The Innovative Solution for Carbon Emission Prediction: Combining Quantum Particle Swarm Optimization and GRU with Dynamic Attention Mechanism

Muhammad Umair*

Georgia State University J. Mack Robinson College of Business, Georgia, USA; mumair1@student.gsu.edu

*Corresponding Author: mumair1@student.gsu.edu

DOI: https://doi.org/10.30212/JITI.202604.008

Submitted: Feb 11, 2026 Accepted: Apr 21, 2026

ABSTRACT

Carbon neutrality, also known as net-zero carbon emissions, is a crucial strategy aimed at addressing the global climate crisis by reducing, offsetting, or capturing carbon emissions associated with human activities. Currently, many countries and organizations are setting carbon neutrality targets and taking action to reduce carbon emissions. However, these efforts require accurate carbon emission predictions to ensure that the formulated policies and measures can be successful. Additionally, traditional carbon emission prediction methods often rely on statistical models and historical data, which struggle to cope with the complexity and variability of the climate system. This results in inaccurate predictions in situations such as extreme weather events, impacting policy development and implementation of mitigation measures. To address this issue, we propose an innovative carbon emission prediction method that combines QPSO and GRU, incorporating an attention mechanism. The advantage of the QPSO algorithm is its ability to search for the optimal solution, optimizing the model's hyperparameters by considering global information, thereby improving the model's adaptability and prediction performance. Meanwhile, the GRU takes full account of the temporal nature of time-series data, better capturing dynamic patterns within the data. Experimental results indicate the outstanding performance of the QPSO-GRU model in carbon emission prediction. It can more accurately capture trends and changes in carbon emissions, providing governments and businesses with more reliable data for formulating emission reduction policies and planning carbon neutrality measures. This not only enhances the efficiency of climate change mitigation but also contributes to achieving global carbon neutrality goals.

Keywords: Carbon neutrality, Artificial intelligence, Energy optimization, Emergency management, QPSO-GRU, Sustainable development

1. Introduction

Carbon neutrality is one of the key strategies for addressing the global climate crisis [1,2]. However, achieving carbon neutrality faces several major challenges. However, carbon neutrality

faces several significant challenges. Firstly, achieving carbon neutrality requires substantial reductions in carbon emissions on a large scale, involving efforts across various industries and societal sectors, including energy production, transportation, industrial manufacturing, and construction [3]. Secondly, while carbon offsetting and carbon capture technologies are continually advancing, their feasibility, cost-effectiveness, and environmental impacts remain subjects of debate [4]. Thirdly, achieving carbon neutrality requires global collaborative action, involving the development and enforcement of international policies and regulations, which presents a complex challenge. Currently, deep learning has become a crucial tool in researching carbon neutrality and addressing climate change. Utilizing deep learning techniques, researchers have developed highly accurate carbon emission prediction models. These models can analyze extensive meteorological, energy usage, and industrial data, aiding governments and businesses in formulating effective emission reduction strategies. Furthermore, deep learning can be applied to optimize carbon capture and storage systems, enhancing their efficiency and feasibility [5]. This contributes to the development of more advanced carbon capture technologies, ultimately reducing carbon emissions from industrial processes. Lastly, deep learning's applications in environmental monitoring and climate modeling have also made significant progress. It can assist scientists in more accurately predicting the impacts of climate change, providing additional insights and information for climate change mitigation efforts. Accurate carbon emission predictions hold immense significance in several ways. Firstly, they facilitate the formulation of targeted emission reduction policies by governments and international organizations. This helps in the equitable allocation of resources, the promotion of emission reduction actions, and the monitoring of policy effectiveness. Additionally, improving indoor environments to reduce high carbon emissions not only benefits the environment but also helps in saving energy and resource [6]. Enhanced energy efficiency in building designs and operations can lead to reduced energy bills, increased energy utilization efficiency, and a decreased demand for finite energy resources. These advantages are economically beneficial for both businesses and individuals while also contributing to long-term sustainability.

Currently, some indoor spaces often require a significant supply of electricity for lighting, air conditioning, and other purposes. They have not adequately incorporated natural light and ventilation into their architectural designs, resulting in unnecessary energy consumption. With the advancement of technology, people have raised higher expectations for sustainability and environmental impact. Therefore, there is an urgent need for an efficient and accurate method to predict carbon emissions. Below, we will introduce some of the latest developments in research in this area:

LSTM is a specialized type of RNN designed specifically for handling sequential data [7]. In the field of carbon emissions forecasting, LSTM demonstrates greater accuracy in predicting future energy consumption trends when processing energy consumption data. However, like traditional RNNs, LSTM may also encounter the issue of vanishing or exploding gradients during the training process, particularly when dealing with long sequences of data and long-term dependencies.

Temporal Convolutional Network (TCN) is a time series model based on convolutional neural networks, with its name derived from its use of convolutional operations along the time dimension

[8]. In carbon emissions prediction, TCN can manage larger datasets, making it more suitable for predicting energy consumption in such contexts. Furthermore, TCN can capture patterns and dependencies over extended time ranges, which aids in more accurately forecasting energy consumption trends. However, when energy consumption data involves complex nonlinear relationships, TCN's performance might decline.

Transformer is a potent neural network model initially designed for natural language processing tasks, later extended to sequence data analysis and prediction [9]. In carbon emissions prediction, the Transformer can process sequence data in parallel, significantly enhancing computational efficiency. Nonetheless, it lacks an explicit sequential processing mechanism, which means that for certain sequence data heavily reliant on the order information, it might not effectively capture their inherent temporal dependencies.

The CNN-LSTM model combines CNN and LSTM for processing sequence data [10]. It can effectively capture both spatial and temporal features, thereby improving the accuracy of energy consumption forecasts. This model finds widespread application in energy consumption prediction. However, for the CNN-LSTM model, additional data may be necessary, especially when employing multiple convolutional and LSTM layers, to prevent overfitting.

Based on the limitations of previous research, this paper proposes a method that utilizes a dynamic attention mechanism to optimize the QPSO-GRU model. By introducing the dynamic attention mechanism, the model can adaptively focus on different features and their importance when processing sequential and spatial data. Through this approach, the model can better capture the spatiotemporal correlations in the data, thus enhancing prediction accuracy. This method proves highly effective in handling various data types and generating data that satisfies specific conditions. Accurately predicting carbon emissions can assist governments and businesses in formulating environmental policies and reducing carbon footprints. Furthermore, this model holds significant potential for applications in climate change modeling, energy management, urban planning, and club management, among other fields. In our forthcoming research, we will further validate and optimize this method to ensure its robustness and scalability across different domains and datasets.

This study delivers several notable advancements, detailed as follows:

- This paper introduce a novel dynamic attention component, which empowers the QPSO-GRU framework to dynamically prioritize features of differing significance during the analysis of sequential and spatial datasets. This adjustment boosts the model's adaptability and predictive precision when addressing intricate data scenarios.
- QPSO-GRU adopts the quantum-behaved particle swarm optimization algorithm, which has stronger global search and parameter tuning abilities than traditional methods, facilitating optimal parameter combinations and improving time-series prediction performance.
- This method enables precise carbon emissions forecasting, which facilitates the development of more effective environmental policies and reduces carbon footprints.

2. Related Work

2.1 Deep Learning-Based Carbon Emission Forecasting Research

In the fields of climate science and environmental studies, deep learning is playing an increasingly vital role, particularly in carbon emission forecasting and climate modeling.

As a class of deep learning models tailored for sequence data, RNNs demonstrate outstanding performance in processing time series and meteorological data [11]. Their unique memory mechanism endows the model with the ability to capture internal temporal dependencies within data, a characteristic that lends significant application value in climate modeling—enabling precise predictions of future meteorological conditions such as temperature, humidity, and wind speed. Such predictive information is crucial for ensuring the accuracy of climate models, thereby providing key technical support for analyzing climate change trends and issuing early warnings for extreme weather events. CNNs are primarily used for image processing and have also found extensive application in the field of environmental scienc [12]. For example, satellite imagery and Geographic Information System (GIS) data can be used to monitor environmental factors such as land use changes, forest coverage, and urban expansion. CNNs can automatically extract features from this data, helping scientists better understand environmental changes. Additionally, CNNs can be used for pattern recognition in atmospheric and ocean simulations, aiding in the prediction of various phenomena in meteorology and oceanography, such as hurricanes, floods, and ocean currents. Generative Adversarial Networks (GANs) are another deep learning model comprising two parts: a generator and a discriminator [13]. GANs have been used to generate climate data such as temperature, rainfall, and wind speed. This is particularly useful for simulating uncertainty in meteorological data and predicting extreme events. By training the generator and discriminator in a competitive manner, GANs can generate meteorological data similar to real data, providing more data samples for climate modeling and analysis.

