Abstract—To reduce environmental pollution and improve the efficiency of cascaded energy utilization, regional integrated energy system (RIES) has received extensive attention. An accurate multi-energy load prediction is significant for RIES as it enables stakeholders to make effective decisions for carbon peaking and carbon neutrality goals. To this end, this paper proposes a multivariate two-stage adaptive-stacking prediction (M2ASP) framework. First, a preprocessing module based on ensemble learning is proposed. The input data are preprocessed to provide a reliable database for M2ASP, and highly correlated input variables of multi-energy load prediction are determined. Then, the load prediction results of four predictors are adaptively combined in the first stage of M2ASP to enhance generalization ability. Predictor hyper-parameters and intermediate data sets of M2ASP are trained with a metaheuristic method named collaborative atomic chaotic search (CACS) to achieve the adaptive staking of M2ASP. Finally, a prediction correction of the peak load consumption period is conducted in the second stage of M2ASP. The case studies indicate that the proposed framework has higher prediction accuracy, generalization ability, and stability than other benchmark prediction models.

Index Terms—Collaborative atomic chaotic search (CACS), multivariate two-stage adaptive-stacking prediction (M2ASP) framework, prediction error correction, regional integrated energy system (RIES).

I. INTRODUCTION

The accelerated construction of regional integrated energy system (RIES) improves the comprehensive utilization efficiency of various energy forms and contributes to environmental protection. The RIES utilizes the advanced technologies of physical information systems and novel management models to integrate heterogeneous energy sources such as electricity and heat within a volume of time and space [1], [2]. As a result, the high-quality operation of diversified energy can be systematically achieved. With the promotion of RIES, multi-energy load prediction has become an increasingly vital prerequisite for the operation effects of energy networks as it enables stakeholders to make effective decisions such as dispatch and maintenance. The accuracy, flexibility, and efficiency of multi-energy load prediction are significant to realize the balance between supply and demand in RIES. However, the volatility, uneven distribution, and seasonal changes of renewable energy sources (RESs) bring challenges to the reliable supply of RIES and the judgment of multi-energy prediction systems [3]. As the coupling relations among different integrated energy subsystems are challenging to determine, the representative influencing factors of multi-energy load prediction are more complex than that of the traditional distribution networks.

Many studies have been performed on short-term load prediction because of its indispensable role in dispatch. The existing single prediction models of short-term load prediction can be broadly divided into parametric and data-driven algorithms. However, uncertain meteorological and sociological events bring many uncertainties to the multi-energy load in RIES, and the parametric method is challenging to solve nonlinear issues and complex influencing factors. Data-driven algorithms like random forest (RF) [4], support vector regression (SVR) [5], and artificial neural networks can achieve satisfactory predictive performance in short-term load prediction with low time consumption. Reference [6] presents a wavelet neural network (WNN) model optimized by the improved particle swarm optimization (IPSO) and chaos optimization algorithm (COA) for short-term load prediction of RIES. Although an existing single model can predict multi-energy load with acceptable accuracy, the restrictions of a particular model and large hypothesis space will lead to poor generalization ability and over-fitting problems [7].

To this end, ensemble learning is adopted to predict RIES loads. Multiple data-driven models are integrated into an ensemble prediction framework to obtain better performance than any single model with a complementary mechanism.
A suitable combination strategy will provide a better fitting performance and prediction accuracy and alleviate the overexpression in the hypothesis space. Reference [8] presents a multivariate ensemble forecast framework (MEFF) with a systematic combination of three multiple predictors: Elman neural network, feedforward neural network, and radial basis function neural network. Reference [9] presents a hybrid model integrated with long short-term memory (LSTM), encoder-decoder, and gradient boosting decision tree to improve the prediction precision of multi-energy loads. Among the ensemble techniques, the stacking ensemble framework (SEF) [10] is well-known for its favorable robustness and prediction accuracy, which has been gradually employed in the multi-energy load prediction domains.

It is worth noting that limited research has been published on improving the generalization ability and training efficiency of SEF. When the traditional SEF generates the intermediate data set that is used to integrate all base-predictor knowledge, the weighted average of each predictor’s results can weaken the performance of strong predictors due to the interference of weak ones. Each predictor model’s correlation is ignored. In addition, a massive increase of predictors may in fact reduce the training speed of the entire SEF with limited accuracy improvement. The enormous computational cost makes it challenging to applying the SEF to large-scale RIES. Meanwhile, the prediction error of multi-energy load mainly comes from the peak period, for which most existing predictors could not address well due to the meteorology variation, coupling relations, and other uncertainties. Therefore, it is worthwhile to dynamically change the SEF structure to accommodate the characteristics of the adopted predictor models. The prediction correction of peak load based on a separate SEF with strong generalization ability can be an effective way to improve prediction accuracy.

It was found that the randomness of random search can match the stacking framework pattern very well [7]. Reference [11] indicates that randomly chosen trials are more efficient for hyper-parameter optimization than trials on a grid. To exploit the potential capabilities of SEF, a suitable random search method is necessary to optimize the inputs and hyper-parameters for the SEF. However, the search space in such a problem grows exponentially, and the conventional optimization methods often produce only suboptimal solutions [12]. To this end, gradient-free mechanisms enable metaheuristic algorithms to bypass the calculation of derivatives of the search space. Some of the population-based metaheuristic algorithms such as genetic algorithm (GA), grey wolf optimizer (GWO), and particle swarm optimization (PSO) [6] are popular in hyper-parameter optimization. More recently, [13] presents atomic orbital search (AOS), a novel metaheuristic algorithm based on the principles of the quantum-based atomic model and quantum mechanics, which outperforms other metaheuristic algorithms in converging to the global optimum in multiple mathematical functions. However, the AOS in dealing with complex optimization problems remains challenging [13]. Photon determination and magnetic field strategies of the original AOS method cannot balance the exploration and exploitation well.

In this paper, a multivariate two-stage adaptive-stacking prediction (M2ASP) framework is investigated, whose structure can be adaptively changed based on the performance of each predictor. An ensemble-learning-based module is employed to construct a data-driven short-term load prediction model as follows: ① a preprocessing module based on ensemble learning is first proposed. Triangular interpolation (TI) is adopted to fill in the null data. Isolation forest (IF) is applied to identify abnormal data. Adaptive variational mode decomposition (AVMD) is used to extract load feature components. Self-organizing maps (SOMs) are adopted to fuse approximate load components. The RF with recursive feature elimination (RF-RFE) is employed to determine the highly correlated input variables for M2ASP; ② the M2ASP is proposed to predict the multi-energy load and correct the prediction errors. A complete M2ASP contains two independent multivariate adaptive-stacking prediction frameworks (MASP); and ③ a collaborative atomic chaotic search (CACS) method is employed to optimize the M2ASP stacking structure and predictor hyperparameters. Although the optimization results of each run of the metaheuristic method are different, the difference in hyper-parameters of each independent predictor model can enhance the generalization ability of MASP. The major contributions of this paper can be highlighted as follows.

1) An M2ASP is proposed, alleviating the blind stacking of the predictor number in SEF. Multi-energy loads of RIES are initially predicted in the first stage, and peak load errors are reduced in the second stage.

2) A preprocessing module based on ensemble learning is first proposed, where the perceived multi-energy load data is restructured by TI, IF, AVMD, and SOM, and representative input variables of M2ASP are determined by RF-RFE.

3) For the correlated exogenous variables and uncertain electricity consumption behavior of users, the prediction errors of the peak load consumption period can be effectively reduced by M2ASP due to the strong generalization ability of MASP.

4) A population-based metaheuristic algorithm named CACS is adopted to optimize the hyper-parameters and intermediate data of M2ASP, which achieves the adaptive staking of M2ASP predictors. The proposed CACS is suitable for the M2ASP structure.

5) The case studies demonstrate that the proposed framework achieves at least an 8.5% improvement in multi-energy load prediction accuracy, fitness, generalization ability, and stability compared with benchmark prediction models.

