This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2025.3581958

CWGWO-N-BEATSx: An improved time series prediction method with multiple external variables for situation prediction

Rui Huang, Chundong Wang, Hao Lin, Haolong Zhang, Hongjing Ma

***Abstract*—Situation prediction, as a critical component of the situation awareness framework, plays a vital role in decision-making systems such as traffic management and cybersecurity. However, most existing studies still focus primarily on univariate time series, lacking effective modeling and optimization mechanisms for multivariate situation data with exogenous variables.In view of the above research status, this paper proposes a novel prediction method called Improved Grey Wolf Optimizer - Neural Basis Expanded Analysis of Time Series with Exogenous Variables (CWGWO-N-BEATSx),specifically designed for time series situation prediction tasks involving multivariate exogenous variables. The proposed approach integrates the powerful modeling capability of N-BEATSx with an improved grey wolf optimization algorithm (CWGWO), aiming to achieve joint optimization of model architecture, critical hyperparameters, and exogenous variable selection. Within the CWGWO-N-BEATSx framework, N-BEATSx is, for the first time, applied to situation prediction. Key hyperparameters influencing the performance of N-BEATSx are identified and quantified, and a CWGWO algorithm — enhanced through the incorporation of chaotic mapping and adaptive weighting mechanisms—is introduced to optimize these hyperparameters. A multi-objective fitness function is constructed by jointly considering prediction accuracy and model complexity, and extensive empirical evaluations are conducted on real-world situation and time series datasets.Experimental results demonstrate that CWGWO outperforms eight mainstream metaheuristic algorithms. Compared with five state-of-the-art (SOTA) methods, CWGWO-N-BEATSx reduces the average MAE on situation datasets and time series datasets by 17.614% and 17.55%, respectively; the average SMAPE by 16.27% and 8.932%; and the MSE by 23.488% and 43.87%, respectively. In addition, CWGWO-N-BEATSx maintains relatively low model complexity, validating its superior performance and strong potential for practical application.**

***Index Terms*—Deep learning, multivariate time series, Time series analysis, hyperparameter optimization, predictive models**

(*Corresponding author:Chundong Wang*)

The authors are with the School of Computer Science and Engineering, Tianjin University of Technology, Tianjin 300384, China (e-mail: hrrr9898@163.com, michael3769@163.com, suzukaze_aoba@126.com, haolong0910@163.com, m161611666@163.com)

I. INTRODUCTION

SITUATION awareness refers to the comprehensive understanding of an individual or system of its surroundings, including the current state, future trends, and possible impacts. The concept was initially applied in military and aviation fields, but nowadays it is widely used in many fields such as cybersecurity, traffic management, healthcare, industrial automation, and so on.

As illustrated in Figure 1, situation awareness typically comprises three hierarchical levels: perception, comprehension, and prediction. Situation prediction represents the third level, which builds upon the perception and understanding of the environment or system state to forecast future situations using computational models and methods. The accuracy of situation prediction directly influences the scientific validity and operational efficiency of decision-making processes, making it a critical component in modern intelligent sensing systems and a prominent focus of recent research.

In practical applications, situation prediction problems are often modeled as time series forecasting tasks. Although recent studies have made notable progress in this field, most efforts remain limited to univariate time series modeling, with insufficient capability to predict complex situation data involving multivariate exogenous variables. This limitation hinders the ability to meet real-world demands for handling high-dimensional heterogeneous information.

To address this problem, some studies have attempted to introduce deep learning models capable of handling multivariate exogenous inputs. Among them, N-BEATSx[1], a fully connected deep neural network architecture with a dual-level residual connection structure, has attracted considerable attention due to its strong predictive performance, simple architecture, and flexibility. Notably, the N-BEATSx model has demonstrated exceptional performance across various forecasting domains beyond situation prediction[2][3][4][5][6]. However, directly applying N-BEATSx to complex situation prediction tasks still faces numerous challenges, particularly regarding the coupling between hyperparameter optimization and exogenous variable modeling.

Firstly, the predictive performance of N-BEATSx is highly dependent on several structural hyperparameters (e.g., stack types, the number of exogenous channels (ENC), block depth, etc.); however, there is still a lack of systematic research on their configuration. Secondly, although existing hyperparameter optimization strategies — such as grid search[7] , Bayesian optimization[8], and basic meta-heuristic algorithms[9] — exhibit certain effectiveness, they generally suffer from limitations such as narrow search space, low efficiency, and a tendency to fall into local optima, making them inadequate for optimizing the high-dimensional and complex hyperparameter space of the N-BEATSx model.

More importantly, the selection of exogenous variables not only influences the input dimensionality and forecasting capability of the model but also directly affects several associated hyperparameters. Nevertheless, current research typically treats these two components in isolation, lacking a unified optimization mechanism, which constrains further improvements in model performance.

In summary, current situation prediction methods still exhibit significant modeling limitations and optimization bottlenecks when confronted with multivariate and highly complex data scenarios. Therefore, there is an urgent need to explore a method that jointly optimizes the selection of exogenous variables and the configuration of key hyperparameters tailored to the structural characteristics of the N-BEATSx model, in order to further enhance its performance and adaptability in complex situation prediction tasks.

Therefore, the motivation for the research in this paper is as follows:

- As a cutting-edge prediction model, N-BEATSx lacks empirical studies regarding its modeling capabilities on complex situation prediction data, and it remains unclear whether its structural characteristics are suitable for forecasting environments characterized by non-stationarity and multivariate interference.
- The predictive performance of N-BEATSx is influenced by multiple hyperparameters; however, existing studies predominantly rely on empirical settings and lack systematic exploration of the key hyperparameters and their optimal combinations for situation prediction.
- The incorporation of exogenous variables is a key modeling approach for N-BEATSx in situation prediction, and the selection of these variables significantly influences the optimal configuration of the model ' s hyperparameters, necessitating their joint optimization alongside the model hyperparameters.
- This study urgently requires the development of an efficient, robust, and generalizable optimization strategy to explore the potential of N-BEATSx, enhance its performance in situation prediction tasks, and address the shortcomings of existing methods in automated modeling and joint optimization capabilities.

This study introduces an enhanced heuristic algorithm, termed CWGWO, designed to optimize the hyperparameters of N-BEATSx, thereby improving its structural configuration and overall performance. CWGWO stands for Chaotic Mapping and Weight Improvement Grey Wolf Optimization Algorithm, and aims to optimize the initial population distribution and improve the searching capability. In addition, we propose CWGWO-N-BEATSx to predict the future of the situation by combining the two. The main contributions of our work are summarized below:

- This study is the first to apply N-BEATSx to situation prediction tasks, exploring its modeling potential in this domain.
- Key hyperparameters affecting the performance of N-BEATSx were identified and quantified, and an improved heuristic algorithm was introduced for the first time to enable automatic hyperparameter optimization.
- CWGWO integrating a weight adjustment mechanism was proposed to improve population initialization quality and global search capability. Meanwhile, a multi-objective fitness function balancing prediction accuracy and model complexity was designed specifically for hyperparameter tuning of N-BEATSx.
- The CWGWO-N-BEATSx framework was constructed and empirically validated on situation prediction and time series datasets. To enable effective evaluation, an extended situation dataset incorporating multivariate exogenous variables was developed based on a univariate situation dataset.

The subsequent structure of this paper is as follows: Chapter 2 provides a detailed analysis of recent research on situation prediction, the N-BEATSx framework, and its optimization. Chapter 3 introduces the situation prediction task and presents the architecture of N-BEATSx. Chapter 4 outlines the proposed CWGWO-N-BEATSx model and its core component — the CWGWO algorithm — offering an in-depth description of the hyperparameters to be optimized and the performance characteristics of CWGWO. Chapter 5 presents the experimental details, including ablation studies and comparative evaluations against state-of-the-art models, followed by a thorough analysis of the results. Finally, the concluding chapter discusses the limitations of the proposed model and offers recommendations for future research in the field of situation prediction.

II. Related work

A. Situation prediction

Situation prediction, as the last stage of situation awareness, ensures its accuracy in order to make the right decisions and treatments for subsequent behaviors.To this end, we conduct research on situation prediction methods from two perspectives: traditional approaches and modern approaches.

1) Traditional Methods

In situation prediction research, traditional methods have long held a dominant position. These mainly include Bayesian Networks, Markov Models, Autoregressive Integrated Moving Average (ARIMA) models, and Vector Autoregression (VAR) models. These methods offer advantages such as simplicity in modeling, well-established theoretical foundations, and strong interpretability, making them particularly suitable for stationary time series with relatively clear patterns of change. Specifically, Bayesian Networks represent conditional probabilistic dependencies among variables by constructing directed acyclic graphs, enabling the integration of prior knowledge and uncertainty reasoning. Markov Models, based on the assumption that the current state of a system depends only on the previous state, use state transition probabilities to effectively model short-term dependencies in time series. ARIMA is a linear time series analysis method adept at handling univariate data with trends or seasonal components. VAR is its multivariate extension, suitable for modeling interactions among multiple related time series and capturing dynamic influences between variables within a system.

Although these traditional methods provide theoretical rigor and a certain level of interpretability, their performance is significantly constrained when dealing with highly nonlinear, noise-intensive, or high-dimensional real-world scenarios. Therefore, enhancing the adaptability of prediction models to non-stationary and complex dynamic environments has become a critical challenge in current situation awareness and prediction research.