However, deep learning still faces certain limitations in the field of environmental science. First, deep learning models typically require large amounts of data for training, and data collection may be constrained in certain environmental domains. Additionally, deep learning models are often regarded as black-box models that make decision-making processes difficult to interpret. In certain application scenarios, such as weather forecasting and climate modeling, model interpretability is crucial. Finally, deep learning models may require support from large-scale computing clusters, which poses a challenge for some research teams.

2.2 Time Series-Based Carbon Emission Forecasting Research

In research related to time series carbon emission forecasting, researchers typically explore various deep learning models, including Informer, BiGRU, and Transformer, among others. These models have demonstrated robust performance in modeling and predicting time series data.

To begin with, the Informer model is specifically designed for time series data [14]. It effectively captures long-term dependencies and seasonal patterns within time series data. Consequently, researchers have widely applied the Informer model to carbon emission forecasting, where it demonstrates exceptional performance, particularly when handling time series data spanning extended intervals. Next, the BiGRU model (Bidirectional Gated Recurrent Unit) can simultaneously

integrate both forward and backward temporal information propagation [15]. This makes it highly useful for modeling temporal relationships within time series data. In the field of carbon emission forecasting, researchers often employ BiGRU because of its effectiveness in handling gradually changing patterns in time series data, capturing dynamic variations. Lastly,the Transformer model, originally designed for natural language processing, has now been applied to time series data modelin [13]. The Transformer model excels at capturing long-range dependencies and global patterns within time series data, making it the preferred solution for carbon emission forecasting. It demonstrates particularly strong application potential for tasks requiring the integration of information across multiple time steps.

Each of these models has its own advantages and suitable scenarios. Researchers can choose the most appropriate model based on the nature of their data and the requirements of their forecasting tasks. The widespread application of deep learning models offers new tools and methodologies for carbon emission forecasting, with the potential to enhance our understanding and prediction of environmental impacts.

3. Ethodology

3.1 Overview of Our Network

Our hybrid approach (for energy consumption prediction and similar sequence tasks) centers on the QPSO-GRU framework, refined with dynamic attention. We integrate QPSO (Quantum Particle Swarm Optimization) with the GRU (Gated Recurrent Unit), supplementing this combination with the attention component to boost the model's predictive accuracy.

QPSO iterates to identify optimal solutions using its own experience and alignment with the global optimum. For energy consumption prediction, it optimizes GRU hyperparameters (e.g., hidden layer size, learning rate) to boost the model's adaptation to training data and predictive performance. GRU regulates information flow via reset and update gates, mitigating the vanishing gradient issue and excelling in sequence data modeling. Finally, integrating a dynamic attention mechanism into the energy consumption prediction framework lets the model prioritize key time steps or features, strengthening its capture of critical information and further elevating prediction accuracy.

Optimizing the QPSO-GRU model through a dynamic attention mechanism is a hybrid approach that combines optimization algorithms and neural networks, enhancing the model's ability to focus on sequence data. In the future, we will continuously refine and optimize this algorithm to make it more effective in various domains and bring greater value to practical applications. Additionally, we will collaborate with researchers and practitioners in related fields to further validate the algorithm's effectiveness and promote its implementation and dissemination in real-world applications.

Figure 1 is the overall flow chart.

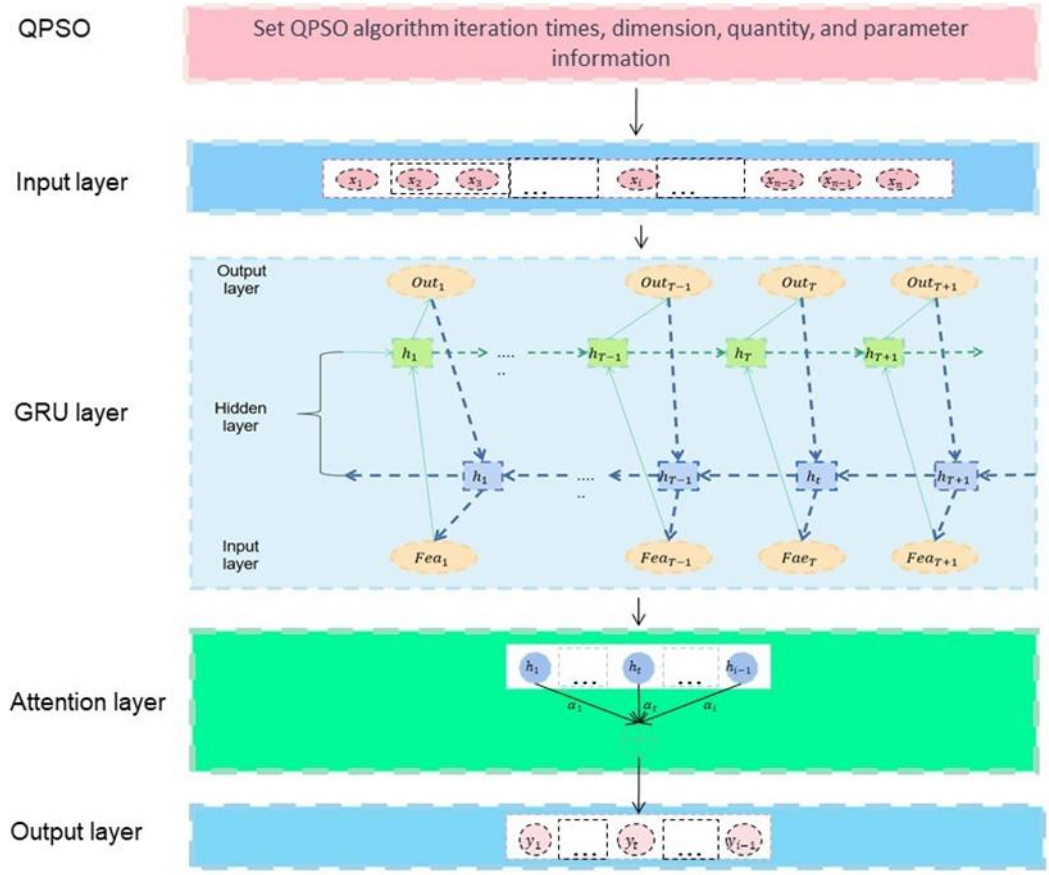

Figure 1. Overall flow chart of the model

3.2 GRU Model

GRU (Gated Recurrent Unit) is a Recurrent Neural Network (RNN) variant, developed to mitigate core limitations of traditional RNNs (e.g., gradient-related issues)[16]. By incorporating gating mechanisms into the RNN framework, GRU better captures long-term patterns in sequential data while reducing model complexity. Figure 2 displays the structural flowchart for GRU.

The structure of GRU includes the following key components:

Update Gate: Determines whether the information from the previous time step should be passed to the current time step. It outputs a value between 0 and 1 through a sigmoid activation function, where 1 indicates fully updating the previous time step's state, and 0 indicates completely ignoring the previous time step's state.