The remaining paper is organized as follows. Section II describes the framework of multi-energy load prediction in RIES. Section III proposes a preprocessing module based on ensemble learning. Section IV describes the working of CACS. Section V describes the principles of the M2ASP. Section VI provides the results of multi-energy load prediction in RIES using the proposed method and the comparison with benchmark prediction methods for validation. And Section VII concludes this paper.
II. FRAMEWORK OF MULTI-ENERGY LOAD PREDICTION IN RIES

To handle the complex coupling relationships among different energy forms, this paper proposes a framework of multi-energy load prediction in RIES, as shown in Fig. 1. The framework includes five layers, including RIES structure, data reconstruction, variable dimension reduction, load prediction, and error correction. First, the underlying RIES passes the raw data to the preprocessing module in the second layer for data reconstruction. Then, the input variable dimension can be dimensionally reduced by the variable dimension reduction of the third layer to eliminate irrelevant information. In the load prediction stage at the fourth layer, the outputs of base-predictors are adaptively combined to predict the multi-energy loads preliminarily. The error correction stage in the fifth layer conducts a prediction correction in the peak load consumption period. The final prediction result is the sum of the multi-energy load prediction result calculated by the first stage of M2ASP and the prediction error calculated by the second stage of M2ASP. Based on the above processes, the proposed method can predict the multi-energy loads one hour later. Therefore, the prediction results can provide accurate information for the operation and maintenance of RIES. The detailed introduction of the five layers in Fig. 1 is as follows, where AdaBoost is short for adaptive boosting.

A. **RIES Structure**

As an effective measure to solve the fossil energy crisis, climate changes, carbon peaking and carbon neutrality goals, and environmental pollution, the construction of RIES has received extensive attention. The first layer of Fig. 1 describes a brief RIES energy relationship. RIES connects different resources through multiple technologies to supply cold, heat, and electricity to the whole region with higher efficiency. The multi-energy load of RIES is not only periodic but also affected by the coupling between different types of loads.

![Fig. 1. Framework of multi-energy load prediction in RIES.](image)

B. **Data Reconstruction**

To detect essential features in RIES, a preprocessing module based on ensemble learning is employed to conduct the data reconstruction and variable dimension reduction. The essence of data reconstruction is to preprocess, decompose, and recombine the multi-energy loads, whose structure is shown in the second layer of Fig. 1.

C. **Variable Dimension Reduction**

Assume \( L_c(w, d, t) \), \( L_e(w, d, t) \), and \( L_t(w, d, t) \) are the electri-
cal, heating, and cooling loads of a particular hour of the day (HD) in a week. 27 possible influencing factors that may affect the multi-energy loads are considered as follows: \( L_i(w, \text{d}_1, t-\text{c}) \), \( L_i(w, \text{d}_2, t-\text{c}) \), and \( L_i(w, \text{d}_3, t-\text{c}) \) are the c-hour-ahead \((c \in \{1, 2, 3, 4\})\) electrical, heating, and cooling loads of the same day of the same week; \( L_i(w, \text{d}_4, t-1, \text{c}) \), \( L_i(w, \text{d}_5, t-1, \text{c}) \), and \( L_i(w, \text{d}_6, t-1, \text{c}) \) are the electrical, heating, and cooling loads of the same hour of the previous day of the same week; \( L_i(w-1, \text{d}_1, t) \), \( L_i(w-1, \text{d}_2, t) \), and \( L_i(w-1, \text{d}_3, t) \) are the electrical, heating, and cooling loads of the same hour of the same day of the previous week; \( \text{d}_1 \) humidity (H); \( \text{d}_2 \) wind speed (WS); \( \text{d}_3 \) solar radiation (SR); \( \text{d}_4 \) temperature (T); \( \text{d}_5 \) day type (off day or working day); \( \text{d}_6 \) HD; \( \text{d}_7 \) day of the week (DW); \( \text{d}_8 \) month of the year (MY); and \( \text{d}_9 \) week of the month (WM). By eliminating irrelevant interfering information, the preprocessing module based on ensemble learning selects input variables that are highly correlated with output variables. Multiple exogenous and meteorological variables are condensed into representative variables of M2ASP. The structure of this section is shown in the third layer of Fig. 1.

D. Load Prediction

Based on the results of data reconstruction and variable dimension reduction, an independent MASP model is proposed to predict RIES multi-energy loads, and a CACS algorithm is proposed to achieve the adaptive stacking of MASP parameters. The structure of load prediction is shown in the fourth layer of Fig. 1.

E. Error Correction

The objective of this subsection is to correct the RIES prediction error based on preliminarily predicted load information, whose structure is shown in the fifth layer of Fig. 1. The prediction error can be effectively alleviated in the fifth layer of Fig. 1.

III. PREPROCESSING MODULE BASED ON ENSEMBLE LEARNING

A. Data Preprocessing

In fact, load data loss is common due to unexpected failure or device maintenance, and it is necessary to preprocess the input data of M2ASP. TI is suitable for filling in null data in the training and test sets [14]. Due to equipment failure, noise interference, and communication delays, the measured data may deviate from the actual condition. Therefore, IF is adopted to eliminate abnormal historical load data in RIES under the condition that state estimation is inconvenient to conduct, which is an anomaly detection method suitable for continuous numerical data [15]. By randomly dividing the features of the historical load data, the child forest anomaly detectors are established, and the base forest anomaly detector is formed [15]. Data that can be easily separated out are considered as abnormal data. Then, TI replaces the abnormal data filtered by IF based on the daily load curves of the same day in different weeks.

B. Data Decomposition

Separately predicting each component of load through data decomposition is an effective way for RIES load prediction [16]. As an adaptive signal processing method suitable for non-stationary and nonlinear signals [17], AVMD overcomes the hard band limits of wavelet approaches and has more mathematical theoretical support than empirical mode decomposition [17]. The target of AVMD is to decompose the multi-energy loads into a discrete number of components, called intrinsic mode functions (IMFs) \( u_i \), corresponding to different frequencies and a residue. The \( k \)-th IMF must be compact around a center pulsation \( \omega_k \) [18]. The constrained variational problem is as follows:

\[
\min_{\{\hat{u}_i, \lambda\}} \left\{ \sum_{t=1}^{T} \left[ \hat{c}_i \left[ \delta(t) + \frac{1}{\pi t} \right] u_i(t) e^{j\omega_i t} \right]^2 \right\} + \left\| f(t) - \sum_{k=1}^{K} \hat{u}_k(t) \right\|^2_2
\]

(1)

where \( K \) is the number of modes decomposed; \( \{u_i\} = \{u_1, u_2, ..., u_K\} \) represents the set of all IMFs; \( \{\omega_k\} = \{\omega_1, \omega_2, ..., \omega_K\} \) represents the center frequencies of all IMFs; \( \delta(t) \) is the impulse function; and \( \hat{c}_i \) is the partial derivative of \( t \).

To render the equation unconstrained, the constraints reconstructed by employing a quadratic penalty term \( \alpha \) and Lagrangian multipliers \( \lambda(t) \) are as follows:

\[
L(\{u_i\}, \{\omega_k\}, \lambda) = \alpha \sum_{t=1}^{T} \left[ \hat{c}_i \left[ \delta(t) + \frac{1}{\pi t} \right] u_i(t) e^{j\omega_i t} \right]^2 + \left\| f(t) - \sum_{k=1}^{K} \hat{u}_k(t) \right\|^2_2 + \left\langle \lambda(t), f(t) - \sum_{k=1}^{K} \hat{u}_k(t) \right\rangle
\]

(2)

Based on the AVMD proposed in [19], \( K \), and initial \( \omega_k \), of IMFs can be determined with the guidance of the convergent tendencies of the initial \( \omega_k \) changing from small to large. After calculating the above problem, the multi-energy load sequence can be decomposed into \( K \) IMFs. Although the prediction of each IMF may improve the accuracy, the increase of output variables observably reduces the computation speed of the M2ASP model. Therefore, the cluster integration of IMFs is considered when making subsequent load predictions.

C. Cluster Fusion

Although the separate prediction of each IMF may improve the accuracy, the increase of predicted variables of the M2ASP model observably reduces the computation speed. Therefore, the SOM clustering algorithm [20] is adopted to combine the IMFs decomposed by AVMD into different groups. The regularity of the same group of IMFs becomes more similar. According to the cluster labels, the same group of IMFs is linearly added to form several new IMF sequences. Therefore, the amount of integrated data is reduced while the regularity of the IMF sequences becomes more prominent. Finally, these reconstructed IMF sequences are transferred to the M2ASP. The output variable of M2ASP is replaced by the IMF sequences of electrical, heating, and cool-
ing loads. For example, M2ASP is adopted to predict future IMF sequences decomposed by electrical loads rather than forecasting the values of electrical loads. At the end of the forecast, the prediction results of IMF sequences for the same load are added linearly to form the final results.

D. Feature Extraction

As the RF method has the in-built feature evaluation mechanisms that can measure the contribution of each variable to the prediction results, the nonsignificant features can be eliminated. To decrease the redundant information between features (variables), a hybrid RF-RFE method is employed to reduce the input variable dimensions and select highly correlated input variables, which evaluates variable importance and calculates feature weights based on the in-built feature evaluation mechanisms of RF. Through iterative screening, the combination of input variables with the highest prediction accuracy is retained, as shown in Fig. 2.

