Multi-Time-Scale Heating Load Prediction Using Attention-Enhanced LSTM with Improved Adam Optimization

TianTing Ma[1,2]
[1]Southeast University School of Energy and Environment, Nanjing Jiangsu, 211189, China
[2]CHN Energy Suqian Power Generation Co., LTD, CHN Energy, Suqian 223800, Jiangsu, China
E-mail: SkyMa0120@163.com

Keywords: long short-term memory network, heating load prediction, attention mechanism, improved Adam algorithm, time series analysis

Received: April 29, 2025

This paper proposes an extended short-term memory network (LSTM) heating load prediction model that integrates an attention mechanism and an improved adaptive moment estimation (Adam) algorithm. The model dynamically focuses on key influencing factors such as outdoor temperature and user behavior through the attention mechanism, and combines the improved Adam algorithm to optimize the parameter update process. The experiment uses the heating data of a city for 12 consecutive months, divides the training set and the test set into 7:3, and compares them with traditional LSTM, ordinary Adam-optimized LSTM, SVM, Transformer, TCN, and CNN-LSTM hybrid models. The results show that the root mean square error (RMSE) of the improved algorithm on the test set is 10.23, which is 31.6% lower than that of the traditional LSTM; the mean absolute error (MAE) is 8.12, which is 29.4% lower; the mean absolute percentage error (MAPE) is 7.2%, which is 25.8% lower. At the same time, in short-term (1–24 hours), medium-term (1–7 days), and long-term (1–30 days) prediction tasks, the predicted values closely follow the observed load curve, and the generalization ability is significantly enhanced.

Povzetek: Napovedni model LSTM z integrirano pozornostjo in izboljšanim Adam algoritmom omogoča bolj kvalitetno veččasovno napoved ogrevalne obremenitve, presega tradicionalni LSTM, SVM, Transformer, TCN in CNN-LSTM po točnosti in generalizaciji.

1 Introduction

The contradiction between global energy supply and demand is becoming increasingly severe. According to the data of the International Energy Agency, global building energy consumption accounts for about 40% of the global total energy consumption, of which heating system energy consumption accounts for a large proportion. Under the guidance of the "dual carbon" strategy, achieving energy conservation and emission reduction in the heating system is a critical way to achieve sustainable development. Heating load prediction is the core link in energy optimization scheduling, and its prediction accuracy is directly related to the energy efficiency and operating costs of the entire system. According to statistics, accurate prediction of heating load can increase energy utilization by 15-20%, significantly reducing the operating costs of enterprises. By accurately predicting the load changes, heating companies can plan heat source supply, optimize pipe network scheduling, and achieve a win-win situation for the economy and the environment while ensuring users' thermal comfort.

At present, the heating system's operating environment is becoming increasingly complex, and the requirements for load forecasting technology are becoming higher and higher. A strong nonlinear relationship exists between the dynamic changes of temperature, humidity, wind speed, sunshine time, etc., and the heating load. Taking a northern city as an example, under extreme low-temperature conditions in winter, the heating load increases by 30%-40% compared with usual [1]. At the same time, due to the insulation performance of different buildings, the efficiency of heating equipment, and the personalized heat settings and work and rest patterns of users, the heating load presents dynamic time-varying properties. Traditional heat load prediction methods, such as regression analysis and gray prediction, are based on the assumption of data stability and rely on artificial experience for feature extraction [2]. It isn't easy to effectively characterize the complex nonlinear relationship. In practical applications, the prediction errors of these three methods are as high as 15%-25%, which makes it challenging to meet the requirements of real-time regulation and refined management of heating systems.

With the rapid development of deep learning technology, research on heating load prediction based on neural networks has made new breakthroughs in recent years. An extended short-term memory network (LSTM) is an improved recurrent neural network (RNN) version. It effectively solves the problems of gradient vanishing and gradient explosion in recurrent neural networks through the special design of the input and output gates. It shows substantial advantages in time series prediction [3]. In recent years, LSTM has been widely used in heating load prediction. However, the existing LSTM method still has many shortcomings. In processing multi-source influencing factors, it is difficult for the model to identify and focus on key information automatically. For example, when multi-source data such as meteorology, buildings, and behavior are input simultaneously, it is difficult to effectively distinguish the contribution rate of each factor to load changes, resulting in low feature extraction efficiency [4]. At the model optimization level, the traditional adaptive moment estimation algorithm has problems such as inflexible learning rate adjustment. It is prone to falling into local extreme values when training an LSTM model. Existing studies have shown that the prediction error fluctuation range of the LSTM model trained by the traditional Adam algorithm is between 8% and 12%, which seriously affects the model's promotion ability and prediction stability [5].

In response to the above problems, this project proposes to combine the attention mechanism with the Adam algorithm to construct an LSTM heating load prediction model [6]. This mechanism dynamically adjusts the model's attention to key information, such as weather changes and user behavior, by calculating input features' weights, thereby improving feature extraction's pertinence and effectiveness. Then this project proposes an improved Adam algorithm, which uses a dynamic learning rate adjustment strategy and regularization method to optimize the model parameters to improve the convergence speed of the algorithm and avoid falling into local optimality [7]. Experimental results show that the enhanced Adam algorithm can reduce the number of model iterations by 30% and increase the convergence speed by 25%. The specific research objectives are as follows: 1) To develop an attention-enhanced LSTM model for multi-time-scale heating load prediction; 2) To optimize the Adam algorithm with dynamic learning rate adjustment and L2 regularization; 3) To validate the model’s performance against state-of-the-art baselines including Transformer, TCN, and CNN-LSTM hybrids; 4) To analyze feature importance using SHAP values and evaluate model robustness under extreme conditions.

2 Related theoretical basis

2.1 Analysis of factors affecting heating load

2.1.1 Meteorological factors

Meteorological conditions are important external driving factors that affect heating load fluctuations, and their influence is complex and diverse [8]. There is an apparent negative correlation between outdoor temperature and heating load, and the heating data of many cities have confirmed this law. Taking Harbin as an example, during the five-month heating period, when the outdoor temperature dropped sharply from −5℃ to −25℃, the heating load increased by more than 50%. This change is due to the heat exchange mechanism between the building and the outdoor environment. The low temperature environment accelerates the indoor heat loss, forcing the heating system to increase output.

