Cui *et al. Journal of Cloud Computing* (2024) 13:67
https://doi.org/10.1186/s13677-024-00624-w

Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH

Open Access

Short-term forecasting of surface solar incident radiation on edge intelligence based on AttUNet

Mengmeng Cui[1*†], Shizhong Zhao[1†] and Jinfeng Yao[2]

Abstract

Solar energy has emerged as a key industry in the field of renewable energy due to its universality, harmlessness, and sustainability. Accurate prediction of solar radiation is crucial for optimizing the economic benefits of photovoltaic power plants. In this paper, we propose a novel spatiotemporal attention mechanism model based on an encoder-translator-decoder architecture. Our model is built upon a temporal AttUNet network and incorporates an auxiliary attention branch to enhance the extraction of spatiotemporal correlation information from input images. And utilize the powerful ability of edge intelligence to process meteorological data and solar radiation parameters in real-time, adjust the prediction model in real-time, thereby improving the real-time performance of prediction. The dataset utilized in this study is sourced from the total surface solar incident radiation (SSI) product provided by the geostationary meteorological satellite FY4A. After experiments, the SSIM has been improved to 0.86. Compared with other existing models, our model has obvious advantages and has great prospects for short-term prediction of surface solar incident radiation.

Keywords Solar energy, Edge intelligence, Attention mechanism, AttUNet

Introduction

Against the backdrop of a series of ecological and environmental issues caused by the large-scale development and utilization of traditional energy, solar energy has gradually become one of the key industries in the field of new energy due to its universality, harmlessness, and durability [1]. Renewable energy refers to energy that is constantly updated and inexhaustible in nature, and its use will not cause sustained damage to the environment. Renewable energy includes various forms such as solar energy, wind energy, hydro energy, geothermal energy, etc. Solar energy has become an increasingly important source of clean energy by capturing sunlight and converting it into electrical or thermal energy. The World Energy Outlook predicts that by 2040, approximately two-thirds of global investment in new power plant construction will be focused on renewable energy, with the largest portion coming from solar energy.

Solar power generation harnesses the shortwave radiation emitted by the sun, converting it into electrical energy through atmospheric propagation and scattering, either directly or indirectly. This process is characterized by its environmental friendliness, as it generates no pollutants and releases no greenhouse gases, particulate matter, or harmful substances. Compared to traditional fossil fuel power generation, solar power generation has a significantly lower impact on atmospheric quality and the environment, contributing to reductions in air and

[†]Mengmeng Cui and Shizhong Zhao contributed equally to this work.

*Correspondence:
Mengmeng Cui
cuimengmeng@nuist.edu.cn
[1] School of Software, Nanjing University of Information Science and Technology, Nanjing 210044, China
[2] China Meteorological Administration Public Meteorological Service Center, Beijing 100081, China

water pollution, carbon emissions, and efforts to mitigate climate change.

Moreover, the operational expenses associated with solar power generation systems are relatively modest [2]. Although the initial investment may be relatively high, in the long term, solar power generation can lead to decreased energy costs and yield substantial economic benefits. Photovoltaic power generation stands as the most prevalent technology in the solar power generation sector, with solar radiation serving as its primary energy source. However, it is important to note that solar energy bears resemblance to hydroelectric energy and is subject to atmospheric conditions, resulting in some challenges such as the intermittent nature and volatility of power generation, as well as potential fluctuations in voltage and frequency [3]. Presently, the utilization of energy storage systems represents a common approach to address the instability of photovoltaic power generation. However, challenges including high costs, limited lifespan, energy conversion losses, and environmental impacts still require further resolution [4].

Therefore, accurate short-term prediction of solar radiation plays an important role in photovoltaic power generation, which can optimize energy scheduling, improve energy supply reliability, improve photovoltaic power generation efficiency, and help optimize the operation of photovoltaic power generation projects economically.Solar radiation forecasting can be segmented based on the projected time frames, encompassing extremely short-term, short-term, medium-term, and long-term predictions. While extremely short-term forecasts of 5 to 30 minutes prove invaluable for power system management and network stability, accurate short-term forecasts spanning hours to several days are vital for informed decision-making, supply equilibrium, and meticulous scheduling.Therefore, extremely short and short-term accurate solar radiation forecasts are crucial for the successful operation of different solar applications [5].

With the rapid development of meteorological satellites, facing the demand for massive meteorological data processing, the existing cloud computing service mode can no longer meet the performance requirements of meteorological satellite data quality control. It is crucial to introduce edge cloud collaboration into meteorological data quality control and expand it into edge devices with computing and storage capabilities.In summary, the main contributions of this paper are as follows:

- Introducing edge-cloud collaboration into meteorological data quality control and extending it to edge devices with computational and storage capabilities.
- Proposing the EDA-AttUNet prediction model based on the Encoder-Translator-Decoder architecture, where the encoder extracts spatial features through multi-layer stacking. The translator introduces a UNet network architecture, forming a standalone encoder-decoder structure. Additionally, an attention mechanism is incorporated into the translator to allow the network to capture both local details and global contextual information simultaneously. The decoder reconstructs the actual surface solar incident radiation through multi-layer stacking.
- Conducting a comprehensive evaluation of the predictive performance and generalization ability of the model through the analysis of SSI data for the China region from FY4A during the entire year of 2021-2022, as well as predictions for different seasons.

Related work

At present, the adoption of edge cloud models in the field of meteorology is gaining momentum, exerting a significant influence on the management of weather-related big data [6]. Within the realm of intelligent services for edge cloud collaboration, a central challenge lies in the real-time control of data quality and the acquisition of highly dependable raw data. In our forthcoming endeavors, we plan to integrate the physical resources of the meteorological network with densely deployed edge servers in 5G environments, aiming to facilitate cross-network resource sharing and further enhance the overall quality of user experience.

In recent years, the methods for short-term and imminent forecasting of solar energy resources can be mainly divided into numerical weather forecasting methods, statistical model methods, and artificial intelligence methods based on the different data used.The numerical prediction model is one of the effective means for conducting solar energy resource assessment and prediction. It can use methods such as solar radiation output from the model, other variables, model prediction, and observation data to establish prediction models for solar energy assessment and prediction [7]. Based on the initial meteorological field data and the establishment of simulation domain and grids, the atmospheric dynamical processes within the specified region are simulated. By combining the simulated atmospheric dynamics with solar radiation data, solar energy resource forecasts are generated. However, due to factors such as data quality and the highly complex and dynamic nature of the atmospheric system, this method has certain limitations and uncertainties. System errors caused by physical processes have not been properly addressed in numerical weather prediction models [8].

The main method of artificial intelligence is machine learning. Machine learning models can solve problems that cannot be represented by explicit algorithms, and

with strong extraction ability for nonlinear features, They have demonstrated commendable accuracy in their solar energy forecasting endeavors [9]. Guijo Rubio et al. The performance of various evolutionary neural networks in predicting solar radiation in Toledo was evaluated by [10]. In their experimental testing, they found that the best model was achieved through the design of the S-shaped unit with evolutionary training.Huang and Liu [11] used set wavelet transform to decompose input data and predicted solar radiation based on an autoregressive model of an external input neural network [12].