2) Modern Methods

Modern methods primarily encompass hybrid architecture approaches and neural network approaches in situation prediction. The hybrid architecture approach is one in which the model incorporates other methods in addition to the use of neural network structures as a means of improving prediction accuracy. Examples include Autoregressive Integral Moving Average (ARIMA) methods, Vector Autoregressive (VAR) methods, Sequence to Sequence (Seq2Seq) methods, and so on ; the neural network approach refers to the use of only neural network structures led by recurrent neural networks (RNN) in the model and the use of some machine learning algorithms to optimize the hyperparameters in the model. Table 1 lists the relevant literature for the study.

Hybrid architecture methods integrate diverse types of neural networks into a single model to leverage the strengths of each approach and enhance overall performance[10]. For instance, Qiu et al.[11] combined grid-based and graph-based traffic flow prediction methods, utilizing Topological Graph Convolutional Networks (ToGCN) and a Sequence-to-Sequence (Seq2Seq) framework to forecast time-dependent future traffic flow and density. Similarly, Moishin et al.[12] integrated Convolutional Neural Networks (CNN) with Long Short-Term Memory networks (LSTM) to capture hydrological influences through a mathematically represented Flood Index (IF), predicting future flood indices using historical and real-time rainfall data. Jin et al.[14] introduced a novel ARIMA-LSTM hybrid method, parallelized via regression coefficient weighting, and applied it to COVID-19 transmission forecasting. Yuan et al.[15] employed CNN to extract embedded spatial features from traffic condition data and utilized a dual-layer GRU to learn hidden information affecting traffic dynamics, thereby predicting terminal area traffic conditions under weather influences using a CNN-GRU model. Zhao et al.[17] proposed a new method combining enhanced Bidirectional LSTM, Vector Autoregression, and multi-scale optimization. In their Attention-based LSTM Network for Cybersecurity Situation Prediction (ALSNAP) scheme, they integrated Vector Autoregression and multi-scale convolution with branch attention into an intelligent cybersecurity situational awareness system. Zhang et al.[20] developed a Situation Awareness Multi-Graph Convolutional Recurrent Network (SA-MGCRN) model for real-time prediction of travel demand during wildfire evacuations, employing Graph Convolutional Networks (GCN) to capture spatial dependencies and Gated Recurrent Units (GRU) for temporal forecasting. While hybrid architectures can amalgamate the advantages of different methods, their complex structures and lack of interpretability make them less ideal for situation prediction tasks.

The neural network structure led by RNN as the main method for situation prediction, combined with the use of machine learning algorithms to optimize the hyperparameters in the model, can significantly improve the model prediction performance. Chen[13] et al. in order to explore the effectiveness of machine learning algorithms in situation prediction, radial basis function (RBF) neural networks were used as the main object of research, and the RBF was optimized by the simulated annealing (SA) algorithm and the hybrid hierarchical genetic algorithm (HHGA), to construct a prediction model of RBF neural network based on SA-HHGA optimization. Sun[16] et al. optimized the ELM using an improved SSA algorithm in order to ensure high prediction accuracy of the ELM with the optimal number of nodes in the hidden layer. Xi et al.[18] addressed the deficiencies in air combat situation assessment. By integrating a multiple regression model with an implicit logical process, a weight optimization model based on the grey prospect theory, a weight mapping model (AE-ELM) based on autoencoders and extreme learning machines, and a prediction model for air combat situation characteristic parameters based on dynamic weights, they proposed a dynamic air combat situation assessment model online extreme learning machine (DWOSELM) based on situation knowledge extraction and weight optimization. In order to improve the prediction accuracy, Wu[19] et al. proposed a model fusing convolutional neural network (CNN) and bi-directional long and short-term memory network (ABiLSTM) and used particle swarm optimization (PSO) algorithm to optimize its hyperparameters. Although the neural network approach has a simple structure and excellent prediction results, there is a lack of research on the relationship between target and exogenous variables.

B. N-BEATSx for time series prediction

N-BEATSx, an advanced forecasting model, has exhibited exceptional nonlinear mapping capabilities across a wide range of time series prediction tasks, including electricity price forecasting, stock volatility, runoff forecasting, agricultural product price forecasting, hydrogen consumption forecasting, etc. N-BEATSx is highly interpretable, adaptable to the target domain without modification, fused with multiple sources of data, and fast to train. BEATSx authors Olivares[1] et al. demonstrated that N-BEATSx can efficiently solve the problem of electricity price forecasting for a wide range of years and markets by including external variables and integrating multiple sources of useful information. Souto[21] et al. demonstrated the effectiveness of N-BEATSx in predicting electricity prices for a wide range of years and markets by using N-BEATSx in conjunction with several other commonly used forecasting models, i.e., LSTM, TCN, HAR, GARCH, and GJR-GARCH models), together to forecast daily stock realized volatility at different time steps and compare the forecasts to conclude that N-BEATSx consistently produces statistically more accurate and robust forecasts than the other models considered. However, it should be noted that the advantage of N-BEATSx in terms of forecasting accuracy is not significant when applied to developing countries' stock indices. To improve the accuracy of soybean futures price forecasting, An[22] et al. constructed a sentiment index (ICSI) considering investor attention based on news text, Baidu search volume and price series, and further constructed a social media-based multimodal integrated forecasting (SMMEF) model that uses N-BEATSx in its second stage to capture the complementary information between the two modalities for final forecasting. Nayak[23] et al. explored the ability of N-BEATSx to forecast agricultural commodity prices using price data of basic crops from major markets in India supplemented with corresponding weather data (precipitation and temperature). Hu[24] et al. developed a model of a hydrogen storage system, constructed a new estimation methodology based on mass difference prediction (PMD method) based on the N-BEATSx model and validated the PMD methodology prediction accuracy under urban and high-speed conditions.

C. Optimised N-BEATSx

Currently, only a small number of studies have made improvements to N-BEATSx. To address the difficulty of achieving accurate results in traditional time series trend extrapolation of historical power data — caused by the high stochasticity and variability of offshore wind power — Shuxin[25] et al. proposed a novel N-BEATSx model enhanced with Copula entropy for short-term offshore wind power forecasting. Wang[26] et al. proposed an advanced multi-modal fusion forecasting model, WT-N-BEATSx-LSTM-RF (WNLR), which integrates N-BEATSx with wavelet transform (WT), long short-term memory (LSTM), and random forest (RF) to capture the rich temporal features of runoff sequences. Wang[27] et al. introduced a text-based deep learning forecasting model, enhancing the N-BEATSx model by incorporating weight coefficients and optimizing both the designed weight coefficients and hyperparameters through the use of Optuna. To overcome the limitation of N-BEATSx in handling external variables, Jiang et al.[28] proposed an interpretable probabilistic tariff prediction model, L-N-BEATSx, which combines N-BEATSx with LassoNet to enable automatic selection of relevant external variables. However, this model does not make any improvement to the values of hyperparameters, and consumes a lot of computational resources. Xu[29] et al. used the Sparrow Search Algorithm (SSA) to optimize the selection of a small number of hyperparameters in the N-N-BEATSx prediction model, and used the optimized values of hyperparameters to predict one-day tariffs in the market of Shanxi Province.

III. Preliminaries

A. Situation prediction tasks

Definition 1. (Time series of situation values):
denote a set of time series of situation values , $t \in T$ by st_t, denoted ST. T denotes a set of timestamps and the time interval is denoted by Δt.

Definition 2. (Exogenous variable value time series):
denote by EX_t^n the exogenous variable value time series corresponding to ST_t, $n \in N(x)$, $t \in T$. N_x is the number of exogenous variables. The rest is the same as Definition 1.

Definition 3. (situation Prediction):
a situation prediction is an estimate of the future value of a situation. The timestamp $T = \{1, 2, \ldots, t\}$ within the set of situation values can be represented as $ST = \{st_1, st_2, \ldots, st_t\}$. The goal of the situation prediction is to predict the situation values that are followed by a length of SL, i.e., {t+1, t+2, ... , the set of situation values of t + SL}, denoted as $ST' = \{st'_{t+1}, st'_{t+2}, \ldots, st'_{t+SL}\}$. SL denotes the length of the time series of the situation values to be predicted.

B. N-BEATSx

The basic architecture of N-BEATSx consists of blocks and stacks. N-BEATSx contains multiple stacks, and each stack contains multiple blocks, as shown in Fig.1.