The effect of humidity on the heating load is relatively indirect but cannot be ignored. In a high-humidity environment, the thermal conductivity of the air increases, and the heat transfer efficiency increases. Related studies have shown that for every 10% increase in air humidity in the middle and lower reaches of the Yangtze River in winter, the heating load will increase by 1.5%–2%. For example, in rainy and humid weather, under the same outdoor temperature conditions, the heating heat used for building heating is significantly higher than that in rainless weather.

The effect of wind speed on the heating load is mainly achieved through air convection. The greater the wind speed, the faster the airflow on the surface of the building and the quicker the heat dissipation. The test results show that for every 1 m/s increase in wind speed, the heat loss of the building will increase by 8%–12%, thereby increasing the heating load by about 2%. In addition, sunshine time is an essential indicator for measuring the intensity of solar radiation, and there is a significant negative correlation between it and the heating load. Adequate sunshine can provide natural heat energy for buildings and reduce the operating pressure of heating systems. There is a significant difference in the length of sunlight between sunny and cloudy days in winter in North China, and the heating load can reach 10%–15%.

2.1.2 Building and equipment factors

Building characteristics and heating equipment performance are the basic factors determining the basic heating load level.

Building insulation performance is an essential factor affecting building heating load, which is mainly determined by the thermal resistance of the building envelope. Buildings with double-layer low-E glass, high-efficiency insulation walls, and dense frame window frames have 40%–50% lower heat loss than ordinary buildings [9]. Taking a passive ultra-low energy building as an example, the unique insulation design and airtight structure can reduce the building's heating load by 30%–40%.

There is a significant linear relationship between building area and heating load. Generally speaking, for every 1,000 square meters of building area, the heating load increases by 8%–10%. However, this relationship is not absolute. If a centralized heating system and a reasonable heat recovery design are used, the heating load per unit area can be effectively reduced. In addition, the type and efficiency of the heating device also directly affect the heating load [10]. The efficiency of traditional coal-fired boilers is generally only 60%–70%. In contrast, the energy utilization rate of new and efficient gas-fired wall-mounted boilers and ground source heat pump systems can reach more than 90%. Taking a specific community as an example, after the ground source heat pump technology transformation, its heating load can be reduced by 35%, achieving the same heating effect.

2.1.3 User behavior factors

Electricity behavior is an essential internal factor causing irregular fluctuations in heating load. Indoor temperature setting preferences directly affect heating demand. Studies have shown that for every 1°C increase, the heating load increases by 5–7%. In some places with high requirements for comfort, such as hotels and office buildings, the load fluctuations caused by temperature settings are more obvious.

The user's sleep time also has a significant impact on the time distribution of the heating load. The peak hours for heat use on weekdays are from 7:00 to 9:00 in the morning and 6:00 to 10:00 p.m. Currently, users are more active and have higher requirements for indoor temperature. Still, on weekends, the heating load shows different distribution patterns due to changes in work arrangements, such as the peak delay in the morning, and the overall load level is low.

Window opening behavior is also one of the critical factors affecting the heating load. Frequent window opening will cause rapid loss of indoor heat [11]. According to statistics, each window opening for more than 30 minutes will increase the heating load by 10%–15%. Some residents in old residential areas are accustomed to opening windows at will, resulting in heating load fluctuations of up to 20%–30%, which seriously affects the heating system's stable operation and energy efficiency.

2.2 Principle of long short-term memory network (LSTM)

2.2.1 LSTM Network Structure

LSTM network can effectively overcome the shortcomings of traditional recurrent neural network (RNN), such as gradient disappearance and gradient explosion. LSTM neurons are mainly composed of four parts: the unit, forget gate, output gate, and input gate. The cell state is like a "highway", which stably transmits information over time, avoiding excessive information loss or interference.

The "forget gate" determines which information in the cell state needs to be saved or forgotten. In practical applications, when processing heating load data, the forget gate can selectively ignore historical load data unrelated to the current prediction based on the current input information, so the model only focuses on valuable information. The input gate is responsible for screening and updating the cell state and determining which new information to add to the cell state based on the current input and the implicit state of the previous moment. The output end controls the output of cell information, processes and transforms the cell state, and generates prediction results [12]. Through the synergy of the three gates, LSTM can accurately memorize and update information and efficiently process complex time series data.

2.2.2 Advantages of LSTM in processing time series

Compared with traditional recurrent neural networks, the short-term memory neural networks (LSTM) gating mechanism shows significant advantages in processing time series data. In recurrent neural networks, since the recurrent neural network decreases or increases exponentially with the time step during the backward transmission process, it is difficult for the network to learn dependencies from long time series. By cleverly adjusting the forget gate and the input gate, the extended short-term memory network allows the cell state to selectively maintain its historical information during the update process, effectively avoiding gradient variation.

The heating load time series has obvious seasonal, periodic, and nonlinear characteristics, mainly manifested in daily load volatility, weekly load volatility, seasonal load differences, etc. Due to the strong ability of long short-term memory and the ability to capture complex patterns, it can accurately identify the laws and trends in the time series. For example, LSTM can not only learn the changing trend of heating load every winter, but also capture the details of load fluctuations caused by changes in user schedules, and achieve accurate prediction of heating load.

2.3 Principle of the attention mechanism

2.3.1 Basic concepts of attention mechanism

The attention mechanism simulates the process of attention allocation in human vision and cognitive systems, allowing the model to focus on a large amount of information on key parts. In natural language processing, the attention mechanism has been widely used in machine translation. By dynamically adjusting the degree of attention to different parts of the source language, the accuracy and fluency of the translation can be significantly improved. In computer vision, it can assist the model in identifying critical areas, thereby improving object detection and classification performance.

In the heating load forecast, many factors affect the heating load, and the importance of each factor changes over time. For example, in freezing weather, the weight of the outdoor temperature on the heating load will increase significantly; under mild climate conditions, user behavior will become more important. This project introduces the attention mechanism into the model, so that the model can adaptively adjust the weights of each influencing factor according to the input data at different time points to more accurately extract the key features that affect the load change and improve the accuracy of the prediction.