![Fig. 2. Implementation process of RF-RFE.](image)

IV. CACS

To optimize the predictor hyper-parameters and intermediate dataset of M2ASP, this paper proposes a new CACS algorithm to enhance the AOS algorithm. Based on collaborative behaviors, chaotic orbits, and dynamic photon strategy, the proposed CACS can overcome the limitations of AOS. CACS is a general meta-heuristic method that outperforms other benchmark optimizers in terms of adaptive stacking of M2ASP.

A. Quantum-based Atomic Model

Concepts borrowed from the inspiring discipline are shown in Table SAI and Fig. S1A of Supplementary Material A. The electron cloud around the nucleus is regarded as the search space. Each electron around the nucleus is considered as a solution candidate. The electron position can be defined by the decision variables. The energy state of each electron is regarded as the objective function value of each solution candidate. Besides, the electrons with lower energy levels represent the solution candidates with better objective function values. The schematic diagram of the quantum-based atomic model is shown in Fig. 3. Electrons are distributed in various imaginary spherical layers L. The radius $R_i$ of the $i$th layer demonstrates the way of distributing these layers around the nucleus, where the layer with the smallest radius represents the nucleus layer $L_n$ and the layers with larger radius $L_i$ (i.e., $i=1, 2, ..., n$) represent the 1st to the $n$th spherical layers $L_n$. By considering the imaginarily created layers around the nucleus, the PDF is utilized for determining the position of solution candidates in these layers. The solution candidate with the best objective function value represents the electron position with the lowest energy level $LE$. The electrons around the nucleus can be excited by the impact of photons, interaction with particles, or magnetic fields, which result in absorption or emission of energy. If an electron absorbs an amount of energy less than its binding energy, it will undergo a transition to an exciting energy level in the outer orbital. Besides, an electron is supposed to absorb an amount of energy more than its binding energy. In that case, it will be repositioned toward the lower energy level in the inner orbital.

![Fig. 3. Schematic diagram of quantum-based atomic model.](image)

B. AOS

Assume the solution candidates are as follows:

$$S = \begin{bmatrix}
S_{11} & S_{12} & \cdots & S_{1j} & \cdots & S_{1m} \\
S_{21} & S_{22} & \cdots & S_{2j} & \cdots & S_{2m} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
S_{q1} & S_{q2} & \cdots & S_{qj} & \cdots & S_{qm}
\end{bmatrix}
$$

$$i=1, 2, ..., q, j=1, 2, ..., m$$

(3)

where $S$ represents the electrons around the nucleus; $S_{ij}$ is the position of the $i$th solution candidate in the atom; $s_{ij}$ is the $j$th decision variable of the $i$th solution candidate; $q$ is the number of electrons inside the electron cloud; and $m$ is the dimension of electron position. The initial electrons are randomly determined as:
\[ s_{j,0} = s_{j,\text{min}} + R_0(s_{j,\text{max}} - s_{j,\text{min}}) \quad i = 1, 2, \ldots, q, j = 1, 2, \ldots, m \]  
where \( s_{j,0} \) is the initial position of the solution candidates; \( s_{j,\text{min}} \) is the minimum bound of the \( j^{th} \) decision variable for the \( i^{th} \) solution candidate; \( s_{j,\text{max}} \) is the maximum bound of the \( j^{th} \) decision variable for the \( i^{th} \) solution candidate; and \( R_0 \) is a random number in the range of \([0,1]\). The energy levels of different electrons are expressed as:

\[ \mathbf{O} = [O_1 \ O_2 \ \ldots \ O_i \ \ldots \ O_q]^T \quad i = 1, 2, \ldots, q \]  
where \( \mathbf{O} \) is a vector containing the objective function values of all electrons; and \( O_i \) is the energy level of the \( i^{th} \) electron. Each imaginary spherical layer has partial solution candidates. In this paper, \( \mathbf{O} \) is computed by the prediction error of the multi-energy loads, which can be quantified by the mean absolute percentage error (MAPE) of the prediction results.

The positions and energy levels of electrons in different imaginary layers can be expressed as:

\[
\mathbf{S}_1 = \begin{bmatrix}
S_{1,1} & S_{1,2} & \dots & S_{1,n} \\
S_{2,1} & S_{2,2} & \dots & S_{2,n} \\
\vdots & \vdots & \ddots & \vdots \\
S_{d,1} & S_{d,2} & \dots & S_{d,n}
\end{bmatrix}
\]

\[
\mathbf{O}_\lambda = \begin{bmatrix}
O_{1,1} & O_{1,2} & \dots & O_{1,n} \\
O_{2,1} & O_{2,2} & \dots & O_{2,n} \\
\vdots & \vdots & \ddots & \vdots \\
O_{d,1} & O_{d,2} & \dots & O_{d,n}
\end{bmatrix}
\]

where \( \mathbf{S}_1 \) is the vector of electrons in the \( k^{th} \) imaginary layer; \( S_{i,j} \) is the \( j^{th} \) electron in the \( i^{th} \) imaginary layer; \( \mathbf{O}_\lambda \) is the energy level of the electrons in the \( k^{th} \) imaginary layer; \( d \) is the total electron number in the \( k^{th} \) imaginary layer; and \( O_{i,k} \) represents the energy level of \( S_{i,k} \).

The binding state \( BS^k \) and binding energy \( BE^k \) in the \( k^{th} \) imaginary layer are calculated as:

\[ BS^k = \frac{\sum_{i=1}^{q} S_{i,k}}{q} \quad k = 1, 2, \ldots, n \]  
\[ BE^k = \frac{\sum_{i=1}^{q} O_{i,k}}{q} \quad k = 1, 2, \ldots, n \]  

The binding energy of each electron in every imaginary layer is compared to \( BE^k \) to decide the emission or absorption of photons for happening. If \( O_{i,k} \geq BE^k \), the electrons tend to emit a photon with a certain amount of energy to reach the energy \( LE \) and the state \( BS^k \) of the atom, which is expressed as:

\[ S_{i,k}(\lambda + 1) = S_{i,k}(\lambda) + a_i(\beta_i(LE - \gamma; BS^k))/k \]  

where \( S_{i,k}(\lambda) \) and \( S_{i,k}(\lambda + 1) \) are the \( \lambda^{th} \) and \( (\lambda + 1)^{th} \) iteration positions for the \( i^{th} \) electron of the \( k^{th} \) layer, respectively; and \( a_i, \beta_i, \) and \( \gamma_i \) are the randomly generated vectors in the range of \((0, 1)\) to quantify the amount of energy that will be exchanged.

If \( O_{i,k} < BE^k \), the electrons tend to absorb a photon with a certain amount of energy to reach the state \( BE^k \) and the electron position with the lowest energy level in the \( k^{th} \) imaginary layer \( (LE^k) \), which is expressed as:

\[ S_{i,k}(\lambda + 1) = S_{i,k}(\lambda) + a_i(\beta_i(LE^k - \gamma; BS^k))/k \]  

where \( LE^k \) is the solution candidate with the best objective function values in the \( k^{th} \) imaginary layer.

The probability of a photon acting on an electron depends on the randomly generated number \( \phi \) and the photon rate \( PR \). If \( \phi < PR \), the probability of a photon acting on an electron is 0. Therefore, the movement of electrons will mainly depend on magnetic fields, which are expressed as:

\[ S_{i,k}(\lambda + 1) = S_{i,k}(\lambda) + r_i \quad i = 1, 2, \ldots, q, k = 1, 2, \ldots, n \]  

where \( r_i \) is a randomly generated scalar in the range of \([0,1]\).

C. CACS

Although AOS has high search efficiency, AOS in dealing with complex optimization problems remains challenging [13]. Photon determination and magnetic field strategies of the original AOS method cannot balance the exploration and exploitation well. To overcome the limitation of the AOS algorithm and systematically utilize the photon rate, the CACS algorithm is proposed for adaptive stacking of MASP. Collaborative behaviors, chaotic orbits, and dynamic photon strategy are innovatively integrated into the AOS algorithm to enhance optimization capabilities, which are expressed as follows.