In addition to their widespread application in time series analysis, recurrent neural networks (RNNs), particularly LSTM networks, have gained prominence for their adeptness in capturing both short-term and long-term dependencies [13]. Amit et al. [14] employed a combination of CNN and BiLSTM for predicting mid-term solar radiation. The assessment conducted at three distinct stations of varying locations demonstrated the robustness of this approach. Similarly, it was observed that the CNN-LSTM architecture exhibits favorable performance across different seasons and weather conditions. Furthermore, CNN has found utility in conjunction with various models for solar radiation prediction. In addition to combining with RNN, CNN is also used to predict solar radiation in conjunction with other models [15]. Omaima et al. The CNN-MLP model was used for solar radiation prediction, and it achieved a stable determination coefficient between 0.99 and 0.94. These results demonstrate its ability to deliver good performance even in cloudy weather [16]. In [17], the research demonstrates that by synergistically integrating two powerful deep learning techniques, namely CNN and LSTM, the resulting approach surpassed the performance of different benchmark methods in predicting Global Solar Radiation. This superiority was evident in terms of accuracy, forecasting speed, and the stability of prediction outcomes.The combined model showcased its potential for significantly improving solar radiation prediction, presenting a notable advancement in this field. Nielsen et al. A combination of quantitative measurements used inspired [18] to propose a new transformer-based framework. The results showed that IrradianceNet, inspired by the latest developments in deep learning spatiotemporal prediction models based on post feature level fusion, used SARAH-2.1 satellite data to predict surface solar irradiance in Europe for the next 4 hours, demonstrating superior performance over persistent models and optical flow methods. Zhang et al. [19] compared the sky imager image with the classic CNN model of solar radiation, and the transformer-based framework, incorporating early feature-level prediction, demonstrated notable enhancements in slope event balance accuracy. Specifically, it achieved an improvement of 9.3% at the 2-minute scale and 3.91% at the 6-minute scale. Furthermore, propelled by advancements in deep learning and harnessing the strengths of CNNs, the deep fully convolutional neural network has found extensive applications in diverse domains, including image segmentation and classification [20]. These methods are gradually beginning to be applied to satellite images. Zhang et al. [21] introduced a specially designed deep fully convolutional network to learn depth patterns for detecting clouds and snow from multispectral satellite images. Numerous experiments have shown that the proposed depth model outperforms the most advanced methods in both quantitative and qualitative performance [22].

Preliminary

Convolutional neural network

Convolutional Neural Network(CNN) is a feedforward neural network that is particularly suitable for processing data with grid structures, such as images and videos. As shown in Fig. 1, it usually consists of multiple

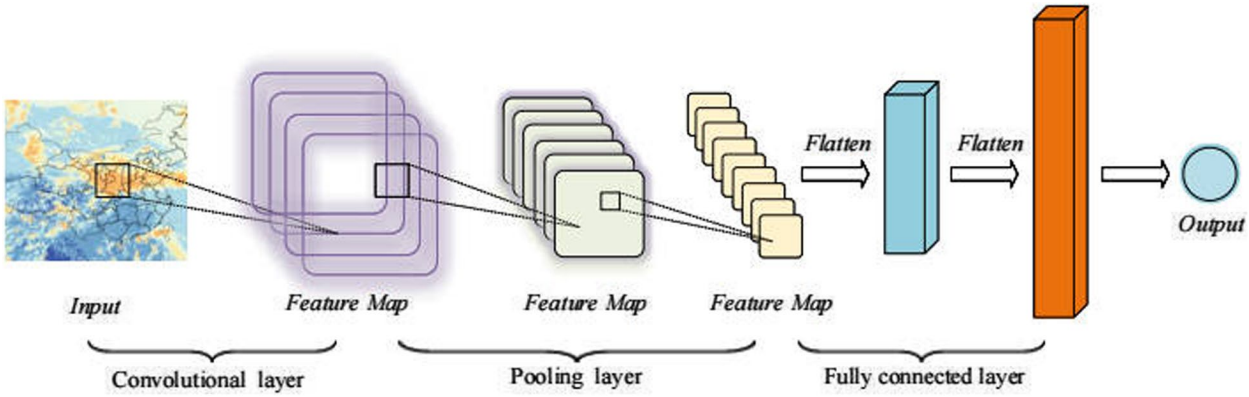

Fig. 1 Basic architecture of a CNN

convolutional layers, pooling layers, and fully connected layers. By stacking multiple layers for feature extraction and abstraction, it can automatically learn and extract features from input data, and has certain robustness to changes such as translation, scaling, and rotation.

Convolutional layers are the core components of CNN. In the convolutional layer, feature maps are generated through linear convolutional filters and nonlinear activation functions (corrector, sigmoid, tanh, etc.). These convolutional kernels can extract spatial features of images, such as edges, textures, etc [23]. Taking a linear rectifier as an example, the calculation method for feature mapping is as follows:

$$f_{i,j,k} = \max\left(w_k^T x_{i,j}, 0\right) \tag{1}$$

Where (i, j) is the pixel index in the feature map, $x_{i,j}$ is the input patch centered on position (i, j), k is the channel index of the feature map, and f represents the output feature values after activation function.

The CNN pooling layer plays a role in reducing dimensionality, extracting important features, translation invariance, and reducing overfitting in convolutional neural networks, helping to improve network efficiency, extract more representative features, and possess certain image spatial invariance. The mathematical expression for pooling layers can be represented as follows: where P represents pooling operations (such as maximum pooling or average pooling), S represents the step size of pooling, f represents the input feature map, while the resulting output feature map is represented as Y:

$$Y(i,j,k) = P\left(f\left(i_s : i_{s+k}, j_s : j_{s+k}, K\right)\right) \tag{2}$$

Among them, i and j represent the position coordinates of the output feature map, and k represents the channel of the output feature map divided by the depth. K represents the size of the pooling window, usually a square. In maximum pooling, the P operation selects the maximum value in the input window as the output; In average pooling, the P operation calculates the average value in the input window as the output. Finally, the Fully Connected Layer flattens the feature map into a one-dimensional vector and integrates the features from various positions. The formula can be expressed as follows:

$$y = F(WY + b) \tag{3}$$

Among them, y stands for the output of the fully connected layer, and Y signifies the one-dimensional vector derived from the output feature map of the convolutional layer. W stands for the weight matrix of the fully connected layer, and b denotes the bias vector. F denotes the activation function, typically using nonlinear functions such as ReLU, Sigmoid, or Tanh.

Depthwise separable convolution

Deepwise Separable Convolution (DWConv) is a special convolutional operation used in convolutional neural networks [24, 25]. As shown in Fig. 2, deep separable convolution divides the convolution operation into two independent steps: deep convolution and point by point convolution. Compared with traditional convolution operations, it greatly reduces the number of parameters, especially when there are many input channels, the reduction in the number of parameters is very significant. The deep convolution stage of deep separable convolution only performs convolution operations on each channel, avoiding computational redundancy between channels and reducing the risk of overfitting while maintaining model performance.

When the input feature map is X and the output feature map is Y, the calculation formula for depth separable convolution can be expressed in the following form: For each channel k of input feature map X, a deep filter $D(k)$ is used for convolution operation. Assuming the size of

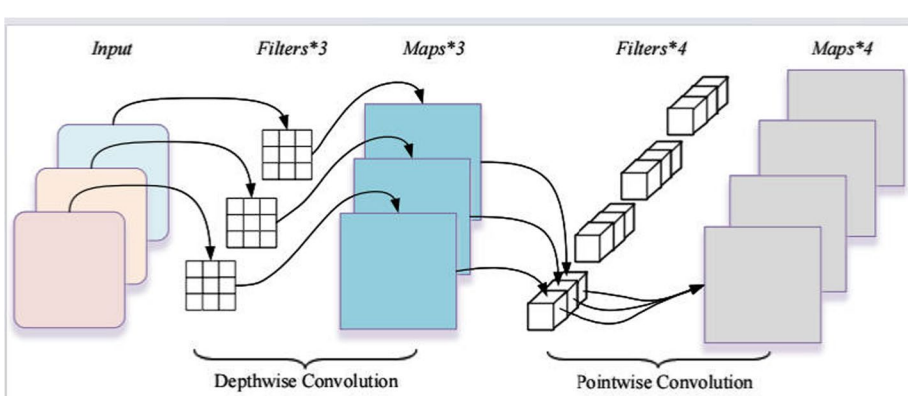

Fig. 2 Deepwise Separated Convolution work process diagram

the input feature map k is $H \times W$, the number of channels is C, and the size of the depth filter is $K \times K$. The calculation formula for deep convolution is:

$$Q = \sum_{i,j=1}^{H,W} \sum_{p,q=1}^{K} \sum_{k=1}^{C} X(i+p, j+q, k) \tag{4}$$

$$Y(i,j,k) = sum \sum_{p,q=1}^{K} \sum_{k=1}^{C} (D(k)(p,q) * Q) \tag{5}$$

Among them, $Y(i,j,k)$ represents the value of the element with position (i,j) and channel k in the output feature map Y. $D(k)(p,q)$ represents the value of depth filter $D(k)$ at position (p,q). $X(i+p, j+q, k)$ represents the value of input feature map X at position $(i+q, j+q, k)$. S represents a sum operation. Perform point by point convolution on the output feature map of deep convolution using a 1×1 Convolutional kernel of. Assuming the output feature map size of deep convolution is $H' \times W'$, the number of channels is C. The calculation formula for point by point convolution is:

$$Z(i',j',k') = sum \sum_{i,j=1}^{H',W'} \sum_{k'}^{C'} \left(W(c,k') * Y(i',j',k')\right) \tag{6}$$

Among them, $Z(i',j',k')$ represents the value of the element with position (i',j') and channel k' in the output feature map Z of point by point convolution. $W(c,k')$ represents the value of position (c,k') in the weight matrix of point by point convolution. $Y(i',j',k')$ represents the value of the element with position (i',j') and channel k' in the output feature map of deep convolution. *sum* represents the sum operation.

Convolutional block attention module

The Convolutional Block Attention Module (CBAM) is an attention mechanism employed to augment the capabilities of CNNs [26]. As shown in Fig. 3, this introduces two modules: channel attention and spatial attention. These modules enable the network to dynamically select and adjust crucial information within the feature map, thereby enhancing the model's expressive capacity and overall performance.

The channel attention module serves to discern the relationships and significance of feature maps within the channel dimension. This is achieved by learning channel attention weights through global average pooling and fully connected layers, which are then applied to each channel within the input feature map. On the other hand, the spatial attention module is designed to grasp the relationships and importance of feature maps in the spatial dimension. It accomplishes this by learning spatial attention weights through a combination of maximum pooling and average pooling operations, and subsequently applying these weights to each spatial position within the input feature map.By concatenating channel attention modules and spatial attention modules, CBAM can simultaneously consider the importance of both channels

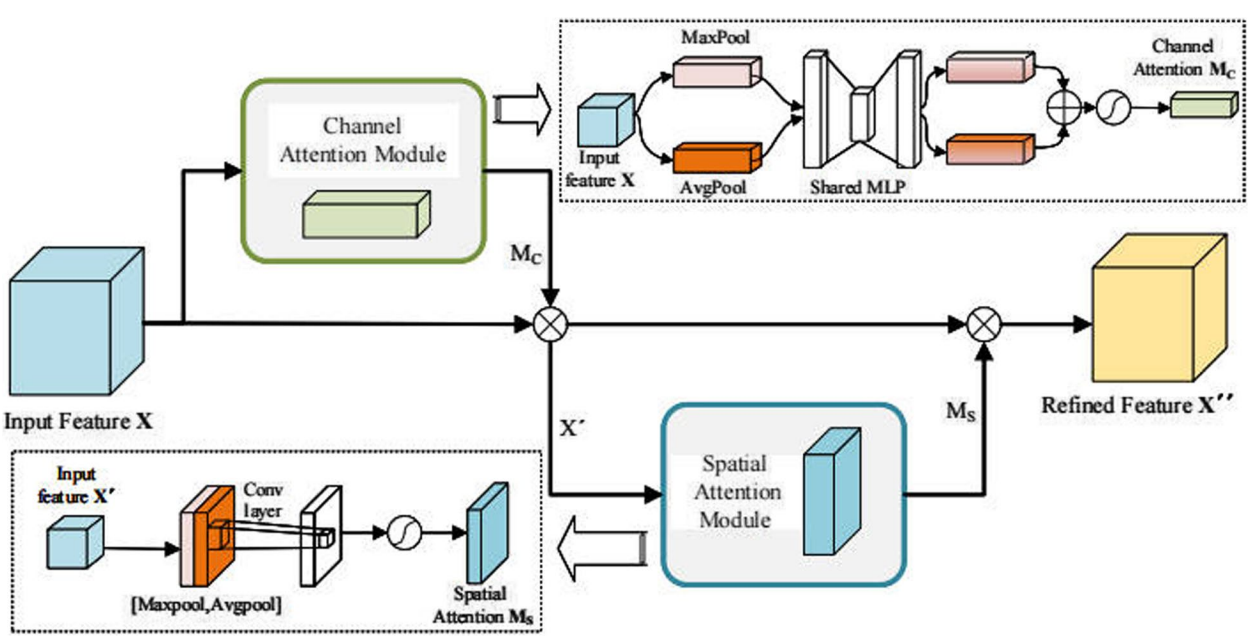

Fig. 3 Convolutional Block Attention Module structure diagram

and spaces, thereby improving the performance of the network in various computer vision tasks. Assuming the input feature map is X, where M_C and M_S represent the channel attention and spatial attention functions, the expression for this attention can be expressed as follows:

$$M_C(X) = \sigma\left(MLP\left(AvgPool(X) + MLP(MaxPool(X))\right)\right) \quad (7)$$

$$X' = M_C(X) \otimes X \quad (8)$$

$$M_S(X) = \sigma\left(f^{b\times b}[AvgPool(X), MaxPool(X)]\right) \quad (9)$$

$$X'' = M_S\left(X'\right) \otimes X' \quad (10)$$

Among them, *MaxPool* and *AvgPool* represent maximum pooling and average pooling operations, respectively, *MLP* represents shared weight multi-layer perceptron, σ represents the Sigmoid function, $\otimes$ represents element by element multiplication, X' represents the output feature map of channel attention, and X'' represents the output feature map of spatial attention.

Atrous spatial pyramid pooling

ASPP (Atrous Spatial Pyramid Pooling) is a deep learning technique used for semantic segmentation tasks. As shown in Fig. 4, ASPP can capture contextual information of different scales and expand the receptive field by using parallel convolutional branches with different sampling rates to process input feature maps. This multi-scale perception ability enables ASPP to better understand objects of different scales and improve its understanding of complex scenes. Secondly, ASPP adopts dilated convolution operation to avoid information loss and resolution reduction, while retaining more detailed information. The effectiveness of this feature representation helps to improve the performance and accuracy of the model. In addition, ASPP also combines global pooling operations to aggregate features across a larger range of contextual information, providing a more global perspective. This helps the model to better understand the overall structure and contextual relationships.

Taking the input feature map Z and Dilation rate list [r1, r2, r3, r4] as an example, the formula can be expressed as:

$$X_r = Pooling(Conv(Z, W, dilation\_rate = r)) \quad (11)$$

$$Z' = Concatenate(X_1, X_2, X_3, X_4) \quad (12)$$

Among them, W represents the convolutional kernel weight corresponding to the void ratio, and X_r represents the result of convolution operation and pooling operation for each Dilation ratio r and the weight W of void convolution Rate represents porosity, Pooling represents pooling operation, Concatenate represents cascading operation, and Sum represents adding operation by channel.Z' represents concatenating all pooled feature maps to obtain the final ASPP output feature map.