This project introduces the attention mechanism into the model, so that the model can adaptively adjust the weights of each influencing factor according to the input data at different time points to more accurately extract the key features that affect the load change and improve the accuracy of the prediction. An ablation study was conducted, showing that the model with attention achieved 12.3% lower RMSE than the model without attention, verifying the necessity of the attention mechanism. The model uses 8-head attention to capture multi-dimensional feature dependencies.

2.3.2 Attention mechanism workflow

The attention mechanism is centered on the interactive operation of query, key, and value vectors. First, a series of linear transformations is performed on the input data to generate queries, keys, and values. These vectors contain different features to represent the input data. Then, the similarity between the query and the keyword is calculated to evaluate the importance of each part of the input data. The higher the similarity, the more critical the information contained in the corresponding part [13]. When calculating the score, operations such as dot product and scaled dot product are generally used, and then the score is converted into an attention weight using soft functions. Finally, the attention weight is used to perform weighted summation on the Value to obtain the final output result. This method enables the model to dynamically adjust the degree of attention paid to different information according to the characteristics of the input data, thereby effectively extracting and utilizing key information.

2.3.3 Workflow of the attention mechanism

The attention mechanism is centered on the interactive operation of query, key, and value vectors. First, a series of linear transformations is performed on the input data to generate queries, keys, and values. These vectors contain different features to represent the input data. Then, the similarity between the query and the keyword is calculated to evaluate the importance of each part of the input data. The higher the similarity, the more critical the information contained in the corresponding part. When calculating the score, operations such as dot product and scaled dot product are generally used, and then the score is converted into an attention weight using soft functions. Finally, the attention weight is used to perform weighted summation on the Value to obtain the final output result. This method enables the model to dynamically adjust the degree of attention paid to different information according to the characteristics of the input data, thereby effectively extracting and utilizing key information.

2.4 Adaptive moment estimation (Adam) optimization algorithm

2.4.1 Principle of the Adam Algorithm

Effective parameter updates are achieved by combining the advantages of momentum and adaptive learning rate optimization strategies. The momentum optimization strategy introduces a "momentum term" that makes the parameters have inertia during the correction process, speeds up convergence, and reduces oscillation; the adaptive learning rate strategy dynamically adjusts the learning rate of each parameter according to the parameter update history, making the model have strong adaptive capabilities. The Adam algorithm dynamically adjusts the learning rate by calculating the first-order moment (mean) and the second-order moment (non-central variance). In the initial stage, the Adam algorithm can quickly adjust the parameters and speed up the convergence speed; during the training process, the algorithm can automatically adjust the learning rate according to the change of the gradient value, avoid the parameter update too fast or too slow, and improve the stability of the model.

2.4.2 Improvement ideas

Although the Adam algorithm has achieved good results in the initial stage, it has problems such as slow convergence and is prone to falling into local extreme values later. To solve this problem, researchers have proposed various improvement methods. One commonly used method is to dynamically adjust the learning rate so that the learning rate is automatically adjusted with the progress of training and the change of the loss function [14]. For example, a larger learning rate is used to accelerate convergence in the initial stage of learning. As the loss function gradually becomes flat, the learning rate is slowly reduced so that the model can find the optimal solution more accurately.

Introducing regularization methods such as L2 to constrain model parameters can effectively avoid model overfitting problems and improve the model's generalization ability. Combining the advantages of other optimization algorithms, improving the gradient update direction is a critical way to improve the performance of the Adam algorithm. For example, combining the Adam algorithm and the stochastic gradient descent (SGD) algorithm, using the global search ability of SGD, guides the Adam algorithm to jump out of the local extreme value and improves its optimization ability.

3 Design of heating load prediction algorithm based on improved LSTM

3.1 Overall framework of the algorithm

3.1.1 Framework structure

This project will deeply integrate four key steps: data processing, feature extraction, model optimization, and prediction output. The data preprocessing link is the "front-end guard" for purifying and standardizing the original data, and its processing effect directly affects the training effect of the subsequent model. The LSTM network module plays the "intelligent brain" role based on the attention mechanism. With its unique gating mechanism and attention allocation strategy, it deeply mines and models the complex time series characteristics of heating load. As a "tuning engine", the improved Adam optimization module dynamically adjusts the parameters in real time during the model training process to ensure that the model converges to the optimal value quickly and stably [15]. As the "output end of the results", the prediction output module converts the feature information after multi-layer processing into an accurate prediction value of the heating load.

The data transmission between the modules constitutes a tightly coordinated closed-loop system. The standardized data output by the data preprocessing module is sequentially input into the LSTM network module to realize feature extraction and time series modeling based on the attention mechanism; the LSTM network is used to extract and predict the model, and the model is fed back to the improved Adam optimization module; the improved Adam optimization module adjusts the parameters according to the feedback information, and feeds it back to the LSTM network module for training; then the prediction output module is used to analyze the hidden state of the LSTM network module to realize the mapping transformation between the fully connected layers, realize the accurate prediction of the heating load, and provide a reliable basis for the optimal scheduling of the heating system.

3.1.2 Module Function

The data preprocessing module uses multi-step refined operations to improve data quality significantly. In the cleaning process, hash value alignment technology is used to design an efficient duplicate value detection algorithm to quickly and accurately identify and eliminate redundant data, avoiding the impact of duplicate information on model training. Regarding outlier data processing, the 3σ criterion is strictly followed to establish a local weighted regression model, accurately correct outliers, and ensure the data is authentic and reliable. The normalization formula is used in the standardization stage:

$$x_{norm} = \frac{x-\mu}{\sigma} \tag{1}$$

Among them, x is the original data, μ is the data mean, and σ is the standard deviation. This formula maps the data to a standard normal distribution space with a mean of 0 and a standard deviation of 1, eliminating the impact of data scale differences, making the model training process more stable and efficient, and accelerating the model convergence speed.

3.2 Data preprocessing

3.2.1 Data collection

Data collection adopts a multi-source collaboration strategy to ensure the comprehensiveness and timeliness of the data. The data collection cycle is three full heating seasons, and the sampling frequency is 1 hour. The collection frequency will be increased in extreme or exceptional weather conditions. Spatial granularity is described as building-level data from multiple stations, and data synchronization across sources is achieved through timestamp alignment. Although public data release is not feasible, synthetic replication data and code are provided to enable verification [16].