1) Collaborative Behaviors

Aiming to maintain a reasonable balance of exploration and exploitation, a collaborative atom model is proposed to evolve the electron search-ability based on the state-of-the-art knowledge of multiple atoms. Assume an atom group contains a leader atom and multiple follower atoms. The leader atom is responsible for exploration, and the follower atoms are responsible for exploitation. In an atom group, the atom containing the electron with the lowest energy level in the collaborative atom model is regarded as the leader atom. Other atoms are regarded as follower atoms. The position updating equations of the electrons in the follower atoms remain consistent with the original AOS method. Each electron of the leader atom updates its position based on the knowledge of itself and the follower atoms, using the historical best position of itself and the follower electrons as follows.

Assume \( L0_{i,k} \) represents the energy level of the \( i^{th} \) electron in the \( k^{th} \) imaginary layer for the leader atom and \( BE_{i,k} \) represents the binding state in the \( k^{th} \) imaginary layer for the leader atom. If \( L0_{i,k} \geq BE_{i,k} \), the leader electrons tend to emit a photon with multi-atomic cooperation. The solution candidates aim to reach the binding state of the considered atoms and the state of the electron with the lowest energy level among all the atoms. The position updating equation for a collaborative version is as follows:

\[ LS_{i,k}(\lambda + 1) = LS_{i,k}(\lambda) + a_i(2\beta_i; LE - \gamma; BS^k - \sigma; BS^k)/k \]  

where \( LS_{i,k}(\lambda) \) and \( LS_{i,k}(\lambda + 1) \) are the \( \lambda^{th} \) and \( (\lambda + 1)^{th} \) iteration positions for the \( i^{th} \) electron of the \( k^{th} \) layer, respectively; and \( a_i, \beta_i, \) and \( \gamma_i \) are the randomly generated vectors in the range of \((0, 1)\) to quantify the amount of energy that will be exchanged.

If \( L0_{i,k} < BE_{i,k} \), the electrons tend to absorb a photon with a certain amount of energy to reach the state \( BE_{i,k} \) and the electron position with the lowest energy level in the \( k^{th} \) imaginary layer \( (LE_{i,k}) \), which is expressed as:

\[ LS_{i,k}(\lambda + 1) = LS_{i,k}(\lambda) + a_i(\beta_i(LE_{i,k} - \gamma; BS^k))/k \]  

where \( LE_{i,k} \) is the solution candidate with the best objective function values in the \( k^{th} \) imaginary layer.

The probability of a photon acting on an electron depends on the randomly generated number \( \phi \) and the photon rate \( PR \). If \( \phi < PR \), the probability of a photon acting on an electron is 0. Therefore, the movement of electrons will mainly depend on magnetic fields, which are expressed as:

\[ LS_{i,k}(\lambda + 1) = LS_{i,k}(\lambda) + r_i \quad i = 1, 2, \ldots, q, k = 1, 2, \ldots, n \]  

where \( r_i \) is a randomly generated scalar in the range of \([0,1]\).
where \(LS_{s,i} (\lambda)\) and \(LS_{s,i} (\lambda+1)\) are the \(\lambda\)th and \((\lambda+1)\)th iteration positions of the \(i\)th electron in the \(k\)th layer for the leader atom, respectively; \(\sigma\) is the randomly generated vectors in the range of \((0,1)\); \(BS^s\) represents the binding state for the leader atom; \(d_L\) is the number of electrons inside the leader atom; \(BS^F\) represents the binding state for all the follower atoms; and \(q_F\) is the number of electrons inside the follower atoms.

If \(LO_{i,k} < BE^{L,k}\), the electrons tend to absorb a photon based on multi-atomic cooperation. The solution candidates aim to reach the binding state of the layer and the state of the electron with the lowest energy level inside the considered leader layer. The collaborative position updating equations are as follows:

\[
LS_{s,i} (\lambda+1) = LS_{s,i} (\lambda) + a_c \cdot (2 \beta \cdot LE^{L,k} - \gamma \cdot BS^{L,k} - \sigma \cdot BS^{F,k}) \\
BS^{L,k} = \frac{\sum_{i=1}^{d_L} S_{s,i} \cdot d_L}{k = 1, 2, \ldots, n} \tag{18}
\]

\[
BE^{F,k} = \frac{\sum_{i=1}^{d_F} O_{s,i} \cdot d_F}{k = 1, 2, \ldots, n} \tag{20}
\]

where \(d_L\) and \(d_F\) are the total electron numbers of the leader atom and follower atoms in the \(k\)th imaginary layer, respectively; \(LE^{L,k}\) represents the solution candidates with the best objective function values of the leader atom in the \(k\)th imaginary layer; \(BS^{L,k}\) is the binding state of the leader atom in the \(k\)th imaginary layer; and \(BS^{F,k}\) is the binding state of the follower atom in the \(k\)th imaginary layer. If \(\varphi < PR\), electrons will enter chaotic orbits to search for optimal solutions.

2) Chaotic Orbits

The original AOS simulates the effects of magnetic fields on electrons by randomly generated vectors \(r\). However, adding randomly generated vectors can miss significant points in the search space. The previously accumulated exploration experience is easy to be wasted. Due to the easy implementation and remarkable ability to avoid the local optimum, chaos theory has been applied to optimization. The seemingly disordered process of chaotic variables has inherent regularity, which employs the randomness, ergodicity, and regularity of chaotic variables to search for the global optimum. Chaos motion can traverse all states without repetition according to its own law within a certain range. Therefore, replacing \(r\) with a chaotic orbit \(c\) can effectively enhance the search efficiency and help electrons get rid of the local optimum. To achieve efficient exploration, electrons perform a chaotic local search in chaotic orbits to explore an irregular space. The entry process of chaotic orbits can be represented as follows.

**Step 1**: chaotic space mapping. Assume \(S_{s,i} (\lambda) = [s_{g,1}(\lambda), s_{g,2}(\lambda), \ldots, s_{g,j}(\lambda), \ldots, s_{g,n}(\lambda)]\) is the position vector of the \(i\)th electron in the chaotic orbit at the \(\lambda\)th iteration. Each dimension \(s_{g,j}(\lambda)\) in the \(S_{s,i} (\lambda)\) is mapped to the chaotic variable \(c_{g,j}(\lambda)\) (\(1 \geq c_{g,j}(\lambda) \geq 0\)), which is expressed as:

\[
c_{g,j}(\lambda) = (s_{g,j}(\lambda) - s_{g,j,max})/(s_{g,j,max} - s_{g,j,min}) \quad \forall j = 1, 2, \ldots, m \tag{21}
\]

where \(s_{g,j}(\lambda)\) is the \(j\)th decision variable of the \(\lambda\)th solution candidate at the \(\lambda\)th iteration; and \(s_{g,j,max}\) and \(s_{g,j,min}\) are the upper and lower limits of the \(j\)th decision variable, respectively.

**Step 2**: chaotic local search. An iterative logistic equation is adopted as:

\[
c_{g,j}(\lambda + 1) = \mu c_{g,j}(\lambda) (1 - c_{g,j}(\lambda)) \tag{22}
\]

where \(\mu\) is the control coefficient. When \(\mu\) is 4, the chaotic dynamics are characterized by iterative logistic equations. With enough iterations, the chaotic orbit of an electron can travel ergodically over the whole search space, which exhibits pseudo-randomness, ergodicity, and irregularity, respectively.

**Step 3**: solution space mapping. The chaotic variable \(c_{g,j}(k)\) can be mapped to the original solution space as:

\[
s_{g,j}(\lambda + 1) = (s_{g,j,max} - s_{g,j,min}) c_{g,j}(\lambda + 1) + s_{g,j,min} \tag{23}
\]

3) Dynamic Photons

This paper presents a dynamic photon strategy based on the optimization demand of different energy levels. By dynamically adjusting \(PR\), fine electrons tend to search locally, and weak electrons tend to perform large modifications such as chaotic orbits. Assume \(O_1(\lambda)\) and \(O_{\text{ave}}\) are the objective function values of the \(i\)th electron at the \(\lambda\)th iteration and the average objective function value of the \(i\)th electron, respectively.

1) When \(O_1(\lambda) > O_{\text{ave}}\), the studied electron is far from the electron position with the lowest energy level. The electron should perform large modifications to explore wider space for exploration. The photon rate \(PR\) becomes larger to lead the electron into the chaotic orbits, which is expressed as:

\[
PR = (PR_{max} - PR_{min})/2 + r_{\text{pr}} (PR_{max} - PR_{min})/2 \tag{24}
\]

where \(PR_{max}\) and \(PR_{min}\) are the maximum and minimum photon rates, respectively; and \(r_{\text{pr}}\) is a random coefficient in [0,1].