Method

Figure 5 illustrates the research scheme adopted in this article. Establish a deep learning model for predicting surface solar incident radiation based on satellite images. This method is mainly divided into three parts: the data preprocessing part, which performs region selection, quality control, interpolation, and normalization processing on the original data, and finally groups it into a format that conforms to deep learning training; The second part is to train the model, which inputs the allocated training set during preprocessing into the model for training. With the powerful learning ability of convolutional neural networks, the model can gradually improve

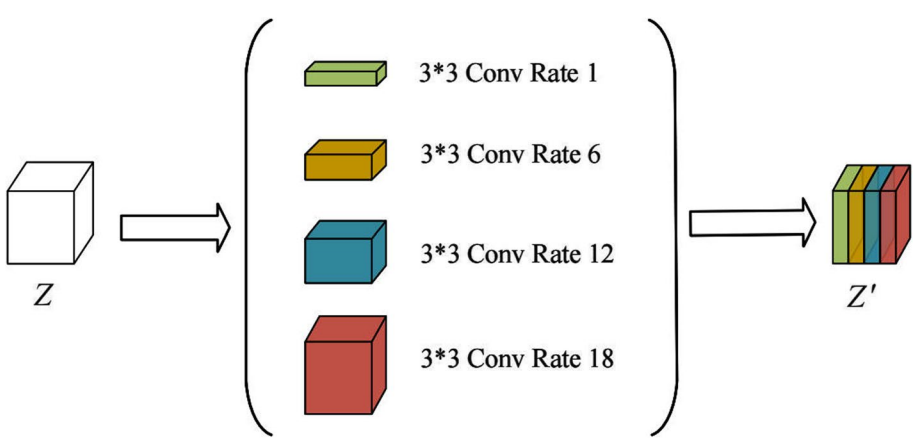

Fig. 4 Atrous Spatial Pyramid Pooling structure diagram

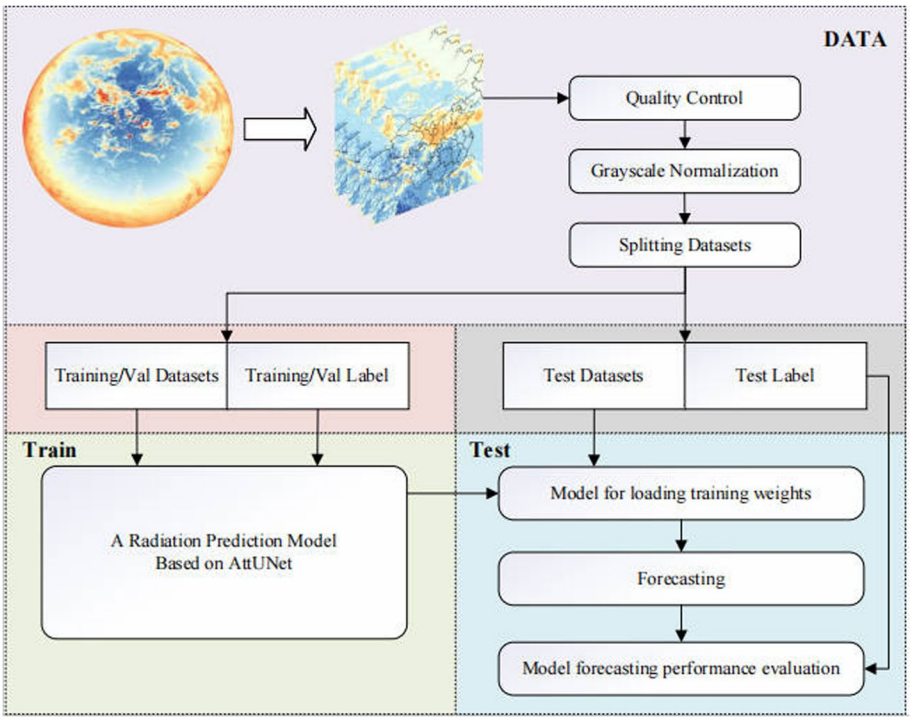

Fig. 5 Overall framework for predicting surface solar incident radiation

the accuracy and generalization ability of predictions. Each round of training will undergo a validation set test, and based on the performance indicators on the validation set, hyperparameters can be adjusted and network structure modified to improve the performance of the model. The final section compares the predicted results of the input model output of the test set with the actual results, and evaluates the performance and generalization ability of multiple models in real scenarios by evaluating them on the test set.

EDA-AttUNet

This subsection will describe the method behind the EDA-AttUNet model, as shown in Fig. 6, which is our spatiotemporal prediction model based on encoder-translator-decoder.

Encoder The encoder extracts spatial features by stacking residual blocks composed of DWConv, LayerNorm, and LeakyRelu. Assuming the input data time step is T, the number of channels is C, and the height and width of the image are H and W, respectively, that is, the input feature shape is *(T, C, H, W)*. The expression for the encoder can be represented as follows:

$$X_i' = X_i \odot \left(\sigma\left(LayerNorm(DWConv(X_i))\right)\right) \tag{13}$$

Among them, the shapes of input X_i and output X_i' are *(T, C, H, W)*. σ represents the Sigmoid function, $\odot$ represents the Hadamard product.

Translator By introducing AttUNet [27, 28], which includes a skip connection mechanism, the translator constructs an Encode-Decode structure separately. The encoder part plays a role in extracting temporal features, while the decoder part is used to restore the feature map to the resolution of the original image. By connecting feature maps at different levels in the encoder and decoder, the fusion of low-level and high-level features is achieved. This feature fusion capability helps to improve the accuracy of segmentation results and the ability to retain details. By using skip connections and feature fusion, the model can simultaneously utilize feature information at different levels, enabling it to capture contextual information at different scales. And an attention module combining CBAM was introduced between the encoder and decoder. These attention modules are used to calculate the importance weights of features and apply them to the feature representation of the decoder. Taking the *(t-th)* layer as an example, the upsampling output of the *(t-th)* layer can be expressed as:

$$X_t' = concat\left(deconv(X_{t-1}), A\left(X'_{t-1}, X_t\right)\right) \tag{14}$$

