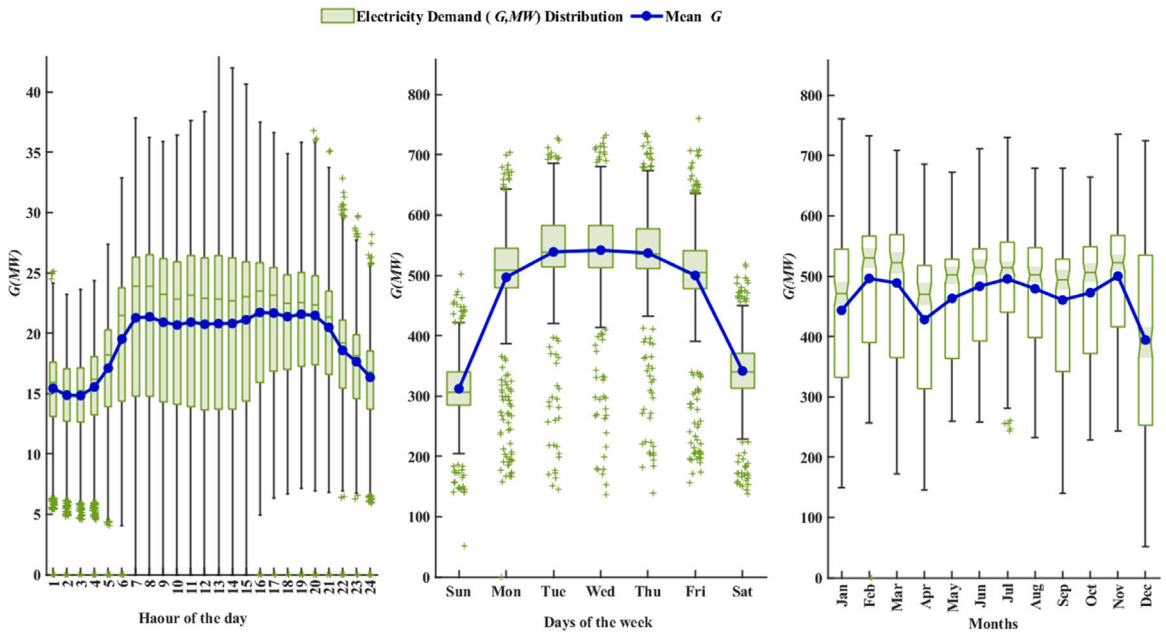


Fig. 8. Box-plot of G, MW distribution for Acacia Ridge, Queensland. (a) hour of the day, (b) days of the week, (c) monthly during the period 1/07/2011–30/06/2021.

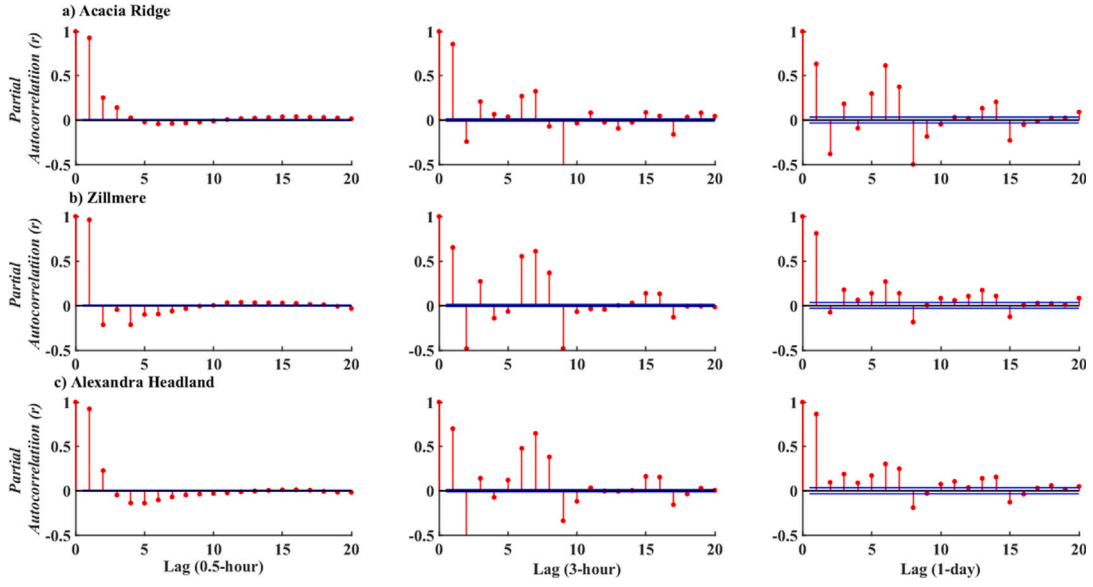


Fig. 9. Statistically significant lags of historical (G) data for three study stations.

Table 1
Training, validation and testing data for half hourly, 3-hourly and daily GH prediction.

Dataset	Training			Validation	Testing			
	Period	Data point	Interval	Percentage of train data	Period	Data point	Interval	Percentage
Half-hourly Prediction	01-Jul-2011 to 30-Jun-2017	105 212	30 min	20%	01-Jul-2017 to 30-Jun-2021	70 122	30 min	40%
3-hourly Prediction	01-Jul-2011 to 30-Jun-2017	17 526	3-h	20%	01-Jul-2017 to 30-Jun-2021	5843	3-h	25%
Daily Prediction	01-Jul-2011 to 30-Jun-2019	2915	1-day	20%	01-Jul-2019 to 30-Jun-2021	722	1-day	20%

*i*7 computer (3.3 GHz speed/32 GB memory) was used. In experimentations carried out, all of the predictive model codes were written in Python 3.7.5 (Keras and TensorFlow 2.0) libraries [115]. In this study, a univariate model was developed where the electricity demand was the only input variable (see Sections 3.2.1 and 3.2.2). The methodology is provided in Fig. 11.(a).

In detail, the model development phase is described as follows:

- Data preprocessing (normalization and data segregation) was followed by direct training of the predictive model using lagged G training data. Input data dimensions were then reshaped (without changing the values) to match the predictive model parameters.
- The training data is subdivided as multiple sets of sub-samples.
- A residual block with convolution layer was combined to create a new network structure to memorize the temporal feature of G

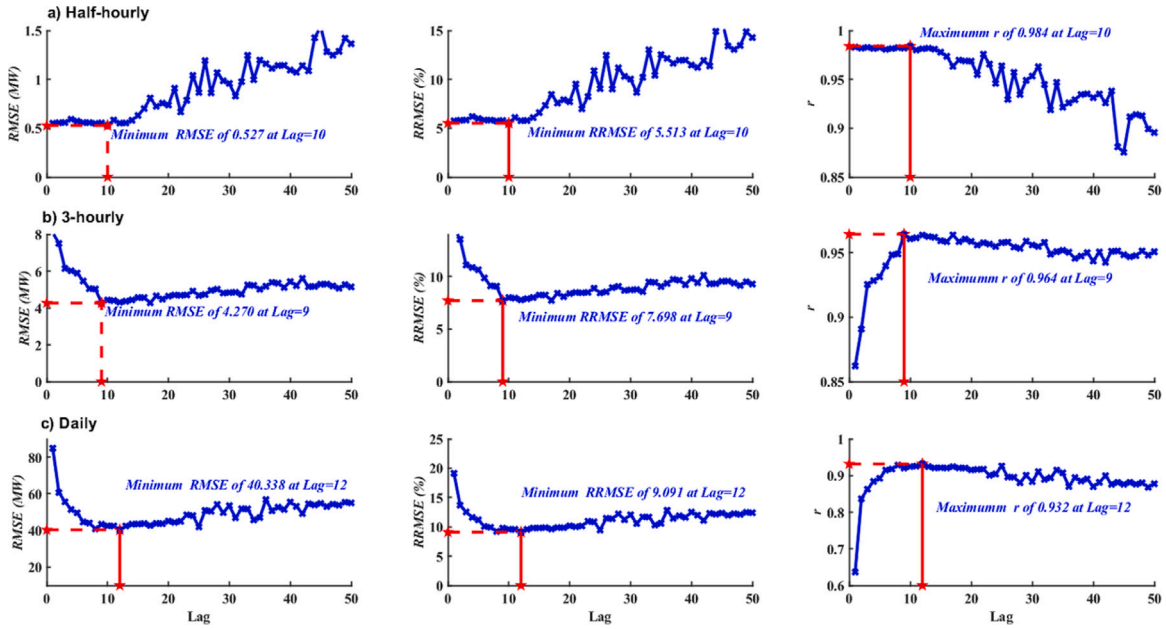


Fig. 10. For (a) half-hourly, (b) 3-hourly, and (c) daily electricity prediction, the selection of the optimal lag period based on $RMSE$ (MW), $RRMSE$ (%) and r. (Site name:- Acacia Ridge).

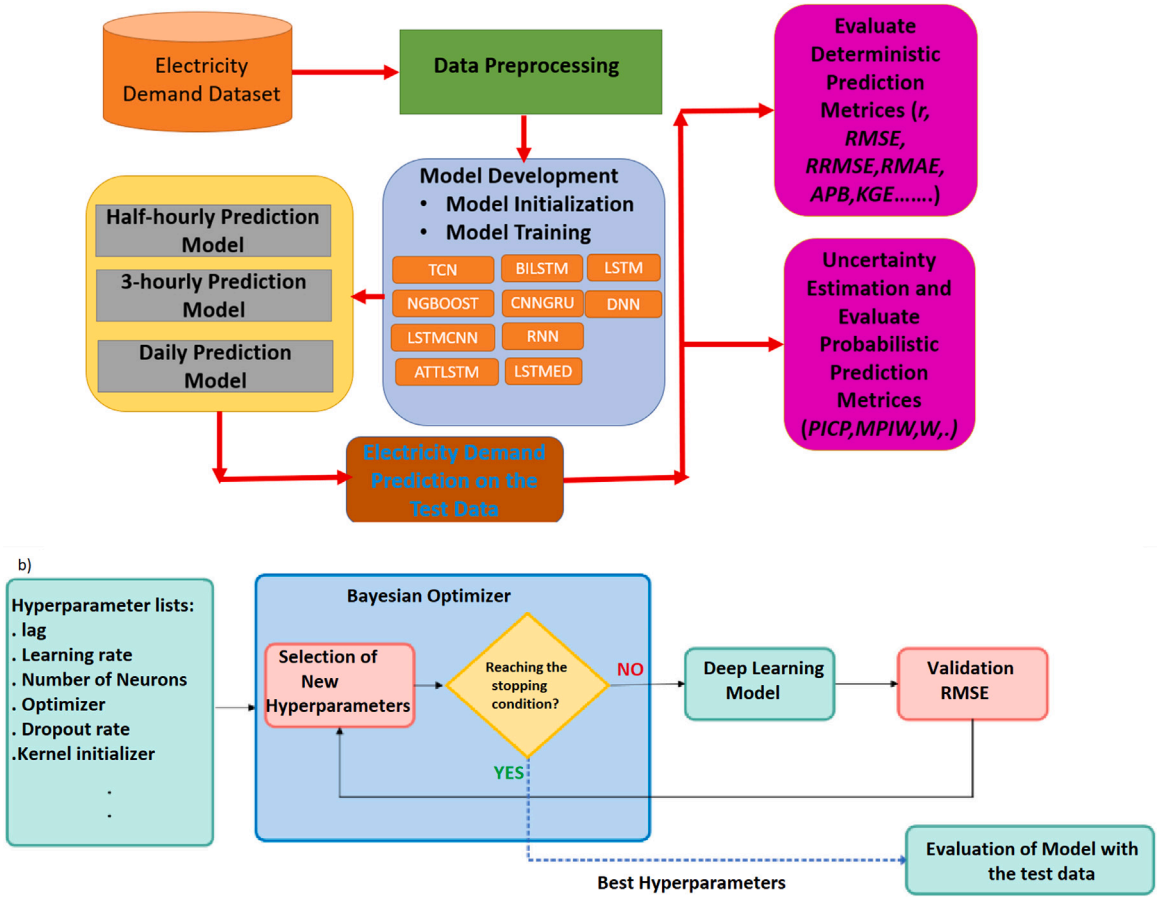


Fig. 11. (a) The schematic of G prediction model. (b) Hyperparameter optimization.

variable and further fit in and predict the non-linearity effects expected in the input G data series.

- Adaptive moment estimation (i.e., the Adam algorithm) [116] was used for network weight adjustments progressively during

feature analysis (learning rate fix to *0.001*; exponential decay ≈ *0.9* as the first moment and *0.999* as the second moment) and rectified linear unit (ReLU) [117] for activation.

- Over-fitting was prevented by using dropout rate, as per earlier works, [118] was set to be at the end of the dilated convolutions in the proposed TCN-based model, and after each layers in the other deep learning models used for comparison.
- During the predictive model training phase, an Early Stopping (*ES*) strategy was used to reduce over-fitting problems and the training time [119]. This *ES* algorithm also aimed to monitor the accuracy on the validation dataset and therefore interrupted the training phase if this metric did not improve after a set number of epochs being relatively (but not too) large.
- To speed up the performance and enable model convergence we adopted Keras ReduceLROnPlateau, which was a learning rate scheduler as a monitoring system. This reduced the learning rate in case no reduction in the loss on validation data occurred. We used a threshold of 15 epoch, a decrease factor of 10^{-1} and delta $= 10^{-4}$ in accordance with [120].

In Tables 2a–2b shows details of hyperparameters of the proposed TCN as well as the comparative models using Bayesian hyperparameter optimization algorithm (BHPO) [121,122]. By using historical data, the BHPO evaluated the performance of hyperparameter sets (Fig. 11.(b)) in such a way that a new set of hyper-parameters were selected and the process was repeated continuously until the global optimal hyperparameter was identified [123]. It should be noted that in previous studies, the BHPO method has been established as a superior approach relative to the HPO algorithms like the conventional Grid search method [124]. In contrast to the grid search method, the BHPO method deduces optimal hyperparameters by just using few iterations [125], and therefore was selected in this research.

In order to quantify the *EU*, we applied the Monte Carlo (MC) dropout (MCDO) method. In principle, the MCDO method was able to specifically make predictions with low uncertainty using a set dropout procedure. When the dropout was enabled during the testing period, the model can deliver the predictions' uncertainty by propagating the weight uncertainty into the output uncertainty. In this study, each of our model was tested on an independent (test) data repeatedly for *1000* times with the dropout being enabled sufficiently to quantify the *EU* values for *G* prediction.

Similarly, to fully capture *EU* for predicted *G*, we used the *NLL*as a loss function. Notably, a traditional deep learning network can optimize the weights to minimize mean squared error (*MSE*) in training phase but only outputs a single value ($\mu(x)$) but does not capture the predictive uncertainty. The *NLL* in these developed predictive models therefore replaced the mean squared error(*MSE*) metric and therefore was able to produce two outputs: the mean ($\hat{\mu}$) prediction and the variance ($\hat{\sigma}^2$) of the *G* prediction. The variance ($\hat{\sigma}^2$) was later utilized to create the prediction intervals and derive the probabilistic metrics of *G* forecasts produced using the proposed TCN and comparative counterpart models.

3.2.4. TCN model detail structure

TCN model is meticulously designed to capture temporal patterns in time series forecasting through its architecture and training process. The model utilizes multiple stacked TCN layers, leveraging dilated convolutions to effectively capture both short-term and long-term dependencies within the input data. This design choice allows the network to learn complex temporal patterns across various scales, which is crucial for accurate forecasting. The parameters such as the number of filters, kernel size, and dilation rates are deduced using the Bayesian hyperparameter optimization algorithm. Adaptive moment estimation (Adam) algorithm was utilized for model training, Adam algorithm adjusts the learning rate dynamically to minimize the loss function.