2) When \(O_1(\lambda) < O_{\text{ave}}\) the studied electron surrounds the electron with the lowest energy level. The electron should gradually search locally as the iteration proceeds for exploitation. The nonlinear decrease of the photon rate \(PR\) can be expressed as:

\[
PR = PR_{\lambda} = \lambda (PR_{max} - PR_{\lambda})/PR_{\lambda} + \lambda (PR_{\lambda} - PR_{min})/PR_{min} \tag{25}
\]

where \(\lambda_{max}\) is the maximum number of iterations. The improvement of CACS for AOS is mainly due to the adoption of collaborative behaviors, chaotic orbits, and dynamic photon strategy.

V. M2ASP Framework

A complete SEF contains predictors for multi-energy load prediction and a stacking method that fuses predictors. The proposed M2ASP contains two independent MASP models. The load prediction results of three base-predictors and a meta-predictor are adaptively combined in the first stage of M2ASP, and a prediction correction of the peak load consumption period is adopted in the second stage of M2ASP. The final prediction result of M2ASP is the sum of the multi-energy load prediction result calculated by the first MASP model and the prediction error calculated by the second MASP model. The schematic diagram of the MASP is provided in Fig. 4.
Fig. 4. Schematic diagram of MASP framework.
A. MASP Framework

MASP is a key submodel of M2ASP. The essence of MASP is to obtain preliminary prediction information through the base-predictors, dynamically generate new datasets based on the performance of each base-predictor, and integrate appropriate prediction knowledge to obtain more accurate prediction results than a single predictor.

1) Base-predictor Training

To avoid overfitting, the intermediate data sets of MASP are generated based on 5-fold cross-validation. First, the original training data is divided into five folds (training K), as shown in Fig. 4 (K=[1, 2, 3, 4, 5]). Then, RF, SVR, and LSTM are trained on the data of four folds to predict the remaining 1-fold training data (predict K (by model K)) and test set (predict K (base-predictor)). Each fold of the training set to be predicted corresponds to a separate predictor model (model K). Therefore, the new one-fold training data and new test set can be generated by base-predictors. For example, RF will repeat the above process five times and obtain a new 5-fold training set (predict 1-5) and five new test data (predict 1-V) calculated by the models 1-V. The process for other base-predictors is similar. After the base-predictors complete training, the new training set of a base-predictor is concatenated from the prediction results of the 5-fold training data set as:

$$\begin{align*}
\mathbf{TRS}_i = \begin{bmatrix}
TRP_1(l) & TRP_2(2) & TRP_3(3) & TRP_4(4) & TRP_5(5)
\end{bmatrix}^T
\end{align*}$$

(26)

where \(\mathbf{TRS}_i\) is the matrix containing new training set generated by the \(i\)th base-predictor; and \(TRP_i(K)\) is the new 1-fold training set generated by model K of the \(i\)th base-predictor.

The new test set of a base-predictor is obtained by weighting the five groups of predicted test data, which are defined as:

$$TES_i = \sum_{j=1}^{5} TEP_i(j) \cdot W_i(j)$$

(27)

where \(TES_i\) is the new test set generated by the \(i\)th base-predictor; \(TEP_i(j)\) is the \(j\)th new test set predicted by the \(i\)th base-predictor; and \(W_i(j)\) is the weight of the \(j\)th new test set predicted by the \(i\)th base-predictor. The weights of the new test set are adaptively adjusted considering the characteristics and performance of all base-predictors. As a result, three groups of new training and test sets incorporating the features of RF, SVR, and LSTM can be transmitted to meta-predictor to be systematically integrated.

2) Meta-predictor Prediction

The meta-predictor prediction is to summarize the knowledge provided by the base-predictors of base-predictor training and obtain the preliminary prediction results through a meta-predictor. First, the three groups of new training datasets obtained in base-predictor training are concatenated into an intermediate training set. Next, the original training labels and coalescent training set are brought into the AdaBoost method for training. Finally, the AdaBoost method predicts future multi-energy loads using the new test set obtained in base-predictor training.

3) Adaptive Stacking

The hyper-parameter setting is more complicated when the MASP is integrated with more predictors. Besides, the determination of the weight \(W_i(j)\) remains a controversial issue. As the performance and correlation of each base-predictor model are ignored, the stacking solely on the base-predictor number may result in unacceptable calculation speed and limited accuracy improvement. To achieve the adaptive stacking in different scenarios and improve the generalization ability, \(W_i(j)\) should be associated with the performance of different base-predictor models in the MASP. Therefore, the hyper-parameters of the four predictors and the 15 weights shown in Fig. 6 are selected as decision variables, prediction accuracy is selected as optimization objectives, and CACS is selected as the optimization method for adaptive stacking of MASP. The hyper-parameter difference of each independent base-predictor model by each run of CACS can improve the generalization ability of MASP.

B. Predictors of MASP

In ensemble learning, it is necessary to select the prediction methods with noticeable differences as the base-predictors. Based on case study results, three base-predictors and one meta-predictor with better stacking prediction performance are selected as follows: ① RF is an ensemble method consisting of decision trees in that each decision tree depends on the values of a random vector sampled independently [4], which is not prone to overfitting when predicting and adopted as one of the base-predictors of M2ASP; ② SVR is widely accepted as an efficient supervised learning algorithm in solving regression [21], which can support a quantitative response to the input variables and predict unknown information; ③ LSTM is a particular type of recurrent neural network that can learn long-term dependency information, suitable for predicting events with long intervals and delays in time series [9]; and ④ AdaBoost is a weighted weak classifier ensemble method based on boosting technology [22]. Since AdaBoost can reuse differently weighting combinations of the same training samples, the requirements for training data volume are acceptable. The powerful generalization ability makes AdaBoost serve as a meta-predictor of M2ASP, which is utilized for training the intermediate data set that gathers the knowledge of the three base-predictors mentioned above.

C. M2ASP

Affected by the irregular electricity consumption behaviors of users and the high penetration of DGs, the current prediction method remains challenging to eliminate the prediction error of the peak multi-energy loads. Due to the reinforced generalization ability of MASP, peak value correction based on the MASP is an effective way to handle this issue. Therefore, a two-stage MASP model named M2ASP is presented for load prediction and peak load correction. Figure 5 illustrates the implementation process of M2ASP. At the first stage of M2ASP, the multi-energy loads are preliminarily predicted by an independent MASP model. At the second stage of M2ASP, the peak load errors are effectively alleviated by another independent MASP model due to the strong generalization ability.
1) First Stage (First MASP Model)

M2ASP contains two independent MASP models, one for load prediction and the other for error correction. After the process of the preprocessing module based on ensemble learning, future multi-energy loads in RIES can be preliminarily predicted with the first MASP model. Although the prediction accuracy of the MASP is improved compared with a single predictor, the existing prediction errors are mainly concentrated around the peak of the multi-energy loads. Therefore, the preliminary prediction results and training errors of the first MASP model are transferred to the second stage.

2) Second Stage (Second MASP Model)

To improve the results of the first stage, the peak load errors are estimated in the second stage of M2ASP as follows.

Step 1: data preprocessing. Based on the preprocessing module in Section II, the training error obtained in the first stage and influencing factors are processed to meet the requirements of the following processes.

Step 2: variable dimension reduction. Based on the preprocessing module in Section II, the representative influencing factors on the prediction error of peak loads are selected using the RF-RFE method.

Step 3: error estimation. According to the load prediction (the first stage of M2ASP) results, the peak load range with a large prediction error can be identified. The data range of the second MASP model only includes peak load periods with large prediction errors. When training the second MASP model, the original training labels are replaced with preliminary training errors of the first MASP model. The prediction error of peak multi-energy loads on the test set can be estimated.

Step 4: error correction. The prediction error on the test set is fed back to the first MASP model to correct the prediction result. IMF sequences for each type of load are added linearly. The final prediction result of M2ASP is the sum of the multi-energy load prediction result calculated by the first MASP model and the prediction error calculated by the second MASP model.

D. Example

Combining the proposed preprocessing module based on ensemble learning, Fig. 6 illustrates the implementation process of the proposed framework. In this subsection, an example procedure of the proposed method is systematically illustrated, where the input data is processed by the preprocessing module and M2ASP to predict the multi-energy loads one hour later. Because multi-energy loads include electrical, heating, and cooling loads, three independent MASP models are constructed to predict the IMF sequences of electrical, heating, and cooling loads, respectively. To reduce peak load errors, three additional MASP models are constructed to estimate the prediction errors of IMF sequences. Therefore, a total of six MASP models are established. The prediction process of each model is similar, except that there are differences in the prediction objective and label value. The following is an example of an MASP model for preliminarily prediction heating loads.