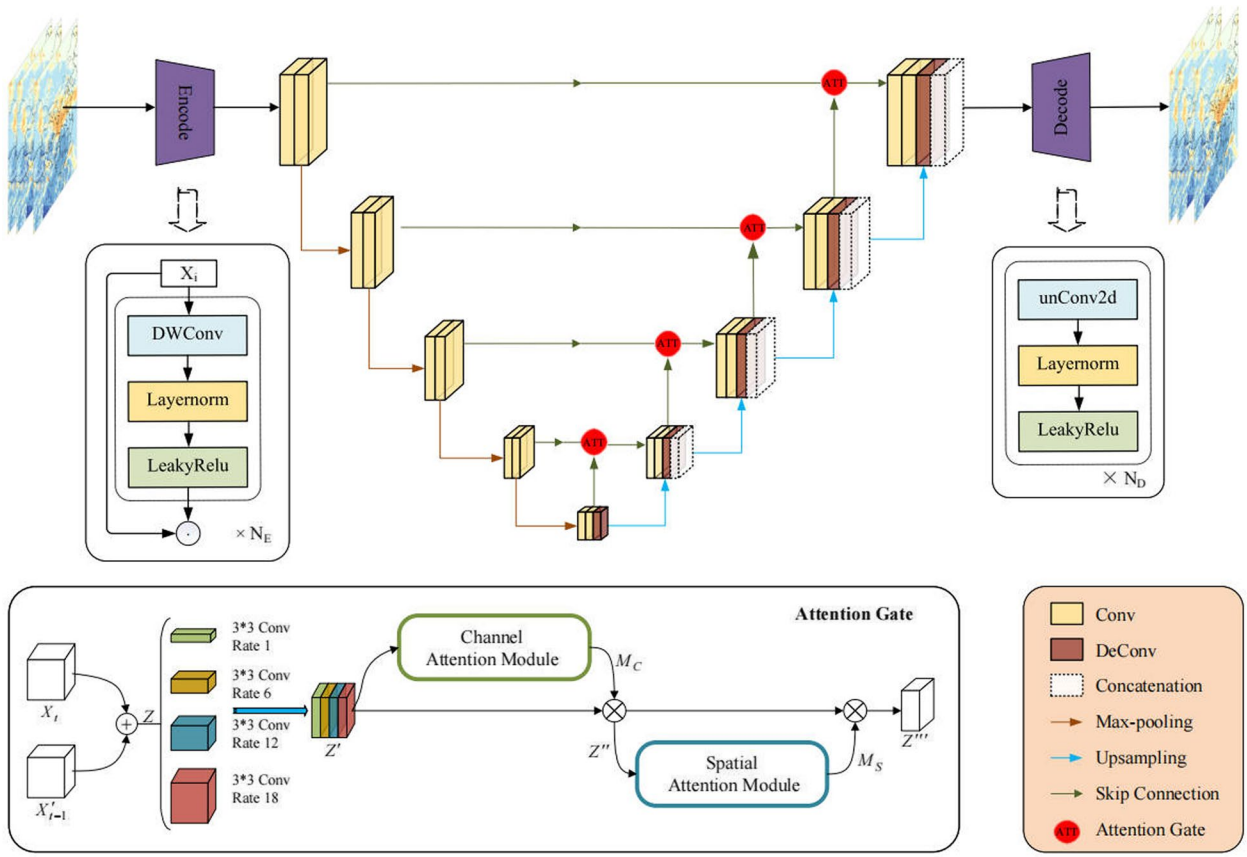

Fig. 6 Surface solar incident radiation prediction model EDA-AttUNet based on Encoder-Translator-Decoder

Among them, X_t represents the feature map of encoder t-layer, X'_{t-1} represents the feature map of encoder *t-1* layer, and X'_t represents the feature map of encoder *t*-layer. Here, *A* represents the Attention Gate function.

As shown in Fig. 5, the Attention Gate first undergoes an ASPP (Hole Space Pyramid Pooling) to convolution the input features using different hole rates, while maintaining computational efficiency while obtaining multiple receptive fields of different scales. This enables the network to simultaneously capture local details and global contextual information, improving the performance of the model. Then, the ASPP output results are fed into the channel attention mechanism and spatial attention mechanism to adaptively learn the importance of features and improve the model's expressive and perceptual abilities. Taking the t-layer as an example, the calculation formula for Attention Gate is as follows:

$$Z = Conv\left(X'_{t-1}\right) + Conv(X_t) \tag{15}$$

$$Z' = concat(b1, b6, b12, b18, mean(Conv(upsample(Z)))) \tag{16}$$

$$Z'' = M_C\left(Z'\right) \otimes Z' \tag{17}$$

$$Z''' = M_S\left(Z''\right) \otimes Z'' \tag{18}$$

Among them, X_t represents the feature map of the encoder *t-layer*, X'_{t-1} represents the feature map of the encoder *t-1* layer, *concat* represents stitching,$b1$,$b6$,$b12$ and $b18$ are the outputs of different partition rates in ASPP, *mean* represents the adaptive average pooling layer, and *upsample* represents the upsampling operation.

Decoder The decoder reconstructs the real surface solar incident radiation by stacking blocks composed of ConvTranspose2d, LayerNorm, and LeakyRelu. The expression for the decoder can be represented as follows:

$$X_k = \sigma\left(LayerNorm\left(unConv2d\left(X_{k-1}\right)\right)\right) \tag{19}$$

Among them, the input X_{k-1} and output shapes of X_k are *(T, C, H, W)*.The ConvTranspose2d mentioned in the article is represented in the formula as unConv2d.

Dynamic weighted loss function

The loss function used in this article is the sum of the weighted mean square error(MSE) and the mean absolute error(MAE) . By multiplying by the given weight, high radiation data with fewer samples can have a greater "contribution" and improve prediction accuracy. The formula is as follows:

$$loss = \frac{1}{N}\sum_{n=1}^{N}\sum_{p,q}\left(w_{n,i,q}\left(\left(\hat{y}_{n,p,q} - y_{n,p,q}\right)^2 + \left|\hat{y}_{n,p,q} - y_{n,p,q}\right|\right)\right) \tag{20}$$

The equation involves variables where $w_{n,p,q}$ denotes the weight of the radiation value at position (p, q) in the nth image, $y_{n,p,q}$ represents the radiation value at position (p, q) in the nth image, and $\hat{y}_{n,p,q}$ stands for the ground truth radiation value at position (p, q) in the nth image. The value of dynamic weight W is shown in (19):

$$W(y) = \begin{cases} 1 & 0 < y(i,j) < 200 \\ 5 & 200 <= y(i,j) < 600 \\ 20 & 600 <= y(i,j) < 1000 \\ 50 & 1000 <= y(i,j) \end{cases} \tag{21}$$

In the given context, The variable y represents the solar radiation value, measured in W/m^2. When $W = 1$, it represents a relatively low solar radiation value at locations (i, j), indicating a lower photovoltaic power generation efficiency. When $W = 5$, it represents a gradually improving solar radiation value at locations (i, j), enabling the photovoltaic system to generate a considerable amount of electricity. When $W = 20$, it represents a high solar radiation value at locations (i, j), allowing the photovoltaic system to generate a large amount of electricity. When $W = 50$, it represents a solar radiation value at locations (i, j) that maximizes the efficiency of the photovoltaic system, resulting in the highest power output.

Experiments

Study area and data

The data used in this experiment is the surface solar incident radiation(SSI) full disk data product provided by the geostationary meteorological satellite FY4A. This product considers parameters such as clouds, aerosols, water vapor content, surface albedo, and surface elevation, which can better grasp the impact of different weather conditions on solar radiation and make up for the shortage of radiation observation data in photovoltaic power generation meteorological forecasting services. The time resolution of this product is generally 1 hour, with a maximum of 15 minutes. The experimental preprocessing interpolates the parts less than 15 minutes. This experiment extracts China's regional data for training from the full disk data center based on the product manual provided by the China Meteorological Data Network. Due to the limitations of the radiation transfer software package plane parallel algorithm currently used by FY-4A, when the solar zenith angle is greater than 70 degrees, the plane parallel mode is no longer applicable due to the influence of Earth's curvature. Therefore, in the inversion process, to ensure the accuracy of the calculation results, Set the critical value of the solar zenith angle to 70, and when the critical value is exceeded, there will be no output of irradiance products. This has resulted in a large area of high latitude areas being without radiation for a long time from the end of December to the beginning of February, resulting in a small number of samples.

Therefore, utilizing the computing and storage capabilities of edge devices [29, 30], performing data quality control tasks, and optimizing data transmission and processing through edge cloud collaboration, such as transmitting data results processed on edge devices to the cloud for further analysis and storage [31]. Ensure that the edge cloud collaboration system has real-time and scalability by optimizing data transmission and processing latency, and dynamically adjusting the workload of the edge and cloud. Through this approach, edge cloud collaboration can effectively introduce meteorological data quality control and extend it to edge devices with computing and storage capabilities, improving the accuracy and reliability of meteorological data [32, 33].

We select the data for the entire year 2021, with the last 10 days of each month as the validation and testing sets, and the rest as the training set. The data preprocessing involved using bilinear interpolation to interpolate the temporal resolution of the hourly data to every 15 minutes. As a result, the final total number of samples is 28,670, with 6,820 samples in spring, 8,530 samples in summer, 8,040 samples in autumn, and 5,280 samples in winter.Each sequence has 16 radiation data in chronological order within two hours. In the experiment, the model uses the radiation maps from the first 8 hours as input to predict the radiation maps from the next 8 hours. The initial size of each radiation map is 386×256, downsampling will be performed to improve the performance of the model. Due to the large amount of solar radiation data and varying peak values at different time periods, in order to facilitate processing, it is necessary to normalize the data before training. The specific normalization method for converting the data into the 0-1 range is shown in expression (20) as follows:

$$I_{norm} = \frac{I - \min(I)}{\max(I) - \min(I)} \tag{22}$$

Among them, $\min(I)$ denotes the lowest recorded solar radiation value within the entirety of the dataset under consideration, while $\max(I)$ signifies the highest

recorded solar radiation value within the same dataset. Here, I represents a solar radiation data point.