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z) \tag{1}$$

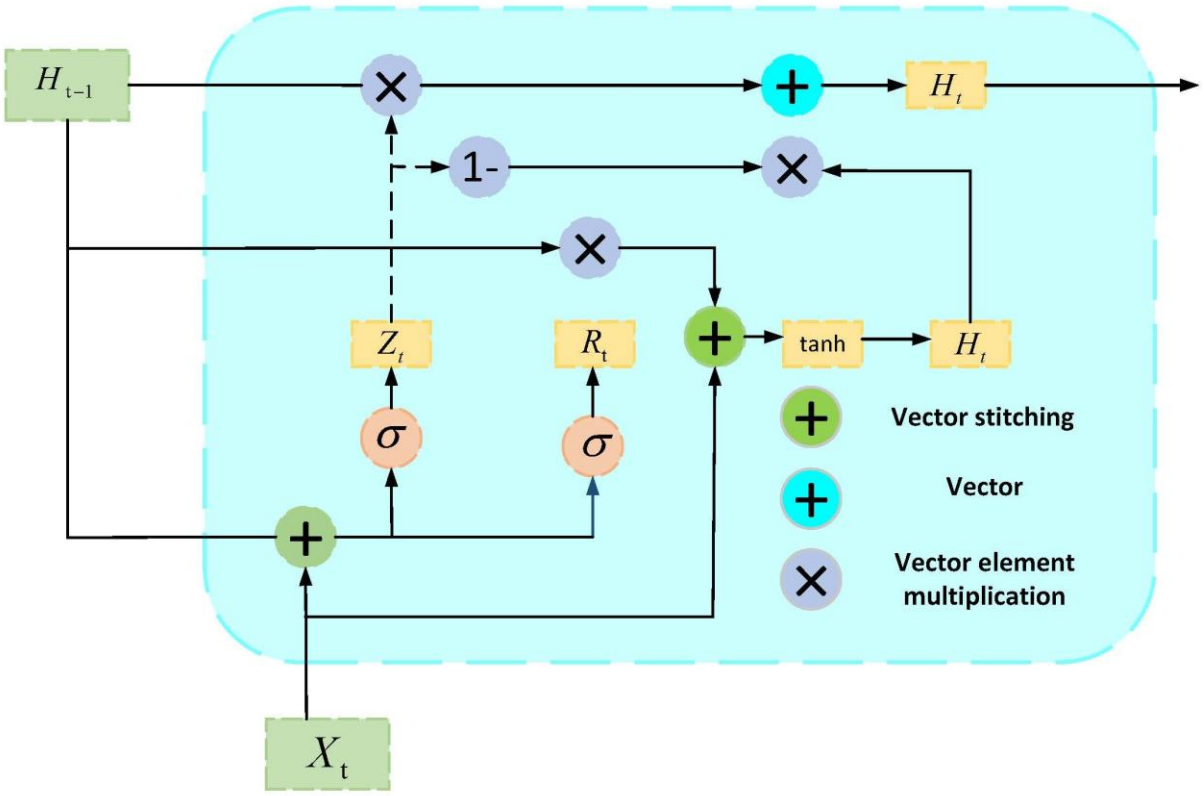

Figure 2. The architecture of GRU cell

Reset Gate: It determines how much past information is forgotten at the current time step, ontrolling the amount of information from the previous time step's hidden state that can be passed to the current time step.

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \tag{2}$$

Candidate Hidden State: At each time step, the GRU calculates a candidate hidden state, which is used to update the hidden state at the current time step. This candidate hidden state is computed by combining the current input and the hidden state from the previous time step.

$$\hat{h}_t = \tanh(W_h \cdot [r_t \odot h_{t-1}, x_t] + b_h) \tag{3}$$

Updating Hidden State: Using the update gate, the model performs a weighted average of the candidate's hidden state and the hidden state from the previous time step, resulting in the hidden state at the current time step. This process allows the model to retain or forget past information based on the needs of the task and capture dependencies more effectively, especially in long sequences.

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \hat{h}_t \tag{4}$$

Where h_t represents the current hidden state at time step t, x_t denotes the input at the current time

step, W_z, W_r, and W_h are parameter matrices, b_z, b_r, and b_h are bias vectors, σ is the sigmoid function, and $\odot$ denotes element-wise multiplication.

In this study, the GRU component is pivotal to the optimized QPSO-GRU framework (integrated with dynamic attention). Traditional QPSO (Quantum-behaved Particle Swarm Optimization)-GRU setups rely on static attention weights, which fail to adjust to input data's varying relevance. By contrast, the GRU's gating mechanism lets the framework dynamically tune attention weights using input context—strengthening capture of temporal data dependencies, and boosting both predictive performance and generalization. Additionally, the GRU component leverages historical energy consumption data to identify usage patterns, forecast future demand, and inform optimized energy utilization strategies.

3.3 OPSO model

QPSO (Quantum-behaved Particle Swarm Optimization) is a swarm intelligence-based optimization algorithm, which emulates quantum particle behaviors and interactions to identify optimal solutions [16]. In the QPSO framework, a set of particles is placed in the search space; each particle represents a candidate solution. These particles update their positions by referencing both other particles' strong solutions and their own historical best results, gradually exploring toward potential optimal outcomes.Figure 3 displays the flowchart for QPSO.

The following describes the updating of particle positions and velocities, as well as the computation of the fitness function:

Position Update Formula:

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{5}$$

Where, $x_i(t+1)$ represents the position of particle i at time step $t+1$, $x_i(t)$ represents the position of particle i at time step t, and $v_i(t+1)$ represents the velocity of particle i at time step $t+1$.

Velocity Update Formula:

$$v_i(t+1) = \chi(v_i(t) + c_1 \cdot r_1 \cdot (pbest_i(t) - x_i(t)) + c_2 \cdot r_2 \cdot (gbest(t) - x_i(t))) \tag{6}$$

Where, $v_i(t+1)$ represents the velocity of particle i at time step $t+1$, χ is the velocity limitation factor, c_1 and c_2 are learning factors, r_1 and r_2 are random numbers, $pbest_i(t)$ represents the individually best solution of particle i at time step t, and $gbest(t)$ represents the global best solution.

In this study, QPSO leverages its global search capability to find potential optimal solutions. Simultaneously, by integrating the GRU model, the system can handle time series data more effectively, thereby improving the efficiency and accuracy of energy consumption management.

3.4 Dynamic Attention Mechainism

The Dynamic Attention Mechanism is a type of attention mechanism used to enhance neural network models. Traditional attention mechanisms are typically static, meaning that the attention weights remain fixed throughout the entire process and do not change with variations in input data [17]. This implies that the model focuses on the same parts regardless of the content of the input sequence. However, in certain cases, this static attention mechanism may not be sufficient to

effectively capture the complex dependencies between different parts of the input sequence, leading to suboptimal performance.