1) Training Set and Test Set

Assume the input data of the raw training set is a 27×500 matrix, where 27 represents the dimension of the input variables, and 500 means that the input data contains 500 hours of sampling points. RF-RFE selects a suitable combination of input variables to achieve the dimensionality reduction of input variables. Assume that only eight input variables are retained.

Therefore, the dimension of the input data of the training set is 8×500. Assume the time dimension of the test set is 100 hours. In the same way, the test set is processed into an 8×100 matrix of input data.
The raw training labels are 500 hours of heating load samples (1×500 matrix). Then, the training labels (predicted objective) are decomposed into 11 IMFs by AVMD. SOM reorganizes 11 IMFs into 4 IMFs sequences. As a result, training labels become a 500×4 matrix.

2) Base-predictor Training

Taking RF as an example, 5-fold cross-validation establishes five sub-models (I-V), as shown in Fig. 6. Model I uses 400 hours of input data as the training set (400×8 matrix) to predict the remaining 100 hours of IMF sequences to generate a new training set (100×4 matrix). The new training sets generated by the five models are spliced into a 500×4 matrix. Three base-predictors will produce three new training sets, which are concatenated into a 500×12 matrix as the input data for the training set of the meta-predictors. Meanwhile, RF’s model I uses 400 hours of input data as the training set (400×8 matrix) to predict the IMF sequence of the test set to generate an intermediate test set (100×4 matrix). The prediction results of the five models (models I-V) on the test set are weighted to obtain a new test set (100×4 matrix). Three base-predictors will produce three new test sets, which are concatenated into a 100×12 matrix as the input data for the test set of the meta-predictors.

3) Meta-predictor Prediction

The training and prediction of the meta-predictor are based on a new training set (500×12 matrix) and a new test set (100×12 matrix) generated by the base-predictor. Training labels remain a 500×4 matrix. The meta-predictor is trained based on the new training set (500×12 matrix) and training labels (500×4 matrix). Next, the meta-predictor will make predictions on the new test set (100×12 matrix) to obtain the predicted IMF sequence (100×4 matrix). Each column of the matrix is added linearly to get prediction results of the heating loads for 100 hours (100×1 matrix).

4) Adaptive Stacking

The purpose of this step is to determine the best hyperparameters and intermediate weights (new test set weights) for the model, which is generally done before training. The hyperparameters of the four predictors and the 15 weights shown in Fig. 4 are selected as decision variables. The objec-
tive function is the average MAPE of the heat load prediction results. Based on the above process, CACS repeatedly calls MASP to predict the heat load in the validation set. The optimal parameter corresponds to the lowest average MAPE.

VI. CASE STUDIES

A. Dataset Description

The climate data are perceived from the China National Weather Station database. The distribution diagram of multi-energy loads of distribution networks in Binhai District, Tianjin, China, is shown in Fig. SA2 of Supplementary Material A. To achieve the adaptive stacking of M2ASP, the validating data of the predictor hyper-parameter setting and the intermediate dataset weights are from February, May, August, and November, 2017. The training data are from January, March, April, June, July, September, October, and December, 2017 for predictor training and peak error correction. The testing data are the representative days of the four quarters in 2017 that are not part of the validating and training sets for performance tests. The time resolution of the electrical, heating, and cooling loads is set to an hour, represented by time $t$. The range limits of $W_t(j)$ are $[0,1]$, which are normalized to ensure that the sum of the weights of each base-predictor is 1. In other words, a day can be divided into 24 intervals represented by time $t$. The peak load range is selected by the M2ASP in the first stage. The objective of the proposed framework is to predict the multi-energy loads one hour later. In addition to historical multi-energy loads, exogenous and meteorological variables are considered in predictions such as $T$, $H$, $WS$, $SR$, $HD$, $DW$, $MY$, $WM$, and day type. One-hot encoding is used to quantify calendar information.

B. Performance Evaluation Indicators

Each evaluation indicator has different emphases and limitations. Therefore, it is necessary to carry out comparative experiments on the proposed M2ASP using various evaluation indicators. Three evaluation indicators selected from the machine learning field that can overall reflect the performance of the prediction methods are as follows.

1) MAPE is adopted to compare the performance of the proposed framework and benchmark prediction models. MAPE normalizes the prediction error for each measurement and weakens the interference of individual outliers on the absolute error, which is expressed as:

$$ MAPE = \frac{1}{M_t} \sum_{i=1}^{M_t} \left| \frac{L_{act} - L_{pre}}{L_{act}} \right| $$

(28)

where $M_t$ is the number of observation points; and $L_{act}$ and $L_{pre}$ are the predicted load and actual load, respectively.

2) Root mean square error (RMSE) utilizes an average prediction error that is sensitive to abnormal values. The value of RMSE is more sensitive to outliers than MAPE, which can evaluate the overall stability of the prediction method. RMSE is expressed as:

$$ RMSE = \sqrt{\frac{1}{M_t} \sum_{i=1}^{M_t} (L_{act} - L_{pre})^2} $$

(29)

3) Although the values of RMSE and MAPE are sufficient to evaluate the prediction accuracy, they cannot evaluate how well the predicted values fit the actual values. R-square ($R^2$) is the coefficient of determination ($R^2 \in [0, 1]$), which quantifies the fitting degree of the prediction model and can be expressed as:

$$ R^2 = 1 - \frac{\sum_{i=1}^{M_t} (L_{act} - L_{pre})^2}{\sum_{i=1}^{M_t} (L_{act} - \bar{L}_{act})^2} $$

(30)

C. Data Decomposition and Reconstruction

In the preprocessing module based on ensemble learning, the original time-series of multi-energy load can be decomposed into multiple IMFs and a residue component mode by AVMD. The numbers of IMFs for electrical, heating, and cooling loads are 6, 13, and 6, respectively. A schematic diagram of IMFs for multi-energy loads decomposed by AVMD is shown in Fig. 7. Therefore, irregular time series with significant fluctuations can be decomposed into IMFs to obtain vital regularization information. To improve the computation speed and avoid excessive-decomposed, the SOM clustering method is adopted to fuse approximate IMFs into IMF sequences. As a result, the sequence numbers of electrical, heating, and cooling loads are 4, 5, and 4, respectively. The output variables of the prediction model are replaced by the IMF sequences of the RIES loads.

D. Ablation Analysis

Because multi-energy loads include electrical, heating, and cooling loads, three independent MASP models are constructed to predict the IMF fusions of electrical, heating, and cooling loads, respectively. To reduce peak load errors, three additional MASP models are constructed to estimate the prediction errors of IMF fusion. Therefore, a total of six MASP models are established, which form a total of three M2ASP models to predict multi-energy loads. To determine six groups of highly correlated input variables of different MASP models, RF-RFE is used to analyze the importance of the above 27 influencing factors and select the optimal input variable combinations. Through the iteration of RF-RFE, the relationship between the number of retained input variables and the RMSE of the prediction results is shown in Fig. 8. When estimating the electrical, heating, and cooling load prediction errors, the estimation accuracy is no longer significantly improved after the numbers of input variables exceed 12, 10, and 11, respectively. Therefore, in order of variable importance, the optimal input variable combination of the MASP models can be determined in Table SAI of Supplementary Material A.

E. Sensitivity Analysis

To further study the impact of input variable changes on multi-energy load prediction, the determined input variables are subjected to sensitivity analysis. Since $WS$ and $SR$ are not considered in the input variables, this paper only analyzes the environmental characteristics of $H$ and $T$. The values of the two variables vary by $[-15\%, 15\%]$. When conducting the sensitivity analysis, only a single input variable can be changed, and other input variables remain unchanged.
Compared with H, the proposed framework is more sensitive to T. When the offset of T is 15%, the RMSEs of electrical, heating, and cooling loads increase by 8.41%, 12.48%, and 12.29%, respectively, compared with the initial prediction value after the change. When the offset of T is −15%, the RMSEs of electrical, heating, and cooling loads increase by 6.46%, 14.48%, and 7.93%, respectively, compared with the initial prediction value after the change. Compared with the other two loads, environmental factors have less influence on electricity load. Therefore, the precision of dry bulb temperature should be strictly required in subsequent studies. Otherwise, excessive environmental data errors will lead to poor accuracy of RIES load prediction.

Fig. 7. Schematic diagram of IMFs for multi-energy loads decomposed by AVMD. (a) IMFs of electrical load. (b) IMFs of heating load. (c) IMFs of cooling load.