Experiment setup and evaluation metrics

This experiment analyzes the performance of the SSI Dataset for predicting ground incident solar radiation in different seasons throughout the year 2021-2022.

In order to test the performance of the model in predicting radiation tasks, some typical benchmark models were selected for comparative experiments, including ConvLSTM [34], PhyDNet [35], E3D-LSTM [36], Traj-GRU [37], PredRNN [38], PredRNN++ [39]. All models are built using the Python framework, and equivalent parameters are used for all models in each experiment to ensure the fairness of test results. The encoder and decoder as well as N_E and N_D in the model proposed in this article are all 4. The model uses an Adam optimizer to optimize parameters. Each model uses early stop and sets the number of iterations to 50. The initial learning rate and batch size are set to 0.001 and 8, respectively. All experiments were conducted on a personal computer equipped with a Windows 10 operating system, 64.0 GB of RAM, 3.60GHz Intel (R) Core (TM) i7-11700KF CPU, and NVIDIA GeForce RTX 3090 GPU.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|\hat{y}_i - y_i\right| \tag{23}$$

$$RMSE = \frac{1}{N}\sum_{i=1}^{N}\sqrt{\left(\hat{y}_i - y_i\right)^2} \tag{24}$$

$$nRMSE = \frac{\frac{1}{N}\sum_{i=1}^{N}\sqrt{\left(\hat{y}_i - y_i\right)^2}}{y_{avg}} \times 100\% \tag{25}$$

$$SSIM\left(\hat{y}_i, y_i\right) = \frac{\left(2\mu_{y_i}\mu_{\hat{y}_i} + C_1\right)\left(2\sigma_{\hat{y}_i y_i} + C_2\right)}{\left(\mu_{y_i}^2 + \mu_{\hat{y}_i}^2 + C_1\right)\left(\sigma_{y_i}^2 + \sigma_{\hat{y}_i}^2 + C_2\right)} \tag{26}$$

In the formula, $\hat{y}_i$ represents the *i-th* predicted value, and y_i represents the *i-th* true value. y_{avg} and $\hat{y}_{avg}$ represent the average predicted value and the average true value, while N represents the current total number of observations. μ is the average value, and σ is the standard deviation. C_1 and C_2 are constants used to prevent the denominator from reaching zero when μ and σ are too small.

Experiment results and analysis

In Fig. 7, we show the visualization of the predicted results of each model within two hours, with a time resolution of 15 minutes. From the graph, it can be seen that the PhyDNet and ConvLSTM models have weaker ability to extract spatiotemporal changes. Although the ConvLSTM model made adjustments to the changes on the right side after one hour, it is clear that the model did not truly learn the spatiotemporal characteristics of radiation variations. It is unable to extract effective spatiotemporal state changes at 60 minutes, and no corresponding movement state changes are made in the following hour. Although the PredRNN model made adjustments to the changes in spatiotemporal state within 60 minutes, it was not outstanding and to some extent relied on the performance of the previous moment. The PredRNN++ model, with its stacked structure compared to the single-layer recurrent prediction units of PredRNN, exhibits better temporal modeling capabilities, resulting in improved predictive performance. However, it is worth noting that the training time for PredRNN++ is twice as long as PredRNN. Finally, the model proposed in this article not only significantly predicted the true distribution of solar radiation within 60 minutes, but also predicted the distribution changes of radiation more accurately after 60 minutes.

The results of various indicators for all models from 2021 to 2022 are displayed in Table 1. From the table, it can be seen that the performance indicators of our model have achieved the best.The SSIM metric reached 0.86, and both PredRNN and PredRNN++ achieved high levels of SSIM, but from our experimental process, it is evident that they required significantly more training time compared to EDA-AttUNet. ConvLSTM, PhyDNet, and Traj GRU are far inferior to the other models in terms of both visual results and experimental indicators.

The comparison of root mean square error indicators for 8 predicted time steps of surface solar incident radiation using different models from 2021 to 2022 is shown in Table 2. From the table, it can be seen that our model not only achieved good results within 1 hour, but also achieved better results within 1-2 hours. Although the PredRNN and PredRNN++ models performed slightly better in the initial prediction time steps, their performance declined more significantly in subsequent predictions. In contrast, EDA-AttUNet demonstrated greater stability as the prediction horizon increased. Similarly, PhyDNet performed better than Traj-GRU in predicting within 1 hour, but with increasing prediction frequency, the root mean square error increased more significantly. The E3D-LSTM model also struggles to accurately predict the distribution of radiation as the prediction horizon increases.

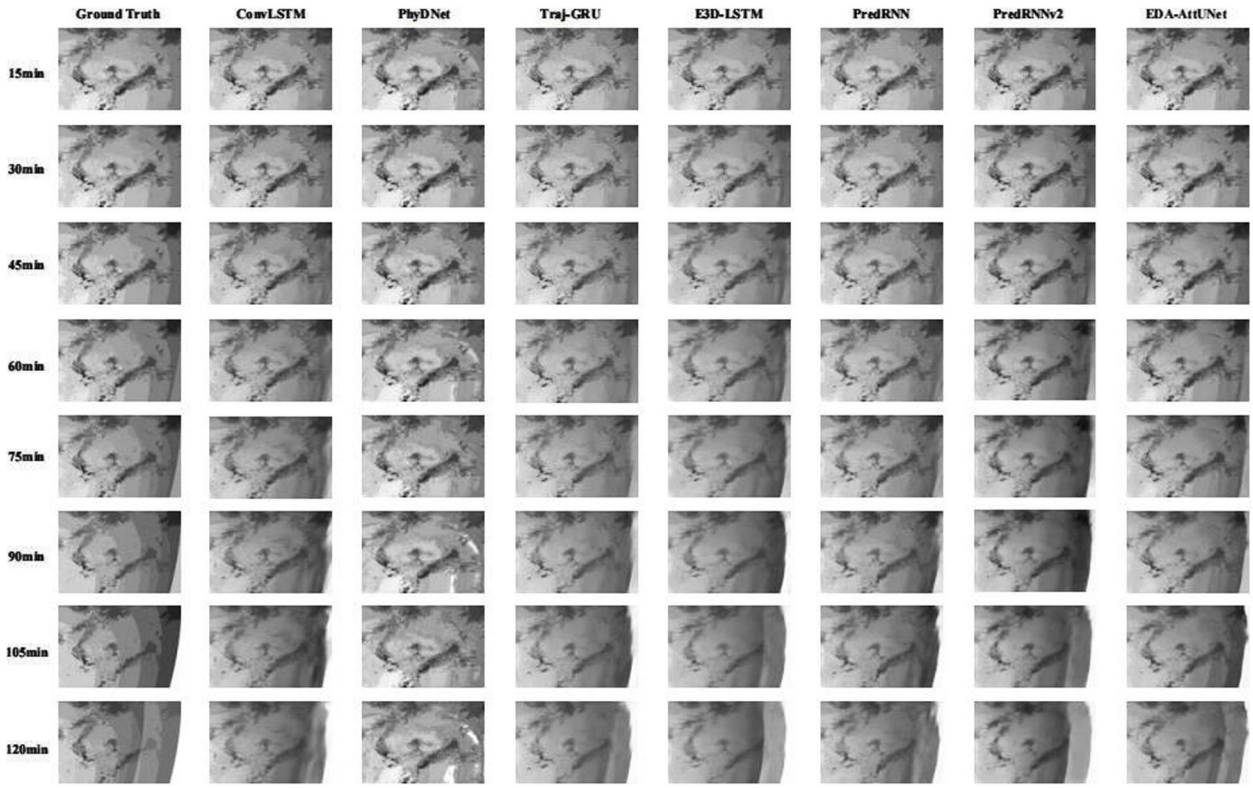

Fig. 7 Prediction examples of each model relative to the true value of SSI. 15 minutes, 30 minutes, 45 minutes, 60 minutes, 75 minutes, 90 minutes, 105 minutes, and 120 minutes refer to the future predicted time relative to the initial start time at 2021-07-22 09:59