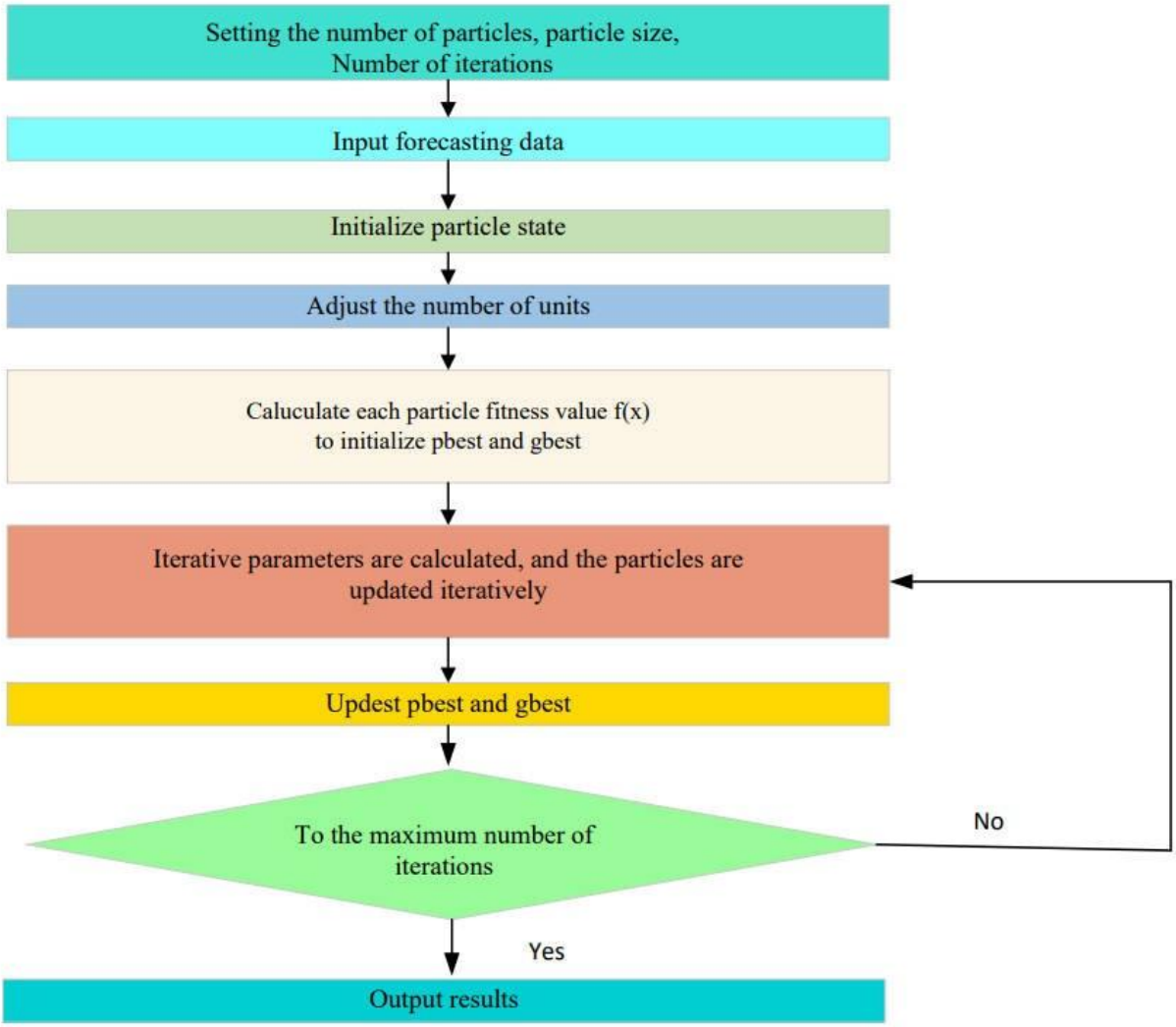

Figure 3. The flow chart of the QPSO

The Dynamic Attention Mechanism addresses this issue by introducing a dynamic way of computing attention weights. These weights are dynamically generated based on the content of the input data and the current state of the model (e.g., hidden states in RNN or Transformer). By dynamically calculating attention weights at different time steps or positions, the model can better adapt to the features and contextual information present in different parts of the input sequence.

In summary, the Dynamic Attention Mechanism is a more flexible and powerful attention mechanism that can better handle the dependencies between different parts of a sequence in sequence modeling tasks, thereby improving model performance. In many sequence generation tasks, models using the dynamic attention mechanism often achieve better results. As shown in Figure \ref{attention}, it is the flow chart of Dynamic attention mechanism.

Suppose we have an input sequence X, where $X = \{x_1,x_2,...,x_n\}$, and each position x_i is a vector. At time step t, the current model state is represented as h_t, and we want to compute dynamic attention

weights and the weighted context vector.

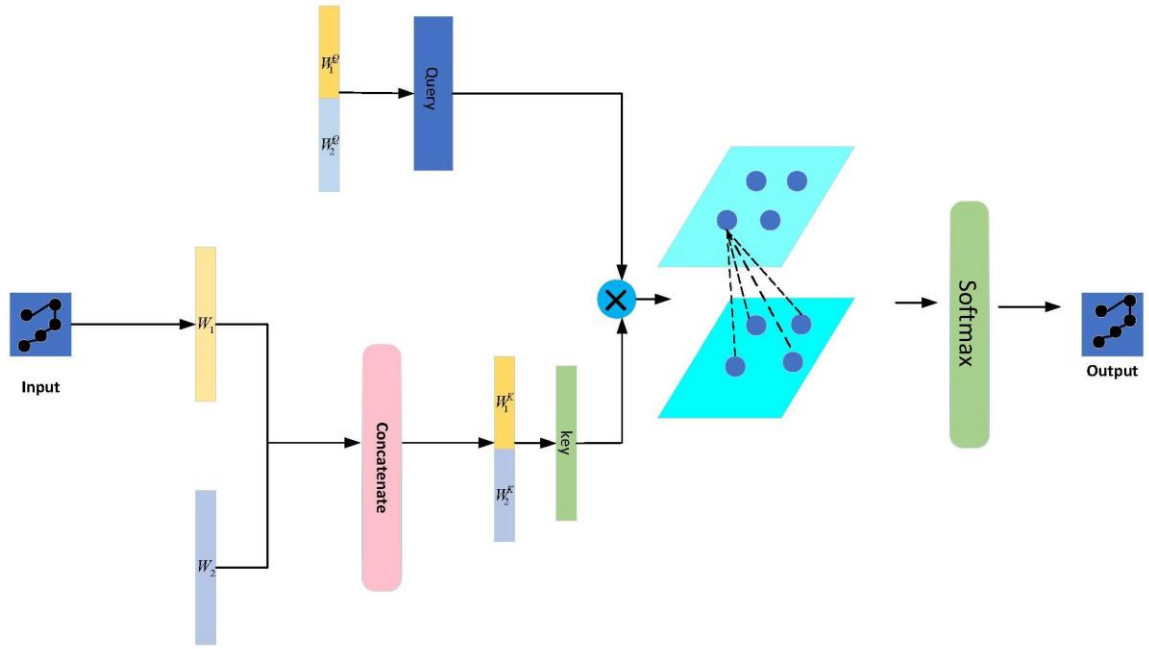

Figure 4. Dynamic attention mechanism

Calculation of Additive Attention Scores: As before, we introduce learnable attention weight vector W_a and activation function f. At time step t, for position i, the additive attention score a_i is defined as:

$$a_i = f(W_a \cdot (h_t + x_i)) \tag{7}$$

Where, h_t represents the current model state at time step t.

Calculation of Dynamic Attention Weights: In the dynamic attention mechanism, the attention weights are dynamic and depend on the input sequence and model state. For this purpose, we introduce learnable attention weight vector W_d and activation function g to map the additive attention scores into the range $(0,1)$. At time step t, for position i, the dynamic attention weight d_i is defined as:

$$d_i = g(W_d \cdot (h_t + x_i)) \tag{8}$$

Where, g is an activation function, such as the sigmoid function, used to map the scores into the range $(0,1)$.

$$D_t = \{d_1, d_2, \ldots, d_n\} \tag{9}$$

$$D_t = \text{softmax}(D_t) \tag{10}$$

Computation of Weighted Context Vector: Finally, we use the normalized dynamic attention weights D_t to compute the weighted sum of the input sequence, resulting in the weighted context

vector c_t:

$$c_l = \sum_{i=1}^{n} (D_l \cdot x_i) \tag{11}$$

The accuracy and efficiency of the QPSO-GRU model in energy consumption forecasting are significantly enhanced through the application of a dynamic attention mechanism. By dynamically focusing on key information at different time steps, this model can better adapt to various energy consumption patterns and adjust weights accordingly based on the data's significance. This enables more accurate predictions of future energy demands and consumption. Furthermore, the dynamic attention mechanism contributes to a deeper understanding of energy consumption patterns and trends in carbon neutrality tasks. By dynamically attending to historical energy consumption data, researchers can identify peak and off-peak periods of energy usage, allowing for the development of tailored energy-saving strategies.

4. Experiment

4.1 Datasets

In this study, a total of four datasets were used as follows:

Building Performance Database (BPD) Dataset: This dataset [18] is provided by the U.S. Department of Energy (DOE) and aggregates large-scale building performance data from thousands of buildings in the United States. It includes energy consumption data, building characteristics, and energy management information. The BPD dataset is widely used for building performance assessments, energy efficiency analyses, and building energy-saving research, providing valuable resources for researchers, policymakers, and engineers. These data allow for in-depth analysis of energy usage in buildings and help drive the development of sustainable building and energy-saving measures.

ASHRAE Great Energy Predictor III (GEP III) Dataset: ASHRAE, the American Society of Heating, Refrigerating and Air-Conditioning Engineers [19], is committed to promoting energy efficiency and indoor environmental improvements. The GEP III dataset is collected through an energy prediction competition organized by ASHRAE. This dataset includes historical energy consumption data, building characteristics, and environmental conditions for real buildings. Its goal is to promote the development and comparison of building energy consumption prediction models, potentially offering more sustainable solutions for the building industry and helping reduce energy waste and environmental impact.

Non-Intrusive Load Monitoring (NILM) Dataset: The NILM dataset is used for studying non-intrusive load monitoring technology [20], which aims to disaggregate total electrical energy consumption into the energy consumption of individual electrical appliances by monitoring the overall electrical load pattern. This dataset includes electricity consumption data for various electrical appliances and can be used for developing appliance recognition and energy monitoring algorithms. These data are crucial for intelligent management and energy conservation in both residential and

industrial electricity usage and have the potential to drive the development of smart grids and energy efficiency.