Thereby, the influence of environmental factors on RIES load prediction can be analyzed. The MAPE of the prediction results under the change of environmental variables is shown in Fig. 9.

Fig. 8. RF-RFE results of prediction results prediction framework.

Fig. 9. Sensitivity analysis results.

F. Optimization Method Comparison

1) Hyper-parameter Setting

The predictor hyper-parameters and intermediate dataset weights of M2ASP are optimized by CACS. In this subsection, the hyperparameter configuration of CACS is discussed. The hyperparameters of CACS include the maximum number of iterations, \( PR \), \( q_{L} \), and \( q_{F} \). The range of photon rate \( PR \) is set to be \([0, 0.3]\) to limit the photon dynamics. After more than 100 iterations, the iterative curve of CACS generally converges. To ensure that the optimizer can con-
To study the difference between AOS and CACS, the total number of electrons and iterations of AOS and CACS should be consistent. CACS in M2ASP is replaced by AOS to compare the performance of the two optimizers in adaptive stacking. The average MAPEs of the multi-energy load prediction in different months are shown in Table I. Although CACS has a 10% increase in computation time compared with AOS, CACS has an average error lower than AOS in most months. Besides, the Wilcoxon signed-rank test [23] is employed to verify the difference between CACS and AOS in Table I. “+” indicates that the hypothesis is wrong (there is a clear difference between the two algorithms), and CACS shows better performance than AOS. “−” indicates that the hypothesis is wrong (there is a clear difference between the two algorithms), and CACS shows worse performance than AOS. The performance of CACS in M2ASP is stronger than AOS in other months. From the p-values and winner indexes of each month, it can be observed that CACS differs from AOS and the improvement strategies of CACS are effective.

### Table I

**Performance Comparison Between CACS and AOS**

<table>
<thead>
<tr>
<th>Month</th>
<th>MAPE</th>
<th>Time (s)</th>
<th>Wilcoxon signed-rank</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CACS</td>
<td>AOS</td>
<td>CACS</td>
</tr>
<tr>
<td>January</td>
<td>0.0463</td>
<td>0.0451</td>
<td>436</td>
</tr>
<tr>
<td>March</td>
<td>0.0392</td>
<td>0.0417</td>
<td>451</td>
</tr>
<tr>
<td>April</td>
<td>0.0431</td>
<td>0.0437</td>
<td>421</td>
</tr>
<tr>
<td>June</td>
<td>0.0490</td>
<td>0.0513</td>
<td>461</td>
</tr>
<tr>
<td>July</td>
<td>0.0538</td>
<td>0.0559</td>
<td>451</td>
</tr>
<tr>
<td>September</td>
<td>0.0472</td>
<td>0.0482</td>
<td>422</td>
</tr>
<tr>
<td>October</td>
<td>0.0439</td>
<td>0.0425</td>
<td>411</td>
</tr>
<tr>
<td>December</td>
<td>0.0581</td>
<td>0.0459</td>
<td>419</td>
</tr>
</tbody>
</table>

2) **Optimizer Comparison**

To demonstrate the superiority of the proposed CACS method, TSA [24], BCA [25], IPalloSO-COA [6], HBA [12], SDSA [26], and AOS are introduced into M2ASP for performance comparison. To ensure a fair comparison, the population size and the maximum number of iterations for each optimizer are set to be 60 and 100, respectively. The hyperparameters of the remaining benchmark optimizers are random sampling to determine a suitable value. The task of adaptive stacking is undertaken by the above benchmark optimizers to predict the electricity loads on January 15, 2017. MAPE is chosen as the fitness function of the optimizer. When optimizing predictor hyper-parameters and intermediate dataset weights of M2ASP, the performance of different optimizers is shown in Table II. Except for $R^2$, the other evaluation indicators are the smaller the better.

### Table II

**Performance of Different Optimizers**

<table>
<thead>
<tr>
<th>Optimizer</th>
<th>MAPE</th>
<th>RMSE</th>
<th>$R^2$</th>
<th>Time (s)</th>
<th>Average rank</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CACS</td>
<td>0.0471</td>
<td>0.1103</td>
<td>0.9732</td>
<td>437</td>
<td>2.00</td>
<td>5.13×10⁻²</td>
</tr>
<tr>
<td>IPalloSO-COA</td>
<td>0.0486</td>
<td>0.1093</td>
<td>0.9692</td>
<td>592</td>
<td>3.00</td>
<td>7.34×10⁻⁴</td>
</tr>
<tr>
<td>AOS</td>
<td>0.0526</td>
<td>0.1123</td>
<td>0.9513</td>
<td>401</td>
<td>3.25</td>
<td>1.21×10⁻⁵</td>
</tr>
<tr>
<td>HBA</td>
<td>0.0584</td>
<td>0.1212</td>
<td>0.9501</td>
<td>467</td>
<td>5.00</td>
<td>5.47×10⁻⁵</td>
</tr>
<tr>
<td>SDSA</td>
<td>0.0763</td>
<td>0.1321</td>
<td>0.9402</td>
<td>398</td>
<td>5.00</td>
<td>9.53×10⁻⁴</td>
</tr>
<tr>
<td>BCA</td>
<td>0.0541</td>
<td>0.1201</td>
<td>0.9689</td>
<td>589</td>
<td>4.25</td>
<td></td>
</tr>
<tr>
<td>TSA</td>
<td>0.0804</td>
<td>0.1367</td>
<td>0.9308</td>
<td>359</td>
<td>5.50</td>
<td></td>
</tr>
</tbody>
</table>

As the heuristic optimization algorithm is only a tool for M2ASP to achieve adaptive stacking, this paper mainly focuses on whether CACS adapts to M2ASP, and the performance of CACS in other optimization domains will not be discussed. If CACS-based M2ASP has better accuracy than other benchmark predictors and CACS outperforms other benchmark optimizers in M2ASP, CACS will be employed as the adaptive stacking method of M2ASP. As shown in Table II and Fig. 11, CACS performs well across all evaluation indicators, ranking first in MAPE and $R^2$. With an acceptable computation time of 437 s, CACS achieves a better average rank than the other six optimizers. Besides, the experimental results demonstrate that collaborative behaviors, chaotic orbits, and dynamic photons only increase the computation time by 4.5%. Compared with AOS, CACS reduces MAPE and RMSE by 10.46% and 1.78%, and improves $R^2$ by 2.3%. Therefore, CACS is an effective tool for M2ASP to achieve dynamic stacking, which contributes to the improvement of M2ASP in accuracy and degree of fitting. To demonstrate proper characterization of CACS, the results of the Wilcoxon signed-rank test [27] are shown in Table II. In the process of repeated experiments, the optimization performance of CACS is more stable than other optimizers, and there are fewer cases of non-convergence. The p-values are based on the hypothesis of the benchmark optimizer and CACS. Obviously, there is a difference in the optimization performance between CACS and other benchmark optimizers.
G. Method Comparison

To verify the performance of the proposed framework, eleven prediction models that perform well in the field of machine learning, LightGBM [28], GBR [29], XGBoost [30], GBRT [31], KNN [32], RF [33], SVR [34], LSTM [35], AdaBoost [22], light-stacking framework (LSF) [7], MEFF [8], and M2ASP, are compared. Both LSF and MEFF are state-of-the-art stacked generalization methods, which are introduced to justify the proposed framework. According to the example of Section V-D, M2ASP processes raw data through the preprocessing module based on ensemble learning, and then predicts the multi-energy load through two stages. Except for M2ASP, other prediction methods directly forecast the raw multi-energy load data. All methods keep the same input variables in predicting electricity, heating, and cooling loads, which are set to be 10 input variables according to the results of ablation analysis. The prediction target of M2ASP is the IMF sequence of multi-energy load, and the prediction target of other methods is multi-energy load. Besides, this paper proposes an efficient CACS as an optimizer for M2ASP, which outperforms other benchmark predictors in M2ASP and strengthens the generalization performance ability. Compared with the existing SEF methods, the structure of M2ASP has been dynamically changed, and the stacking method of each base-predictor is changed to adaptive stacking. With strong generalization ability, M2ASP can solve the drawback of large prediction error during peak electricity consumption in a targeted manner.