Table 1 Comparison of all statistical indicators for all models throughout the year

Metric	ConvLSTM	PhyDNet	Traj-GRU	E3D-LSTM	PredRNN	PredRNN++	EDA-AttUNet
MAE	114.74	103.19	94.57	97.49	96.25	92.65	**89.56**
RMSE	167.82	148.26	157.63	150.22	141.34	138.77	**134.2**
nRMSE	35.52	31.48	33.81	31.97	30.88	30.04	**29.47**
SSIM	0.73	0.82	0.78	0.80	0.83	0.84	**0.86**

Table 2 Comparison of Root Mean Square Error indicators for prediction results of different time steps of various models throughout the year

Model	Step1	Step2	Step3	Step4	Step5	Step6	Step7	Step8
ConvLSTM	100.08	109.65	121.06	133.25	154.93	183.92	211.48	235.49
PhyDNet	96.52	103.82	115.22	128.47	151.91	178.44	195.59	227.88
Traj-GRU	97.88	104.77	119.31	130.21	150.32	174.46	190.93	224.31
E3D-LSTM	95.24	101.92	116.01	125.44	142.09	163.44	180.19	207.23
PredRNN	**89.29**	**96.48**	114.61	127.23	145.37	169.23	187.04	211.19
PredRNN++	**87.11**	**93.22**	**107.42**	**124.23**	141.50	162.91	184.75	209.41
EDA-AttUNet	90.47	97.84	109.37	124.74	**139.93**	**158.37**	**175.74**	**192.93**

Due to the seasonal influence of solar radiation, the results of various performance indicators in spring, summer, and autumn from 2021 to 2022 are presented in Table 3. In the inversion algorithm of FY-4A, the solar zenith angle of 70 degrees is the critical value, which results in a small number of effective samples in some regions of China during winter. Therefore, seasonal testing is not conducted here. From the table, it can be seen that the error in spring is higher than that in summer and autumn. It is worth noting that the PredRNN and PredRNN++ models have lower MAE values in summer compared to the proposed model, with a difference of 0.59 and 2.74, and lower RMSE values with a difference of 13.79 and 15.83. We tentatively attribute this to the significantly larger number of effective samples and higher radiation values during the summer season. The stacked structure of PredRNN++ clearly has an advantage in handling such data.

Conclusion

The paper introduces a novel encoder-decoder based on AttUNet, which incorporates an attention mechanism. This enhancement aims to capture the spatial variations and temporal dependencies of radiation motion, thereby improving the model's ability to evolve with radiation dynamics. Compared to traditional methods, this model can better capture the complex spatial and temporal characteristics of radiation motion, thereby improving the accuracy of prediction. In addition, this method also has good generalization ability and is suitable for radiation prediction in different regions and time scales. The experimental results of the proposed model demonstrate its effectiveness in practical radiation forecasting.Future research will focus on advancing the integration of satellite data with ground observations, as well as considering the impact of weather conditions on solar radiation to enhance the accuracy of radiation prediction.

Table 3 Comparison of all performance indicators for spring, summer, and autumn 2021-2022

Season	Model	MAE	RMSE	nRMSE	SSIM
Spring	ConvLSTM	118.25	171.49	39.80	0.71
	PhyDNet	107.01	160.38	36.21	0.76
	Traj-GRU	111.24	165.91	38.04	0.74
	E3D-LSTM	108.14	160.33	36.04	0.77
	PredRNN	101.78	155.05	34.26	0.78
	PredRNN++	100.78	153.05	32.26	0.80
	EDA-AttUNet	**97.22**	**147.71**	**32.55**	**0.81**
Summer	ConvLSTM	112.11	167.28	38.12	0.74
	PhyDNet	104.37	151.29	34.83	0.83
	Traj-GRU	106.33	157.44	35.14	0.80
	E3D-LSTM	104.24	155.91	37.32	0.76
	PredRNN	93.61	135.71	29.55	0.87
	PredRNN++	**91.46**	**133.22**	**28.43**	**0.89**
	EDA-AttUNet	94.20	149.05	31.16	0.84
Autumn	ConvLSTM	109.48	166.44	37.32	0.74
	PhyDNet	99.88	148.65	32.35	0.82
	Traj-GRU	103.71	153.29	34.04	0.81
	E3D-LSTM	101.24	147.91	33.01	0.79
	PredRNN	91.98	144.01	32.15	0.83
	PredRNN++	89.28	140.21	31.33	0.85
	EDA-AttUNet	**87.27**	**130.22**	**27.15**	**0.88**

Authors' contributions

Cui: Conceptualization, Methodology, Software, Data curation. Zhao: Investigation, Writing-Reviewing and Editing. Yao:Formal analysis,Supervision, Writing-Reviewing and Editing.

Funding

This work was supported by the National Natural Science Foundation of China under Grant (No. 92267104 and 62372242), China Meteorological Service Association Meteorological Technology Innovation Project (No. CMSA2023MD004) and in part by Natural Science Foundation of Jiangsu Province of China under Grant (No. BK20211284).

Availability of data and materials

The dataset and materials generated in this research process can be obtained from the corresponding author upon reasonable request.

Declarations

Ethics approval and consent to participate

The research in this paper does not involve any illegal or unethical practices.

Consent for publication

The authors read and approved the final manuscript.

Competing interests

The authors declare no competing interests.

Received: 25 October 2023 Accepted: 1 March 2024
Published online: 22 March 2024

References

1. Gürel AE, Ağbulut Ü, Biçen Y (2020) Assessment of machine learning, time series, response surface methodology and empirical models in prediction of global solar radiation. J Clean Prod 277:122353. https://doi.org/10.1016/j.jclepro.2020.122353
2. Osman AI et al (2023) Cost, environmental impact, and resilience of renewable energy under a changing climate: a review. Environ Chem Lett 21(2):741–764. https://doi.org/10.1007/s10311-022-01532-8
3. Jung J, Onen A, Arghandeh R, Broadwater RP (2014) Coordinated control of automated devices and photovoltaic generators for voltage rise mitigation in power distribution circuits. Renew Energy 66:532–540. https://doi.org/10.1016/j.renene.2013.12.039
4. Paulescu M, Paulescu E, Gravila P, Badescu V (2013) Weather Modeling and Forecasting of PV Systems Operation. in Green Energy and Technology. Springer London, London. https://doi.org/10.1007/978-1-4471-4649-0
5. Kumari P, Toshniwal D (2021) Extreme gradient boosting and deep neural network based ensemble learning approach to forecast hourly solar irradiance. J Clean Prod 279:123285. https://doi.org/10.1016/j.jclepro.2020.123285