The 1st block of the 1st stack in the model receives the target and exogenous variables of the backtracking window length as inputs, and all the subsequent blocks have the predicted residuals and exogenous variables generated by the previous block as inputs. The kth block of the ith stack receives $\left(y_{i,k-1}^{back}, X_{i,k-1}^{back}\right)$ as input data and goes through four fully connected layers according to (1) to get the hidden unit $h_{i,k} \in R^{N_h}$, the hidden unit then goes through a backward linear layer and a predictive linear layer according to (2) to get the backward prediction $\theta_{i,k}^{back}$ and the forward prediction $\theta_{i,k}^{for}$ respectively, and they then go through the base expansion operation between the learning coefficients of the backward basis functions and the base vectors of the blocks $g_{i,k}^{back}$ and $g_{i,k}^{for}$ as shown in (3) to get the backward prediction components and respectively. $\hat{y}_{i,k}^{back}$ and forward prediction component $\hat{y}_{i,k}^{for}$.

$$h_{i,k} = FC_{i,k}\left(y_{i,k-1}^{back}, X_{i,k-1}^{back}\right) \tag{1}$$

$$\theta_{i,k}^{back} = LIN^{back}\left(h_{i,k}\right), \theta_{i,k}^{for} = LIN^{for}\left(h_{i,k}\right) \tag{2}$$

$$\hat{y}_{i,k}^{back} = g_{i,k}^{back}\left(\theta_{i,k}^{back}\right), \hat{y}_{i,k}^{for} = g_{i,k}^{for}\left(\theta_{i,k}^{for}\right) \tag{3}$$

N-BEATSx uses a two-layer residual structure to organize multiple blocks into stacks. The input to the k+1st block of the ith stack is then the difference between the input of the kth block and the output of the backtracking prediction component of the kth block, i.e., (4). The advantage of this structure is that each time, the prediction error is put into the next block for retraining and re-prediction, and with the layer-by-layer training of blocks and stacks, the smaller and smaller error between the real and predicted values is continuously learnt, so that the model learns sufficiently from the situation data, thus greatly improving the prediction accuracy.

$$y_{i,k+1}^{back} = y_{i,k}^{back} - \hat{y}_{i,k}^{back} \tag{4}$$

As shown in (5), if there are M blocks in the ith stack, the sum of the forward prediction components of M blocks is the forward prediction result of the ith stack. If there are N stacks in the model, the sum of forward prediction of N stacks is the final prediction result of the model, as shown in (6).

$$\hat{y}_{i}^{for} = \sum_{k=1}^{M} \hat{y}_{i,k}^{for} \tag{5}$$

$$\hat{y}^{for} = \sum_{i=1}^{N} \hat{y}_{i}^{for} \tag{6}$$

In terms of interpretability, the N-BEATSx model allows each block to perform their projection directly using harmonic functions, polynomial trends and exogenous variables. When the base $V_{i,k}^{for}$ is $\mathrm{T} = [1, \mathrm{t}, ..., \mathrm{t}^{N_{pol}}] \in R^{H\times(N_{pol}+1)}$, the coefficients are those of the polynomial model of the trend. When the base $V_{i,k}^{for}$ is the harmonic $S = [1, \cos\left(2\pi \frac{t}{N_{hr}}\right), ..., \cos\left(2\pi \lfloor H/2-1 \rfloor \frac{t}{N_{hr}}\right), ..., \sin\left(2\pi \frac{t}{N_{hr}}\right), ..., \sin\left(2\pi \lfloor H/2-1 \rfloor \frac{t}{N_{hr}}\right)] \in R^{H\times(H-1)}$, the coefficient vector $\theta_{i,k}^{for}$ can be interpreted as Fourier transform coefficients, with the hyperparameters N_{hr} controlling the harmonic oscillation. The exogenous basis expansion can be considered as a time-varying local regression when the basis $V_{i,k}^{for}$ is the matrix $X = [X_1, ..., X_{N_x}] \in R^{H\times N_x}$, where N_x is the number of exogenous variables. Where the time vector $t^T = [0, 1, 2, ..., H-2, H-1]/H$ is discretely defined; N_{pol} is the maximum polynomial number. The number of interpretable bases of the model and the way they are combined can be flexibly adapted.

In the generic configuration, the basis of the non-Lindauer projection in (3) corresponds to the canonical vector, i.e., $g_{i,k}^{for} = I_{H\times H}$, a unitary matrix of dimension equal to the projected range H, which matches the bases of the coefficients $\left|\theta_{i,k}^{for}\right| = H$.

The overall framework of N-BEATSx studied in this paper is illustrated in the figure, which is expanded using three types of stacks, modelling trending, seasonal and exogenous factors respectively. This allows the model to both capture different parts of the time series and make the logic behind the forecasts more transparent. Within each module of N-BEATSx, the output layer of the Fully Connected Neural Network (FCNN) deploys a number of neurons H, where the value of H is taken in a one-to-one mapping with the time span of the forecast. That is, each neuron assumes a clear division of labour corresponding to a precise point in time during the prediction period and is responsible for generating the corresponding prediction value.

$$\hat{y}_{i,k}^{trend} = \sum_{j=0}^{N_{pol}} t^{i} \theta_{i,k,j}^{trend} \equiv T\theta_{i,k}^{trend} \tag{7}$$

$$\hat{y}_{i,k}^{season} = \sum_{j=0}^{\lfloor H/2-1 \rfloor} \cos\left(2\pi i \frac{t}{N_{hr}}\right)\theta_{i,k,j}^{season} + \sin\left(2\pi i \frac{t}{N_{hr}}\right)\theta_{i,k,j+\lfloor H/2 \rfloor}^{season} \equiv S\theta_{i,k}^{season} \tag{8}$$

$$\hat{y}_{i,k}^{exo} = \sum_{j=0}^{N_x} X_i \theta_{i,k,j}^{exo} \equiv X\theta_{i,k}^{exo} \tag{9}$$

Relative to NBEATS, exogenous blocks are a newly added class of stacks that learn context vectors from time-dependent covariates with encoder convolutional substructures $C_{i,k}$, as specified in (10), where the temporal convolutional network is chosen as the encoder. This allows the model to capture complex patterns in the time series while utilising the exogenous variables to provide additional information, thus improving the prediction performance. For generic configuration and exogenous blocks the ordering is determined by data-driven hyperparameter tuning.

$$\hat{y}_{i,k}^{exo} = \sum_{j=1}^{N_c} C_{i,k,j} \theta_{i,k,j}^{for} \equiv C_{i,k}\theta_{i,k}^{for} \quad with \quad C_{i,k} = TCN\left(X\right) \tag{10}$$

IV. Methodology

The overall process of the situation prediction method proposed in this paper is depicted in Fig. 2. It consists of three key sections: the data processing module, the hyperparameter optimization module, and the situation prediction module.

Firstly, the historical situation data is time-serialised and divided into a training set and a validation set. Secondly, the CWGWO algorithm proposed in this paper is used in the task of optimising the hyperparameters of N-BEATSx. Finally, the optimised N-BEATSx with optimised hyperparameter values is used to predict the future situation.

In the next section, we provide a detailed description of CWGWO and CWGWO-N-BEATSx.

A. perparameters of N-BEATSx

The objective of CWGWO is to identify the optimal combination of hyperparameters that maximizes the performance of N-BEATSx. Through comprehensive analysis and experimentation, we have compiled a list of all the hyperparameters that significantly influence the performance of N-BEATSx, as presented in Table 2. A description of the optimisation of some of the hyperparameters is given below:

- $size_2$, $size_3$ represent the number of neurons in the first and second hidden layer in the block, respectively.
- B_2 and B_3 represent the number of blocks in the second and third stacks, respectively.
- STT stands for the selection of the stack type, and in this study, after several experiments, it was decided that for the selection of the stack type, it is consistent with the proposed N-BEATSx model in the literature.
- NO_1 and NO_2 represent the selection of normalisation methods for the target and exogenous variables, respectively, and here the median normalisation method and the constant normalisation method were selected. normalisation method and invariant normalisation method were selected here.
- IO stands for weight initialisation methods, here three methods were selected, namely orthogonal initialisation, He normal distribution initialisation and Glorot normal distribution initialisation three methods.
- AO represents the activation function of the block and here four activation functions, Softplus, SELU, PReLU and Sigmoid, are selected.
- TF stands for whether batch normalisation is used.
- ENC represents the number of channels of the exogenous variable.
- DPT, DPE represent N-BEATS generic configuration dropout probability and exogenous variable dropout probability, respectively.
- TF_1, TF_2, and TF_3 represent whether the input contains data from the first 3, 4, and 8 time steps of the target variable y.
- TF_4, TF_5, and TF_6 represent whether the input contains data from the first 1, 2, and 8 time steps of exogenous variable 1 (TF_7~TF_{21} same as above).

Other hyperparameters are more common and will not be repeated here.

B. fitness function

The fitness function is used as an important index for evaluating the solution, and this paper takes two aspects into account for its design, which are prediction accuracy and model space complexity. In this paper, the mean value of MAE is used as an indicator for assessing the prediction accuracy of the model, and the overall number of parameters of the normalised model is used as an indicator for assessing the spatial complexity of the model, which is defined as follows:

$$fitness = MAE\_mean + Params\_total \tag{11}$$

$$MAE\_mean = \left(\frac{1}{N_{sample}} \sum_{i=1}^{N_{sample}} \frac{1}{SL} \sum_{m=1}^{SL} \left\| \tilde{y}_m^i - y_m^i \right\| \right) \times 100\% \tag{12}$$

$$Params\_total = Params_{in\_fir} + Params_{fir\_sec} + Params_{sec\_out} \tag{13}$$

We evaluate the effect of univariate point-in-time predictions carrying exogenous variables using the fitness function proposed above. The prediction error is measured using a widely adopted fitness function known as MAE (Mean Absolute Error). The model space complexity is quantified by the normalised total number of parameters of the model, Params_T. Where $\tilde{y}$ denotes the predicted value of the model for the target variable y. SL denotes the length of the time series of the target variable y in the test set. In this way, the average error of the model's prediction of the target variable at each time point can be derived. m refers to the total number of target variables in the test set.