In terms of convergence, the model incorporates an EarlyStopping callback that halts training if the validation loss does not improve after 50 epochs, thereby preventing overfitting and conserving computational resources. Additionally, the ReduceLROnPlateau callback reduces the learning rate by a factor of 0.1 if no improvement in validation loss is observed after 15 epochs, allowing the model to converge more smoothly. The choice between probabilistic and deterministic outputs further tailors the model's predictions. When $probabilistic = True$, the model outputs both the mean and variance, which is beneficial for uncertainty quantification, providing confidence intervals around predictions. Conversely, $probabilistic = False$ yields direct forecasts, suitable for scenarios requiring deterministic predictions.

Regarding parameter settings, the model features 2 hidden layers of TCN blocks, facilitating the learning of complex features. However, this depth necessitates a balance to avoid overfitting, particularly if the dataset is small or noisy. The look-back window of 10 time steps (half-hourly) dictates that the model uses the past 10 time steps to make future predictions. Choosing the appropriate look-back window is critical, as it influences the model's ability to capture significant patterns without introducing excessive noise.

During training, the model processes the training data through forward and backward passes, where convolutional filters extract temporal features, and the Adam optimizer updates weights based on the loss (mean Squared Error) gradient. The learning rate adjustment mechanism ensures that the model fine-tunes its weights carefully as it approaches convergence. Overall, the TCN model is designed to efficiently handle sequential data, with its use of dilated convolutions and skip connections enhancing its ability to capture both local and long-term patterns. The implemented callbacks support stable and efficient training, making the model highly suitable for time series forecasting tasks.

Fig. 12 shows two plots from the training of a TCN model for *G* predictions at "Acacia Ridge". The left plot depicts the training and validation loss over epochs, where both losses decrease and stabilize, indicating effective learning (model convergence). The right plot illustrates the reduction of the learning rate over time, which is progressively lowered to fine-tune the model. These plots provide insights into the model's training and optimization process.

3.3. Performance metrics

To assess the proposed TCN model's performance quantitatively, deterministic, statistical as well as probabilistic model evaluation metrics computed with simulated *G* in the independent testing phase are now introduced.

3.3.1. Deterministic metrics

TCN and benchmark models were evaluated using the following deterministic metrics, as per earlier studies [73,126,127]:

$$r = \frac{\sum_{i=1}^{n}(G^a - \langle G^a \rangle)(G^p - \langle G^p \rangle)}{\sqrt{\sum_{i=1}^{n}(G^a - \langle G^a \rangle)^2}\sqrt{\sum_{i=1}^{n}(G^p - \langle G^p \rangle)^2}} \tag{14}$$

where r is the correlation coefficient between G^a and G^p that represent actual and predicted electricity demand and $\langle G^a \rangle$ and $\langle G^p \rangle$ represent actual and predicted mean electricity demand and N is the test points.

$$RMSE(MW) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(G^p - G^a)^2} \tag{15}$$

$$MAE(MW) = \frac{1}{N}\sum_{i=1}^{N}|G^p - G^a| \tag{16}$$

where *MAE* is the mean absolute error and *RMSE* is the root mean square error to evaluate the model using errors computed for predicted and actual electricity demand.

$$RRMSE(\%) = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(G^p - G^a)^2}}{\langle G^a \rangle} \cdot 100 \tag{17}$$

Table 2a
TCN model *vs.* LSTM, BILSTM, CNNGRU and LSTMCNN models for half-hourly G prediction. ReLU = Rectified Linear Units; Adam = adaptive moment estimation.

Predictive Models	Model Hyperparameters	Hyperparameter Selection	Acacia Ridge	Alexandra Headland	Zillmere
TCN *- proposed model*	Filters1	[50, 80, 100, 200]	100	200	80
	Filters2	[40, 50, 60, 70, 80]	50	40	100
	Dilation rate	[1, 2, 4, 6]			
	Padding	['causal']			
	Activation	['ReLU']			
	Epochs	[1000]			
	Solver	['adam']			
	Batch Size	[50]			
LSTM	LSTM units1	[100, 200, 300, 400, 500]	300	100	400
	LSTM units2	[20, 30, 40, 50, 60, 70]	20	70	20
	LSTM units3	[10, 20, 30, 40, 50]	50	30	10
	Activation	['ReLU']			
	Epochs	[1000]			
	Solver	['adam']			
	Batch Size	[50]			
BILSTM	BILSTM units1	[100, 200, 300, 400, 500]	100	300	100
	BILSTM units2	[20, 30, 40, 50, 60, 70]	40	70	60
	BILSTM units3	[10, 20, 30, 40, 50, 100]	100	50	100
	Activation	['ReLU']			
	Epochs	[1000]			
	Solver	['adam']			
	Batch Size	[50]			
CNNGRU	CNN Filter 1	[100, 200, 300, 400, 500]	400	400	100
	CNN Filter 2	[20, 30, 40, 50, 60, 70]	50	20	70
	GRU units 1	[100, 200, 300, 400, 500]	200	200	100
	GRU units 2	[20, 30, 40, 50, 60, 70]	70	60	50
	Activation	['ReLU']			
	Epochs	[1000]			
	Solver	['adam']			
	Batch Size	[50]			
LSTMCNN	LSTM units1	[100, 200, 300, 400, 500]	100	200	100
	LSTM units2	[20, 30, 40, 50, 60, 70]	50	40	70
	CNN Filter 1	[100, 200, 300, 400, 500]	100	300	100
	CNN Filter 2	[20, 30, 40, 50, 60, 70]	50	40	30
	Activation	['ReLU']			
	Epochs	[1000]			
	Solver	['adam']			
	Batch Size	[50]			

$$RMAE(\%) = \frac{1}{N}\sum_{i=1}^{N}\frac{|G^a - G^p|}{G^p}\cdot 100 \tag{18}$$

where *RRMSE* represents relative root mean square error and *RMAE* represents the root mean absolute error. In Eqs. (18)–(17), the model is evaluated using percentage errors, enabling a direct comparison of our model with a model tested elsewhere with different datasets.

$$WI = 1 - \frac{\sum_{i=1}^{N}(G^a - G^p)^2}{\sum_{i=1}^{N}\left(|G^p - \langle G^a\rangle| + |(G^o - \langle G^p\rangle|)\right)^2} \tag{19}$$

$$LM = 1 - \frac{\sum_{i=1}^{N}|G^p - G^a|}{\sum_{i=1}^{N}|G^a - \langle G^a\rangle|} \tag{20}$$

$$NS = 1 - \frac{\sum_{i=1}^{N}(G^a - G^p)^2}{\sum_{i=1}^{N}(G^a - \langle G^a\rangle)^2} \tag{21}$$

where *WI* is the Willmott's index, *NS* is the Nash–Sutcliffe equation and *LM* is the Legates and McCabe's index. In Eqs. (19), (20) and (21), a normalized and unit-less form of model evaluation is used as a universal measure of comparing the model in different datasets and geographical scenarios.

We adopted Global Performance Indicator *(GPI)* following the study of [128] which aimed to unify six different metrics.

$$GPI = \sum_{j=1}^{6}\alpha_j(g_j - y_{ij}) \tag{22}$$

where g_j is the median scaled value of jth metric, $y_{i,j}$ takes a scaled value of jth metric for i_{th} model, $\alpha_j = -1$ for $j = 4$ (r^2) and equalled 1 for the remaining statistical metrics. A relatively large value of the GPI would indicate the best predictive model.

In this study, we also adopted Kling–Gupta Efficiency following the recommendations of (KGE) [129] and the Absolute Percentage Bias ($APB;\%$) following [130] where a coefficient of variation (CV) is used.

$$KGE = 1 - \sqrt{(r-1)^2 + \left(\frac{\langle G^p\rangle}{\langle G^a\rangle} - 1\right)^2 + \left(\frac{CV_p}{CV_a}\right)^2} \tag{23}$$

$$\mathrm{APB}(\%) = \left|\frac{\sum_{i=1}^{n}(G^a - G^p)}{\sum_{i=1}^{n}G^a}\right|\cdot 100 \tag{24}$$

where r was the correlation coefficient and CV was the coefficient of variation.

In order to evaluate the performance of the TCN model statistically, we used the Diebold–Mariano (DM) test, in which a model generating the DM statistic of >0 would be superior [131–133].

3.3.2. Probabilistic metrics

A set of specific evaluation indicators for the quality of Prediction Interval (PI) was applied in this problem of electricity demand prediction. The value of PI represents an interval consisting of an upper and a lower bound covering a prescribed probability $100(1-\alpha)$ [134]. The prediction interval nominal confidence *PINC* includes α that represents the confidence interval so from the perspective of reliability and sharpness, it is essential to evaluate PI quality holistically. As a result, quality PIs will be highly reliable and sharp [135]. To determine the reliability of

Table 2b
The architecture of ATTLSTM, CLSTMED, DNN, RNN and NGBOOST models used to benchmark the proposed TCN model.

Predictive Models	Model Hyperparameters	Hyperparameter Selection	Acacia Ridge	Alexandra Headland	Zillmere
ATTLSTM	LSTM units1	[200, 150, 100, 50]	150	50	100
	LSTM units2	[100, 50, 40, 30]	100	100	30
	LSTM units3	[100, 50, 40, 30]	50	100	40
	Activation	['relu']			
	Epochs	[1000]			
	Solver	['adam']			
	Batch Size	[50]			
CLSTMED	LSTM units1	[200, 150, 100, 50]	150	100	150
	LSTM units2	[100, 50, 40, 30]	100	100	30
	CNN Filter 1	[100, 50, 40, 30]	30	50	100
	CNN Filter 2	[100, 50, 40, 30]	40	30	50
	Activation	['relu']			
	Epochs	[1000]			
	Solver	['adam']			
	Batch Size	[50]			
DNN	Hidden Neuron 1	[200, 150, 100, 50]	150	200	50
	Hidden Neuron 2	[100, 50, 40, 30]	100	40	100
	Hidden Neuron 3	[100, 50, 40, 30]	40	30	100
	Activation	['relu']			
	Epochs	[1000]			
	Solver	['adam']			
	Batch Size	[50]			
RNN	RNN units1	[200, 150, 100, 50]	50	200	150
	RNN units2	[100, 50, 40, 30]	100	50	30
	Kernel_initializer	glorot_uniform'			
	Dropout	[0.1, 0.2, 0.3]	0.1	0.1	0.2
	Activation	['relu']			
	Epochs	[1000]			
	Solver	['adam']			
	Batch Size	[50]			
NGBOOST	tree learner	Decision Tree'			
	criterion	['friedman_mse']			
	min_samples_split	[1, 2, 5, 10, 15]	2	5	5
	max_depth	[10, 20, 30, 40, 50]	20	10	40

PIs, we refer to the prediction interval coverage probability *PICP* that indicates which targets fall within the PIs:

$$PICP = \frac{1}{N}\sum_{i=1}^{N} c_i \tag{25}$$

where $if\, L\left(G_i\right) < G_i < U\left(Gi\right) \rightarrow c_i = 1; else \rightarrow c_i = 0$. $U(G_i)$ and $L(G_i)$ represent respectively the upper and lower bounds of electricity demand PIs corresponding to the ith sample and N is the number of test data.

Studies have shown, however, that *PICP* does not provide sufficient information about PI reliability [136]. Thus, the average coverage error *(ACE)* utilized to quantify reliability is:

$$ACE = PICP - PINC \tag{26}$$

It is noteworthy that when the ACE value of a PI is near zero, it is considered to be more reliable and of higher quality [137].

Furthermore, even when their*PICP* values are satisfactory, wide PIs are still meaningless as long as the sharpness is low and therefore they give little information about the variation of the target. Therefore, as a measure of sharpness, the normalized mean prediction interval width *(PINAW)* [138] is commonly used and can be defined as follows:

$$PINAW = \frac{1}{N * R}\left(\sum_{i=1}^{N}\left(U\left(G_i\right) - L\left(G_i\right)\right)\right) \tag{27}$$

where R = range of the target. In addition to the *PINAW* metrics this study also utilizes the e average relative interval length *(ARIL)* as other sharpness measurement metrics. A smaller *ARIL* value indicates a narrower band of predicted uncertainty. Therefore, the lower the value, the lower the model uncertainty [139]. The *ARIL* can be defined as:

$$ARIL = \frac{1}{N}\sum_{i=1}^{N}\frac{\left(U\left(G_i\right) - L\left(G_i\right)\right)}{G^{obs}{}_{,i}} \tag{28}$$

where $G^{obs}_{,i}$ is the ith observed value (here, actual electricity demand G at the ith time step).

In addition, we evaluated the quality and sharpness of the PIs using Winkler score (WS) [140], defined as:

$$WS = \begin{cases} \Delta_i & L\left(G_i\right) \le y_i \le U\left(G_i\right) \\ \Delta_i + 2\left(L\left(G_i\right) - y_i\right)/\alpha & y_i < L\left(G_i\right) \\ \Delta_i + 2\left(y_i - U\left(G_i\right)\right)/\alpha & y_t > U\left(G_i\right) \end{cases} \tag{29}$$

where $\Delta_i = U(G_i) - L(G_i)$ is the PI width at the ith time step. A lower *WS* is preferred when constructing PIs.

4. Results

Here, we evaluate the results obtained from the proposed TCN-based model, as well as standalone DLs (BILSTM, DNN, LSTM, RNN), as well as hybrid DLs (ATTLSTM, CLSTMED, CNNGRU, and LSTMCNN). These model evaluations are carried out after predicting the electricity demand (G) at three forecast horizons of half-hourly, three-hourly and daily time-steps cross-validated for three diverse sites (Acacia Ridge, Alexandra Head and Zillmere) in South-east Queensland, Australia. Considering all performance metrics, the TCN model is significantly better than the other methods.

Using conventional metrics such as r, *RMSE*, and*MAE*, we show in Table 3 how the proposed TCN compares with comparative models. In predicting G values, the proposed TCN model registered the highest magnitude of r at all three study sites. Interestingly, the Acacia Ridge study site recorded the same magnitude of $r \approx 0.965$ for TCN as well as the RNN model for half-hourly step. As a result, the TCN model also achieved the lowest error values at all three sites and for all three time intervals.