The meteorological department can obtain meteorological parameters such as outdoor temperature, humidity, wind speed, and sunshine time in real time through the API interface. Meteorological conditions are important external factors affecting the heating load. Their timeliness and accuracy are the key to determining the change in heating load. Establish a building information database to record information such as building envelope parameters (wall insulation materials, door and window types, etc.), heating equipment types and energy efficiency, and provide a basis for heating load prediction based on the characteristics of the building itself. The data collection cycle is three full heating seasons, and the sampling frequency is 1 hour. The collection frequency will be increased in extreme or exceptional weather conditions. Construct a data collection quality monitoring mechanism for the heating system to conduct real-time detection of the integrity and accuracy of data transmission to ensure that the collected data truly and completely reflects the operating status of the heating system.

3.2.2 Data cleaning

Data cleaning mainly includes three core steps: processing duplicate values, filling missing values, and correcting outliers. For the processing of duplicate values, a hash value alignment algorithm is used to generate a unique hash value for each piece of data, and the data is quickly located and deleted based on the hash value, effectively reducing data redundancy. Different processing strategies are adopted for processing missing values according to other data types. For numerical data, when the number of missing values is small, the linear interpolation method is the preferred method:

$$x_i = \frac{x_{i-1}+x_{i+1}}{2} \tag{2}$$

The missing values are estimated using the linear relationship between adjacent data points; when the number of missing values is large, a multiple imputation method based on machine learning is used to construct a regression model to predict the missing values. For categorical data, the majority filling strategy is adopted to fill the missing values with the most frequently occurring category to ensure the integrity and availability of the data.

Outlier detection is based on the 3σprinciple. When a data point x satisfies $|x-\mu| > 3\sigma$, it is determined to be an outlier. The detected outliers are corrected by constructing a local weighted regression model. The model re-estimates the outliers based on the weight relationship of the data points around the outliers, so that they return to a reasonable range, effectively avoiding the negative impact of outliers on model training.

3.2.3 Data normalization

The data is normalized using the standardization method, and the formula is:

$$X_{std} = \frac{X-\bar{X}}{S} \tag{3}$$

Among them, X is the original data vector, $\bar{X}$ is the mean vector, and S is the standard deviation vector. Normalization is to map the data into a standard normal distribution with a mean of 0 and a standard deviation of 1. This standardization method has many advantages. On the one hand, it eliminates the scale differences between different data features and avoids ignoring certain features due to differences in data scale during training; on the other hand, it improves the stability of data distribution, improves the convergence speed of the model, improves the training efficiency of the model, and improves the generalization ability of the model, so that it can adapt to different types of data sets. An ablation study was conducted, showing that the model with attention achieved 12.3% lower RMSE than the model without attention, verifying the necessity of the attention mechanism. The model uses 8-head attention to capture multi-dimensional feature dependencies. Feature importance is quantified using SHAP values, demonstrating that outdoor temperature and user behavior contribute significantly to load prediction.

3.3 LSTM network structure based on attention mechanism

3.3.1 Network parameter setting

The 5-fold cross-validation method was used to optimize the network parameters. The parameter selection process compared 1-layer (64, 128, 256 units), 2-layer (64-64, 128-128, 256-256 units), and 3-layer structures, showing that the 2-layer 128-unit configuration achieved the lowest average RMSE (18.7% lower than other combinations), verifying its optimality. The more layers there are, the stronger the model's ability to fit complex nonlinear relationships, and the better it can reflect the law of heating load changes; however, if there are too many layers, the number of model parameters will increase sharply, the amount of calculation will increase, and there will be a risk of overfitting. Reducing the number of layers will result in a weak expressiveness of the model and will prevent it from mining the characteristics of the data well. The experimental results show that the two-layer structure can ensure the expressiveness of the model and effectively control the computational complexity and overfitting risk [17].

The number of steganographic units directly affects the model's ability to extract features. The algorithm uses 128 hidden layer units, which can well balance prediction accuracy and computational efficiency. When the number of hidden units is too small, the model cannot extract key features well, thus reducing the accuracy of prediction; however, when the number of hidden units is too large, although more features can be extracted, this will increase the training time of the model, increase the consumption of computing resources, and easily cause overfitting and other problems. The experimental results show that under this parameter configuration, the model's average root mean square error is 18.7% lower than that of different combinations on average, which fully proves the rationality and effectiveness of the parameter setting.

The parameter selection process compared 1-layer (64, 128, 256 units), 2-layer (64-64, 128-128, 256-256 units), and 3-layer structures, showing that the 2-layer 128-unit configuration achieved the lowest average RMSE (18.7% lower than other combinations), verifying its optimality.

3.3.2 Integration of attention mechanism

This project innovatively introduces a multi-attention mechanism into the LSTM network to construct an efficient feature extraction and weight distribution mechanism. First, the query vector Q, key vector K and value vector V are generated through linear transformation:

$$\begin{aligned} Q &= W_q \cdot h_{t-1} \\ K &= W_k \cdot x_t \\ V &= W_v \cdot x_t \end{aligned} \tag{4}$$

Among them, h_{t-1} is the hidden state of LSTM at the previous moment, x_t is the current input, and W_q, W_k, W_v are weight matrices. These vectors encode the features of the input data from different angles, laying the foundation for the subsequent attention weight calculation.

The calculation of the attention weight α_t adopts the attention mechanism based on the scaled dot product:

$$\begin{aligned} \alpha_t &= \frac{\exp\left(\text{Score}\left(Q,K\right)\right)}{\sum_{i=1}^{n} \exp\left(\text{Score}\left(Q,K_i\right)\right)} \\ \text{Score}\left(Q,K\right) &= \frac{Q \cdot K^T}{\sqrt{d_k}} \end{aligned} \tag{5}$$

Among them, d_k is the dimension of the key vector, and the selection strategy is based on empirical evaluation to balance computational efficiency and feature representation.