In order to evaluate the complexity of the model and ensure its applicability in different tasks, we designed a detailed method for calculating the parameter quantities. The method not only considers the connections between input, hidden and output layers, but also adjusts the dimensions of θ vectors according to different stack types. Through normalisation, we ensure the comparability and stability of parameter quantities.

Since all the N-BEATSx models are fully connected layers, each neuron in a fully connected layer is connected to all the neurons in the previous layer. Therefore the following formula is used to calculate the number of parameters in each layer:

$$Params_{i\_i+1} = size_i \times size_{i+1} + size_{i+1} \quad i \in [1,4],\ i \in Z^+ \tag{14}$$

$size_i$ indicates the input dimension of the i layer, whose size is determined by the dictionary ex_{dict} that specifies the exogenous variables and the input dimension of the next layer, $size_{i+1}$. ex_{dict} determines which exogenous variables should be included in N-BEANSx to participate in the model training, and thus its length len_{ex}, together with $size_{i+1}$, determines the size of $size_i$, as shown in (15), where m_{is} is the hyperparameter that controls the input dimension.

$$size_{c_i} = \begin{cases} m_{is} \cdot size_{out}, & len_{ex} = NULL \\ len_{ex} \cdot m_{is} \cdot size_{out}, & len_{ex} \neq NULL \end{cases} \tag{15}$$

The number of stacks is then determined based on the stack type to determine the overall number of parameters. The total number of parameters Params_T is obtained by normalising the number of parameters per layer using the Min-Max normalisation method.

It is noteworthy that there exists a significant difference in the numerical scale between *MAE_mean* and *Params_T*; however, no additional scaling was applied. This discrepancy was intentionally preserved to reflect the inherent emphasis on prediction accuracy in the optimization objectives of this study. This "implicit weighting" design allows the complexity term to occupy a relatively small proportion of the overall fitness function while still exerting a substantive influence on the structural search. Essentially, the allocation of weights can be adjusted based on specific needs: in scenarios with limited computational resources, the weight of Params_T can be increased, whereas in accuracy-critical scenarios, the weight of MAE_mean can be given greater emphasis. Experimental results also demonstrate that this approach provides discriminative evaluation and consistently reflects changes in model performance. Therefore, this study adopts a dimensionally unaligned but strategically clear weighting scheme, which also lays the foundation for subsequent dynamic weight adjustment mechanisms.

C.iGWO

Compared with the traditional Grey Wolf Optimizer (GWO), iGWO has been enhanced and improved in several aspects to improve search efficiency and optimization performance. Specifically, the following parts have been improved:

The initial population of the classical grey wolf algorithm is randomly generated. iGWO supports the use of an initial solution to initialize the population, which allows the user to provide a better initial solution based on existing knowledge or experience, thus accelerating convergence.

In classical GWO, the new position of each wolf is obtained based on the positions of the three head wolves, α, β and δ, being added together with equal weights and then divided by 3. Specifically, the new position X of each wolf is calculated using (21). where X_α, X_β, and X_δ denote the positions of α, β, and δ, respectively; A_1, A_2, and A_3 are coefficient vectors controlling the speed and direction at which the other wolves are approaching the head wolf; and D_α, D_β, and D_δ denote the distance between the current wolf and α, β, and δ, respectively. This method is simple and intuitive, but lacks flexibility in that the influence of all head wolves is the same and the relative importance between them is not taken into account.

The linear decay coefficient vector A and the random coefficient vector C can be computed in the following way:

$$A = 2 \cdot a \cdot r_1 - a \tag{16}$$

$$C = 2 \cdot r_2 \tag{17}$$

Grey wolves rounding up prey can be modelled in the following form:

$$X_1 = X_\alpha - A \cdot \left| C \cdot X_\alpha - X_j \right| \tag{18}$$

$$X_2 = X_\beta - A \cdot \left| C \cdot X_\beta - X_j \right| \tag{19}$$

$$X_3 = X_\delta - A \cdot \left| C \cdot X_\delta - X_j \right| \tag{20}$$

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3} \tag{21}$$

Relying solely on (22), however, can diminish the population diversity of the GWO in the later stages of the solution search process. This tends to cause the GWO to fall into a local optimum.

In order to enhance the population diversity in GWO, the iGWO design introduces a local search mechanism to improve the position of individuals, thereby enhancing the global search capability of the algorithm and avoiding premature convergence to local optimal solutions. Specifically, for each individual, Eq.(22) is used to calculate its Euclidean distance from other individuals in the population, and the individuals whose distance is smaller than the current moving distance are selected for local search to generate Near(t). Where Di,j denotes the Euclidean distance between wolf i and wolf j, and aik and ajk are the coordinates of wolf i and wolf j in the kth dimension, respectively, where k dimension refers to the dimension of the objective function. XGWO(t+1) is updated according to (21).

$$D_{i,j} = \sqrt{\sum_{k=1}^{n} \left(a_{ik} - a_{jk}\right)^2} \tag{22}$$

The $X_{CWGWO}(t+1)$ position is then updated according to (23), where $X(t)$ is the position of wolf i, $X_n(t)$ is randomly sampled from $Near(t)$, and $X_r(t)$ is randomly sampled from the original wolf matrix, and r is a random number between 0 and 1:

$$X_{CWGWO}(t+1) = X(t) + r \times \left(X_n(t) - X_r(t)\right) \tag{23}$$

Finally $X_{GWO}(t+1)$ and $X_{CWGWO}(t+1)$ are compared according to (24):

$$X(t+1) = \begin{cases} X_{GWO}(t+1), & fitness(X_{GWO}) < fitness(X_{CWGWO}) \\ X_{CWGWO}(t+1), & fitness(X_{GWO}) \geq fitness(X_{CWGWO}) \end{cases} \tag{24}$$

In addition, iGWO introduces a linear decay coefficient α, see (25), which decreases linearly from 2 to 0. This coefficient is used to control the search behaviour of the grey wolves, with larger values encouraging exploration in the early stages, and smaller values in the later stages promoting exploitation. This dynamic adjustment mechanism allows the algorithm to automatically balance exploration and exploitation at different iteration stages to avoid falling into local optimal solutions.

$$\alpha = 2 - count \times \frac{2}{iterations} \tag{25}$$

Finally, iGWO has added support for hyperparameter target values. If the target value is set and the fitness value of the current optimal solution is already at or better than the target value in a given iteration, the algorithm terminates early and does not continue iterating. This saves computational resources, especially if the target value is known.

D. Initial solution generation of CWGWO

In general, the initial solution of the grey wolf algorithm is randomly generated, while iGWO adds an initial solution setting on the basis of random generation, so that the user can set the value of the initial solution according to the a priori knowledge, and then randomly generate the initial solution for the initial solution that is not set. However, this kind of population initialization requires a large amount of a priori knowledge to support, which increases the preliminary workload and does not significantly reduce the training time. Therefore, in this paper, a new initialization scheme is selected, in which the outputs of the Tent mapping function and the sine mapping function are combined in a weighted manner, so as to enhance the diversity of the initial solutions.

The Tent mapping is a simple discrete-time chaotic system whose mathematical expression is given in (26), where xn is the value of the nth iteration, usually in the interval [0, 1]. μ is a control parameter that determines the shape and behaviour of the mapping, and usually takes values in the range of $0<\mu<1$.

$$x_{n+1}=\begin{cases}\dfrac{x_n}{\mu}, & if\ 0\le x_n<\mu\\ \dfrac{2-x_n}{2(1-\mu)}, & if\ \mu\le x_n\le 1\end{cases}\tag{26}$$

Sine mapping is a nonlinear mapping based on sinusoidal functions, which generates sequences with complex dynamic behaviour through the periodic and nonlinear properties of sinusoidal functions. Unlike Tent mapping, the graph of sinusoidal mapping is not segmented linear but smooth curves, so it can introduce more nonlinear variations. Its mathematical expression is given in (27). where xn is the value of the nth iteration, usually in the interval [0, 1]. is the square of a sinusoidal function, ensuring that the output value is always in the range [0, 1].

$$x_{n+1}=\sin^2(\pi x_n)\tag{27}$$

The sequences generated by the Tent mapping are well traversed and random, and can be distributed uniformly in the search space, but their behaviour is "sharp", especially around the boundaries. This means that the solutions generated by the Tent mapping may be too concentrated in certain regions, leading to a lack of population diversity. Sine mapping generates smoother sequences with periodic variations, which can remain stable in certain regions for a longer period of time. Although it can also cover the entire search space, its smoothness may result in overly dense solutions in some regions and fewer solutions in others.

The weighted combination of the outputs of these two mapping functions retains the traversal and randomness of the Tent mapping while introducing the smoothness and nonlinear variation of the sinusoidal mapping. This combination can better cover the whole search space and avoid overly concentrated or sparse solutions in certain regions, thus improving the diversity of the population.

$$X_{j,i}=0.5\cdot Tent\_map(u)+0.5\cdot Sinusoidal\_map(u)\tag{28}$$

Fig.4 compares the distribution of initial solutions generated by the Tent mapping alone, the Sine mapping alone, and the combination of the two mappings by weight. The figure shows that the "Tent-Sine" mapping combines the advantages of both mappings and generates a more diverse population.