This suggests that TCN is superior to standalone DL and hybrid DL models yielding high r/low *RMSE*/*MAE*. The CLSTMED, compared with

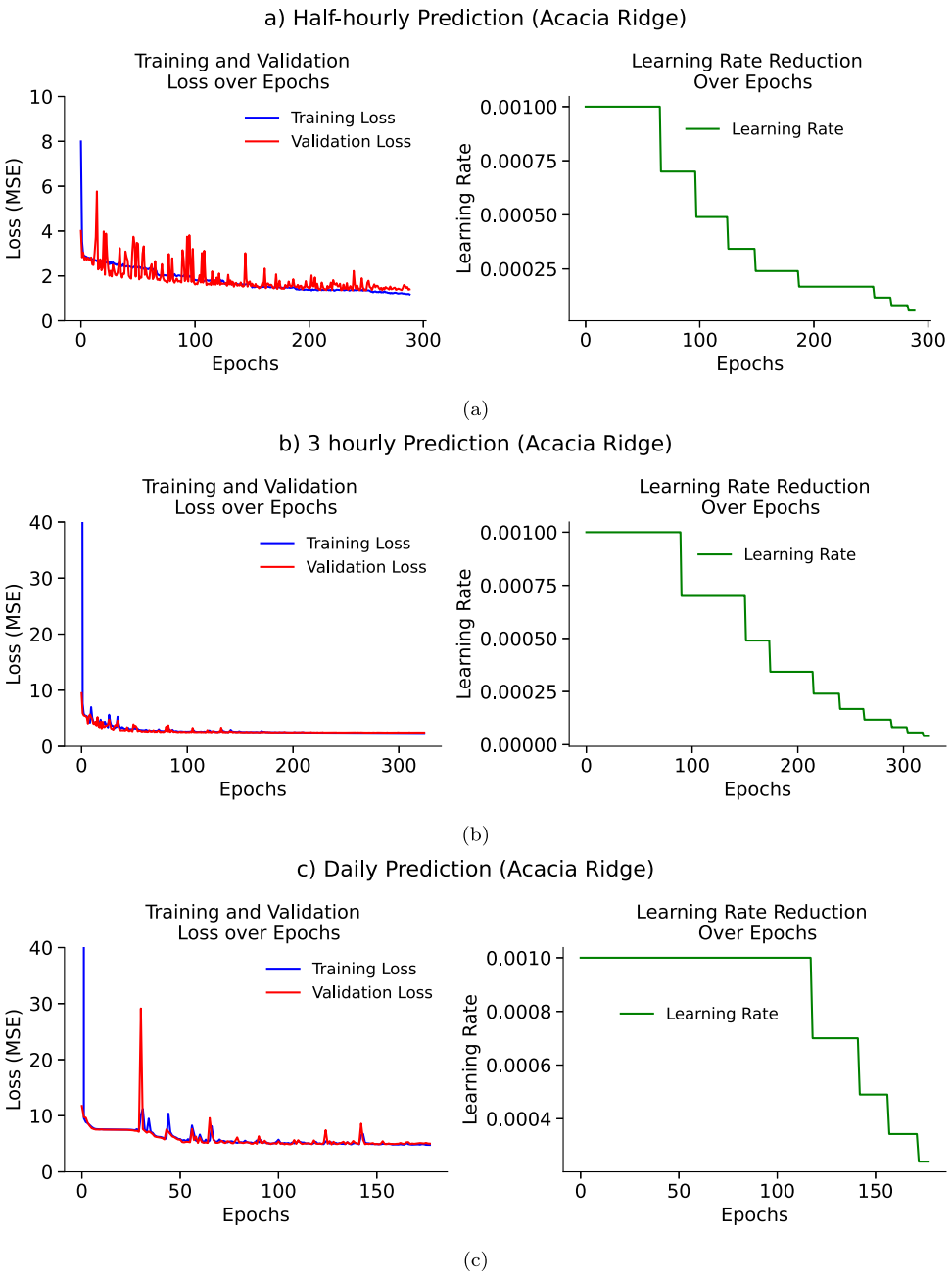


Fig. 12. TCN model convergence and the learning rate reduction plot for Acacia Ridge sub-station during training, where an epoch refers to one complete pass through the entire training dataset. During each epoch, the model processes every training sample and adjusts its internal parameters (weights and biases) to minimize the loss Mean Squared Error function.

TCN, yielded $r \approx 0.962$, $RMSE \approx 0.515$, $MAE \approx 0.387$, whereas the LSTMCNN model generated $r \approx 0.939$, $RMSE \approx 3.852$, $MAE \approx 2.817$ and the LSTM model resulted in $r \approx 0.893$, $RMSE \approx 34.197$, $MAE \approx 21.309$ for the half-hourly, three-hourly and daily G prediction steps, respectively for Acacia Ridge study site.

Based on Table 3, the TCN-based model registered highest r and lowest *RMSE* and *MAE*. However, DL hybrid models like CLSTMED, LSTMCNN and NGBOOST, as well as probabilistic-based natural gradient boosting models, have also been shown to be good predictions of G values for all of the tested time intervals at all of the study sites. Nevertheless, based on the *r*, *RMSE*, and*MAE* values alone, it may not provide the whole picture of its efficacy in predicting the electricity demand values, and therefore, we have also utilized the other model evaluation criteria.

We assessed the efficacy of TCN model based on *WI*, *NS* and *LM* (Table 4) that ideally takes a value of unity for a perfect model. Compared with the TCN-based model, the magnitude of *WI*,*NS* and *LM* were lower for all of the DL, as well as the hybrid models at all prediction time intervals. In other words, the TCN's prediction accuracy is considerably higher. In order to evaluate predictive model's performance over geographically diverse study sites, key relative percentage measures such as the *RRMSE (%)* and the *RMAE (%)* are shown in Table 5.

Consistent with *WI*, *NS* and *LM*, the lowest value of *RRMSE* and *RMAE* was achieved for the TCN model with $RRMSE \approx 5.336\%$/ $RMAE \approx 4.275\%$, $RRMSE \approx 6.684\%$/ $RMAE \approx 5.502\%$ and $RRMSE \approx 7.547\%$/ $RMAE \approx 5.243\%$ at half hourly, three hourly and daily prediction, respectively for Acacia Ridge study site. A similar trend was consistently demonstrated by this objective model at Alexandra Head and Zillmere study sites.

Table 3
Using Eqs. (14)–(16), we evaluate TCN *vs.* benchmark models via deterministic metrics: r, $RMSE$, and MAE. Boldface indicates the best model according to performance metrics.

Predictive Models	Prediction Time-Step	Acacia Ridge			Alexandra Head			Zillmere		
		r	$RMSE$ (MW)	MAE (MW)	r	$RMSE$ (MW)	MAE (MW)	r	$RMSE$ (MW)	MAE (MW)
TCN	Half-hourly	**0.965**	**0.494**	**0.369**	**0.952**	**1.800**	**1.388**	**0.952**	**0.943**	**0.588**
	3-hourly	**0.942**	**3.736**	**2.766**	**0.940**	**11.672**	**8.363**	**0.890**	**8.541**	**5.856**
	Daily	**0.899**	**33.204**	**21.035**	**0.893**	**80.531**	**59.420**	**0.745**	**52.331**	**39.593**
ATTLSTM	Half-hourly	0.945	0.618	0.479	0.899	3.057	2.077	0.917	1.232	0.855
	3-hourly	0.919	4.427	3.184	0.925	13.071	9.908	0.888	8.647	5.931
	Daily	0.825	43.780	29.271	0.866	90.483	69.923	0.722	54.431	42.036
BILSTM	Half-hourly	0.941	0.641	0.500	0.905	2.524	1.991	0.922	1.256	0.911
	3-hourly	0.931	4.090	2.942	0.922	13.309	10.273	0.885	8.745	6.168
	Daily	0.845	42.328	24.473	0.811	107.335	85.123	0.715	55.258	40.833
CLSTMED	Half-hourly	0.962	0.515	0.387	0.900	2.597	2.029	0.940	1.104	0.781
	3-hourly	0.935	3.979	2.896	0.934	12.235	9.011	0.887	8.655	6.149
	Daily	0.874	37.192	23.807	0.880	85.505	65.226	0.734	53.425	40.203
CNNGRU	Half-hourly	0.946	0.617	0.476	0.903	2.552	1.991	0.919	1.281	0.920
	3-hourly	0.933	4.037	2.906	0.916	13.779	10.297	0.844	10.189	7.150
	Daily	0.828	43.329	29.199	0.867	90.147	71.224	0.729	53.972	40.531
DNN	Half-hourly	0.936	0.670	0.522	0.913	2.423	1.935	0.920	1.272	0.932
	3-hourly	0.834	6.341	4.532	0.918	13.686	10.066	0.869	9.323	6.519
	Daily	0.794	48.813	30.173	0.688	137.862	108.908	0.690	57.679	42.575
LSTMCNN	Half-hourly	0.950	0.590	0.457	0.899	2.598	2.061	0.933	1.165	0.835
	3-hourly	0.939	3.852	2.817	0.927	12.906	9.718	0.888	8.626	6.103
	Daily	0.858	39.408	25.025	0.868	89.633	69.277	0.731	53.749	40.518
LSTM	Half-hourly	0.950	0.495	0.371	0.938	2.032	1.612	0.948	1.028	0.716
	3-hourly	0.934	3.997	2.899	0.914	14.008	10.519	0.885	8.757	6.288
	Daily	0.893	34.197	21.309	**0.893**	80.548	**59.354**	0.743	52.539	**39.551**
NGBOOST	Half-hourly	0.958	0.540	0.391	0.938	2.047	1.680	0.950	1.004	0.726
	3-hourly	0.914	4.571	3.545	0.823	20.051	12.864	0.821	10.896	8.078
	Daily	0.873	37.259	24.115	0.871	88.748	67.297	0.731	53.695	40.255
RNN	Half-hourly	**0.965**	0.498	0.374	0.935	2.446	1.642	0.885	1.455	1.048
	3-hourly	0.920	4.392	3.152	0.928	12.819	9.405	0.868	9.376	6.621
	Daily	0.889	34.782	22.277	0.886	83.486	61.858	0.739	52.926	40.103

According to Table 3, 4 and 5, both the dimensional and non-dimensional metrics corroborate the suitability of the proposed TCN model for G prediction. Additionally, to validate the proposed methods' predictive skill for G prediction, the APB and the KGE metrics were also used. These results, as per Table 6, advocate TCN model as consistently outperforming DL models with varying magnitudes, both standalone and hybrid, such that the KGE was higher and the APB was lower for all prediction time intervals at all sites. Accordingly, the proposed TCN model provides a high level of accuracy in G prediction tasks.

To assess the capability of the standalone DL as well as the hybrid models for G prediction in a test period, further analyses of the absolute prediction error ($|PE|$) was conducted. Note that $|PE| = |predicted - actual\mathrm{G}|$ so ideally, the $|PE|$ should be 0 for the best model. Also, on an overall basis, this metric is likely to have a higher frequency of $|PE|$ values that are closer to 0.

For the Alexandra Headland study site, Fig. 13 plots the frequency distribution of $|PE|$ computed in a number of error brackets.

In general, for the half-hourly G prediction interval, 20%–26% of $|PE|$ for all models lie in less than $0.05MW$ error bar except for the case of ATTLSTM, BILSTM, NGBOOST and RNN where they are 18%, 18%, 17% and 19%, respectively. Importantly, the TCN model has $|PE|$ (26%) in the first bin ($|PE| < 0.05MW$) whereas the hybrid LSTMCNN as 23%, the CLSTMED and CNNGRU have 21% and the standalone DNN and LSTM models have 20%. Similarly for the three-hourly and daily G prediction intervals, we saw about 38% and 39% of $|PE|$ values registered for the case of the proposed TCN model in less than $6.0MW$ and $40.0MW$, respectively. These results validate the TCN model for G prediction at short- (half-hourly, 3-hourly) and long-term (daily) periods.

We now examine GPI as a rank metric where any model with highest GPI is the most accurate. As per Fig. 14, the TCN-based model has the best performance with $GPI \approx 1.592$, ≈ 1.388, ≈ 0.666 for the half-hourly, three-hourly and the daily prediction horizons for Alexandra Headland. Likewise, for Acacia ridge study site, $GPI \approx 3.1308$ (half-hourly), ≈ 1.104 (three-hourly, ≈ 0.8837 (daily), and for the Zillmere study site, it was $GPI \approx 7.6623, \approx 1.7132, \approx 0.1866$, respectively. These results show the TCN as best model with a high GPI value compared to standalone/hybrid DL models.

To accord with Section 3.3.1, we conducted one-sided DM tests to determine statistically the model performance. Table 7 shows results based on the DM test. The objective model performs better than the comparative model when it has a positive DM value, and the lower the DM value, the better it performs. In accordance with Table 7, the DM statistic value is always positive for the case of the TCN model tested at all time intervals. Also, this would demonstrate that the TCN model predicted the G value significantly better than the other models.

As per Section 3.3.2 we now quantify the uncertainties inherent in G data (i.e., aleatoric, AU) by utilizing a maximum likelihood method. The accuracy of any DL model could also dependent on actual amount of data available, thus creating epistemic uncertainty (EU) which in this study are captured by a Monte Carlo Dropout (MCDO) method. Fig. 15 shows the relationship between the uncertainties and absolute value of prediction error ($|PE|$) for a DL as well as hybrid models emulated in the testing phase for Alexandra Headlands study site.

Evidently, Fig. 15 shows that the total uncertainty as well as the values of AU and EU are very low for the TCN model compared with the other comparative models used in this G prediction problem. Additionally, we investigate the relation between Root Mean Square Error ($RMSE, MW$) and corresponding predicted total uncertainty (TU) (sum of AU and EU) in Fig. 16) presented for three time intervals. Notably, the results show that the BILSTM, CNNGRU, LSTMCNN and the NGBOOST models have registered large total uncertainties in G prediction.

Table 4
Evaluation of TCN *vs.* benchmark models using WI, NS, and LM. The best model based on performance metrics is boldfaced.

Prediction Models	Prediction Time-Steps	Acacia Ridge			Alexandra Head			Zillmere		
		WI	LM	NS	WI	LM	NS	WI	LM	NS
TCN	Half-hourly	**0.991**	**0.832**	**0.965**	**0.988**	**0.793**	**0.949**	**0.988**	**0.830**	**0.950**
	3- hourly	**0.985**	**0.786**	**0.942**	**0.984**	**0.775**	**0.933**	**0.970**	**0.702**	**0.873**
	Daily	**0.973**	**0.755**	**0.888**	**0.972**	**0.714**	**0.888**	**0.924**	**0.454**	**0.670**
ATTLSTM	Half-hourly	0.986	0.783	0.945	0.972	0.719	0.882	0.979	0.757	0.919
	3- hourly	0.979	0.748	0.914	0.980	0.729	0.913	0.970	0.704	0.874
	Daily	0.951	0.647	0.795	0.960	0.615	0.817	0.900	0.290	0.447
BILSTM	Half-hourly	0.985	0.769	0.938	0.975	0.699	0.895	0.979	0.745	0.913
	3- hourly	0.982	0.768	0.928	0.978	0.716	0.907	0.969	0.691	0.874
	Daily	0.959	0.730	0.835	0.940	0.492	0.698	0.914	0.429	0.621
CLSTMED	Half-hourly	0.990	0.825	0.962	0.974	0.691	0.891	0.984	0.788	0.937
	3- hourly	0.983	0.771	0.932	0.982	0.761	0.927	0.970	0.692	0.875
	Daily	0.967	0.734	0.867	0.966	0.656	0.848	0.921	0.449	0.661
CNNGRU	Half-hourly	0.986	0.785	0.946	0.975	0.704	0.897	0.978	0.741	0.908
	3- hourly	0.983	0.774	0.931	0.978	0.724	0.907	0.954	0.630	0.794
	Daily	0.945	0.590	0.723	0.961	0.611	0.821	0.919	0.441	0.650
DNN	Half-hourly	0.983	0.755	0.931	0.976	0.696	0.900	0.979	0.745	0.916
	3- hourly	0.951	0.596	0.783	0.978	0.731	0.906	0.965	0.675	0.854
	Daily	0.940	0.629	0.738	0.886	0.204	0.281	0.895	0.372	0.483
LSTMCNN	Half-hourly	0.987	0.787	0.948	0.973	0.685	0.891	0.983	0.774	0.932
	3- hourly	0.984	0.782	0.938	0.980	0.736	0.916	0.970	0.693	0.874
	Daily	0.963	0.724	0.854	0.963	0.629	0.839	0.913	0.412	0.591
LSTM	Half-hourly	0.991	0.831	0.965	0.984	0.753	0.933	0.987	0.806	0.947
	3- hourly	0.983	0.773	0.931	0.976	0.710	0.896	0.969	0.684	0.870
	Daily	0.972	0.761	0.888	0.971	0.701	0.878	0.922	0.452	0.655
NGBOOST	Half-hourly	0.989	0.812	0.953	0.982	0.716	0.917	0.987	0.792	0.943
	3- hourly	0.976	0.702	0.897	0.957	0.686	0.845	0.939	0.493	0.672
	Daily	0.966	0.720	0.858	0.963	0.649	0.833	0.920	0.447	0.658
RNN	Half-hourly	0.991	0.827	0.964	0.983	0.779	0.926	0.969	0.695	0.877
	3- hourly	0.979	0.752	0.917	0.981	0.758	0.923	0.963	0.653	0.840
	Daily	0.971	0.749	0.883	0.969	0.692	0.869	0.924	0.462	0.680

Table 5
Comparing the TCN with benchmark models using $RRMSE$ and $RMAE$ (%) with best model indicated in boldface.