Hybrid Atmospheric Reconnaissance (HAR) Dataset: The HAR dataset encompasses atmospheric data from multiple observation stations and meteorological sensors, including temperature, humidity, wind speed, precipitation, and various other indicators [21]. These data can be utilized in applications such as weather forecasting, climate research, and environmental monitoring. The completeness and diversity of this dataset make it a powerful tool for conducting atmospheric research and are expected to play a significant role in future carbon mitigation efforts.

4.2 Experimental Details

This paper selected four datasets for training, with the training process as follows:

Step1: Data preprocessing

First, the four datasets undergo data cleaning to remove missing values and outliers. Next, feature selection and extraction are carried out to obtain significant features related to energy consumption. Subsequently, energy consumption data is normalized. Data preprocessing helps improve the model's performance, reduce noise and interference, enable the model to better adapt to real-world scenarios, and enhance the accuracy and efficiency of research on energy optimization and prediction.

Step2: Model training

- Introducing Dynamic Attention Mechanism: During model training, this mechanism adjusts attention weights across distinct time steps using the contextual information of input data. This lets the model better capture temporal dependencies and correlations between sequence time steps, boosting its capability to model temporal data.
- Model Optimization: The training set drives the iterative optimization of the QPSO-GRU model. In the training phase, optimization techniques like gradient descent are adopted to minimize the loss function and determine the model's optimal parameter settings. Initially, the QPSO-GRU's parameters undergo random initialization. Subsequently, training data is input into the model to generate predictive outputs, which are then compared against real-world values to calculate the model's loss. Following this, parameter gradients are derived from the computed loss, and such gradient data is utilized to update the model's parameters. This cycle is repeated across numerous epochs until a pre-set termination criterion is satisfied—for instance, when the loss function converges or the maximum iteration count is reached. Via this iterative optimization workflow, the model progressively acquires patterns and inherent regularities from the training dataset, ultimately enhancing its capacity to fit the training data.
- Model Validation: The optimized QPSO-GRU model is evaluated using a test set. The test set contains data that the model has not seen during training and is used to assess the model's generalization ability on new data.
- Model saving: The fully trained combination model ought to be saved on the hard drive once the training process reaches completion.

4.3 Experimental Results and Analysis

As shown in Table 1, the table compares the evaluation results of different models on different datasets, where the performance metrics include Accuracy, Recall, F1 Score, and Area Under the Curve (AUC). The seven models are Elnour et al.[22], Kim et al.[23], Lei et al.[24], Rajesh et al.[25], Farrokhi et al.[26], Xiao et al. [27] and Ours (our model). From the table, it can be observed that the Ours model achieves the highest accuracy on all datasets, with values of 97.59%, 98.36%, 96.8%, and 98.07%, respectively. However, the Kim et al. model has slightly lower accuracy on the BPD Dataset and GEP III Dataset, with values of 88.57% and 92.06%, respectively. Moreover, the Ours model outperforms other models in most datasets in terms of Recall, particularly achieving a high recall rate of 95.36% on the BPD Dataset. Additionally, the Ours model shows the highest F1 Score, especially reaching 93.8% on the NILM Dataset. Furthermore, the Ours model achieves the highest AUC values, with values of 96.01%, 95.63%, 95.49%, and 95.13% on the respective datasets. Figure5 visualizes the data from Table 1, providing a more intuitive comparison of the performance of different models. It is evident from the visualization that Ours model performs the best on these four datasets, with high accuracy, recall, F1 Score, and the largest AUC values.

Table 1. Comparison of different models on different indicators

Model	Datasets															
	BPD Dataset				GEP III Dataset				NILM Dataset				HAR Dataset			
	Accuracy	Recall	F1 Sorce	AUC	Accuracy	Recall	F1 Sorce	AUC	Accuracy	Recall	F1 Sorce	AUC	Accuracy	Recall	F1 Sorce	AUC
Elnour et al. [22]	96.17	89.83	87.16	91.04	85.98	88.44	88.74	86.68	92.73	91.48	88.11	90.54	87.87	86.48	85.03	87.86
Kim et al. [23]	88.57	86.45	83.98	89.57	92.06	91.85	89.71	86.93	88.6	91.07	87.59	87.15	90.32	88.69	87.68	87.98
Lei et al. [24]	86.24	91.6	83.89	92.4	89.91	86.51	84.32	87.9	87.36	93.36	86.99	84.3	90.66	85.08	83.92	86.2
Rajesh et al. [25]	92.64	85.03	86.34	89.91	93.32	87.64	84.08	93.55	95.95	90.13	85.37	86.06	95.59	85.48	86.49	88.58
Farrokhi et al. [26]	96.41	91.89	90.76	91.25	87.22	90.28	89.42	91.67	85.98	89.7	88.18	87.88	89.37	87.1	90.61	92.03
Xiao et al. [27]	93.11	89.72	86.98	87.47	93.53	90.62	85.92	89.27	87.41	84.12	84.23	85.18	89.09	84.81	88.03	93.26
Ours	97.59	95.36	93.8	96.01	98.36	95.55	94.12	95.63	96.8	94.82	93.07	95.13	98.07	94.04	93.8	95.49

As shown in Table 2, we have compared the performance metrics of different models on four different datasets. The metrics include model parameters (Parameters), floating-point operations (Flops), inference time (Inference Time), and training time (Training Time).