E.Solution Weight allocation of CWGWO

Instead of simply averaging the positions of the three head wolves, CWGWO introduces a dynamic position-based weight allocation mechanism based on iGWO. Compared to iGWO which averages the new positions X_1, X_2 and X_3 of the three head wolves, see (29)~(32), CWGWO improves the weight assignment of the three head wolves by calculating the dynamic weights ω_1, ω_2 and ω_3 for X_1, X_2 and X_3, and weightedly averages the new positions of the three head wolves to obtain the final updated position. With dynamic weight adjustment, the algorithm is able to find a balance between global and local search. The weights are calculated based on the absolute value of the current positions of the three wolves, which means that the weights are automatically adjusted with the iterative process. That is, when a wolf's position is closer to the optimal solution, its weight will be larger, thus guiding the other wolves to move in that direction. This allows the algorithm to better adapt to changes in the search space.

$$\omega_1=\frac{|X_1|}{|X_1|+|X_2|+|X_3|+\varepsilon}\tag{29}$$

$$\omega_2=\frac{|X_2|}{|X_1|+|X_2|+|X_3|+\varepsilon}\tag{30}$$

$$\omega_3=\frac{|X_3|}{|X_1|+|X_2|+|X_3|+\varepsilon}\tag{31}$$

$$X(t+1)=clip\left(\frac{\omega_1\cdot X_1+\omega_2\cdot X_2+\omega_3\cdot X_3}{3},\min,\max\right)\tag{32}$$

Where the clip function is used to ensure that the new position is within the given boundaries, i.e., between min and max. ε is a very small value used to prevent the denominator from being zero and to ensure the stability of the calculation.

The specific algorithm flowchart is shown below:

(Fig.5)

F. Convergence proof of CWGWO

As CWGWO is an enhanced version of the convergent GWO algorithm, we demonstrate its convergence by applying a relevant theorem from harness theory, a method commonly employed in the convergence analysis of heuristic algorithms. Initially, we develop a harness theory model specifically for CWGWO.

Definition 4. In GWO and CWGWO, the fitness function, i.e., (11), is the objective function $f(x)$ of the harness theory, $x_\alpha^{(t)}$ is the location of the optimal solution α in the tth iteration, $t\in\{1,2,...,T\}$, and x^* is the global optimal solution.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2025.3581958

Definition 5. In GWO, M_t denotes the gap between the current optimal solution and the global optimal solution. ξ_t contains all the information that can be determined at time t and before, including the current optimal solution location Mt and the fitness value $f(M_t)$.

Definition 6. In GWO, CM_t represents the gap between the current optimal solution and the global optimal solution. ξ_t contains all the information that can be determined at time t and before, including the current optimal solution location CM_t and the fitness value $f(CM_t)$.

Next, we need proof:

Theorem 1: The sequence Mt of harnesses of GWO is a superharness.

First, construct a sequence of harnesses associated with the objective function $f(x)$. Definition:

$$M_t = f\left(x_\alpha^{(t)}\right) - f\left(x^*\right) \tag{33}$$

Obviously, it follows from the existing theory[30] that we need to show that Mt is a superharness, i.e:

$$E\left[M_{t+1} \mid \xi_t\right] \le M_t \tag{34}$$

This will show that the expected value of Mt will not increase, thus proving that the GWO will gradually converge to the globally optimal solution.

First, consider the expected value of M_{t+1}:

$$E\left[M_{t+1} \mid \xi_t\right] = E\left[f\left(x_\alpha^{(t+1)}\right) - f\left(x^*\right) \mid \xi_t\right] \tag{35}$$

According to the update rule of GWO, the optimal solution $x_\alpha^{(t+1)}$ is the best individual selected from the current population. Therefore, $x_\alpha^{(t+1)}$ may be the current optimal solution $x_\alpha^{(t)}$ or the new position of other individuals after position update. To simplify the analysis, we simplify the updating method of $x_\alpha^{(t+1)}$ from

$x_\alpha^{(t+1)} = \frac{1}{3}\left(x_\alpha^{(t)} - A_1 \cdot D_\alpha + x_\beta^{(t)} - A_2 \cdot D_\beta + x_\delta^{(t)} - A_3 \cdot D_\delta\right)$ to the following:

$$x_\alpha^{(t+1)} = x_\alpha^{(t)} - A \cdot D_\alpha \tag{36}$$

Next, we analyse the expectation of $f\left(x_\alpha^{(t+1)}\right)$. Since the objective function $f(x)$ is continuous and differentiable, we can use a Taylor expansion to approximate $f\left(x_\alpha^{(t+1)}\right)$:

$$f\left(x_\alpha^{(t+1)}\right) \approx f\left(x_\alpha^{(t)}\right) + \nabla f\left(x_\alpha^{(t)}\right)^T \left(x_\alpha^{(t+1)} - x_\alpha^{(t)}\right) + \frac{1}{2}\left(x_\alpha^{(t+1)} - x_\alpha^{(t)}\right)^T \nabla^2 f\left(x_\alpha^{(t)}\right)\left(x_\alpha^{(t+1)} - x_\alpha^{(t)}\right) \tag{37}$$

Substituting $x_\alpha^{(t+1)} = x_\alpha^{(t)} - A \cdot D_\alpha$, we get:

$$f\left(x_\alpha^{(t+1)}\right) \approx f\left(x_\alpha^{(t)}\right) - \nabla f\left(x_\alpha^{(t)}\right)^T \left(A \cdot D_\alpha\right) + \frac{1}{2}\left(A \cdot D_\alpha\right)^T \nabla^2 f\left(x_\alpha^{(t)}\right)\left(A \cdot D_\alpha\right) \tag{38}$$

Now, we calculate the expected value of $f\left(x_\alpha^{(t+1)}\right)$:

$$E\left[f\left(x_\alpha^{(t+1)}\right) \mid \xi_t\right] = f\left(x_\alpha^{(t)}\right) - E\left[\nabla f\left(x_\alpha^{(t)}\right)^T \left(A \cdot D_\alpha\right) \mid \xi_t\right] + \frac{1}{2} E\left[\left(A \cdot D_\alpha\right)^T \nabla^2 f\left(x_\alpha^{(t)}\right)^T \left(A \cdot D_\alpha\right) \mid \xi_t\right] \tag{39}$$

Since A_1, A_2, A_3 and C_1, C_2, C_3 are random variables and $E\left[A_i \mid \xi_t\right] = a_t$ and $E\left[C_i \mid \xi_t\right] = c_t$, we have:

$$E\left[\nabla f\left(x_\alpha^{(t)}\right)^T \left(A \cdot D_\alpha\right) \mid \xi_t\right] = \nabla f\left(x_\alpha^{(t)}\right)^T E\left[\left(A \cdot D_\alpha\right) \mid \xi_t\right] = \nabla f\left(x_\alpha^{(t)}\right)^T \left(a_t \cdot D_\alpha\right) \tag{40}$$

Eventually, we get:

$$E\left[f\left(x_\alpha^{(t+1)}\right) \mid \xi_t\right] \approx f\left(x_\alpha^{(t)}\right) - a_t \nabla f\left(x_\alpha^{(t)}\right)^T D_\alpha \tag{41}$$

In order to prove that M_t is superharnessed, we need to satisfy (34), i.e:

$$E\left[f\left(x_\alpha^{(t+1)}\right) - f\left(x^*\right) \mid \xi_t\right] \le f\left(x_\alpha^{(t)}\right) - f\left(x^*\right) \tag{42}$$

From the previous derivation, we have (38), thus:

$$E\left[M_{t+1} \mid \xi_t\right] \approx M_t - a_t \nabla f\left(x_\alpha^{(t)}\right)^T D_\alpha \tag{43}$$

To ensure that $E\left[M_{t+1} \mid \xi_t\right] \le M_t$, we need to choose the right a_t and D_α, making:

$$a_t \nabla f\left(x_\alpha^{(t)}\right)^T D_\alpha \ge 0 \tag{44}$$

Typically, a_t decreases linearly with the number of iterations, i.e., $a_t = \frac{c}{t+1}$, where c is a constant greater than zero. This ensures that the step size decreases as the number of iterations increases, reducing the risk of over-exploration. D_α is the absolute value of the distance between an individual and the optimal solution, and is positive. Therefore, $\nabla f\left(x_\alpha^{(t)}\right)^T D_\alpha$ is usually positive. By the above analysis, we prove that M_t is a super harness, i.e.:

$$E\left[M_{t+1} \mid \xi_t\right] \le M_t \tag{45}$$

Theorem 1 is proved.

Next, we need proof:

Theorem 2: A GWO is convergent if the sequence of harnesses Mt of the GWO is a superharness.

According to the properties of the super harness[31], we know that the expected value of Mt does not increase and that Mt gradually approaches zero as the number of iterations increases. This means:

$$\lim_{t \to \infty} E\left[M_t\right] = 0 \tag{46}$$

To wit:

$$\lim_{t \to \infty} E\left[f\left(x_\alpha^{(t)}\right) - f\left(x^*\right)\right] = 0 \tag{47}$$

This suggests that the GWO converges gradually to the globally optimal solution x^* with appropriate parameter choices.

Theorem 2 is proved.

Finally, we have to prove the following theorem:

Theorem 3: The sequence of harnesses CM_t of CWGWO is a hyperharness.