Prediction Models	Prediction Time-Steps	Acacia Ridge		Alexandra Head		Zillmere	
		$RRMSE$	$RMAE$	$RRMSE$	$RMAE$	$RRMSE$	$RMAE$
TCN	Half-hourly	**5.336**	**4.275**	**6.676**	**5.236**	**6.485**	**4.273**
	3-hourly	**6.684**	**5.502**	**7.291**	**5.314**	**9.900**	**7.038**
	Daily	**7.547**	**5.243**	**6.320**	**4.720**	**7.517**	**5.569**
ATTLSTM	Half-hourly	6.658	5.628	10.099	8.115	8.428	6.284
	3-hourly	7.959	6.312	8.034	6.415	9.948	7.167
	Daily	10.062	7.373	7.011	5.614	7.782	5.927
BILSTM	Half-hourly	6.873	5.909	9.237	7.431	8.682	6.588
	3-hourly	7.377	5.841	8.255	6.776	10.052	7.690
	Daily	10.261	6.559	8.172	6.849	8.090	5.860
CLSTMED	Half-hourly	5.582	4.510	9.442	7.461	7.604	5.608
	3-hourly	7.171	5.743	7.640	5.829	9.837	7.305
	Daily	8.303	5.792	6.668	5.172	7.643	5.627
CNNGRU	Half-hourly	6.647	5.499	9.227	7.283	8.862	6.540
	3-hourly	7.257	6.007	8.559	6.724	11.635	8.416
	Daily	9.738	7.261	6.936	5.674	7.741	5.685
DNN	Half-hourly	7.237	6.067	8.869	7.306	8.742	6.682
	3-hourly	11.606	9.151	8.523	6.405	10.694	7.812
	Daily	11.705	8.282	10.318	8.629	8.332	5.981
LSTMCNN	Half-hourly	6.404	5.327	9.616	7.632	8.020	6.040
	3-hourly	6.904	6.910	8.048	6.332	9.888	7.309
	Daily	8.873	6.202	6.971	5.585	7.721	5.681
LSTM	Half-hourly	5.373	4.329	7.456	6.199	7.046	5.157
	3-hourly	7.229	5.772	8.688	6.797	10.046	7.521
	Daily	7.680	5.287	6.330	4.722	7.552	5.562
NGBOOST	Half-hourly	5.830	4.688	7.447	6.907	6.899	5.231
	3-hourly	8.207	7.101	12.648	224.312	12.517	9.394
	Daily	8.493	6.161	6.875	5.303	7.671	5.627
RNN	Half-hourly	5.407	4.350	8.052	5.588	9.913	7.396
	3-hourly	7.920	6.356	7.940	5.913	10.878	8.166
	Daily	7.805	5.460	6.590	4.898	7.566	5.609

Table 6
Comparing the TCN with benchmark models based on relative APB (%) and KGE values, as per Eqs. (23) and (24). The best model as measured by performance metrics is boldfaced.

Prediction Models	Prediction Time-Steps	Acacia Ridge		Alexandra Head		Zillmere	
		APB	*KGE*	*APB*	*KGE*	*APB*	*KGE*
TCN	Half-hourly	**3.988**	**0.982**	**5.148**	**0.960**	**4.045**	**0.971**
	3- hourly	**4.949**	**0.969**	**5.225**	**0.934**	**6.788**	**0.904**
	Daily	**4.781**	**0.924**	**4.664**	**0.940**	**5.687**	**0.806**
ATTLSTM	Half-hourly	5.160	0.971	6.862	0.904	5.853	0.958
	3-hourly	5.724	0.948	6.090	0.912	6.824	0.917
	Daily	6.728	0.878	5.418	0.822	6.010	0.568
BILSTM	Half-hourly	5.356	0.957	7.288	0.926	6.296	0.934
	3-hourly	5.307	0.959	6.372	0.900	7.090	0.926
	Daily	5.933	0.915	6.481	0.726	5.978	0.788
CLSTMED	Half-hourly	4.195	0.981	7.376	0.929	5.382	0.964
	3-hourly	5.219	0.960	5.627	0.936	6.989	0.921
	Daily	5.315	0.931	5.086	0.860	5.751	0.807
CNNGRU	Half-hourly	5.134	0.972	7.200	0.934	6.365	0.922
	3-hourly	5.223	0.964	6.397	0.928	8.164	0.832
	Daily	6.562	0.718	5.480	0.831	5.813	0.801
DNN	Half-hourly	5.639	0.951	7.084	0.914	6.403	0.954
	3- hourly	8.295	0.834	6.269	0.921	7.478	0.911
	Daily	7.235	0.833	8.151	0.473	6.150	0.665
LSTMCNN	Half-hourly	4.959	0.965	7.627	0.934	5.751	0.965
	3-hourly	5.049	0.967	6.060	0.917	6.996	0.915
	Daily	5.635	0.927	5.387	0.877	5.820	0.727
LSTM	Half-hourly	4.030	0.982	5.915	0.938	4.910	0.971
	3-hourly	5.243	0.962	6.524	0.895	7.213	0.914
	Daily	4.786	0.941	4.665	0.910	5.685	0.790
NGBOOST	Half-hourly	4.214	0.937	6.113	0.839	4.994	0.924
	3-hourly	6.364	0.899	8.114	0.898	9.281	0.637
	Daily	5.497	0.914	5.214	0.850	5.751	0.808
RNN	Half-hourly	4.061	0.980	5.405	0.926	7.135	0.932
	3-hourly	5.685	0.955	5.825	0.949	7.681	0.878
	Daily	4.999	0.936	4.883	0.910	5.733	0.825

Table 7
Statistical evaluation of TCN model using the DM test statistics [131–133]. (a) Half-hourly time interval, (b) 3-hourly time interval, (c) daily interval. Positive results mean the rows outperform the columns; negative results mean the columns outperform the rows. The best model is boldfaced.

Predictive Model	TCN	ATTLSTM	BILSTM	CLSTMED	CNNGRU	DNN	LSTMCNN	LSTM	NGBOOST	RNN
TCN		**44.2945**	**48.7922**	**50.5816**	**50.7311**	**48.2189**	**39.9685**	**26.263**	**13.2494**	**49.2256**
ATTLSTM			−15.3183	−5.8697	−12.771	−19.3286	−2.3568	−52.1467	−35.6315	−15.8704
BILSTM				18.7494	4.9653	−18.757	4.8775	−48.8626	−25.3021	−10.1799
CLSTMED					−6.8644	−40.1081	0.0982	−49.4376	−26.8039	−36.763
CNNGRU						−15.3963	2.5347	−55.6926	−30.8764	−9.6505
DNN							15.1249	−37.6455	−18.1757	44.1264
LSTMCNN								−26.9283	−18.0497	−10.0644
LSTM									1.1029	41.8902
NGBOOST										20.9187

Predictive Model	TCN	ATTLSTM	BILSTM	CLSTMED	CNNGRU	DNN	LSTMCNN	LSTM	NGBOOST	RNN
TCN		**9.8657**	**15.6362**	**7.0274**	**10.8369**	**10.9877**	**10.4793**	**13.4972**	**10.0439**	**9.1939**
ATTLSTM			2.4529	−7.0587	4.9548	2.8887	−1.2271	4.8235	9.3164	−1.934
BILSTM				−11.8781	2.9575	2.4113	−5.0138	5.0699	8.8253	−4.7182
CLSTMED					8.8174	7.5933	6.1837	9.781	9.6861	5.5906
CNNGRU						−0.329	−4.3733	0.9401	8.9788	−5.3881
DNN							−7.2157	9.1168	7.7455	−6.5546
LSTMCNN								11.4364	8.8694	−1.0993
LSTM									7.5866	−9.2594
NGBOOST										−9.0346

Predictive Model	TCN	ATTLSTM	BILSTM	CLSTMED	CNNGRU	DNN	LSTMCNN	LSTM	NGBOOST	RNN
TCN		**2.306**	**1.9937**	**4.639**	**3.5506**	**3.8091**	**2.3832**	**3.7991**	**4.5595**	**3.4825**
ATTLSTM			−1.0582	2.8873	1.8327	0.4815	0.0246	2.3359	4.1636	2.918
BILSTM				3.7438	2.5287	1.6849	1.7008	2.9463	4.4622	3.2619
CLSTMED					−1.4373	−2.7888	−3.3068	−1.5113	4.2398	2.551
CNNGRU						−1.5848	−1.3986	0.6958	4.0839	2.6307
DNN							−0.241	2.1957	4.2164	2.9434
LSTMCNN								2.5048	4.6303	3.4161
LSTM									4.3489	3.0493
NGBOOST										−4.5573

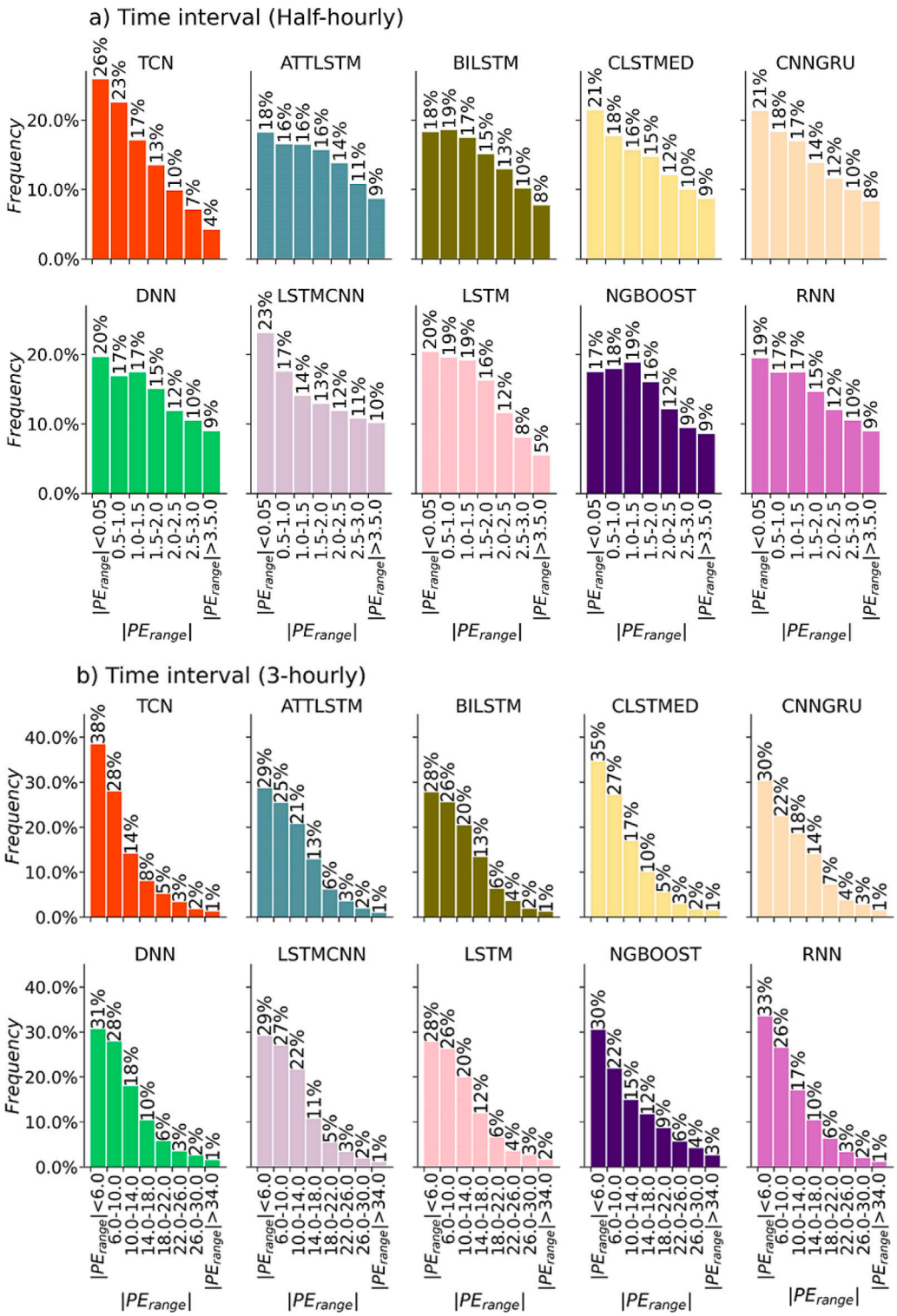


Fig. 13. Cumulative frequency of absolute prediction error($|PE|$) for Alexandra Headland study site. Each error bracket is represented by a bar that indicates the percentage of errors accumulated. Please refer to Table 2 for the names of the models.

This large uncertainty appears to be a result of epistemic uncertainty, as illustrated in Fig. 15, indicating that the model cannot capture G time series patterns during its training phase. With this improvement, we can predict G more accurately than existing machine learning and deep learning models. In the highly competitive current electricity market, even small changes in accuracy of G predictions affect ta company's operating costs and risk strategies, making the proposed method a valuable addition to the proposed method, which is very useful for companies in the electricity market.

Furthermore, we have adopted the evaluation metrics for the proposed model, the *PICP*, the *PINAW*, the *WS*, the *ARIL*, and the *ACE* evaluation metrics (see Section 3.3.2) as well as the prediction interval (PI) based on 95% confidence and $\alpha = 0.05$ significance level. Table 8 provides the prediction results using probabilistic metrics

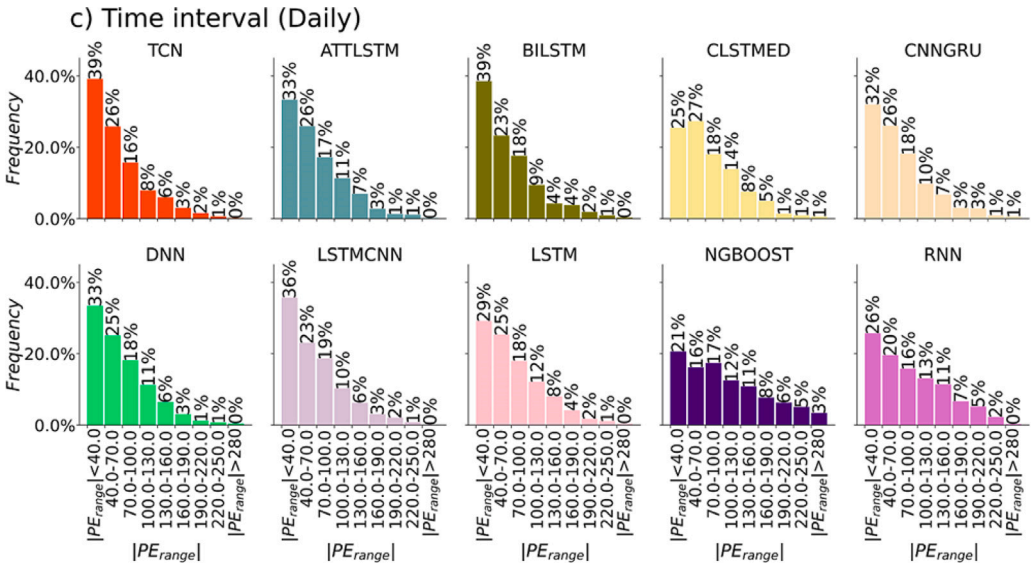


Fig. 13. (*continued*).