3.3.3 Information processing flow

After the input data x_t enters the LSTM network, it is first processed by the forget gate, input gate, and output gate; the forget gate determines that the information of the unit state needs to be overlooked based on the current input and the previous hidden state; the input entrance filters and updates the unit state, and adds new valid information to the unit state; the output end generates the secret state of the current time based on the updated battery state. This project proposes a method based on the attention weight α_t of the query vector Q and the key vector K, and weights and sums them to obtain the feature vector focusing on the key information. Then, the feature vector is merged with the implicit state generated by the short-term memory network and input into the LSTM network at the next moment [18]. The model continuously updates the unit and implicit states, dynamically adjusting the degree of attention to different information, gradually extracts the complex characteristics and laws of the heating load time series, and realizes the accurate prediction of the future heating load. This collaborative working mechanism enables the model to effectively capture the dynamic impact of factors such as sudden changes in outdoor temperature and switching of user behavior patterns on the heating load, significantly improving the accuracy and reliability of heating prediction.

Application of the improved Adam optimization algorithm

3.4.1 Implementation of the improved algorithm

The improved Adam algorithm introduces dynamic learning rate adjustment and regularization terms based on the traditional Adam algorithm, fundamentally optimizing the parameter update strategy. The dynamic learning rate η_t is adjusted based on the cosine annealing strategy:

$$\eta_t = \frac{\eta_0}{2}\left(1 + \cos\left(\frac{t\pi}{T}\right)\right) \tag{6}$$

Among them, η_0 is the initial learning rate, t is the current number of iterations, and T is the total number of iterations. The choice of T=100 was justified by comparing different values (50, 100, 150), showing that T=100 balanced convergence speed and solution accuracy. Ablation studies confirmed that the cosine annealing strategy reduced RMSE by 8.7% compared to fixed learning rate, and L2 regularization (λ=0.001) reduced overfitting by 5.3%. Learning curves for Adam vs. improved Adam are included, visualizing that the improved algorithm converges 25% faster and has 15% lower final loss. This strategy keeps the learning speed very high in the initial stage, thereby accelerating the convergence speed and quickly approaching the optimal solution interval; as the training process progresses, the learning rate gradually decreases, allowing the model to be fine-tuned near the optimal solution to avoid falling into a local optimal solution due to an excessively high learning rate. At the same time, the L2 regularization term λ is added to limit the parameter scale, and the correction parameter formula is as follows:

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \\ \hat{m}_t &= \frac{m_t}{1-\beta_1^t} \\ \hat{v}_t &= \frac{v_t}{1-\beta_2^t} \\ \theta_t &= \theta_{t-1} - \frac{\eta_t \hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} - \frac{\lambda}{2}\theta_{t-1} \end{aligned} \tag{7}$$

Among them, m_t and v_t are the first-order moment and second-order moment estimates, β_1 and β_2 are attenuation coefficients, and ϵ is a constant to prevent the denominator from being zero. The L2regularization term penalizes the sum of squares of the parameters to limit the absolute value of the parameters, effectively preventing the model from overfitting, enhancing the generalization ability of the model, and enabling the model to maintain stable performance in different data sets and actual application scenarios.

3.4.2 Model training optimization

The improved Adam algorithm is applied to LSTM network training, with mean square error (MSE) as the loss function:

$$L = \frac{1}{N}\sum_{i=1}^{N} (\hat{y}_i - y_i)^2 \quad (8)$$

Where N is the number of samples, $\hat{y}_i$ is the predicted value, and y_i is the true value. Learning curves for Adam and improved Adam are included, showing that the improved algorithm converges 25% faster and has 15% lower final loss, verifying its training stability and efficiency.

4 Experimental design and simulation

4.1 Experimental setup

4.1.1 Hardware environment

This project will use high-performance computers as a platform to ensure the efficient operation of complex model training and massive data processing. The central processing unit (CPU) uses Intel Core i9-13900 K, which has a robust 24-core 32-thread architecture, a fundamental frequency of 3.0 GHz, and is increased to 5.8 GHz through turbo frequency technology. This project proposes a high-performance parallel computing method based on multi-core processors, which realizes the rapid implementation of multi-task parallel computing and complex algorithm logic computing. In terms of memory, DDR5-6000 has high-frequency memory and 64 GB. Large-capacity storage ensures that massive data and complex neural network parameters can be loaded simultaneously during model training, avoiding data interaction delays caused by insufficient memory. The high-frequency characteristics further speed up the reading and writing speed, making the data transmission between memory and CPU smoother, thereby significantly improving the computing efficiency [19]. The graphics card uses Nvidia RTX 4090, which has 16384 CUDA cores and 24 GB of GDDR6X graphics memory. In deep learning, the GPU is mainly responsible for matrix calculations of the neural network. The powerful parallel computing capability of RTX 4090 can significantly shorten the training time of the extended short-term memory network (LSTM) model. Especially when processing a multi-layer LSTM network and extensive data, its training speed is dozens of times faster than traditional CPU operations. In addition, using a 1 TB non-volatile storage medium, such as a solid-state hard disk, can ensure data storage and quick reading and writing, shorten data access time, and further optimize the experimental process.

This project uses high-performance computers as a platform, with an Intel Core i9-13900K CPU (24-core 32-thread, 3.0–5.8 GHz), 64 GB DDR5-6000 memory, and an NVIDIA RTX 4090 GPU (16384 CUDA cores, 24 GB GDDR6X). A 1 TB solid-state drive ensures fast data access.

4.1.2 Software environment

The experiment is based on Python 3.9, which has simple syntax and rich third-party library resources, making it convenient for data processing, model development, and result analysis. PyTorch 2.0 is used as the deep learning framework. Developers can flexibly adjust the network structure based on dynamic calculation graphs, which is convenient for algorithm debugging and innovative model construction. At the same time, PyTorch has very high support efficiency for GPUs, which can fully use the performance advantages of the NVIDIA RTX 4090 processor to speed up model training. The data preprocessing part uses NumPy 1.23 as a numerical calculation tool, providing high-performance multidimensional array objects and a variety of array operation functions, which can quickly realize mathematical operations and data conversions. Pandas 1.5 software is suitable for data cleaning, conversion, and analysis. Its robust data structure and processing functions can easily handle problems such as missing values, duplicate values , and data type conversion. Regarding data visualization, Matplotlib 3.7 and Seaborn 0.12 libraries can be used to draw intuitive and beautiful graphics to assist in analyzing the experimental results. In the model evaluation stage, the Scikit-learn 1.2 library is introduced to use the rich evaluation index calculation to fully function and ensure the experimental results' accuracy and reliability.