Since the position update method of GWO is $x_\alpha^{(t+1)} = \frac{1}{3}\left(x_\alpha^{(t)} - A_1 \cdot D_\alpha + x_\beta^{(t)} - A_2 \cdot D_\beta + x_\delta^{(t)} - A_3 \cdot D_\delta\right)$,i.e., $x_\alpha^{(t+1)} = \frac{1}{3}\left(X_\alpha^{(t)} + X_\beta^{(t)} + X_\delta^{(t)}\right)$, similarly according to (29~32) the position update method of CWGWO can be obtained as:

$$x_\alpha^{(t+1)} = \frac{1}{3}\left(\frac{\left|X_\alpha^{(t)}\right|^2 + \left|X_\beta^{(t)}\right|^2 + \left|X_\delta^{(t)}\right|^2}{\left|X_\alpha^{(t)}\right| + \left|X_\beta^{(t)}\right| + \left|X_\delta^{(t)}\right|}\right) \tag{48}$$

Again, to simplify the analysis, we have simplified $x_{\alpha}^{(t+1)}$:

$$x_{\alpha}^{(t+1)} = \frac{1}{3}\left(\frac{3\cdot\left|X_{\alpha}^{(t)}\right|^2}{3\cdot\left|X_{\alpha}^{(t)}\right|}\right) = \frac{\left|X_{\alpha}^{(t)}\right|}{3} = \frac{1}{3}\left(x_{\alpha}^{(t)} - A_1\cdot D_{\alpha}\right) \quad (49)$$

It can be seen that (49) is multiplied by only one more constant than (36), and thus the same reasoning proves that CM_t is a superharness, i.e:

$$E\left[CM_{t+1} \mid \xi_t\right] \le CM_t \quad (50)$$

Theorem 3 is proved.

That is, CWGWO converges gradually to the globally optimal solution x^* with appropriate parameter choices.

V. Experiment

A.Research question

The analyses carried out in our experiments were designed to investigate four main questions (RQs):

RQ1 Is CWGWO the optimal heuristic algorithm for optimizing the hyperparameters of N-BEATSx?

RQ2 Did all the enhancements we designed for CWGWO-N-BEATSx meet the expected outcomes?

RQ3 Is CWGWO-N-BEATSx equally good at predicting situational datasets as well as normal time-series datasets?

RQ4 Do our improvements to N-BEATSx hold an advantage over frontier situation prediction models?

To investigate these four problems, we structured our research into four experimental tasks. In the first task, we evaluated and compared the optimization capabilities of different heuristic algorithms for tuning the hyperparameters of N-BEATSx. In the second task, we conducted ablation studies on the improvements made from GWO to CWGWO. In the third experimental task, we tested the scalability of CWGWO-N-BEATSx prediction for different datasets. In the last experimental task, we compared CWGWO-N-BEATSx with other existing SOTA methods.

B.Data

In this paper, two datasets are used to validate the predictive effectiveness of CWGWO-N-BNEATSx.First, we extend the univariate situation dataset Cncert used in [32] by incorporating multivariate exogenous variables to better support the modeling demands of CWGWO-N-BEATSx under complex environments. Specifically, 17 macroeconomic and social indicators potentially related to situation dynamics were initially collected as candidate exogenous variables. These variables span multiple dimensions, including economic activity, financial flows, and price fluctuations.

During preprocessing, we standardized the temporal granularity of all exogenous variables and the target variable, aligning them on a weekly basis to ensure consistency in time steps. Missing values were imputed using linear interpolation. Furthermore, all variables were normalized using Z-score standardization to eliminate scale differences and prevent any single variable from dominating the training process.

As shown in Figure 4, to identify the most relevant exogenous variables, we computed the Pearson correlation coefficients between the 17 candidate variables and the target variable y, and visualized the results using a correlation heatmap. Based on the correlation distribution (Figure 5), six variables with a strong positive correlation with y were selected as exogenous inputs for the model: Gross Domestic Product per capita (GDP), Consumer Price Index (CPI), Agricultural Product Price Index (ECPI-N), Money Supply (M2), Gold Reserve (GR), and China's Accumulated Local and Foreign Currency Deposits (TDC). Since the situation dataset with exogenous variables was collected only once, another real-world dataset was chosen to evaluate the performance of the model.The BJ dataset collects the Beijing air quality index for the six-year period from 2010 to 2015, in which the target variable is the PM2.5 index for these six years, and the exogenous variables that have an impact on it are also six:which are DEWP, HUMI, PRES, TEMP, cbwd, and Iws.

We set the ratio of the training set size to the test set size to 9:1. the MAE mean, MSE, and SMAPE are used as evaluation metrics:

$$MAE = \frac{1}{N_{sample}}\sum_{i=1}^{N_{sample}}\frac{1}{SL}\sum_{m=1}^{SL}\left\|\widetilde{y}_m^i - y_m^i\right\| \quad (51)$$

$$MSE = \frac{1}{N_{sample}}\sum_{i=1}^{N_{sample}}\left(\widetilde{y}_m^i - y_m^i\right)^2 \quad (52)$$

$$SMAPE = \frac{100}{N_{sample}}\sum_{i=1}^{N_{sample}}\frac{\left|\widetilde{y}_m^i - y_m^i\right|}{\left|\widetilde{y}_m^i\right| + \left|y_m^i\right|}\times 2 \quad (53)$$

C. Convergence ability experiment

The No Free Lunch (NFL) theorem[33] precludes the direct selection of an optimization algorithm for the N-BEATSx hyperparameters. Consequently, we conduct a comparative analysis between CWGWO and other heuristic algorithms to highlight the advantages of CWGWO in optimizing the N-BEATSx hyperparameters. Among the heuristic algorithms considered is the Particle Swarm Optimization Algorithm (PSO), Gravity Search Algorithm (GSA)[34] , Genetic Algorithm (GA), Sine Cosine Algorithm (SCA)[35] , Multiverse Optimization Algorithm (MVO)[36] , Sparrow Search Algorithm (SSA)[37] , Student Psychology Based Optimization(SPBO)[38] and Grey Wolf Algorithm (GWO).

The population size of CWGWO is set to 20. in order to balance the convergence ability of the algorithm, the population size of the other heuristics is set to 40. the maximum number of iterations for all the heuristics is set to 100. in practice, it is recommended to maximise the population size and the maximum number of iterations within the time constraints in order to obtain the best optimisation results. The other parameters of the above heuristic algorithm take the following values: the inertia weight of PSO is 0.9; the cognitive coefficient of PSO is 2; the social coefficient of PSO is 2; the decay factor of PSO is 0.9; the mutation probability of GA is 0.1; the number of elite strategies of GA is 1; the distribution index of GA is 1; the population size factor of GA is 1; the control parameter of SCA is 2; the mutation probability of BBO is 0.1; the number of elite strategies for BBO is 1; and the distribution index for BBO is 3.

The experimental results are shown in Table 4, in the BJ dataset, the best fitness of CWGWO is 12.36% better than the best fitness of the second best SSA, and it is also reduced by 9.88%, 1.22%, and 15% on the remaining three metrics MAE and SMAPE and MSE, respectively. In the Cncert dataset, CWGWO was 8.07% better than the next best GWO in terms of the best fitness, and was also 7.97%, 8.69%, and 7.69% lower on the remaining three metrics, MAE and SMAPE and MSE, respectively. In addition, CWGWO also has the second-smallest number of parameters on the BJ dataset after SSA, and the third-lowest number of parameters on the Cncert dataset, the values of the hyperparameters specifically related to the number of parameters and the length of the dictionary of exogenous variables are shown in Table 5. Which indicates that CWGWO-N-BEATSx not only has good prediction accuracy, but also its model complexity is also low, which makes it a more desirable situation prediction model.

To further visualize and validate the modeling accuracy and generalization capability of the proposed CWGWO-N-BEATSx model in practical multivariate time series situation prediction tasks, we plotted comparison curves between the predicted and actual values on the BJ dataset for the year 2015 and on the Cncert dataset, as shown in Figure 7. It can be observed that CWGWO-N-BEATSx achieves a strong fit to the original situation sequences, with the predicted curves closely aligning with the actual curves in both overall trends and local fluctuations. This indicates the model's strong ability to capture both the trend and periodic variations inherent in time series data.

In addition, the model exhibits consistently low residual fluctuations across different time periods, reflecting its stability and robustness across diverse scenarios. The results presented in the figure further corroborate the conclusions drawn from the aforementioned quantitative experiments, demonstrating that the CWGWO optimization strategy effectively enhances the hyperparameter search process of N-BEATSx. It not only improves prediction accuracy but also controls model complexity, thereby achieving a favorable balance between performance and efficiency.

D. Ablation experiment

The objective of the ablation experiment on the improvement levels was to highlight the necessity of the enhancements we made to CWGWO. These enhancements include:

- W: Cephalopod Weight Distribution
- T: Tent and sinusoidal mapping weighted initialisation population strategy

The experimental results are shown in Table 6. The experimental results show that iGWO_T_W obtains the lowest MAE, SMAPE, MSE and fitness on the situation awareness dataset Cncert and the real-world dataset BJ. Compared with iGWO, the ‘Tent-Sine’ initialisation has the greatest impact on the prediction effect, with the ‘Tent-Sine’ initialisation method reducing the three indicators and fitness values by 2.07%, 2.10%, 5.26% and 3.65% on the BJ dataset, and by 4.27%, 1.82% and 3.65% on the BJ dataset, respectively, 5.26%, and 3.65%; on the BJ dataset, the ‘Tent-Sine’ initialisation method reduces the three metrics and fitness values by 4.27%, 1.82%, 2.00%, and 3.68%, respectively. Nevertheless, both the ‘Tent-sine’ mapping initialisation population strategy and the head wolf weight allocation strategy are also crucial for optimal prediction results.