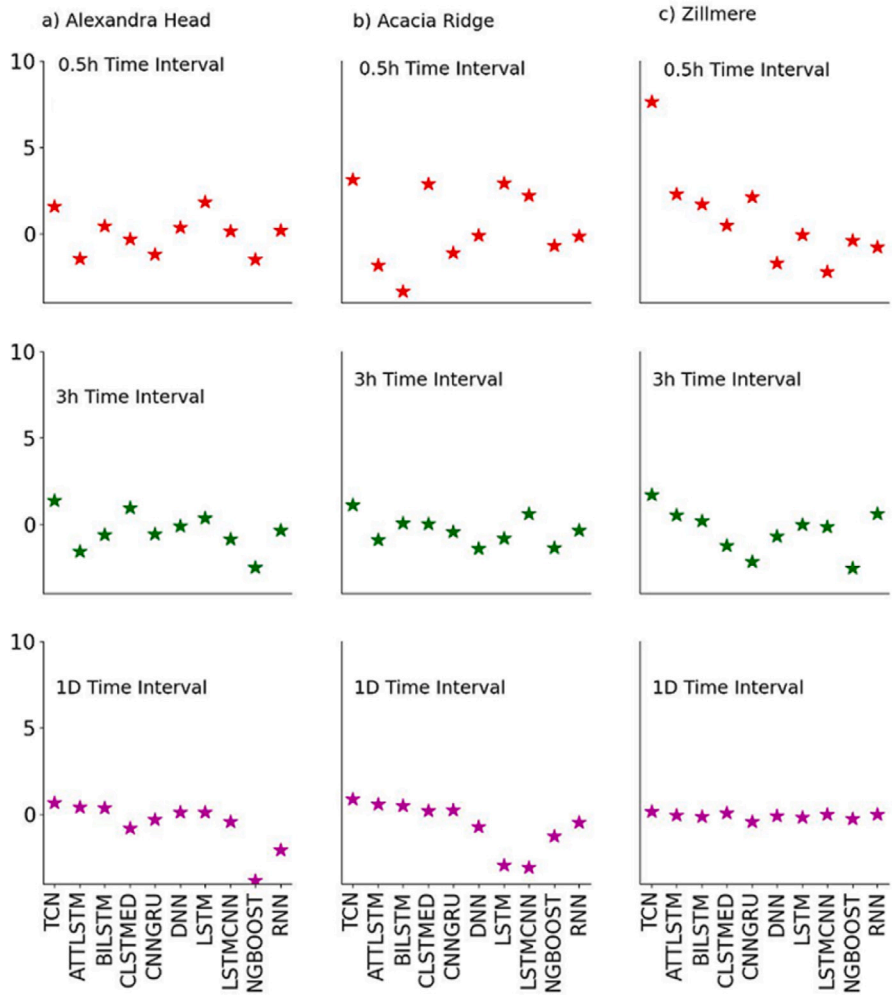


Fig. 14. Global performance indicator (GPI) for TCN *vs.* benchmark models. (a) Acacia Ridge, (b) Zillmere, and (c) Alexandra Headland. See Table 2 for model names.

In accordance with interpret Table 8, the lowest $PINAW$ and the highest $PICP$ computed at the nominated confidence interval of 95% is expected for the best predictive model. Evidently, for all of the proposed study sites, the $PICP$ values were considerably greater at the 95% interval which indicates that all of the standalone and hybrid models were significantly capable of capturing the observed G values. A primary reason for this could be attributable to the G time series dataset that were relatively stable. Furthermore, the width of the PI

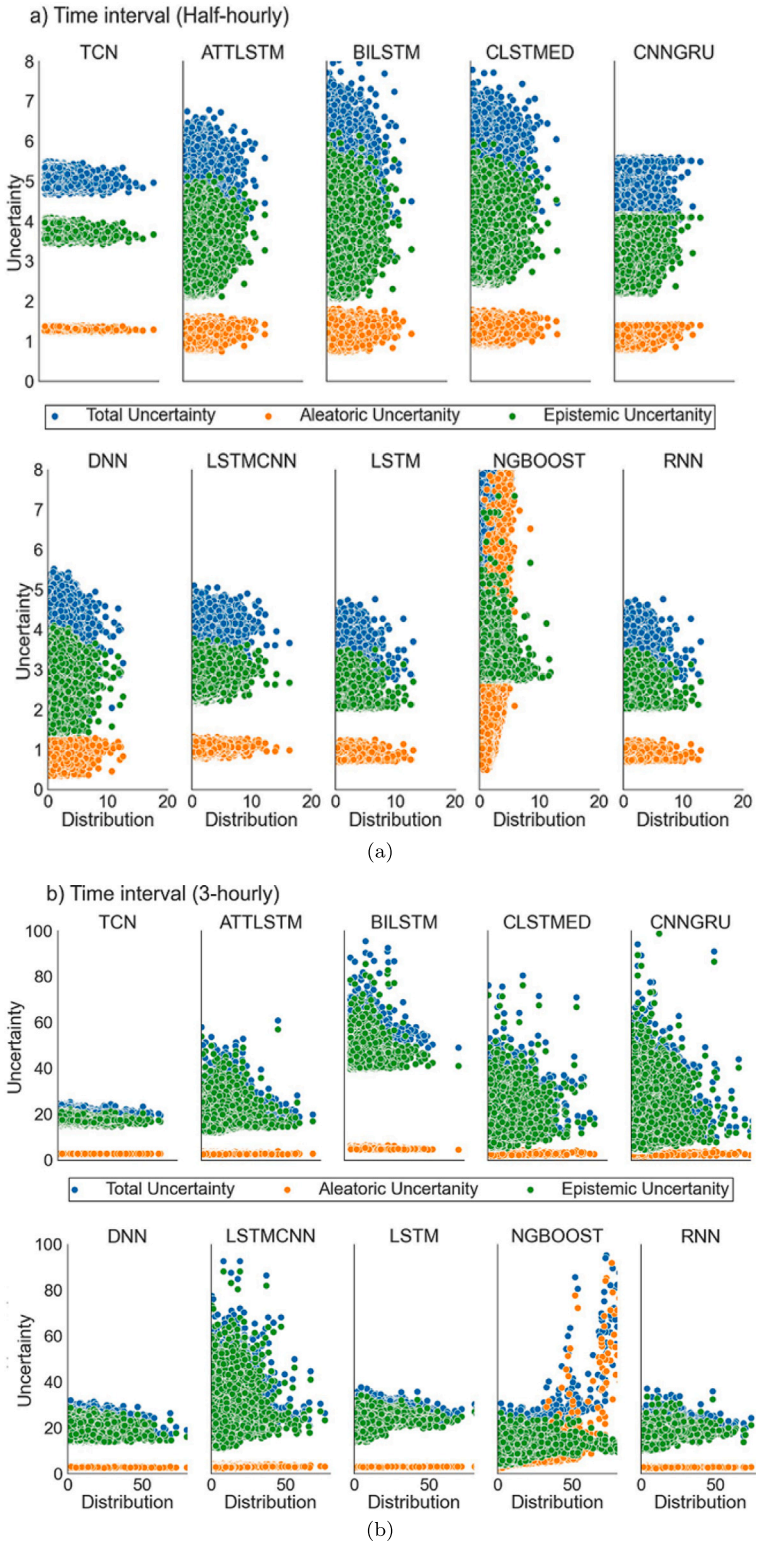
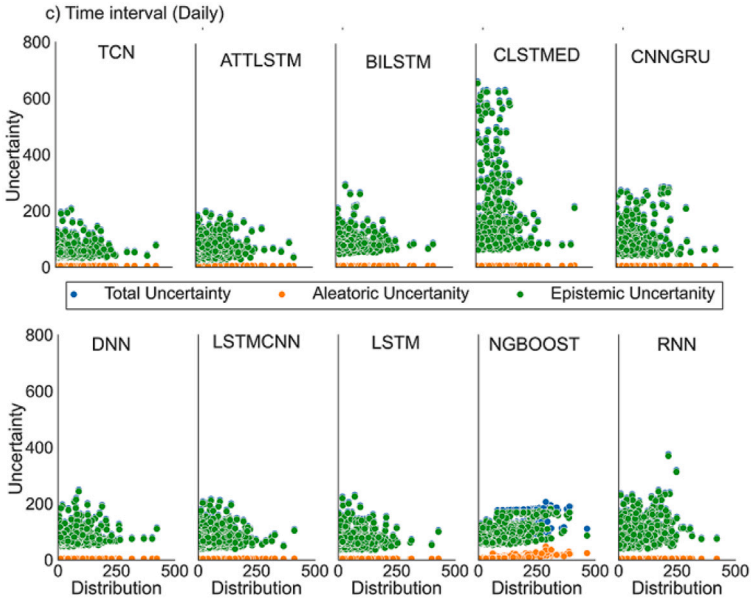


Fig. 15. Scatter-plot showing the relationship between aleatoric (AU) and epistemic (EU) uncertainties including prediction error ($|PE|$) distribution for short-term periods (a) Half-hourly and (b) 3-hourly as well as long-term period (c) daily electricity demand prediction intervals. (test site: Alexandra Headland).

(*PINAW*) is greater if the *PICP* exceeds the preset *PINC*. Both the *PICP* and the *PINAW* metrics behave in a somewhat contrariwise way. This is because, as the generated PI width is increased, the $PICP$ values would also increase.

Thus, the simulation results provided in Table 8 reaffirmed the finding that some predictive models actually had a higher coverage probability, albeit at a wider interval range. At the Acacia Ridge study site, we note that the DNN model tested at half hourly G prediction produced $PICP \approx 1.0$ with $PINAW \approx 0.951$, and a likewise result was obtained at the Alexandra Headland for LSTM model tested at three-hourly G prediction resulting in $PICP \approx 0.999$ with $PINAW \approx 0.343$. For Zillmere study site, RNN for G prediction produced $PICP \approx 0.825$



(c)

Fig. 15. (*continued*).

Table 8
Probabilistic metrics of TCN using relative PICP and PINAW values, as per Eqs. (25) and (27). It is expected that the most accurate model (boldface) will record the lowest PINAW and the highest $PICP$ value.

Predictive Models	Prediction Time-Step	Acacia Ridge		Alexandra Head		Zillmere	
		PICP	*PINAW*	*PICP*	*PINAW*	*PICP*	*PINAW*
TCN	Half-hourly	**0.990**	**0.937**	**0.996**	**0.250**	**0.971**	**0.031**
	3- hourly	**0.963**	**0.185**	**0.985**	**0.243**	**0.898**	**0.160**
	Daily	**0.924**	**4.781**	**0.940**	**4.664**	**0.806**	**5.687**
ATTLSTM	Half-hourly	0.986	0.971	0.989	0.336	0.961	0.040
	3- hourly	0.964	0.227	0.988	0.300	0.944	0.208
	Daily	0.878	6.728	0.822	5.418	0.568	6.010
BILSTM	Half-hourly	0.970	0.957	0.997	0.330	0.925	0.158
	3- hourly	0.958	0.196	0.989	0.284	0.940	0.205
	Daily	0.915	5.933	0.726	6.481	0.788	5.978
CLSTMED	Half-hourly	0.996	0.981	0.998	0.346	0.929	0.127
	3- hourly	0.959	0.192	0.991	0.292	0.934	0.194
	Daily	0.931	5.315	0.860	5.086	0.807	5.751
CNNGRU	Half-hourly	0.992	0.972	0.995	0.320	0.940	0.170
	3- hourly	0.959	0.193	0.974	0.254	0.918	0.209
	Daily	0.718	6.562	0.831	5.480	0.801	5.813
DNN	Half-hourly	1.000	0.951	1.000	0.396	0.952	0.161
	3- hourly	0.982	0.379	0.984	0.295	0.956	0.236
	Daily	0.833	7.235	0.473	8.151	0.665	6.150
LSTMCNN	Half-hourly	0.997	0.965	0.995	0.278	0.914	0.141
	3- hourly	0.957	0.195	0.995	0.410	0.937	0.194
	Daily	0.927	5.635	0.877	5.387	0.727	5.820
LSTM	Half-hourly	0.995	0.982	0.999	0.343	0.943	0.121
	3- hourly	0.960	0.195	0.995	0.390	0.954	0.219
	Daily	0.941	4.786	0.910	4.665	0.790	5.685
NGBOOST	Half-hourly	0.970	0.917	at a	0.401	1.000	0.296
	3- hourly	0.996	0.316	0.935	0.258	0.988	0.335
	Daily	0.914	5.497	0.850	5.214	0.808	5.751
RNN	Half-hourly	0.992	0.980	0.999	0.323	0.972	0.052
	3- hourly	0.972	0.236	0.988	0.288	0.930	0.205
	Daily	0.936	4.999	0.910	4.883	0.825	5.733

with $PINAW \approx 5.733$. Nonetheless, in comparison with the other standalone as well as hybrid models, the newly proposed TCN model produced the narrowest PI values for all of the study sites tested at all three time intervals. It can be deduced that TCN model is more likely to produce the PIs that have a higher accuracy relative to the other models.