1) Performance Analysis in Different Seasons

Considering the strong seasonal and temperature sensitivity of multi-energy loads, the load characteristics are different in different seasons. Representative daily load data are extracted to verify the feasibility of the proposed model. Besides, the prediction effects of benchmark models are compared and analyzed. The comparison chart of prediction results in different seasons is shown in Fig. 12. Obviously, the regularity of the cooling load is less than that of the heating load and the electricity load, which leads to a decrease in the prediction accuracy of the cooling load compared with the other two types of loads. Among all benchmark models, M2ASP has better goodness of fit and a more stable prediction accuracy. With the unique stacking design and strong generalization capability, less non-convergence occurs in M2ASP. Although other prediction methods may exceed the prediction accuracy of M2ASP on a single hour, the comprehensive prediction performance of the four quarters remains worse than that of M2ASP. Especially for the cooling load in October and the heating load in July, benchmark methods have large prediction errors at individual points, which will seriously affect the decision-making effects of RIES managers.

To compare the difference between single predictor and ensemble framework, the average performance of benchmark models in four quarters is compared in Table III. For the single predictor, the MAPEs of RF, SVR, LSTM, and AdaBoost are stronger than other single predictors, which indicates that the base-predictors and meta-predictor of M2ASP are credible. Among them, RF has the highest prediction accuracy, which is 16.92% and 7.84% lower than KNN in MAPE and RMSE, respectively. With the integration of base-predictors, the ensemble framework (M2ASP, LSF, MEFF) outperforms a single predictor performance. Since the ensemble framework requires cross-validation to prevent overfitting, there is a longer time lag when training the model compared with a single prediction model. Due to its lightweight structure and less integrated predictors, the computation time of M2ASP is reduced by 10.25% and 27.36% compared with LSF and MEFF, respectively. In terms of MAPE, RMSE, and $R^2$, the performance of M2ASP is better than other benchmark models.

M2ASP without peak error correction is introduced into the comparison. As shown in Table III, the peak error correction helps M2ASP reduce MAPE and RMSE by 6.52% and 7.6%, respectively, and improves $R^2$ by 0.83%. The substantial reduction in RMSE demonstrates that peak error correction can effectively reduce the chance of significant errors and improve the stability. Even without peak error correction, M2ASP still outperforms other benchmark prediction models. The $p$-values of the Wilcoxon signed-rank test are used to verify the difference between M2ASP and the original technology. The $p$-values in Table III are based on the hypothesis of M2ASP versus other prediction methods. All $p$-values are less than $6.41 \times 10^{-4}$. Therefore, M2ASP differs from other prediction methods in prediction performance. Besides, the improvement in prediction accuracy of M2ASP benefits from the combination of the suitable heuristic algorithm and the adaptive stacking structure. The stacking structure provides a general platform for the fusion of predictors. CACS guarantees the full utilization of the advantages of stacked predictors.

2) Special Day Analysis

Special days refer to days with low frequency and irregular electricity consumption. To test the generalization ability and application value of M2ASP, this paper predicts the multi-energy loads on the Chinese New Year, which is shown in Fig. 13. There are fewer similar data for abnormal days in a year than normal days, which dramatically tests the generalization ability of the prediction method. M2ASP is still more accurate than other benchmark models. Due to the strong generalization ability, the accuracy drop of M2ASP is also smaller than other benchmark models.
TABLE V

<table>
<thead>
<tr>
<th>Model</th>
<th>MAPE</th>
<th>RMSE</th>
<th>$R^2$</th>
<th>Time (s)</th>
<th>$p$-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RF</td>
<td>0.054</td>
<td>0.141</td>
<td>0.957</td>
<td>28</td>
<td>3.66×10^{-6}</td>
</tr>
<tr>
<td>SVR</td>
<td>0.063</td>
<td>0.147</td>
<td>0.951</td>
<td>25</td>
<td>4.67×10^{-6}</td>
</tr>
<tr>
<td>LSTM</td>
<td>0.058</td>
<td>0.146</td>
<td>0.954</td>
<td>26</td>
<td>5.31×10^{-6}</td>
</tr>
<tr>
<td>AdaBoost</td>
<td>0.055</td>
<td>0.143</td>
<td>0.951</td>
<td>32</td>
<td>8.32×10^{-8}</td>
</tr>
<tr>
<td>LightGBM</td>
<td>0.059</td>
<td>0.152</td>
<td>0.952</td>
<td>24</td>
<td>7.12×10^{-6}</td>
</tr>
<tr>
<td>GBRT</td>
<td>0.061</td>
<td>0.158</td>
<td>0.955</td>
<td>26</td>
<td>3.64×10^{-6}</td>
</tr>
<tr>
<td>XGBoost</td>
<td>0.063</td>
<td>0.149</td>
<td>0.949</td>
<td>30</td>
<td>9.31×10^{-6}</td>
</tr>
<tr>
<td>GBRT</td>
<td>0.061</td>
<td>0.148</td>
<td>0.953</td>
<td>26</td>
<td>4.12×10^{-7}</td>
</tr>
<tr>
<td>KNN</td>
<td>0.065</td>
<td>0.153</td>
<td>0.947</td>
<td>23</td>
<td>1.35×10^{-7}</td>
</tr>
<tr>
<td>M2ASP</td>
<td>0.043</td>
<td>0.121</td>
<td>0.977</td>
<td>429</td>
<td></td>
</tr>
<tr>
<td>M2ASP without peak error correction</td>
<td>0.046</td>
<td>0.131</td>
<td>0.969</td>
<td>306</td>
<td></td>
</tr>
<tr>
<td>LSF</td>
<td>0.049</td>
<td>0.137</td>
<td>0.961</td>
<td>478</td>
<td>6.41×10^{-4}</td>
</tr>
<tr>
<td>MEFF</td>
<td>0.047</td>
<td>0.133</td>
<td>0.965</td>
<td>591</td>
<td>4.51×10^{-4}</td>
</tr>
</tbody>
</table>

Fig. 12. Comparison chart of prediction results in different seasons. (a) Electrical load in January. (b) Electrical load in April. (c) Electrical load in July. (d) Electrical load in October. (e) Heating load in January. (f) Heating load in April. (g) Heating load in July. (h) Heating load in October. (i) Cooling load in January. (j) Cooling load in April. (k) Cooling load in July. (l) Cooling load in October.

Fig. 13. MAPE of multi-energy load on Chinese New Year.

VII. CONCLUSION

In this paper, an M2ASP is proposed to predict multi-energy loads in the RIES, which achieves an adaptive combina-
tion of four different predictors. Predictor hyper-parameters and intermediate data sets of M2ASP are trained with a new optimization method named CACS to improve prediction performance. Inputs are processed with a preprocessing module based on ensemble learning, using multiple preprocessing technologies. In the first stage of M2ASP, the outputs of base-predictors are adaptively combined to predict the multi-energy loads preliminarily. The second stage of M2ASP conducts a prediction correction in the peak load consumption period. The final prediction result is the sum of the IMF sequences of various loads. The case studies demonstrate that the proposed framework has better prediction accuracy, generalization ability, and stability than other benchmark prediction models.

REFERENCES


Leijiao Ge received the Ph.D. degree in electrical engineering from Tianjin University, Tianjin, China, in 2016. He is currently a Professor in the School of Electrical and Information Engineering at Tianjin University. His main research interests include smart distribution networks, cloud computing, and big data.
Yuanliang Li received the B.E. degree in electrical engineering and automation from China University of Petroleum, Qingdao, China, in 2016. Currently, he is pursuing the M.S. degree in electrical engineering from Tianjin University, Tianjin, China. His main research interests include multi-energy load forecasting and smart distribution networks.

Jun Yan received the B.Eng. degree in information and communication engineering from Zhejiang University, Hangzhou, China, in 2011, and the M.Sc. and Ph.D. (with excellence in doctoral research) degrees in electrical engineering from the University of Rhode Island, Kingston, USA, in 2013 and 2017, respectively. Currently, he is an Assistant Professor at the Concordia Institute for Information Systems Engineering, Concordia University, Montreal, Canada. He was a Recipient of the IEEE ICC Best Paper Award (2014) and the IEEE WCCI Best Student Paper Award (2016). His research interests include computational intelligence and cyber-physical security with applications in smart grids, smart cities, and other smart critical infrastructures.

Yuanliang Li received the M.S. degree in mechanical engineering from Hohai University, Nanjing, China, in 2019. He is currently pursuing the Ph.D. degree in information systems engineering with Concordia University, Montreal, Canada. His main research interests include smart grid security and reinforcement learning.

Jiaan Zhang is currently a Lecturer in the School of Electrical Engineering at Hebei University of Technology, Tianjin, China. His main research interests include new energy power generation and intelligent system, power system modeling and simulation.

Xiaohui Li is currently a Senior Engineer in the Marketing Service Center at State Grid Tianjin Electric Power Company, Tianjin, China. His main research interests include smart distribution network and electric power measurement.