E. Compare with the SOTA methods

To comprehensively validate the effectiveness of the proposed CWGWO-N-BEATSx model in multivariate time series situation prediction tasks, we conducted comparative experiments using five state-of-the-art models across two datasets: BJ and Cncert. The selected baseline methods and their key characteristics are as follows:

iTransformer: By introducing a time series decomposition module, this model explicitly separates the sequence into trend and seasonal components[40], improving its ability to model long-term dependencies. However, it lacks mechanisms for model complexity control, posing a risk of structural redundancy.

CNN+BiLSTM+Attention: This approach integrates Convolutional Neural Networks, Bidirectional Long Short-Term Memory networks, and the attention mechanism[41]. While it demonstrates competence in extracting temporal features, its capacity to model multivariate exogenous variables is limited, making it inadequate for capturing complex inter-variable dependencies.

iGWO-N-BEATS: This method combines the classic improved Grey Wolf Optimizer (iGWO) with the univariate N-BEATS model[32], offering strong optimization capabilities. However, it does not accommodate exogenous variables, limiting its predictive capacity in multidimensional data environments.

N-BEATSx: The original N-BEATSx model, which has not undergone any improvements, possesses the capability to handle exogenous variables[1] but lacks structural optimization and hyperparameter tuning, resulting in suboptimal performance.

SSA-N-BEATSx: Utilizing the SSA algorithm to optimize certain hyperparameters[29], this model achieves improved accuracy. Nevertheless, its search strategy exhibits limited stability and does not incorporate complexity control mechanisms.

To ensure fairness, all comparison models were retrained on the two datasets using optimized hyperparameters. The comparative results, shown in Table 7, demonstrate that CWGWO-N-BEATSx consistently achieves superior performance across all evaluation metrics:

On the BJ dataset, CWGWO-N-BEATSx reduced average MAE by 17.55%, SMAPE by 8.932%, and MSE by 43.87% compared to the other five SOTA models.

On the Cncert dataset, CWGWO-N-BEATSx reduced average MAE by 17.614%, SMAPE by 16.27%, and MSE by 23.488% compared to the same baselines.

Overall, CWGWO-N-BEATSx effectively inherits the strengths of the N-BEATSx architecture in trend-seasonality decomposition, while achieving breakthroughs in optimization stability, convergence speed, and complexity control through the integration of the CWGWO. Compared to CNN+BiLSTM+Attention, the CWGWO-N-BEATSx model features a simpler structure, is easier to train, and shows significant improvement in SMAPE. Compared to iTransformer, CWGWO-N-BEATSx demonstrates better accuracy and significantly reduces MSE. Relative to iGWO-N-BEATS, this model incorporates support for exogenous variables, greatly enhancing adaptability to multidimensional data. Finally, compared to SSA-N-BEATSx, CWGWO offers a more robust search mechanism and effective model compression, further improving generalization performance.

F. Robustness Evaluation and Significance Analysis

To further verify the robustness and statistical significance of the proposed model, we conducted 10 independent experiments on both the BJ and Cncert datasets, recording the mean and standard deviation (mean ± std) for three key metrics: MAE, SMAPE, and MSE, as presented in Table Ⅷ. Results marked with an asterisk (*) indicate that the corresponding model demonstrates significantly superior performance over other baseline methods in multiple independent trials.

Experimental findings show that the CWGWO-N-BEATSx model consistently achieves the best performance across all metrics on both datasets. On the BJ dataset, CWGWO-N-BEATSx yields the lowest MAE (2.536 ± 0.021), SMAPE (48.775 ± 0.190), and MSE (0.0017 ± 0.0001), significantly outperforming traditional approaches such as CNN+BiLSTM+Attention as well as advanced methods including iTransformer and iGWO-N-BEATS. Similarly, on the Cncert dataset, CWGWO-N-BEATSx again achieves optimal results with MAE (4.690 ± 0.019), SMAPE (11.148 ± 0.160), and MSE (0.0048 ± 0.0001), demonstrating a clear advantage over all comparative models.

Notably, CWGWO-N-BEATSx also exhibits the lowest standard deviations in most metrics, indicating strong stability under repeated experimental conditions. This robustness is primarily attributed to the adaptive optimization capability of the CWGWO algorithm, which enables the model to maintain excellent performance under various hyperparameter settings.

Furthermore, the observed differences among experimental results are statistically significant, further confirming that the superior performance of the proposed model is not due to randomness. In summary, CWGWO-N-BEATSx combines high accuracy with strong stability, making it a generalizable and practically valuable framework for real-world situation prediction applications.

Ⅵ. Discussion

The superiority of the CWGWO-N-BEATSx algorithm is demonstrated through the evaluation of both a situation dataset and a real-world dataset.

Compared with existing methods, the proposed CWGWO-N-BEATSx not only integrates the powerful modeling capabilities of the N-BEATSx architecture with the efficiency of an improved Grey Wolf Optimizer, but also introduces, for the first time, a joint optimization strategy for exogenous variable selection and structural hyperparameter configuration. This enables a well-balanced trade-off between predictive accuracy and model complexity. Experimental results demonstrate that CWGWO-N-BEATSx outperforms current mainstream state-of-the-art (SOTA) methods — such as CNN+BiLSTM+Attention, iTransformer, and SSA-N-BEATSx—on both datasets across multiple evaluation metrics including MAE, SMAPE, and MSE, validating the practicality and advancement of the proposed approach.

Furthermore, this research contributes not merely as a case study, but as a systematic extension to the existing body of work in areas such as structural optimization, joint modeling of exogenous information, and metaheuristic algorithm refinement. Specifically, the dynamic weight allocation mechanism and the Tent-sine initialization strategy proposed within CWGWO demonstrate strong generalizability and portability, offering new insights into the design of optimization strategies for deep learning models.

In terms of computational efficiency, the time complexity of CWGWO-N-BEATSx is primarily influenced by the number of model training iterations and the dimensionality of the hyperparameter search space. CWGWO enhances initial population quality and incorporates adaptive search weighting to effectively narrow the search space and accelerate convergence. In practice, on the same computing platform , CWGWO-N-BEATSx reduces average training time by approximately 14.8% compared to SSA-N-BEATSx, while maintaining superior predictive accuracy and improved computational efficiency.

From a broader perspective, situation prediction plays a strategically significant role in domains such as traffic safety, cybersecurity, and urban management[42]. The accuracy of prediction models directly influences the intelligence and responsiveness of decision-making systems. The CWGWO-N-BEATSx model proposed in this study improves prediction accuracy while reducing model complexity, thereby providing a practical and effective solution for building efficient, intelligent, and interpretable situation awareness systems. This work thus holds substantial research value and promising application potential.

However, there are still limitations of our study, including:

- Since optimising the number of hidden layers of the N-BEATSx model implies that there are more parameters to be trained, which would significantly increase the required computational resources and time, and is prone to overfitting, it is not optimised in this paper. However, this may result in a more superior model structure that is not exploited.

- Although FC-GAGA[39], like N-BEATSx, is an extension of the N-BEATS framework, it possesses a distinct set of hyperparameters. As a result, the conclusions drawn from this study cannot be directly applied to FC-GAGA. The unique hyperparameter configuration of FC-GAGA requires separate analysis to fully understand its performance and optimization needs.

- In this study, the situation prediction task is conceptualized as a time series prediction task incorporating multivariate exogenous variables. However, the situation prediction task also necessitates the integration of spatiotemporal relationships, encompassing factors such as traffic dynamics and geographical location attributes.

In future work, we aim to improve our CWGWO by combining spatio-temporal data to optimize FC-GAGA for situation prediction tasks and achieve more accurate situation awareness.Furthermore, future work will explore resource optimization strategies for deployment on edge devices and embedded environments, aiming to enhance the model ’ s adaptability and inference efficiency on specific hardware platforms in the context of situation prediction.

Ⅶ. Conclusion

This study proposes a new model architecture, CWGWO-N-BEATSx, for situation prediction tasks. The main idea of this work is to globally optimise the hyper-parameters of N-BEATSx using an improved GWO, i.e., CWGWO. Through this optimization, the model not only improves prediction accuracy and convergence speed but also effectively controls model complexity. Experimental results demonstrate that CWGWO-N-BEATSx outperforms traditional methods and several advanced models across multiple metrics, showcasing its broad potential for application in complex situation prediction.

References

[1] Olivares K G, Challu C, Marcjasz G, et al. Neural basis expansion analysis with exogenous variables: Forecasting electricity prices with NBEATSx[J]. International Journal of Forecasting, 2023, 39(2): 884-900.

[2] Souto H G, Moradi A. Introducing NBEATSX to realized volatility forecasting[J]. Expert Systems with Applications, 2024, 242: 122802.

[3] Mamadbekov F. PREDICTING FUTURE DEMAND: ML METHODS FOR DEMAND FORECASTING[J]. 2024.

[4] Wang T, Guo Y, Evgenievna M S, et al. Application of a Multi-Model Fusion Forecasting Approach in Runoff Prediction: A Case Study of the Yangtze River Source Region[J]. Sustainability, 2024, 16(14): 5964.