In a somewhat different manner compared to *PICP* and *PINAW*, the *WS*,*ARIL* and *ACE* metrics can simultaneously consider the PIs sharpness value, width of the uncertainty bounds and the deviations between the *PICP* and the *PINC*, respectively. According to the result obtained for the *ARIL* metrics shown in Table 9, the proposed TCN model is apparently superior over the other standalone as well as hybrid

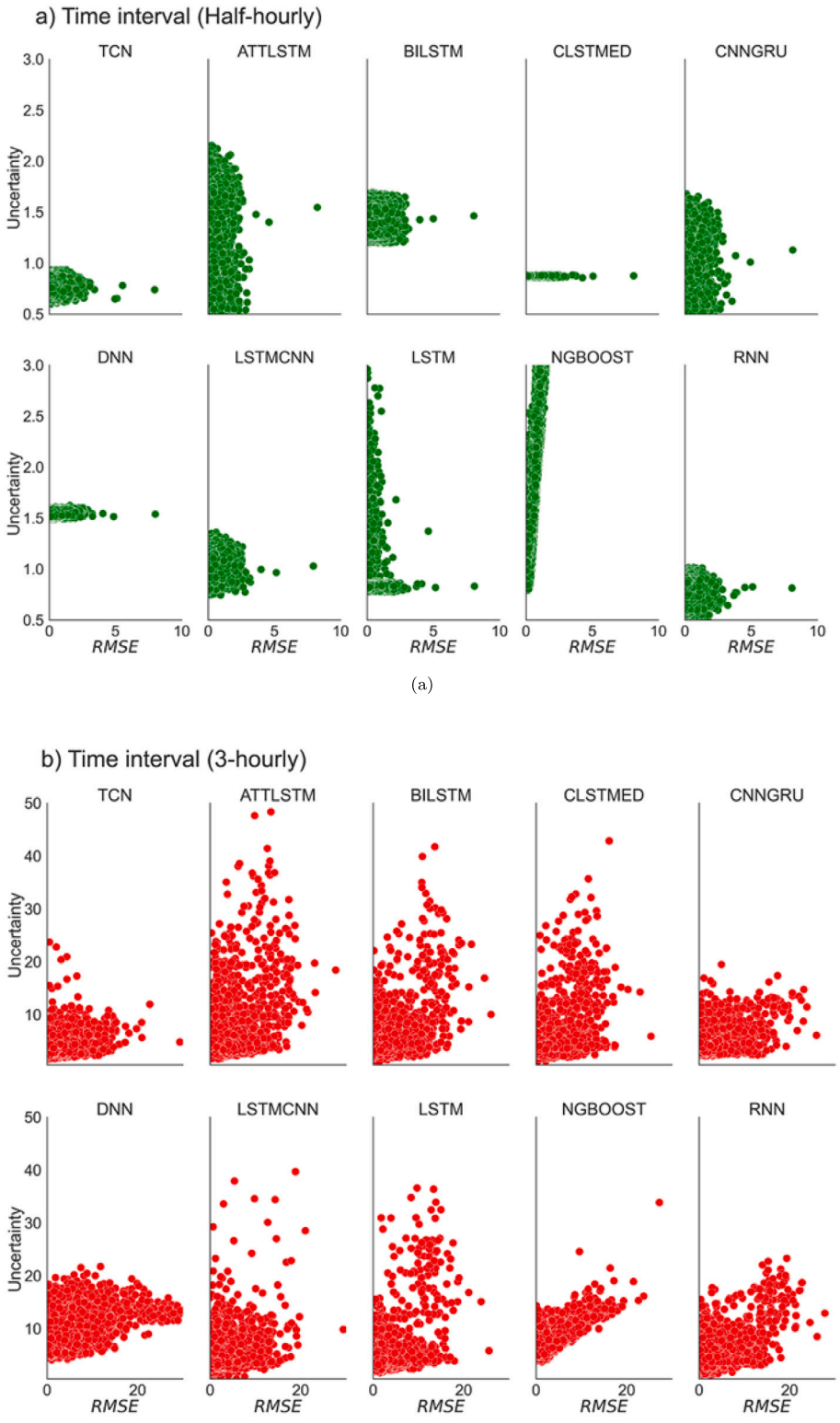


Fig. 16. Scatter-chart showing the relationship between total uncertainty (TU) (sum of AU and EU) and ($RMSE, MW$). (a) Half-hourly, and (b) 3-hourly as well as long-term (c) daily electricity demand prediction (test site name: Acacia Ridge).

models. This conclusion also can be obtained by analysing the results of *WS* metrics. Additionally, the *ACE* value of the proposed model remained similar in its magnitude relative to other comparative models. In summary, considering all of the evaluation metrics (i.e., those arising from the probabilistic as well as deterministic measures of predicted and observed G), the proposed TCN model has outperformed the baseline models tested for half-hourly, three-hourly and daily prediction at all of the three study sites. Fig. 17 depicts the PIs generated by the models for Zillmere, which supports this finding.

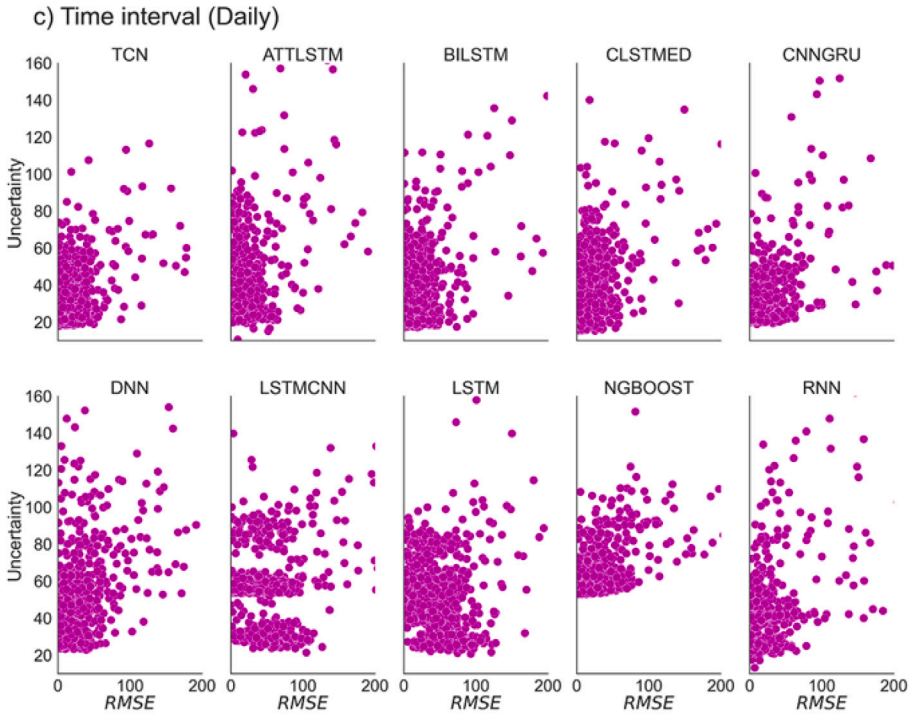


Fig. 16. (*continued*).

Table 9
Probabilistic evaluation of TCN using relative WS, $ARIL$ and ACE values, as per Eqs. (26)–(29). A better performing prediction interval, shown in boldface, is expected to have a lower Winkler score value. A higher reliability of prediction intervals is achieved for any model with an ACE closer to zero.

Prediction Models	Prediction Time-Steps	Acacia Ridge			Alexandra Head			Zillmere		
		WS	$ARIL$	ACE	WS	$ARIL$	ACE	WS	$ARIL$	ACE
TCN	Half-hourly	**12.787**	**0.500**	**0.096**	**3.075**	**0.380**	**0.090**	**4.678**	**0.300**	**0.071**
	3-hourly	**18.704**	**0.336**	**0.063**	**69.504**	**0.465**	**0.085**	**38.931**	**0.343**	**−0.002**
	Daily	**178.375**	**0.360**	**0.063**	**352.545**	**0.187**	**−0.030**	**223.697**	**0.272**	**0.045**
ATTLSTM	Half-hourly	22.246	0.781	0.099	5.097	0.541	0.086	5.798	0.370	0.061
	3-hourly	22.970	0.419	0.064	84.815	0.588	0.088	43.207	0.448	0.044
	Daily	227.485	0.521	0.063	396.948	0.286	0.049	263.255	0.365	0.079
BILSTM	Half-hourly	16.788	0.662	0.097	5.567	0.670	0.100	6.544	0.369	0.025
	3-hourly	20.254	0.358	0.058	80.284	0.538	0.089	42.203	0.444	0.040
	Daily	242.023	0.537	0.070	449.409	0.313	0.016	272.655	0.373	0.081
CLSTMED	Half-hourly	17.595	0.705	0.098	3.453	0.420	0.096	5.567	0.307	0.029
	3-hourly	19.596	0.357	0.059	82.103	0.556	0.091	41.164	0.418	0.034
	Daily	184.489	0.392	0.068	370.982	0.263	0.051	298.725	0.408	0.076
CNNGRU	Half-hourly	16.319	0.646	0.095	3.983	0.443	0.092	6.744	0.399	0.040
	3-hourly	20.036	0.355	0.059	74.326	0.501	0.074	46.215	0.441	0.018
	Daily	266.362	0.637	0.079	533.138	0.442	0.082	232.242	0.286	0.040
DNN	Half-hourly	20.066	0.840	0.100	6.056	0.745	0.100	6.371	0.404	0.052
	3-hourly	36.131	0.721	0.082	84.146	0.558	0.084	46.923	0.511	0.056
	Daily	292.565	0.675	0.068	575.198	0.480	0.092	236.248	0.282	0.040
LSTMCNN	Half-hourly	14.201	0.575	0.095	3.878	0.460	0.097	6.196	0.325	0.014
	3-hourly	18.966	0.341	0.057	113.901	0.816	0.095	40.956	0.417	0.037
	Daily	207.602	0.449	0.063	459.971	0.206	0.098	231.591	0.285	0.047
LSTM	Half-hourly	17.391	0.718	0.099	3.287	0.409	0.095	5.160	0.306	0.043
	3- hourly	19.890	0.360	0.060	108.315	0.755	0.095	43.305	0.471	0.054
	Daily	182.497	0.379	0.066	358.239	0.211	0.000	227.679	0.276	0.045
NGBOOST	Half-hourly	20.331	0.866	0.100	6.386	0.785	0.100	10.176	0.784	0.100
	3-hourly	29.130	0.597	0.096	91.640	0.512	0.035	59.041	0.722	0.088
	Daily	188.069	0.400	0.061	394.887	0.267	0.035	230.673	0.280	0.040
RNN	Half-hourly	19.915	0.710	0.099	3.063	0.355	0.092	7.306	0.495	0.072
	3-hourly	23.712	0.437	0.072	81.711	0.553	0.088	43.652	0.441	0.030
	Daily	177.968	0.369	0.071	375.268	0.279	0.076	230.128	0.285	0.049

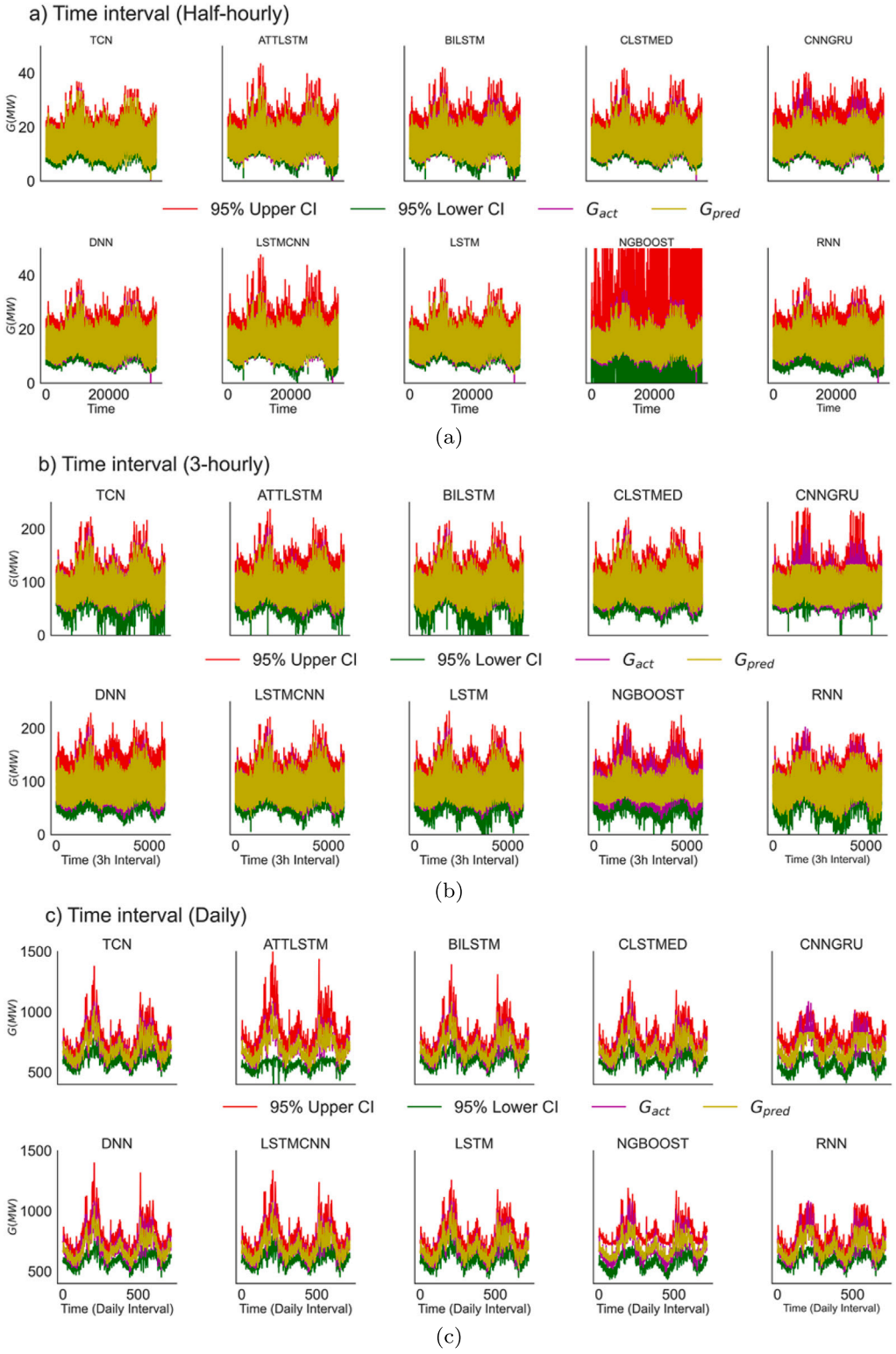


Fig. 17. The scatter chart illustrates the relationship between total uncertainty (comparison showing upper bounds, lower bounds for 95% prediction interval of TCN *vs.* comparative models for short-term (a) half-hourly and(b) three-hourly and long-term (c) daily electricity demand prediction (testing site name: - Zillmere).

5. Conclusions and future outlook

5.1. Conclusions

For reliability and sustainability of the electricity supply system, risk analysis and projections of consumer electricity use, energy providers need accurate and reliable forecasts of electricity demand and the associated uncertainties. In order to address this challenge, this study developed a Temporal Convolutional Neural Network (TCN) model to predict electricity demand in multiple steps (half-hourly, 3-hourly, and daily). In addition, the proposed TCN model was benchmarked against deep learning models (and their variants) using real-world datasets. For

short- and long-term periods of electricity demand prediction intervals, we studied the distribution of uncertainty generated by the TCN model, including aleatoric and epistemic errors. As part of this study, we also examined the average coverage errors of G predictions using the proposed TCN model, along with the relative coverage probability of prediction intervals and normalized mean widths and average relative interval lengths, as well as Winkler scores to gauge prediction interval quality and sharpness. Based on the contributions made, the proposed TCN model was found to successfully predict electricity demand and associated uncertainties and therefore could be employed in decision-making for electricity demand forecasting.

In making the above contributions to knowledge, this study has analysed electricity demand data captured in Southeast Queensland between 2011 and 2020. As the first step, data consolidation and preprocessing tasks were performed on historical G dataset followed by an optimal time-lag (Partial Auto-correlation) determination of G data series for the modelling tasks that served as the input time step. We selected the time lag $(t - n)$ where the current and n-antecedent (i.e. historical) electricity demand had the highest correlation and lowest error selected through an Extreme Learning Machine (ELM) approach. In contrast to prior studies, the proposed TCN model employed causal convolutions and dilation methods, and therefore, was effective with sequential datasets that was time-related and whose receptive fields were large. Accordingly, we compared TCN with the best time-series deep learning models from CNNGRU, CNNADM, DNN, LSTM, NGBOOST, RNN, and encoder–decoder CNNLSTM. The proposed TCN model outperformed the other DL models at multiple time-steps based on the independent test dataset.