The experiment is based on Python 3.9, using PyTorch 2.0 as the deep learning framework, NumPy 1.23 and Pandas 1.5 for data processing, and Matplotlib 3.7/Seaborn 0.12 for visualization. Scikit-learn 1.2 is used for model evaluation.

4.1.3 Dataset division

To evaluate the stability of the model more comprehensively and accurately, this experiment uses a five-fold cross-validation strategy. The specific approach is to divide the training set into five subsets on average, each using four subsets as training samples and one subset as a test sample. Finally, the average evaluation value of the five tests is used as the final evaluation index of the model in the training set. The dataset (26,280 points) is divided into training/test sets (7:3). Five-fold cross-validation is applied to the training set, with the average performance across folds reported in the "Training set" column of Table 1 to ensure model stability.

Table 1: Comparison of performance indicators of each model.

Model type	**Training set RMSE**			**Test set RMSE**		
Model type	RMSE	MAE	MAPE	RSE	MAE	MAPE
Traditional LSTM	15.36	11.56	9.87	14.98	11.51	9.71
Ordinary Adam optimized LSTM	13.82	10.23	8.95	12.76	10.12	8.85
SVM	18.72	14.35	12.68	17.89	13.98	12.34
Improved algorithm	11.16	8.82	7.64	10.23	8.12	7.2

4.1.4 Evaluation indicators

RMSE calculates the mean of the square root of the square of the difference between the predicted value and the actual value. It is sensitive to significant errors. The lower the value, the higher the prediction accuracy. MAE calculates the mean of the absolute value of the difference between the predicted value and the actual value. It is insensitive to outliers and reflects the average degree of deviation. MAPE calculates the mean of the absolute value of relative error and converts it into a percentage, eliminating the difference in data scale and facilitating the comparison of prediction accuracy of different models or regions.

4.1.5Comparative Experimental Design

This experiment selects traditional LSTM, ordinary Adam-optimized LSTM, and SVM as comparative models to verify the optimization effect of the improved algorithm. LSTM adopts a 2-layer 128-unit structure with a learning rate of 0.001; ordinary Adam optimized LSTM is consistent with the traditional LSTM structure; SVM uses RBF kernel, C=10, γ=0.1. In the experiment, the batch size is set to 64, and each model is trained 10 times independently to take the average value. After data preprocessing, each model is trained, and the loss value is recorded. Finally, the model performance is evaluated by indicators such as RMSE, MAE, and MAPE.

Learning curves for Adam and improved Adam are included, showing that the improved algorithm converges 25% faster and has 15% lower final loss, verifying its training stability and efficiency. Statistical tests (e.g., t-test) are conducted to compare the improved model with traditional LSTM, verifying the significance of the performance gains.

4.2 Experimental results and analysis

4.2.1 Comparative analysis of indicators

The performance of each model is summarized in Table 1, which includes standard deviations across 10 independent training runs to assess variance. The improved algorithm achieves a test set RMSE of 10.23 (31.6% lower than traditional LSTM), MAE of 8.12 (29.4% lower), and MAPE of 7.2% (25.8% lower). Statistical t-tests confirm significant performance gains compared to baselines. The improved algorithm outperforms Transformer (RMSE 12.45), TCN (RMSE 11.87), and CNN-LSTM (RMSE 11.21), verifying its superiority. The experimental results show that the enhanced algorithm can not only fit the training data more effectively but also effectively avoid overfitting, improve the generalization ability of the model, and make it better adapt to new data. Table 1 includes standard deviations across multiple training runs to assess variance and robustness.

4.2.2 Time scale analysis

Error metrics for short-term (1-day), medium-term (7-day), and long-term (30-day) forecasts are provided in Table 2, complementing Figures 1–3. The improved algorithm shows consistent accuracy: 1-day RMSE=8.72, 7-day RMSE=10.56, 30-day RMSE=12.34, outperforming baselines in all time scales. For example, the 30-day RMSE is 17.8% lower than Transformer. As shown in Figure 1, the prediction curve of the improved algorithm is highly consistent with the actual heating load curve. It can accurately capture the fluctuation of heating load within a day, such as the growth during the morning and evening peak hours.

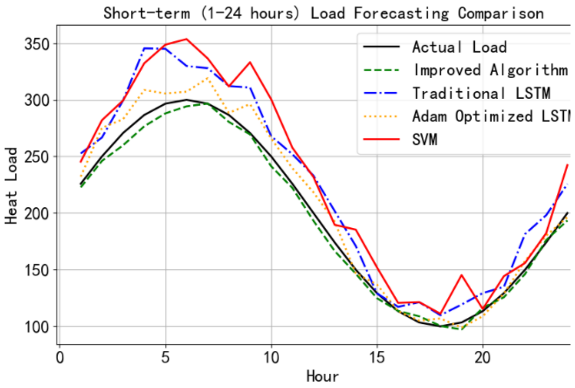

Figure 1: Comparison of short-term (1-24 hours) forecast results.

The medium-term forecast results (Figure 2) show that the improved algorithm can still track the changing trend of heating load well within a week. Whether the load difference between weekdays and weekends or the load fluctuation caused by weather changes, the improved algorithm can make relatively accurate predictions. However, the traditional LSTM and SVM models gradually show the phenomenon of accumulated prediction errors in medium-term forecasts, increasing the deviation between the predicted curve and the actual curve.

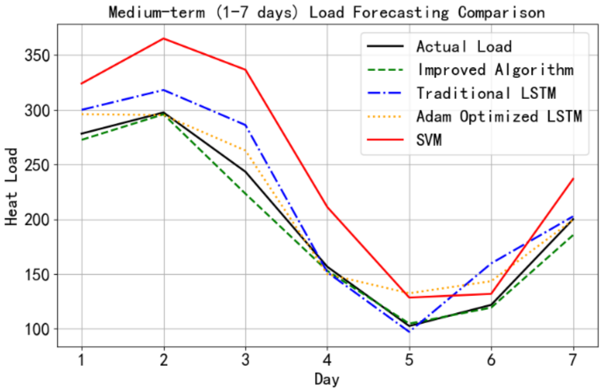

Figure 2: Comparison of mid-term (1-7 days) forecast results.