6. Hu Z et al (2022) Cloud-edge cooperation for meteorological radar big data: a review of data quality control. Complex Intell Syst 8(5):3789–3803. https://doi.org/10.1007/s40747-021-00581-w
7. Thaker J, Höller R (2023) Evaluation of High Resolution WRF Solar. Energies 16(8):3518. https://doi.org/10.3390/en16083518
8. Deo RC et al (2023) Cloud cover bias correction in numerical weather models for solar energy monitoring and forecasting systems with kernel ridge regression. Renew Energy 203:113–130. https://doi.org/10.1016/j.renene.2022.12.048
9. Voyant C et al (2017) Machine learning methods for solar radiation forecasting: A review. Renew Energy 105:569–582. https://doi.org/10.1016/j.renene.2016.12.095
10. Huang X et al (2021) Hybrid deep neural model for hourly solar irradiance forecasting. Renew Energy 171:1041–1060. https://doi.org/10.1016/j.renene.2021.02.161
11. Huang J, Liu H (2021) A hybrid decomposition-boosting model for short-term multi-step solar radiation forecasting with NARX neural network. J Cent South Univ 28(2):507–526. https://doi.org/10.1007/s11771-021-4618-9
12. Li D, Zhang H, Cheng J, Liu B (2024) Improving efficiency of DNN-based relocalization module for autonomous driving with server-side computing. J Cloud Comp 13(1):25. https://doi.org/10.1186/s13677-024-00592-1
13. Van Houdt G, Mosquera C, Nápoles G (2020) A review on the long short-term memory model. Artif Intell Rev 53(8):5929–5955. https://doi.org/10.1007/s10462-020-09838-1
14. Duan J et al (2023) A multistep short-term solar radiation forecasting model using fully convolutional neural networks and chaotic aquila optimization combining WRF-Solar model results. Energy 271:126980. https://doi.org/10.1016/j.energy.2023.126980
15. Pérez E, Pérez J, Segarra-Tamarit J, Beltran H (2021) A deep learning model for intra-day forecasting of solar irradiance using satellite-based estimations in the vicinity of a PV power plant. Sol Energy 218:652–660. https://doi.org/10.1016/j.solener.2021.02.033
16. Feng C, Zhang J (2020) SolarNet: A sky image-based deep convolutional neural network for intra-hour solar forecasting. Sol Energy 204:71–78. https://doi.org/10.1016/j.solener.2020.03.083
17. Wang Q, Wu B, Zhu P, Li P, Zuo W, Hu Q (2020) ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. arXiv. [Online]. http://arxiv.org/abs/1910.03151. Accessed 19 Aug 2023
18. Nielsen AH, Iosifidis A, Karstoft H (2021) IrradianceNet: Spatiotemporal deep learning model for satellite-derived solar irradiance short-term forecasting. Sol Energy 228:659–669. https://doi.org/10.1016/j.solener.2021.09.073
19. Zhang L, Wilson R, Sumner M, Wu Y (2023) Advanced multimodal fusion method for very short-term solar irradiance forecasting using sky images and meteorological data: A gate and transformer mechanism approach. Renew Energy 216:118952. https://doi.org/10.1016/j.renene.2023.118952
20. Ca V, Mannem R, Ghosh PK (2018) Air-Tissue Boundary Segmentation in Real-Time Magnetic Resonance Imaging Video Using Semantic Segmentation with Fully Convolutional Networks. In Interspeech 2018. ISCA, pp 3132–3136. https://doi.org/10.21437/Interspeech.2018-1939
21. Zhan Y, Wang J, Shi J, Cheng G, Yao L, Sun W (2017) Distinguishing Cloud and Snow in Satellite Images via Deep Convolutional Network. IEEE Geosci Remote Sens Lett 14(10):1785–1789. https://doi.org/10.1109/LGRS.2017.2735801
22. Humayun M, Alsirhani A, Alserhani F, Shaheen M, Alwakid G (2024) Transformative synergy: SSEHCET-bridging mobile edge computing and AI for enhanced eHealth security and efficiency. J Cloud Comp 13(1):37. https://doi.org/10.1186/s13677-024-00602-2
23. Lin M, Chen Q, Yan S (2014) Network In Network. arXiv. [Online]. http://arxiv.org/abs/1312.4400. Accessed 18 Sep 2023
24. Chollet F (2017) Xception: Deep Learning with Depthwise Separable Convolutions. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, Honolulu, pp 1800–1807. https://doi.org/10.1109/CVPR.2017.195
25. Xiao L, Cao Y, Gai Y, Khezri E, Liu J, Yang M (2023) Recognizing sports activities from video frames using deformable convolution and adaptive multiscale features. J Cloud Comp 12(1):167. https://doi.org/10.1186/s13677-023-00552-1
26. Woo S, Park J, Lee J-Y, Kweon IS (2018) CBAM: Convolutional Block Attention Module. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y (eds) Computer Vision - ECCV 2018. Lecture Notes in Computer Science, vol 11211. Springer International Publishing, Cham, pp 3–19. https://doi.org/10.1007/978-3-030-01234-2-1
27. Oktay O, et al (2018) Attention U-Net: Learning Where to Look for the Pancreas. arXiv. [Online]. http://arxiv.org/abs/1804.03999. Accessed 21 Sep 2023
28. Ronneberger O, Fischer P, Brox T (2015) U-Net: Convolutional Networks for Biomedical Image Segmentation. arXiv. [Online]. http://arxiv.org/abs/1505.04597. Accessed 21 Sep 2023
29. Xu X, Tang S, Qi L, Zhou X, Dai F, Dou W (2023) CNN Partitioning and Offloading for Vehicular Edge Networks in Web3. IEEE Commun Mag 61(8):36–42. https://doi.org/10.1109/MCOM.002.2200424
30. Li Z, Li G, Bilal M, Liu D, Huang T, Xu X (2023) Blockchain-Assisted Server Placement With Elitist Preserved Genetic Algorithm in Edge Computing. IEEE Internet Things J 10(24):21401–21409. https://doi.org/10.1109/JIOT.2023.3290568
31. Xu X, Liu Z, Bilal M, Vimal S, Song H (2022) Computation Offloading and Service Caching for Intelligent Transportation Systems With Digital Twin. IEEE Trans Intell Transport Syst 23(11):20757–20772. https://doi.org/10.1109/TITS.2022.3190669
32. Zhou X et al (2023) Edge Computation Offloading With Content Caching in 6G-Enabled IoV. IEEE Trans Intell Transport Syst 1–15. https://doi.org/10.1109/TITS.2023.3239599
33. Xu X, Gu J, Yan H, Liu W, Qi L, Zhou X (2023) Reputation-Aware Supplier Assessment for Blockchain-Enabled Supply Chain in Industry 4.0. IEEE Trans Ind Inf 19(4):5485–5494. https://doi.org/10.1109/TII.2022.3190380
34. Shi X, Chen Z, Wang H, Yeung D-Y, Wong W, Woo W (2015) Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. arXiv. [Online]. http://arxiv.org/abs/1506.04214. Accessed 19 Aug 2023
35. Le Guen V, Thome N (2020) Disentangling Physical Dynamics From Unknown Factors for Unsupervised Video Prediction. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, Seattle, pp 11471–11481. https://doi.org/10.1109/CVPR42600.2020.01149
36. Wang Y, Jiang L, Yang M-H, Li L-J, Long M, Fei-Fei L (2019) Eidetic 3D LSTM: A Model for Video Prediction and Beyond. International Conference on Learning Representations (ICLR). https://openreview.net/forum?id=B1lKS2AqtX
37. Shi X, et al (2017) Deep Learning for Precipitation Nowcasting: A Benchmark and A New Model. arXiv. [Online]. http://arxiv.org/abs/1706.03458. Accessed 08 Oct 2023
38. Wang Y, et al (2022) PredRNN: A Recurrent Neural Network for Spatiotemporal Predictive Learning. arXiv. [Online]. http://arxiv.org/abs/2103.09504. Accessed 19 Aug 2023
39. Wang Y, Gao Z, Long M, Wang J, Yu PS (2018) PredRNN++: Towards A Resolution of the Deep-in-Time Dilemma in Spatiotemporal Predictive Learning. arXiv. [Online]. http://arxiv.org/abs/1804.06300. Accessed 31 Jan 2024

Publisher's Note