Our study involved the analysis of all G predictions generated by all models aimed at demonstrating how DL is applicable for electricity demand predictions for insight into decision-making process through the uncertainty analysis. Accordingly, the proposed DL models for half-hourly, three-hourly and daily predictions were evaluated using extensive uncertainty measurements. For all forecast horizons, the TCN model superseded the hybrid model performance as well as the standalone models, to show its ability to as a decision-making tool in electricity modelling area.

5.2. Future research

This study does have some limitations that could be investigated further despite the superior performance of the TCN model. Firstly, the study has utilized the historical behaviour of electricity use datasets as a predictor variable through optimal lagged combinations to build a predictive model. Predicting electricity demand requires a detailed understanding of weather effects, especially air temperatures and seasonality of electricity use e.g, [141] which was not considered in training the proposed TCN model.

As temperature and humidity influence electricity demand, especially in warmer periods that increase the heated air temperatures, the effects of weather must be considered in re-training the proposed TCN model and the Monte Carlo dropout and maximum likelihood used in this study is adoptable in quantifying aleatoric uncertainty, which is the inherent randomness found in modelling data not explained by any other definite way, or epistemic uncertainty that occurs in a model found due to a lack of training data. Secondly, the daily and weekly work patterns (e.g., working days, holidays, inter-holidays, holiday seasons) can moderate the average electricity use profile, and affect both the demand and electricity price e.g, [142]. Lastly, a strong correlation between spot price of electricity and demand could exist in national electricity markets as well as the intermittent supply of renewable electricity into the grid. Therefore, the inclusion of electricity price as a predictor of demand and assessing the impact of renewable electricity into the grid could possibly provide greater insights into the efficacy of the proposed model for predicting demand more accurately. Fourthly, if different variables such as weather data, electricity price, or renewable electricity influx into the grid are considered, the application of explainable artificial intelligence methods where global and local explanations in terms of the impact of each new predictor variable on model response needs to be considered to provide greater confidence into the TCN model outputs for key decision-making.

This study has provided major contributions to time series predictions by comparing a variety of DL models (stacked, encoder–decoder architecture) and CNN models. In future, decision-makers can therefore easily comprehend and implement the proposed methods and associated uncertainty evaluation measures developed in this work as end-to-end learning methods were presented without combined/hybridized network architectures current found in the literature. The benefits of the TCN model also arise from its potential use in the prediction of electricity demand in other regions or countries not tested in this pioneer work to adopt as a decision-making tool in electricity forecasting areas. Finally, future researchers could apply TCN model in other emerging areas of interest, e.g., solar energy prediction, wind energy, rainfall pattern analysis, water quality, agriculture and energy consumption. Using accurate modelling methods, policy departments can manage climate projection scenarios and resolve renewable energy-related problems.

CRediT authorship contribution statement

Sujan Ghimire: Conceptualization, Visualization, Investigation, Methodology, Software, Data curation, Writing – original draft. **Ravinesh C. Deo:** Supervision, Investigation, Writing – review & editing. **David Casillas-Pérez:** Visualization, Investigation, Writing – review & editing. **Sancho Salcedo-Sanz:** Investigation, Writing – review & editing. **Rajendra Acharya:** Writing – review & editing. **Toan Dinh:** Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The required data was provided by Energex. The study received partial funding from the Ministry of Science and Innovation, Spain (Project ID: PID2020-115454GB-C21). Partial support of this work was through the LATENTIA project PID2022-140786NB-C31 of the Spanish Ministry of Science, Innovation and Universities (MICINNU).

Data availability

Energex Site: https://www.energex.com.a.

References

[1] Jiang P, Li R, Liu N, Gao Y. A novel composite electricity demand forecasting framework by data processing and optimized support vector machine. Appl Energy 2020;260:114243.
[2] Cf O. Transforming our world: the 2030 Agenda for sustainable development. United Nations: New York, NY, USA; 2015.
[3] Ghiasi M, Ghadimi N, Ahmadinia E. An analytical methodology for reliability assessment and failure analysis in distributed power system. SN Appl Sci 2019;1(1):1–9.
[4] Yang Z, Ghadamyari M, Khorramdel H, Alizadeh SMS, Pirouzi S, Milani M, et al. Robust multi-objective optimal design of islanded hybrid system with renewable and diesel sources/stationary and mobile energy storage systems. Renew Sustain Energy Rev 2021;148:111295.
[5] Hiruta Y, Ishizaki NN, Ashina S, Takahashi K. Regional and temporal variations in the impacts of future climate change on Japanese electricity demand: simultaneous interactions among multiple factors considered. Energy Convers Manage 2022;14:100172.

[6] Varma R, et al. Bridging the electricity demand and supply gap using dynamic modeling in the Indian context. Energy Policy 2019;132:515–35.
[7] Tso GK, Yau KK. Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks. Energy 2007;32(9):1761–8.
[8] Li R, Jiang P, Yang H, Li C. A novel hybrid forecasting scheme for electricity demand time series. Sustainable Cities Soc 2020;55:102036.
[9] Niu J, Xu Z-h, Zhao J, Shao Z-j, Qian J-x. Model predictive control with an on-line identification model of a supply chain unit. J Zhejiang Univ Sci C 2010;11(5):394–400.
[10] Prol JL, Sungmin O. Impact of COVID-19 measures on short-term electricity consumption in the most affected EU countries and USA states. Iscience 2020;23(10):101639.
[11] Andersen FM, Larsen HV, Gaardestrup R. Long term forecasting of hourly electricity consumption in local areas in Denmark. Appl Energy 2013;110:147–62.
[12] Amjady N. Short-term hourly load forecasting using time-series modeling with peak load estimation capability. IEEE Trans Power Syst 2001;16(3):498–505.
[13] Juberias G, Yunta R, Moreno JG, Mendivil C. A new ARIMA model for hourly load forecasting. In: 1999 IEEE transmission and distribution conference (cat. no. 99CH36333), vol. 1, IEEE; 1999, p. 314–9.
[14] Sigauke C, Chikobvu D. Daily peak electricity load forecasting in South Africa using a multivariate non-parametric regression approach. ORiON 2010;26(2).
[15] Sahraei MA, Duman H, Çodur MY, Eyduran E. Prediction of transportation energy demand: multivariate adaptive regression splines. Energy 2021;224:120090.
[16] Taylor JW, De Menezes LM, McSharry PE. A comparison of univariate methods for forecasting electricity demand up to a day ahead. Int J Forecast 2006;22(1):1–16.
[17] Deb C, Zhang F, Yang J, Lee SE, Shah KW. A review on time series forecasting techniques for building energy consumption. Renew Sustain Energy Rev 2017;74:902–24.
[18] Bernardi M, Petrella L. Multiple seasonal cycles forecasting model: the Italian electricity demand. Stat Methods Appl 2015;24(4):671–95.
[19] Deng J. Control problems of grey system, systems and control letters. vol 1982;1:N0.
[20] Ding S, Hipel KW, Dang Y-g. Forecasting China's electricity consumption using a new grey prediction model. Energy 2018;149:314–28.
[21] Cayir Ervural B, Ervural B. Improvement of grey prediction models and their usage for energy demand forecasting. J Intell Fuzzy Systems 2018;34(4):2679–88.
[22] Xu N, Dang Y, Gong Y. Novel grey prediction model with nonlinear optimized time response method for forecasting of electricity consumption in China. Energy 2017;118:473–80.
[23] Li K, Zhang T. Forecasting electricity consumption using an improved grey prediction model. Information 2018;9(8):204.
[24] Wang J, Du P, Lu H, Yang W, Niu T. An improved grey model optimized by multi-objective ant lion optimization algorithm for annual electricity consumption forecasting. Appl Soft Comput 2018;72:321–37.
[25] Xie N-m, Yuan C-q, Yang Y-j. Forecasting China's energy demand and self-sufficiency rate by grey forecasting model and Markov model. Int J Electr Power Energy Syst 2015;66:1–8.
[26] Deo RC, Ghimire S, Downs NJ, Raj N. Optimization of windspeed prediction using an artificial neural network compared with a genetic programming model. In: Research anthology on multi-industry uses of genetic programming and algorithms. IGI Global; 2021, p. 116–47.
[27] Hasanah RN, OMP RR, Suyono H. Comparison analysis of electricity load demand prediction using recurrent neural network (RNN) and vector autoregressive model (VAR). In: 2020 12th international conference on electrical engineering. IEEE; 2020, p. 23–9.
[28] Leme JV, Casaca W, Colnago M, Dias MA. Towards assessing the electricity demand in Brazil: Data-driven analysis and ensemble learning models. Energies 2020;13(6):1407.
[29] Al-Musaylh MS, Deo RC, Adamowski JF, Li Y. Short-term electricity demand forecasting with MARS, SVR and ARIMA models using aggregated demand data in Queensland, Australia. Adv Eng Inform 2018;35:1–16.
[30] Zhu K, Geng J, Wang K. A hybrid prediction model based on pattern sequence-based matching method and extreme gradient boosting for holiday load forecasting. Electr Power Syst Res 2021;190:106841.
[31] Mukherjee S, Vineeth C, Nateghi R. Evaluating regional climate-electricity demand nexus: a composite Bayesian predictive framework. Appl Energy 2019;235:1561–82.
[32] Ramos D, Faria P, Morais A, Vale Z. Using decision tree to select forecasting algorithms in distinct electricity consumption context of an office building. Energy Rep 2022;8:417–22.
[33] Azadeh A, Saberi M, Ghaderi S, Gitiforouz A, Ebrahimipour V. Improved estimation of electricity demand function by integration of fuzzy system and data mining approach. Energy Convers Manage 2008;49(8):2165–77.
[34] Azadeh A, Saberi M, Seraj O. An integrated fuzzy regression algorithm for energy consumption estimation with non-stationary data: a case study of Iran. Energy 2010;35(6):2351–66.
[35] Ghorbani M, Deo RC, Yaseen ZM, H Kashani M, Mohammadi B. Pan evaporation prediction using a hybrid multilayer perceptron-firefly algorithm (MLP-FFA) model: case study in North Iran. Theor Appl Climatol 2018;133(3):1119–31.
[36] xin Nie R, peng Tian Z, yin Long R, Dong W. Forecasting household electricity demand with hybrid machine learning-based methods: Effects of residents' psychological preferences and calendar variables. Expert Syst Appl 2022;206:117854. http://dx.doi.org/10.1016/j.eswa.2022.117854, URL https://www.sciencedirect.com/science/article/pii/S0957417422011095.
[37] Bengio Y. Learning deep architectures for AI. Now Publishers Inc; 2009.
[38] Ekonomou L. Greek long-term energy consumption prediction using artificial neural networks. Energy 2010;35(2):512–7.
[39] Kialashaki A, Reisel JR. Development and validation of artificial neural network models of the energy demand in the industrial sector of the United States. Energy 2014;76:749–60.
[40] Gajowniczek K, Zabkowski T. Electricity forecasting on the individual household level enhanced based on activity patterns. PLoS One 2017;12(4):e0174098.
[41] Abraham A, Nath B. A neuro-fuzzy approach for modelling electricity demand in victoria. Appl Soft Comput 2001;1(2):127–38.
[42] Zahedi G, Azizi S, Bahadori A, Elkamel A, Alwi SRW. Electricity demand estimation using an adaptive neuro-fuzzy network: a case study from the Ontario province–Canada. Energy 2013;49:323–8.
[43] Hadjout D, Torres J, Troncoso A, Sebaa A, Martínez-Álvarez F. Electricity consumption forecasting based on ensemble deep learning with application to the Algerian market. Energy 2022;243:123060.
[44] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. 2014, arXiv preprint arXiv:1409.0473.
[45] Kumari P, Toshniwal D. Deep learning models for solar irradiance forecasting: A comprehensive review. J Clean Prod 2021;318:128566.
[46] Bashir T, Haoyong C, Tahir MF, Liqiang Z. Short term electricity load forecasting using hybrid prophet-LSTM model optimized by BPNN. Energy Rep 2022;8:1678–86.
[47] Kawaguchi K, Kaelbling LP, Bengio Y. Generalization in deep learning. 2017, arXiv preprint arXiv:1710.05468.
[48] Ghimire S, Nguyen-Huy T, Prasad R, Deo RC, Casillas-Pérez D, Salcedo-Sanz S, et al. Hybrid convolutional neural network-multilayer perceptron model for solar radiation prediction. Cogn Comput 2022;1–27.
[49] He W. Load forecasting via deep neural networks. Procedia Comput Sci 2017;122:308–14.
[50] Gul MJ, Urfa GM, Paul A, Moon J, Rho S, Hwang E. Mid-term electricity load prediction using CNN and bi-LSTM. J Supercomput 2021;77(10):10942–58.
[51] Shi H, Xu M, Ma Q, Zhang C, Li R, Li F. A whole system assessment of novel deep learning approach on short-term load forecasting. Energy Procedia 2017;142:2791–6.
[52] Torres JF, Galicia A, Troncoso A, Martínez-Álvarez F. A scalable approach based on deep learning for big data time series forecasting. Integr Comput-Aided Eng 2018;25(4):335–48.
[53] Kong W, Dong ZY, Jia Y, Hill DJ, Xu Y, Zhang Y. Short-term residential load forecasting based on LSTM recurrent neural network. IEEE Trans Smart Grid 2017;10(1):841–51.
[54] Muralitharan K, Sakthivel R, Vishnuvarthan R. Neural network based optimization approach for energy demand prediction in smart grid. Neurocomputing 2018;273:199–208.
[55] Skomski E, Lee J-Y, Kim W, Chandan V, Katipamula S, Hutchinson B. Sequence-to-sequence neural networks for short-term electrical load forecasting in commercial office buildings. Energy Build 2020;226:110350.
[56] Kim J-Y, Cho S-B. Explainable prediction of electric energy demand using a deep autoencoder with interpretable latent space. Expert Syst Appl 2021;186:115842. http://dx.doi.org/10.1016/j.eswa.2021.115842, URL https://www.sciencedirect.com/science/article/pii/S0957417421012033.
[57] Jin Y, Guo H, Wang J, Song A. A hybrid system based on LSTM for short-term power load forecasting. Energies 2020;13(23):6241.
[58] Jayasinghe WLP, Deo RC, Ghahramani A, Ghimire S, Raj N. Deep multi-stage reference evapotranspiration forecasting model: Multivariate empirical mode decomposition integrated with the boruta-random forest algorithm. IEEE Access 2021;9:166695–708.
[59] Bouktif S, Fiaz A, Ouni A, Serhani MA. Multi-sequence LSTM-RNN deep learning and metaheuristics for electric load forecasting. Energies 2020;13(2):391.
[60] Wang Y, Gan D, Sun M, Zhang N, Lu Z, Kang C. Probabilistic individual load forecasting using pinball loss guided LSTM. Appl Energy 2019;235:10–20.
[61] Xiuyun G, Ying W, Yang G, Chengzhi S, Wen X, Yimiao Y. Short-term load forecasting model of gru network based on deep learning framework. In: 2018 2nd IEEE conference on energy internet and energy system integration. IEEE; 2018, p. 1–4.
[62] Zhang X, Wang R, Zhang T, Wang L, Liu Y, Zha Y. Short-term load forecasting based on RBM and NARX neural network. In: International conference on intelligent computing. Springer; 2018, p. 193–203.
[63] Fu G. Deep belief network based ensemble approach for cooling load forecasting of air-conditioning system. Energy 2018;148:269–82.
[64] Chitalia G, Pipattanasomporn M, Garg V, Rahman S. Robust short-term electrical load forecasting framework for commercial buildings using deep recurrent neural networks. Appl Energy 2020;278:115410.