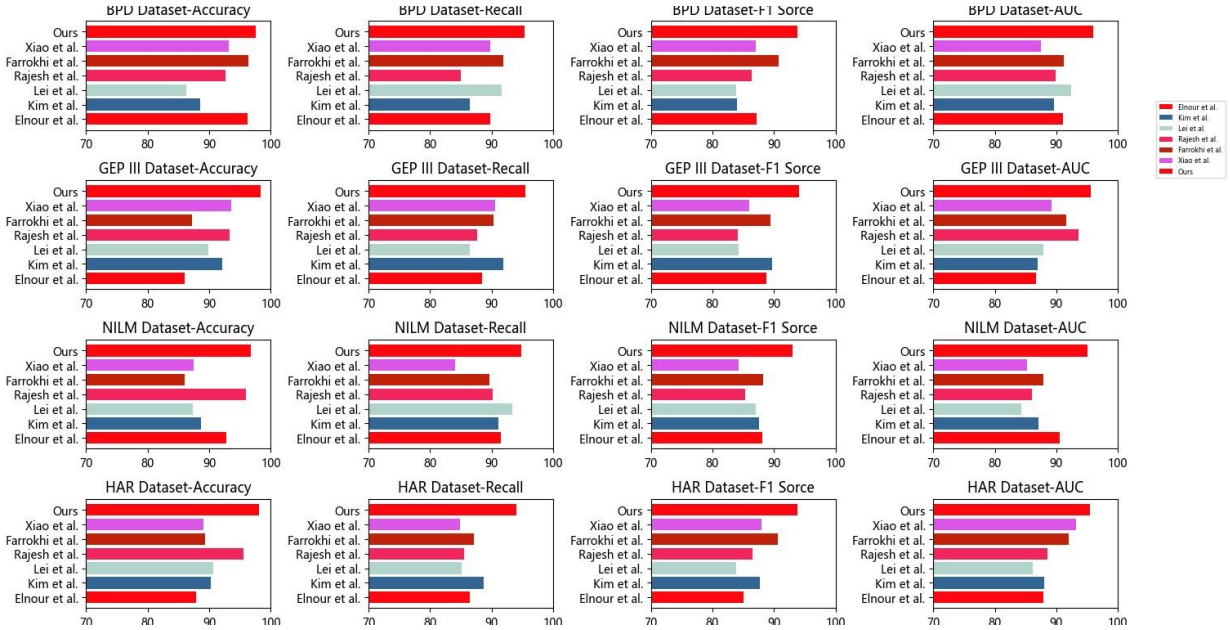

Figure 5.Comparison of different models on different indicators

Table 2. Comparison of different models on different indicators

Model	Datasets															
	BPD Dataset				GEP III Dataset				NILM Dataset				HAR Dataset			
	Parameters (M)	Flops (G)	Inference Time (ms)	Trainning Time (s)	Parameters (M)	Flops (G)	Inference Time (ms)	Trainning Time (s)	Parameters (M)	Flops (G)	Inference Time (ms)	Trainning Time (s)	Parameters (M)	Flops (G)	Inference Time (ms)	Trainning Time (s)
Elnour et al.	311.26	259.21	238.24	339.83	382.21	202.23	219.97	275.91	354.54	247.03	277.97	218.52	380.61	370.77	356.95	463.28
Kim et al.	212.28	219.30	348.63	398.86	228.80	332.33	231.58	216.21	300.35	329.94	310.89	321.51	248.41	359.95	347.41	654.97
Lei et al.	331.26	341.23	246.96	282.56	287.93	339.57	377.91	323.09	376.78	280.17	306.03	230.72	253.24	394.90	207.15	410.28
Rajesh et al.	238.13	263.44	369.54	374.15	311.23	266.68	230.85	263.09	302.96	247.08	231.28	352.28	352.44	302.55	272.15	241.84
Farrokhi et al	223.31	271.29	331.72	276.63	272.43	304.54	272.42	234.69	335.29	217.19	286.97	334.95	281.41	371.41	279.41	217.17
Xiao et al.	301.64	355.33	269.84	285.58	275.12	354.67	363.39	272.04	225.88	379.56	288.02	282.57	208.86	201.28	236.15	274.62
Ours	230.30	182.30	209.82	135.61	139.38	170.66	192.68	112.73	127.44	159.12	142.69	134.06	152.76	222.15	217.68	125.23

In the table, each row represents a different model, while each column represents the corresponding dataset and performance metric. From the table, it can be observed that our model has only 230.30 million parameters, significantly reducing the model's complexity and storage requirements compared to other models, which typically have 300 to 380 million parameters. Additionally, our model only requires 182.30 billion floating-point operations, while other models usually range from 250 to 370 billion operations. This indicates that our model requires fewer computational resources during inference. Furthermore, our model demonstrates fast and efficient inference times. On the BPD Dataset, the inference time is only 209.82 milliseconds, much lower

than the range of 230 to 370 milliseconds observed in other models. Moreover, our model exhibits a clear advantage in training time. On the BPD Dataset, the training time is only 135.61 seconds, much lower than the range seen in other models. Figure 6 visualizes the content of the table using pie charts. From the figure, we can visually observe that our model performs excellently on multiple performance metrics, with fewer parameters, lower floating-point operations, and faster inference and training times. This implies that our model can work more efficiently in practical applications, saving computational resources and time, while maintaining a good level of performance.

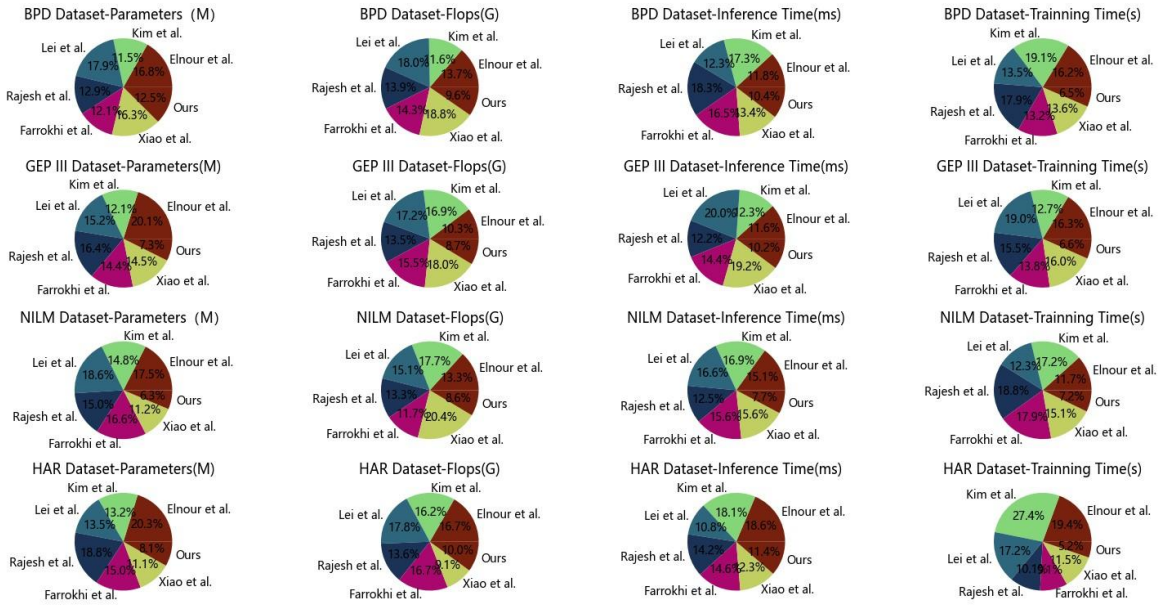

Figure 6. Comparison of different models on different indicators

As shown in Table 3, the table displays the results of ablation experiments on the GRU module, where different models are evaluated on four different datasets. The performance metrics used for evaluation include accuracy, recall, F1 score, and AUC. The *Model* column lists the different types of models evaluated, including RNN [28], CNN [29], LSTM [7], and GRU [30]. We found that the GRU model generally performs the best across all datasets, achieving the highest accuracy. Particularly on the NILM dataset, the GRU model achieved an accuracy of 98.43%.

Table 3. Ablation experiments on the GRU module

Model	Datasets															
	BPD Dataset				GEP III Dataset				NILM Dataset				HAR Dataset			
	Accuracy	Recall	F1 Sorce	AUC	Accuracy	Recall	F1 Sorce	AUC	Accuracy	Recall	F1 Sorce	AUC	Accuracy	Recall	F1 Sorce	AUC
RNN [28]	91.37	86.96	89.44	88.15	93.03	84.01	90.96	88.27	94.69	84.41	86.29	84.65	90.19	93.53	89.4	89.19
RNN [29]	89.57	86.27	88.34	91.93	90.62	88.23	87.67	89.74	85.93	92.15	84.93	89.61	88.98	85.27	88.09	92.41
LSTM RNN	91.4	84.02	85.29	90.77	86.52	86.96	85.05	90.72	93.53	84.76	89.05	92.78	87.79	88.93	88.08	92.64

[17]																
GRU RNN [30]	97.46	94.24	92.03	93.64	97.25	94.91	93	91.81	98.43	94.99	92.5	91.4	97.92	94.23	93.34	91.71

At the same time, the GRU model showed the best recall on the GEP III and HAR datasets, achieving 94.91% and 94.23%, respectively. Although the recall of GRU is slightly lower than other models on the BPD and NILM datasets, it remains relatively high. Additionally, the GRU model achieved the highest F1 score on the NILM dataset, reaching 92.5%, which is an outstanding performance. On the BPD and GEP III datasets, the GRU's F1 scores are also relatively good, at 92.03% and 93%, respectively. Figure 7 visualizes the table content using bar charts, where it can be observed that the GRU model performs remarkably well on these four datasets, with its accuracy, recall, F1 score, and AUC values generally outperforming other models.