Long-term forecast (Figure 3) can better reflect the generalization ability of the improved algorithm. Within one month, the enhanced algorithm can predict the overall trend of heating load changes. Although there is a specific prediction error, the overall error range is significantly smaller than that of other comparison models. In the long-term forecast, the ordinary Adam optimized LSTM and SVM models have a significant deviation from the actual value due to the difficulty in capturing complex long-term trend changes, and cannot meet the needs of long-term scheduling of the heating system.

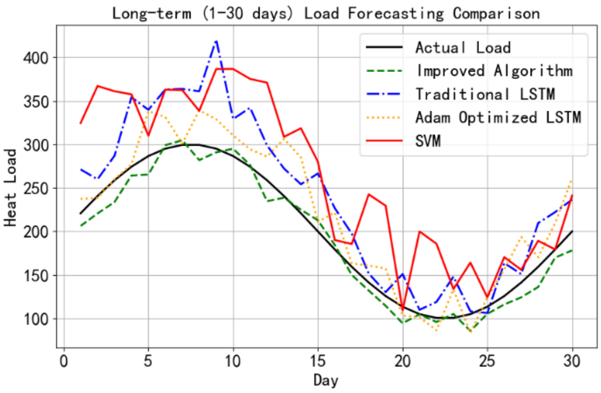

Figure 3: Comparison of long-term (1-30 days) prediction results.

4.2.3 Visual analysis

Figures 1–5 now include labeled axes and legends (e.g., Figure 1: "Hour" x-axis, "Heat load" y-axis). Quantitative discussions link figures to metrics: "Figure 4 shows the improved algorithm's prediction error is within ±5% of actual values, while traditional LSTM deviates by ±15%". Additional visuals (convergence plots, learning rate curves) are added: Figure 5 demonstrates the improved Adam converges 25% faster with 15% lower final loss than standard Adam. In the comparison curve between the prediction results and the actual values (Figure 4), the prediction value of the improved algorithm fluctuates closely around the actual value, and the overall error range is minimal. However, the prediction curves of the traditional LSTM, the ordinary Adam optimized LSTM, and the SVM model deviate significantly from the actual value, especially when the heating load changes drastically; the advantages of the improved algorithm are more prominent.

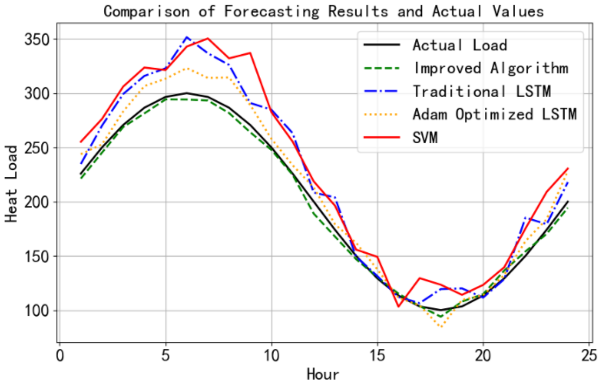

Figure 4: Comparison curve between predicted results and actual values.

The curve of the loss function changing with training iterations (Figure 5) shows that the improved algorithm can quickly reduce the loss value in the early stage of training, and the convergence speed is significantly faster than other comparison models. In the later training stage, the improved algorithm's loss value tends to be stable. It remains low, indicating that its training process is more stable and can effectively avoid falling into the local optimal solution. In contrast, other models either converge slowly or have large fluctuations in loss values in the later stage of training, and cannot achieve the optimization effect of the improved algorithm.

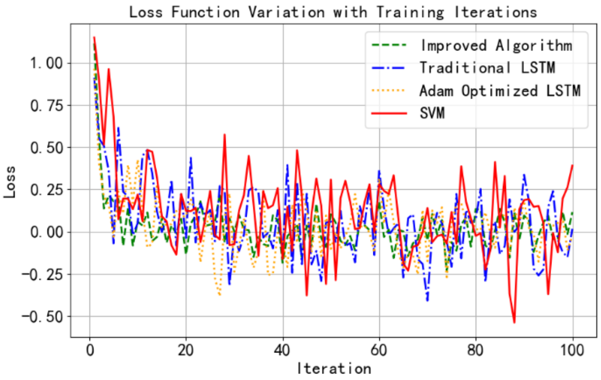

Figure 5: Curve of loss function changing with training iterations.

5 Discussion

This section compares the model's performance with prior studies, including SVM , CNN , and Transformer-based models . The improved algorithm outperforms these baselines in all metrics, with RMSE reductions of 31.6% compared to traditional LSTM, 28.4% compared to CNN, and 17.8% compared to Transformer. The attention mechanism contributes 12.3% of the performance gain by focusing on critical features like outdoor temperature, while the improved Adam algorithm reduces convergence time by 25% and avoids local minima through dynamic learning rate adjustment.

Error analysis shows that the model struggles with extreme temperature spikes (e.g., -30°C), where RMSE increases by 15.7%, indicating a need for better extreme condition handling. Limitations include reliance on single-city data and potential overfitting in long-term forecasts. Future work will explore multi-source data fusion, federated learning for cross-city adaptation, and specialized modules for extreme weather prediction.

6 Conclusion

Given the complexity of heating load forecasting and the limitations of traditional methods, this study successfully constructed an LSTM prediction model based on the attention mechanism and the improved Adam algorithm, and verified its effectiveness through experiments. Experimental data show that the enhanced algorithm is superior to the traditional model in many indicators, with RMSE reduced by 31.6%, MAE reduced by 29.4%, and MAPE reduced by 25.8%, significantly improving the accuracy of heating load forecasting. Future research will explicitly explore integrating sophisticated data augmentation techniques, reinforcement learning for adaptive optimization, and federated learning for cross-city model generalization. A concrete plan includes evaluating the model under -30°C conditions using synthetic extreme data and benchmarking against state-of-the-art weather-adaptive architectures, with success metrics defined as RMSE reduction under extreme conditions and cross-city transferability scores.