[65] Mughees N, Mohsin SA, Mughees A, Mughees A. Deep sequence to sequence bi-LSTM neural networks for day-ahead peak load forecasting. Expert Syst Appl 2021;175:114844. http://dx.doi.org/10.1016/j.eswa.2021.114844, URL https://www.sciencedirect.com/science/article/pii/S0957417421002852.

[66] Du S, Li T, Yang Y, Horng S-J. Multivariate time series forecasting via attention-based encoder–decoder framework. Neurocomputing 2020;388:269–79.

[67] Akinola SA, Thakur P, Sharma MS, Kumar K, Singh G. Deep learning approach to load forecasting: A survey. In: International conference on communication, networks and computing. Springer; 2020, p. 250–62.

[68] Almalaq A, Edwards G. A review of deep learning methods applied on load forecasting. In: 2017 16th IEEE international conference on machine learning and applications. IEEE; 2017, p. 511–6.

[69] Lara-Benítez P, Carranza-García M, Luna-Romera JM, Riquelme JC. Temporal convolutional networks applied to energy-related time series forecasting. Appl Sci 2020;10(7):2322.

[70] Zhu R, Liao W, Wang Y. Short-term prediction for wind power based on temporal convolutional network. Energy Rep 2020;6:424–9.

[71] Lemos VH, Almeida JD, Paiva AC, Junior GB, Silva AC, Neto SM, et al. Temporal convolutional network applied for forecasting individual monthly electric energy consumption. In: 2020 IEEE international conference on systems, man, and cybernetics. IEEE; 2020, p. 2002–7.

[72] Xia M, Shao H, Ma X, de Silva CW. A stacked GRU-rnn-based approach for predicting renewable energy and electricity load for smart grid operation. IEEE Trans Ind Inf 2021;17(10):7050–9.

[73] Ghimire S, Deo RC, Wang H, Al-Musaylh MS, Casillas-Pérez D, Salcedo-Sanz S. Stacked LSTM sequence-to-sequence autoencoder with feature selection for daily solar radiation prediction: A review and new modeling results. Energies 2022;15(3):1061.

[74] Kavzoglu T, Teke A. Predictive performances of ensemble machine learning algorithms in landslide susceptibility mapping using random forest, extreme gradient boosting (xgboost) and natural gradient boosting (ngboost). Arab J Sci Eng 2022;1–19.

[75] Bai S, Kolter JZ, Koltun V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. 2018, arXiv preprint arXiv:1803.01271.

[76] Yan J, Mu L, Wang L, Ranjan R, Zomaya AY. Temporal convolutional networks for the advance prediction of ENSO. Sci Rep 2020;10(1):1–15.

[77] He Z, Zhong Y, Pan J. An adversarial discriminative temporal convolutional network for EEG-based cross-domain emotion recognition. Comput Biol Med 2022;141:105048.

[78] Yin L, Xie J. Multi-feature-scale fusion temporal convolution networks for metal temperature forecasting of ultra-supercritical coal-fired power plant reheater tubes. Energy 2022;238:121657.

[79] Lin Y, Koprinska I, Rana M. Temporal convolutional neural networks for solar power forecasting. In: 2020 international joint conference on neural networks. IEEE; 2020, p. 1–8.

[80] Hu J, Luo Q, Tang J, Heng J, Deng Y. Conformalized temporal convolutional quantile regression networks for wind power interval forecasting. Energy 2022;123497.

[81] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016, p. 770–8.

[82] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning. PMLR; 2015, p. 448–56.

[83] Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 2014;15(1):1929–58.

[84] Kosana V, Teeparthi K, Madasthu S. Hybrid wind speed prediction framework using data pre-processing strategy based autoencoder network. Electr Power Syst Res 2022;206:107821.

[85] Ding P, Liu X, Li H, Huang Z, Zhang K, Shao L, et al. Useful life prediction based on wavelet packet decomposition and two-dimensional convolutional neural network for lithium-ion batteries. Renew Sustain Energy Rev 2021;148:111287.

[86] Kiranyaz S, Avci O, Abdeljaber O, Ince T, Gabbouj M, Inman DJ. 1D convolutional neural networks and applications: A survey. Mech Syst Signal Process 2021;151:107398.

[87] Chen Z, Sun S, Wang Y, Wang Q, Zhang X. Temporal convolution-network-based models for modeling maize evapotranspiration under mulched drip irrigation. Comput Electron Agric 2020;169:105206.

[88] Zhou K, Wen L. Incentive-based demand response with deep learning and reinforcement learning. In: Smart energy management. Springer; 2022, p. 155–82.

[89] Hochreiter S, Schmidhuber J. LSTM can solve hard long time lag problems. In: Advances in neural information processing systems, vol. 9, 1996.

[90] Dudek G, Pełka P, Smyl S. A hybrid residual dilated LSTM and exponential smoothing model for midterm electric load forecasting. IEEE Trans Neural Netw Learn Syst 2021.

[91] Mughees N, Mohsin SA, Mughees A, Mughees A. Deep sequence to sequence bi-LSTM neural networks for day-ahead peak load forecasting. Expert Syst Appl 2021;175:114844.

[92] Zang H, Xu R, Cheng L, Ding T, Liu L, Wei Z, et al. Residential load forecasting based on LSTM fusing self-attention mechanism with pooling. Energy 2021;229:120682.

[93] Gaur K, Singh SK. CNN-bi-LSTM based household energy consumption prediction. In: 2021 3rd international conference on signal processing and communication. IEEE; 2021, p. 233–7.

[94] Yang G, Du S, Duan Q, Su J. Short-term price forecasting method in electricity spot markets based on attention-LSTM-mTCN. J Electr Eng Technol 2022;1–10.

[95] Tawn R, Browell J. A review of very short-term wind and solar power forecasting. Renew Sustain Energy Rev 2022;153:111758.

[96] Tang X, Chen H, Xiang W, Yang J, Zou M. Short-term load forecasting using channel and temporal attention based temporal convolutional network. Electr Power Syst Res 2022;205:107761.

[97] Fang Z, Crimier N, Scanu L, Midelet A, Alyafi A, Delinchant B. Multi-zone indoor temperature prediction with LSTM-based sequence to sequence model. Energy Build 2021;245:111053.

[98] Sajjad M, Khan ZA, Ullah A, Hussain T, Ullah W, Lee MY, et al. A novel CNN-GRU-based hybrid approach for short-term residential load forecasting. Ieee Access 2020;8:143759–68.

[99] Ghimire S, Deo RC, Raj N, Mi J. Deep solar radiation forecasting with convolutional neural network and long short-term memory network algorithms. Appl Energy 2019;253:113541.

[100] Agga A, Abbou A, Labbadi M, El Houm Y. Short-term self consumption PV plant power production forecasts based on hybrid CNN-LSTM, ConvLSTM models. Renew Energy 2021;177:101–12.

[101] Ghimire S, Deo RC, Casillas-Pérez D, Salcedo-Sanz S, Sharma E, Ali M. Deep learning CNN-LSTM-MLP hybrid fusion model for feature optimizations and daily solar radiation prediction. Measurement 2022;111759. http://dx.doi.org/10.1016/j.measurement.2022.111759, URL https://www.sciencedirect.com/science/article/pii/S0263224122009629.

[102] Eaton-Rosen Z, Bragman F, Bisdas S, Ourselin S, Cardoso MJ. Towards safe deep learning: accurately quantifying biomarker uncertainty in neural network predictions. In: International conference on medical image computing and computer-assisted intervention. Springer; 2018, p. 691–9.

[103] Laptev N, Yosinski J, Li LE, Smyl S. Time-series extreme event forecasting with neural networks at uber. In: International conference on machine learning, vol. 34, sn; 2017, p. 1–5.

[104] Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR. Improving neural networks by preventing co-adaptation of feature detectors. 2012, arXiv preprint arXiv:1207.0580.

[105] Cicuttin A, Crespo ML, Mannatunga KS, Garcia VV, Baldazzi G, Rignanese LP, et al. A programmable system-on-chip based digital pulse processing for high resolution X-ray spectroscopy. In: 2016 international conference on advances in electrical, electronic and systems engineering. IEEE; 2016, p. 520–5.

[106] Al-Gabalawy M, Hosny NS, Adly AR. Probabilistic forecasting for energy time series considering uncertainties based on deep learning algorithms. Electr Power Syst Res 2021;196:107216.

[107] Gal Y, Ghahramani Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: International conference on machine learning. PMLR; 2016, p. 1050–9.

[108] Mosser L, Naeini E. Deep probabilistic neural networks for geoscience. In: 82nd EAGE annual conference & exhibition, vol. 2021, European Association of Geoscientists & Engineers; 2021, p. 1–5.

[109] Seitzer M, Tavakoli A, Antic D, Martius G. On the pitfalls of heteroscedastic uncertainty estimation with probabilistic neural networks. In: International conference on learning representations. 2021, p. 1–24.

[110] Qu C, Liu W, Taylor CJ. Bayesian deep basis fitting for depth completion with uncertainty. In: Proceedings of the IEEE/CVF international conference on computer vision. 2021, p. 16147–57.

[111] Goodfellow I, Bengio Y, Courville A. Deep learning. MIT Press; 2016.

[112] Gast J, Roth S. Lightweight probabilistic deep networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 3369–78.

[113] Wan C, Xu Z, Pinson P, Dong ZY, Wong KP. Optimal prediction intervals of wind power generation. IEEE Trans Power Syst 2013;29(3):1166–74.

[114] Ghimire S, Nguyen-Huy T, Deo RC, Casillas-Perez D, Salcedo-Sanz S. Efficient daily solar radiation prediction with deep learning 4-phase convolutional neural network, dual stage stacked regression and support vector machine CNN-REGST hybrid model. Sustain Mater Technol 2022;32:e00429.

[115] Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, et al. {TensorFlow}: A system for {Large – Scale} machine learning. In: 12th USeNIX symposium on operating systems design and implementation. 2016, p. 265–83.

[116] Chang Z, Zhang Y, Chen W. Effective adam-optimized LSTM neural network for electricity price forecasting. In: 2018 IEEE 9th international conference on software engineering and service science. IEEE; 2018, p. 245–8.

[117] Agarap AF. Deep learning using rectified linear units (relu). 2018, arXiv preprint arXiv:1803.08375.

[118] Li H, Jiang Z, Shi Z, Han Y, Yu C, Mi X. Wind-speed prediction model based on variational mode decomposition, temporal convolutional network, and sequential triplet loss. Sustain Energy Technol Assess 2022;52:101980.

[119] Passos D, Mishra P. An automated deep learning pipeline based on advanced optimisations for leveraging spectral classification modelling. Chemometr Intell Lab Syst 2021;215:104354.
[120] Wang H, Yang J, Lin M, Li W, Zhu J, Ren L, et al. Quad-polarimetric SAR sea state retrieval algorithm from Chinese gaofen-3 wave mode imagettes via deep learning. Remote Sens Environ 2022;273:112969.
[121] Law T, Shawe-Taylor J. Practical Bayesian support vector regression for financial time series prediction and market condition change detection. Quant Finance 2017;17(9):1403–16.
[122] Dewancker I, McCourt M, Clark S, Hayes P, Johnson A, Ke G. A stratified analysis of Bayesian optimization methods. 2016, arXiv preprint arXiv:1603.09441.
[123] Abbasimehr H, Paki R. Prediction of COVID-19 confirmed cases combining deep learning methods and Bayesian optimization. Chaos Solitons Fractals 2021;142:110511.
[124] Cornejo-Bueno L, Garrido-Merchán EC, Hernández-Lobato D, Salcedo-Sanz S. Bayesian optimization of a hybrid system for robust ocean wave features prediction. Neurocomputing 2018;275:818–28.
[125] He F, Zhou J, Feng Z-k, Liu G, Yang Y. A hybrid short-term load forecasting model based on variational mode decomposition and long short-term memory networks considering relevant factors with Bayesian optimization algorithm. Appl Energy 2019;237:103–16.
[126] Ghimire S, Deo RC, Casillas-Pérez D, Salcedo-Sanz S. Improved complete ensemble empirical mode decomposition with adaptive noise deep residual model for short-term multi-step solar radiation prediction. Renew Energy 2022.
[127] Ghimire S, Bhandari B, Casillas-Pérez D, Deo RC, Salcedo-Sanz S. Hybrid deep CNN-SVR algorithm for solar radiation prediction problems in Queensland, Australia. Eng Appl Artif Intell 2022;112:104860.
[128] Despotovic M, Nedic V, Despotovic D, Cvetanovic S. Review and statistical analysis of different global solar radiation sunshine models. Renew Sustain Energy Rev 2015;52:1869–80.
[129] Gupta HV, Kling H, Yilmaz KK, Martinez GF. Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling. J Hydrol 2009;377(1–2):80–91.
[130] McKenzie J. Mean absolute percentage error and bias in economic forecasting. Econom Lett 2011;113(3):259–62.
[131] Harvey D, Leybourne S, Newbold P. Testing the equality of prediction mean squared errors. Int J Forecast 1997;13(2):281–91.
[132] Diebold FX, Mariano RS. Comparing predictive accuracy. J Bus Econom Statist 2002;20(1):134–44.
[133] Costantini M, Pappalardo C. Combination of forecast methods using encompassing tests: An algorithm-based procedure. Technical report, Reihe Ökonomie/Economics Series; 2008.
[134] Hu J, Lin Y, Tang J, Zhao J. A new wind power interval prediction approach based on reservoir computing and a quality-driven loss function. Appl Soft Comput 2020;92:106327.
[135] Liu L, Zhao Y, Chang D, Xie J, Ma Z, Sun Q, et al. Prediction of short-term PV power output and uncertainty analysis. Appl Energy 2018;228:700–11.
[136] Golestaneh F, Pinson P, Gooi HB. Very short-term nonparametric probabilistic forecasting of renewable energy generation—With application to solar energy. IEEE Trans Power Syst 2016;31(5):3850–63.
[137] Kabir HD, Khosravi A, Hosen MA, Nahavandi S. Neural network-based uncertainty quantification: A survey of methodologies and applications. IEEE Access 2018;6:36218–34.
[138] Kabir H, Khosravi A, Kavousi-Fard A, Nahavandi S, Srinivasan D. Optimal uncertainty-guided neural network training. 2019, arXiv preprint arXiv:1912.12761.
[139] Ahmadi A, Nasseri M, Solomatine DP. Parametric uncertainty assessment of hydrological models: coupling UNEEC-P and a fuzzy general regression neural network. Hydrol Sci J 2019;64(9):1080–94.
[140] Rodríguez F, Insausti X, Etxezarreta G, Galarza A, Guerrero JM. Very short-term parametric ambient temperature confidence interval forecasting to compute key control parameters for photovoltaic generators. Sustain Energy Technol Assess 2022;51:101931.
[141] Al-Musaylh MS, Deo RC, Adamowski JF, Li Y. Short-term electricity demand forecasting using machine learning methods enriched with ground-based climate and ECMWF reanalysis atmospheric predictors in southeast Queensland, Australia. Renew Sustain Energy Rev 2019;113:109293.
[142] Al-Musaylh MS, Deo RC, Li Y, Adamowski JF. Two-phase particle swarm optimized-support vector regression hybrid model integrated with improved empirical mode decomposition with adaptive noise for multiple-horizon electricity demand forecasting. Appl Energy 2018;217:422–39.