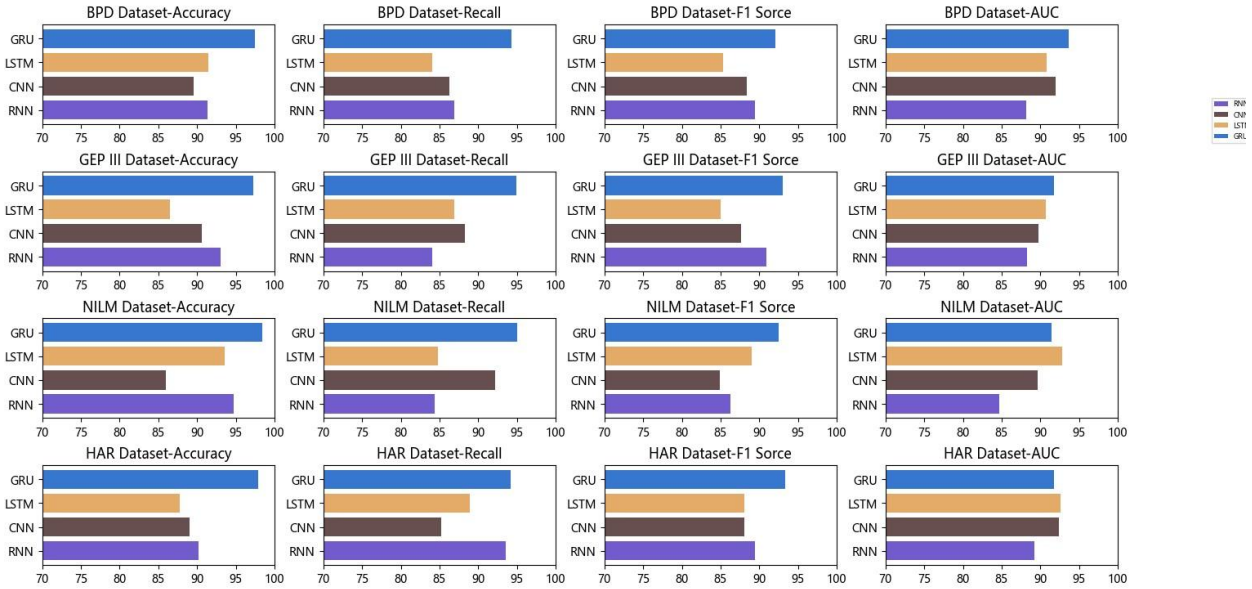

Figure 7. Ablation experiments on the GRU module

As shown in Table 4, the table displays the results of ablation experiments on the *Dynamic AttentionMechanism* module. The compared methods include *Self-Attention Mechanism* [31], *Multi-Head Attention* [32], *Local Attention*[33], and *Dynamic Attention Mechanism* [34]. From the table, it can be concluded that on the *BPD Dataset*, the *Dynamic Attention Mechanism* has the lowest number of parameters, only 231.06 million, compared to other methods such as *Self-Attention Mechanism* with 347.12 million and *Multi-Head Attention* with 263.92 million, indicating fewer parameters.

Table 4. Ablation experiments on the Dynamic Attention Mechanism module.

Model	**Datasets**			
	BPD Dataset	**GEP III Dataset**	**NILM Dataset**	**HAR Dataset**

	Parameters (M)	Flops (G)	Inference Time (ms)	Trainning Time (s)	Parameters (M)	Flops (G)	Inference Time (ms)	Trainning Time (s)	Parameters (M)	Flops (G)	Inference Time (ms)	Trainning Time (s)	Parameters (M)	Flops (G)	Inference Time (ms)	Trainning Time (s)
Self-Attention Mechanism[31]	347.12	275.44	329.99	286.19	346.26	392.19	305.80	381.39	310.40	203.62	356.43	338.72	289.80	364.22	228.67	269.54
Multi-Head Attention[32]	263.92	246.75	274.46	374.98	323.96	273.74	270.66	285.35	319.47	267.29	286.61	206.20	303.04	349.51	261.50	287.26
Local Attention [33]	377.64	276.88	323.52	249.21	265.39	242.01	347.51	269.79	243.99	351.75	224.67	273.94	373.82	322.57	237.17	321.16
Dynamic Attention Mechanism [34]	231.06	214.09	103.49	121.35	111.43	168.81	181.78	111.41	229.14	123.20	136.22	216.35	188.53	142.88	172.35	217.79

Similarly, on the *BPD Dataset*, the *Dynamic Attention Mechanism* shows the lowest FLOPs value, 214.09 billion, while *Self-Attention Mechanism* and *Multi-Head Attention* have 275.44 billion and 246.75 billion, respectively, indicating lower computational cost. Moreover, on the *GEP III Dataset*, the *Dynamic Attention Mechanism* has an inference time of 181.78 milliseconds, while *Self-Attention Mechanism* and *Multi-Head Attention* have 305.80 milliseconds and 270.66 milliseconds, respectively, indicating faster predictions. Additionally, on the *GEP III Dataset*, the *Dynamic Attention Mechanism* has a training time of 111.41 seconds, while *Self-Attention Mechanism* and *Multi-Head Attention* have 381.39 seconds and 285.35 seconds, respectively, meaning it is more efficient in training the model. Figure 8 visualizes the table content through a line chart, which demonstrates that the *Dynamic Attention Mechanism* method performs well in terms of model size, computational cost, inference time, and training time on the *BPD Dataset* and *GEP III Dataset*, fully showcasing the superiority of our model.

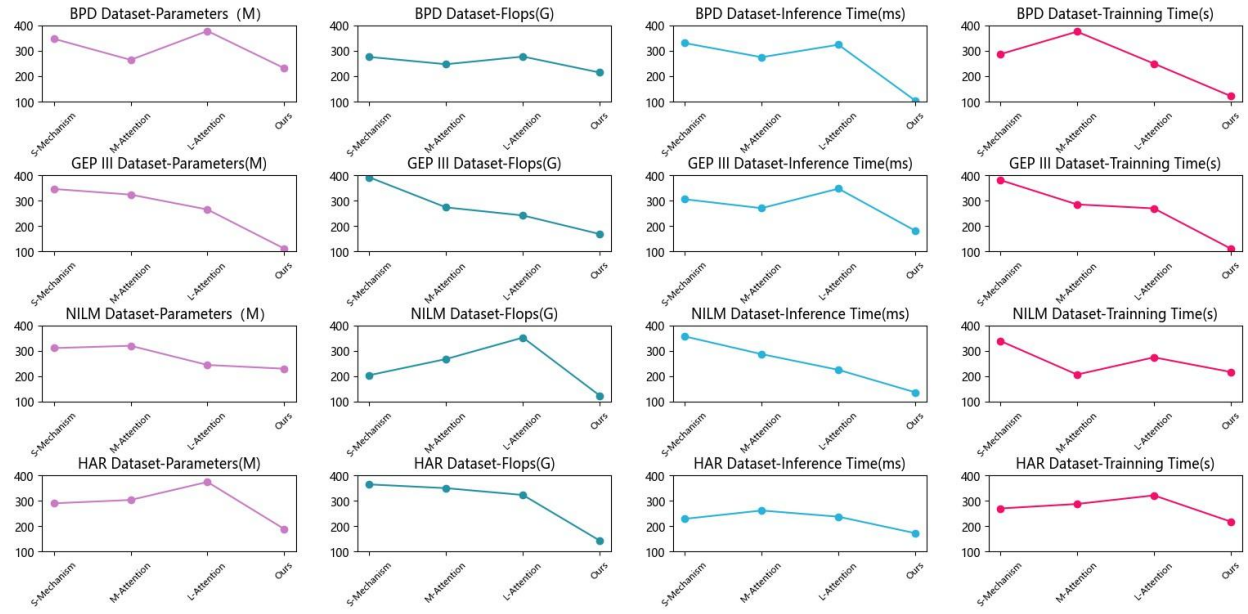

Figure 8. Ablation experiments on the GRU module

5. Conclusion And Discussion

This study addresses the challenges in carbon emission and energy consumption forecasting by proposing an integrated approach of the QPSO-GRU model and a dynamic attention mechanism. This mechanism equips the model with the capability to adaptively modify attention weights for each time step, drawing on contextual information within the input sequence, and thereby precisely capturing temporal correlations and inter-temporal dependencies.The QPSO-GRU model combined with the dynamic attention mechanism demonstrates outstanding performance in energy consumption prediction. This approach helps better capture spatiotemporal features in the data, leading to improved accuracy and stability in energy consumption prediction. Additionally, the method provides a sustainable and environmentally friendly solution for energy management and optimization.

However, our model also has some limitations. While the proposed QPSO-GRU model and dynamic attention mechanism excel in predicting energy consumption, their generalization to other domains or different types of facilities remains unexplored. Furthermore, due to data limitations, we lack more features for comprehensive analysis.

In future research, we can also explore how to further optimize the model's structure and parameters to enhance predictive performance. For instance, we can consider introducing more complex neural network structures or other optimization algorithms to further enhance the model's capabilities. Additionally, we can consider incorporating the expertise of domain specialists to further improve the model's performance. Collaborations with energy experts can help us better understand the impact of different factors on energy consumption, allowing for precise adjustments to the model's parameters and structure. This interdisciplinary collaboration contributes to improving the model's accuracy and practicality. Another direction worth exploring is how to better explain and visualize the model's predictive results in practical applications. Interpretability and explainability have become increasingly important in real-world applications, especially in decision-making contexts.

Therefore, we can explore how to design interpretable methods to help decision-makers understand the model's predictive results and make informed decisions.