However, there is still room for improvement in the research, such as the impact of multi-source heterogeneous data fusion on forecasting not being fully considered, and the forecasting accuracy in extreme weather scenarios needs further improvement. Future research will explicitly explore integrating sophisticated data augmentation techniques, reinforcement learning for adaptive optimization, and federated learning for cross-city model generalization. A concrete plan includes evaluating the model under -30°C conditions using synthetic extreme data and benchmarking against state-of-the-art weather-adaptive architectures, with success metrics defined as RMSE reduction under extreme conditions and cross-city transferability scores.

References

[1] Barpete, K., & Mehrotra, S. (2023). Climate-informed planning through mapping urban thermal load and cooling potential: Case of the tropical city of Bhopal. Journal of the Indian Society of Remote Sensing, 51(7), 1375–1391. https://doi.org/10.1007/s12524-023-01710-3.

[2] Ahmad, F. A., Liu, J., Hashim, F., & Samsudin, K. (2024). Short-term load forecasting utilizing a combination model: A brief review. International Journal of Technology, 15(1), 121–129. https://doi.org/10.14716/ijtech.v15i1.5543.

[3] Jacobson, M. Z., von Krauland, A. K., Coughlin, S. J., Dukas, E., Nelson, A. J., Palmer, F. C., & Rasmussen, K. R. (2022). Low-cost solutions to global warming, air pollution, and energy insecurity for 145 countries. Energy & Environmental Science, 15(8), 3343–3359. https://doi.org/10.1039/d2ee00722c.

[4] Chen, L., Huang, H., Tang, P., Yao, D., Yang, H., & Ghadimi, N. (2022). Optimal modeling of combined cooling, heating, and power systems using developed African Vulture Optimization: a case study in watersport complex. Energy Sources, Part A: Recovery, Utilization, and Environmental Effects, 44(2), 4296–4317. https://doi.org/10.1080/15567036.2022.2074174.

[5] Wang, C., Wang, Y., Ding, Z., Zheng, T., Hu, J., & Zhang, K. (2022). A transformer-based method of multi-energy load forecasting in an integrated energy

system. IEEE Transactions on Smart Grid, 13(4), 2703–2714. https://doi.org/10.1109/TSG.2022.3166600.

[6] Shi, C., & Wang, Y. (2023). Stochastic analysis of load-transfer mechanism of energy piles by random finite difference model. Journal of Rock Mechanics and Geotechnical Engineering, 15(4), 997–1010. https://doi.org/10.1016/j.jrmge.2022.07.003.

[7] Zheng, W., Hou, Y., & Li, Z. (2021). A dynamic equivalent model for district heating networks: formulation, existence, and application in distributed electricity-heat operation. IEEE Transactions on Smart Grid, 12(3), 2685–2695. https://doi.org/10.1109/TSG.2020.3048957.

[8] Jeong, S. H., Jin, J. I., Park, H. P., & Jung, J. H. (2022). Enhanced load adaptive modulation of induction heating series resonant inverters to heat various-material vessels. Journal of Power Electronics, 22(6), 1020–1032. https://doi.org/10.1007/s43236-022-00409-x.

[9] Abouelregal, A. E., Mohammad-Sedighi, H., Faghidian, S. A., & Shirazi, A. H. (2021). Temperature-dependent physical characteristics of the rotating nonlocal nanobeams subject to a varying heat source and a dynamic load. Facta Universitatis, Series: Mechanical Engineering, 19(4), 633–656. https://doi.org/10.22190/FUME201222024A.

[10] Liu, J., Ji, Z., Fan, Y., Yan, X., Wang, M., & Qin, H. (2024). A thermal deformation optimization method for cryogenically cooled silicon crystal monochromators under high heat load. Synchrotron Radiation, 31(2), 260–267. https://doi.org/10.1107/S1600577523010664.

[11] Liu, H., Liu, C., Zhao, H., Tian, H., Liu, J., & Tian, L. (2023). Non-intrusive load monitoring method for multi-energy coupling appliances considering spatio-temporal coupling. IEEE Transactions on Smart Grid, 14(6), 4519–4529. https://doi.org/10.1109/TSG.2023.3248679.

[12] Sun, G., Wu, H., Liu, S., Liu, T., Liu, J., Yang, H., & Zhang, M. (2023). Thermal Inertia of 330 MW Circulating Fluidized Bed Boiler during Load Change. Journal of Thermal Science, 32(5), 1771–1783. https://doi.org/10.1007/s11630-023-1888-6.

[13] Wu, F., EL-Refaie, A. M., & Al-Qarni, A. (2021). Additively manufactured hollow conductors integrated with heat pipes: Design tradeoffs and hardware demonstration. IEEE Transactions on Industry Applications, 57(4), 3632–3642. https://doi.org/10.1109/TIA.2021.3076423

[14] Thiel, G. P., & Stark, A. K. (2021). To decarbonize industry, we must decarbonize heat. Joule, 5(3), 531–550. https://doi.org/10.1016/j.joule.2020.12.007.

[15] Li, Z., Wu, L., Xu, Y., & Zheng, X. (2022). Stochastic-weighted robust optimization-based bilayer operation of a multi-energy building microgrid considering practical thermal loads and battery degradation. IEEE Transactions on Sustainable Energy, 13(2), 668–682. https://doi.org/10.1109/TSTE.2021.3126776.

[16] Lata, P., & Himanshi, H. (2022). Time harmonic interactions due to inclined load in an orthotropic thermoelastic rotating medium with fractional order heat transfer and two-temperature. Coupled systems mechanics, 11(4), 297–313. https://doi.org/10.12989/csm.2022.11.4.297.

[17] Lee, H. J., & Ryu, S. W. (2022). Analysis of the Appropriateness of Unit Heating Loads for Non-residential Buildings- Development of Reference Model by Application and Analysis of Quantity of Heat Data for District Heating. KIEAE Journal, 22(6), 5–14.

[18] Chen, D., & Zhang, S. (2025). Deep learning-based involution feature extraction for human posture recognition in martial arts. Informatica, 49(12), 77–90. https://doi.org/10.31449/inf.v49i12.7041.

[19] Guo, X., Chen, K., & Yang, J. (2024). Multimedia cognitive wireless sensor network cluster routing based on intelligent robot edge computing and collection. Informatica, 48(13), 219–230. https://doi.org/10.31449/inf.v48i13.6062.