In conclusion, our work effectively addresses the challenge of energy consumption forecasting and carbon emissions prediction by combining the QPSO-GRU model with the dynamic attention mechanism. Furthermore, the application of our model offers sustainable and eco-friendly solutions for energy management, contributing to carbon neutrality efforts. Lastly, the introduction of this model opens up new possibilities for related research, exploring its potential transferability and scalability beyond its initial domain. We envision that the impact of this research can significantly contribute to sustainable energy practices and bring about positive transformations and advancements in society.

Acknowledgements

This article received no financial or funding support.

Conflicts of Interest

The author confirms that there are no conflicts of interest.

References

[1] Zhao, X., Ma, X., Chen, B., Shang, Y. and Song, M. Challenges toward carbon neutrality in China: strategies and countermeasures. Resources, Conservation and Recycling, 2022, 176, 105959.

[2] Zhao, X., Ma, X., Chen, B., Shang, Y. and Song, M. Challenges toward carbon neutrality in China: strategies and countermeasures. Resources, Conservation and Recycling, 2022, 176, 105959.

[3] Wu, X., Tian, Z. and Guo, J. A review of the theoretical research and practical progress of carbon neutrality. Sustainable Operations and Computers, 2022, 3, 54-66.

[4] Li, R., Wang, Q., Liu, Y. and Jiang, R. Per-capita carbon emissions in 147 countries: the effect of economic, energy, social, and trade structural changes. Sustainable Production and Consumption, 2021, 27, 1149-1164.

[5] Chen, Z., Yang, Y., Ziya, S., Yang, S., Abulimiti, M. and Wang, Y. Carbon emissions index decomposition and carbon emissions prediction in Xinjiang from the perspective of population-related factors, based on the combination of STIRPAT model and neural network. Environmental Science and Pollution Research, 2022, 1-16.

[6] Shi, C. and Feng, X. Carbon emission factor decomposition and carbon peak prediction based on multi-objective decision and information fusion processing. EURASIP Journal on Advances in Signal Processing, 2021, 2021, 1-18.

[7] Shen, Z., Wu, Q., Qian, J., Gu, C., Sun, F. and Tan, J. Federated learning for long-term forecasting of electricity consumption towards a carbon-neutral future. In: 2022 7th International Conference on Intelligent Computing and Signal Processing (ICSP), 2022, 789-793.

[8] Sheng, Y., Wang, H., Yan, J., Liu, Y. and Han, S. Short-term wind power prediction method based on deep clustering-improved temporal convolutional network. Energy Reports, 2023, 9, 2118-2129.

[9] Chen, Y., Chen, X., Xu, A., Sun, Q. and Peng, X. A hybrid CNN-transformer model for ozone concentration prediction. Air Quality, Atmosphere & Health, 2022, 15, 1533-1546.

[10] Li, W. Cooling, heating and electric load forecasting for integrated energy systems based on CNN-LSTM. In: 2021 6th International Conference on Power and Renewable Energy (ICPRE), 2021, 808-812.

[11] Oyando, H.C., Kanyolo, T.N. and Chang, C.K. RNN-based main transformer OLTC control for SMR integration into a high renewable energy penetrated grid. Journal of Electrical Engineering & Technology, 2023, 1-13.

[12] Tang, J. and Li, J. Carbon risk and return prediction: evidence from the multi-CNN method. Frontiers in Environmental Science, 2022, 10, 2160.

[13] Chen, Y., Chen, X., Xu, A., Sun, Q. and Peng, X. A hybrid CNN-transformer model for ozone concentration prediction. Air Quality, Atmosphere & Health, 2022, 15, 1533-1546.

[14] Liu, T., Xu, C., Guo, Y. and Chen, H. A novel deep reinforcement learning based methodology for short-term HVAC system energy consumption prediction. International Journal of Refrigeration, 2019, 107, 39-51.

[15] Yang, W., Huang, B., Zhang, A., Li, Q., Li, J. and Xue, X. Condition prediction of submarine cable based on CNN-BiGRU integrating attention mechanism. Frontiers in Energy Research, 2022, 10, 1023822.

[16] Feng, H., Hu, Y.J., Li, C. and Wang, H. Rolling horizon optimisation strategy and initial carbon allowance allocation model to reduce carbon emissions in the power industry: case of China. Energy, 2023, 277, 127659.

[17] Yang, W., Huang, B., Zhang, A., Li, Q., Li, J. and Xue, X. Condition prediction of submarine cable based on CNN-BiGRU integrating attention mechanism. Frontiers in Energy Research, 2022, 10, 1023822.

[18] Tian, Z., Zhang, X., Wei, S., Du, S. and Shi, X. A review of data-driven building performance analysis and design on big on-site building performance data. Journal of Building Engineering, 2021, 41, 102706.

[19] Rysanek, A., Nuttall, R. and McCarty, J. Forecasting the impact of climate change on thermal comfort using a weighted ensemble of supervised learning models. Building and Environment, 2021, 190, 107522.

[20] Iqbal, H.K., Malik, F.H., Muhammad, A., Qureshi, M.A., Abbasi, M.N. and Chishti, A.R. A critical review of state-of-the-art non-intrusive load monitoring datasets. Electric Power Systems Research, 2021, 192, 106921.

[21] Islam, M.M., Nooruddin, S., Karray, F. and Muhammad, G. Human activity recognition using tools of convolutional neural networks: a state of the art review, data sets, challenges, and future prospects. Computers in Biology and Medicine, 2022, 106060.

[22] Elnour, M., Himeur, Y., Fadli, F., Mohammedsherif, H., Meskin, N., Ahmad, A.M. and others. Neural network-based model predictive control system for optimizing building automation and management systems of sports facilities. Applied Energy, 2022, 318, 119153.

[23] Kim, A.A., Sunitiyoso, Y. and Medal, L.A. Understanding facility management decision making for energy efficiency efforts for buildings at a higher education institution. Energy and Buildings, 2019, 199, 197-215.

[24] Lei, T., Cai, Z. and Hua, L. 5G-oriented IoT coverage enhancement and physical education resource management. Microprocessors and Microsystems, 2021, 80, 103346.

[25] Rajesh, P., Shajin, F.H., Mouli Chandra, B. and Kommula, B.N. Diminishing energy consumption cost and optimal energy management of photovoltaic aided electric vehicle (PV-EV) by GFO-VITG approach. Energy Sources, Part A: Recovery, Utilization, and Environmental Effects, 2021, 1-19.

[26] Farrokhi, A., Farahbakhsh, R., Rezazadeh, J. and Minerva, R. Application of internet of things and artificial intelligence for smart fitness: a survey. Computer Networks, 2021, 189, 107859.

[27] Xiao-wei, X. Study on the intelligent system of sports culture centers by combining machine learning with big data.

Personal and Ubiquitous Computing, 2020, 24, 151-163.

[28] Oyando, H.C., Kanyolo, T.N. and Chang, C.K. RNN-based main transformer OLTC control for SMR integration into a high renewable energy penetrated grid. Journal of Electrical Engineering & Technology, 2023, 1-13.

[29] Tang, J. and Li, J. Carbon risk and return prediction: evidence from the multi-CNN method. Frontiers in Environmental Science, 2022, 10, 2160.

[30] Lv, Z., Wang, N., Lou, R., Tian, Y. and Guizani, M. Towards carbon neutrality: prediction of wave energy based on improved GRU in maritime transportation. Applied Energy, 2023, 331, 120394.

[31] Liu, Y., Zhang, F., Yang, S. and Cao, J. Self-attention mechanism for dynamic multi-step ROP prediction under continuous learning structure. Geoenergy Science and Engineering, 2023, 212083.

[32] Zou, Y., Wu, H., Yin, Y., Dhamotharan, L., Chen, D. and Tiwari, A.K. An improved transformer model with multi-head attention and attention to attention for low-carbon multi-depot vehicle routing problem. Annals of Operations Research, 2022, 1-20.

[33] Ding, L., Tang, H. and Bruzzone, L. LANet: local attention embedding to improve the semantic segmentation of remote sensing images. IEEE Transactions on Geoscience and Remote Sensing, 2020, 59, 426-435.

[34] Liu, P., Zhang, L. and Gulla, J.A. Dynamic attention-based explainable recommendation with textual and visual fusion. Information Processing & Management, 2020, 57, 102099.

Copyright© by the authors, Licensee Intelligence Technology International Press. The article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution-ShareAlike 4.0 (CC BY-SA).