[5] Shuxin T, Feng Z, Yang F, et al. Short-term prediction of offshore wind power based on an improved N-BEATSX model[C]//2023 IEEE 6th International Electrical and Energy Conference (CIEEC). IEEE, 2023: 3812-3817.

[6] Wang L, An W, Li F T. Text-based corn futures price forecasting using improved neural basis expansion network[J]. Journal of Forecasting, 2024.

[7] Belete D M, Huchaiah M D. Grid search in hyperparameter optimization of machine learning models for prediction of HIV/AIDS test results[J]. International Journal of Computers and Applications, 2022, 44(9): 875-886.

[8] Wu L, Perin G, Picek S. I choose you: Automated hyperparameter tuning for deep learning-based side-channel analysis[J]. IEEE Transactions on Emerging Topics in Computing, 2022.

[9] Nematzadeh S, Kiani F, Torkamanian-Afshar M, et al. Tuning hyperparameters of machine learning algorithms and deep neural networks using metaheuristics: A bioinformatics study on biomedical and biological cases[J]. Computational biology and chemistry, 2022, 97: 107619.

[10] Lin H. Flow Direction Algorithm: A Comprehensive Review[J]. Archives of Computational Methods in Engineering, 2025: 1-17.

[11] Qiu H, Zheng Q, Msahli M, et al. Topological graph convolutional network-based urban traffic flow and density prediction[J]. IEEE transactions on intelligent transportation systems, 2020, 22(7): 4560-4569.

[12] Moishin M, Deo R C, Prasad R, et al. Designing deep-based learning flood forecast model with ConvLSTM hybrid algorithm[J]. IEEE Access, 2021, 9: 50982-50993.

[13] Chen Z. Research on internet security situation awareness prediction technology based on improved RBF neural network algorithm[J]. Journal of Computational and Cognitive Engineering, 2022, 1(3): 103-108.

[14] Jin Y, Wang R, Zhuang X, et al. Prediction of COVID-19 data using an ARIMA-LSTM hybrid forecast model[J]. Mathematics, 2022, 10(21): 4001.

[15] Yuan L, Zeng Y, Chen H, et al. Terminal Traffic Situation Prediction Model under the Influence of Weather Based on Deep Learning Approaches[J]. Aerospace, 2022, 9(10): 580.

[16] Sun H, Wang J, Chen C, et al. ISSA-ELM: A network security situation prediction model[J]. Electronics, 2022, 12(1): 25.

[17] Zhao D, Shen P, Zeng S. ALSNAP: Attention-based long and short-period network security situation prediction[J]. Ad Hoc Networks, 2023, 150: 103279.

[18] Xi Z, Kou Y, Li Y, et al. A Dynamic Air Combat Situation Assessment Model Based on Situation Knowledge Extraction and Weight Optimization[J]. Aerospace, 2023, 10(12): 994.

[19] WU Y A X, ZHAO D M E I. Build IPSO-ABiLSTM Model for Network Security Situation Prediction[J]. Journal of Information Science & Engineering, 2024, 40(1).

[20] Zhang X, Zhao X, Xu Y, et al. Situational-aware multi-graph convolutional recurrent network (sa-mgcrn) for travel demand forecasting during wildfires[J]. Transportation Research Part A: Policy and Practice, 2024, 190: 104242.

[21] Souto H G, Moradi A. Introducing NBEATSX to realized volatility forecasting[J]. Expert Systems with Applications, 2024, 242: 122802.

[22] An W, Wang L, Zeng Y R. Social media-based multi-modal ensemble framework for forecasting soybean futures price[J]. Computers and Electronics in Agriculture, 2024, 226: 109439.

[23] Nayak G H H, Alam M W, Singh K N, et al. Exogenous variable driven deep learning models for improved price forecasting of TOP crops in India[J]. Scientific Reports, 2024, 14(1): 17203.

[24] Hu D, Huang J, Lu D, et al. A real-time hydrogen consumption estimation method for fuel cell vehicles[J]. International Journal of Green Energy, 2024: 1-13.

[25] Shuxin T, Feng Z, Yang F, et al. Short-term prediction of offshore wind power based on an improved N-BEATSX model[C]//2023 IEEE 6th International Electrical and Energy Conference (CIEEC). IEEE, 2023: 3812-3817.

[26] Wang T, Guo Y, Evgenievna M S, et al. Application of a Multi-Model Fusion Forecasting Approach in Runoff Prediction: A Case Study of the Yangtze River Source Region[J]. Sustainability, 2024, 16(14): 5964.

[27] Wang L, An W, Li F T. Text-based corn futures price forecasting using improved neural basis expansion network[J]. Journal of Forecasting, 2024.

[28] Jiang H, Dong Y, Dong Y, et al. Probabilistic electricity price forecasting by integrating interpretable model[J]. Technological Forecasting and Social Change, 2025, 210: 123846.

[29] Xu F, Teng X, Lu J, et al. Prediction of day-ahead electricity price based on N-BEATSx model optimized by SSA considering coupling between features[C]//The Purple Mountain Forum on Smart Grid Protection and Control. Singapore: Springer Nature Singapore, 2022: 178-194.

[30] Tucker, Howard G. A graduate course in probability. Courier Corporation, 2013.

[31] Robbins H, Siegmund D. A convergence theorem for non negative almost supermartingales and some applications[M]//Optimizing methods in statistics. Academic Press, 1971: 233-257.

[32] Lin H, Wang C. DIGWO-N-BEATS: An evolutionary time series prediction method for situation prediction[J]. Information Sciences, 2024, 664: 120316.

[33] Wolpert D H, Macready W G. No free lunch theorems for optimization[J]. IEEE transactions on evolutionary computation, 1997, 1(1): 67-82.

[34] Rashedi E, Nezamabadi-Pour H, Saryazdi S. GSA: a gravitational search algorithm[J]. Information sciences, 2009, 179(13): 2232-2248.

[35] Mirjalili S. SCA: a sine cosine algorithm for solving optimization problems[J]. Knowledge-based systems, 2016, 96: 120-133.

[36] Das B, Mukherjee V, Das D. Student psychology based optimization algorithm: A new population based optimization algorithm for solving optimization problems[J]. Advances in Engineering software, 2020, 146: 102804.

[37] Mirjalili S, Gandomi A H, Mirjalili S Z, et al. Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems[J]. Advances in engineering software, 2017, 114: 163-191.

[38] Simon D. Biogeography-based optimization[J]. IEEE transactions on evolutionary computation, 2008, 12(6): 702-713.

[39] Oreshkin B N, Amini A, Coyle L, et al. FC-GAGA: Fully connected gated graph architecture for spatio-temporal traffic forecasting[C]//Proceedings of the AAAI conference on artificial intelligence. 2021, 35(10): 9233-9241.

[40] Liu Y, Hu T, Zhang H, et al. itransformer: Inverted transformers are effective for time series forecasting[J]. arXiv preprint arXiv:2310.06625, 2023.

[41] Yang, Aimin, et al. "A Network Security Situation Prediction for Consumer Data in the Internet of Things Using Variational Mode Decomposition (VMD) and fused CNN-BiLSTM-Attention." IEEE Transactions on Consumer Electronics (2023).

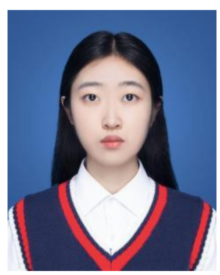
[42] Demin Gao, Haoyu Wang, Shuai Wang, Weizheng Wang, etal, WiLo: Long-Range Cross-Technology Communication from Wi-Fi to LoRa[J], IEEE Transactions on Communications, 2024, DOI:10.1109/TCOMM.2024.3461574

Huang Rui, born in 1999 in Tianjin, China, received her Bachelor of Engineering degree from Tianjin Normal University, Tianjin, China, in 2022. Currently, she is pursuing her Master of Engineering degree in computer technology at Tianjin University of Technology, Tianjin, China. Her research interests include machine learning, data mining, heuristic algorithms, and cyberspace security.

Chundong Wang, a professor and doctoral supervisor, is a member of the China Computer Federation (CCF). He obtained his Master of Engineering degree from Nankai University in 2002 and his Ph.D. in Engineering from Nankai University in 2007. Currently, he is employed at the School of Computer Science and Technology, Tianjin University of Technology. His primary research interests include big data and intelligent computing security, Internet of Things (IoT) and edge computing security, network security situational awareness, machine learning, and related fields.

Hao Lin was born in Tianjin, China, in 1995. He received the B.A. degree in engineering from Tianjin University of Technology and Education, Tianjin, in 2018, and the M.A. degree in engineering from Inner Mongolia University of Technology, Hohhot, China, in 2021. He is currently pursuing the Ph.D. degree in engineering with Tianjin University of Technology, Tianjin.,His research interests include cyberspace security, social engineering attack, and data mining.

Haolong Zhang was born in Henan Province, China, in 2000. He received his Bachelor of Engineering degree from Henan University of Technology in 2022 and his Master's degree in Electronic Information Engineering from Tianjin University of Technology in 2025. His research interests include cybersecurity, data mining, facial expression transfer, and related fields.



Hongjing Ma was born in Hebei Province, China, in 2001. She received the B.A. degree in engineering from Hebei University of Geosciences in 2023. She is currently pursuing the master degree in engineering with Tianjin University of Technology. Her research interests include data security, attack prevention, and trajectory protection.

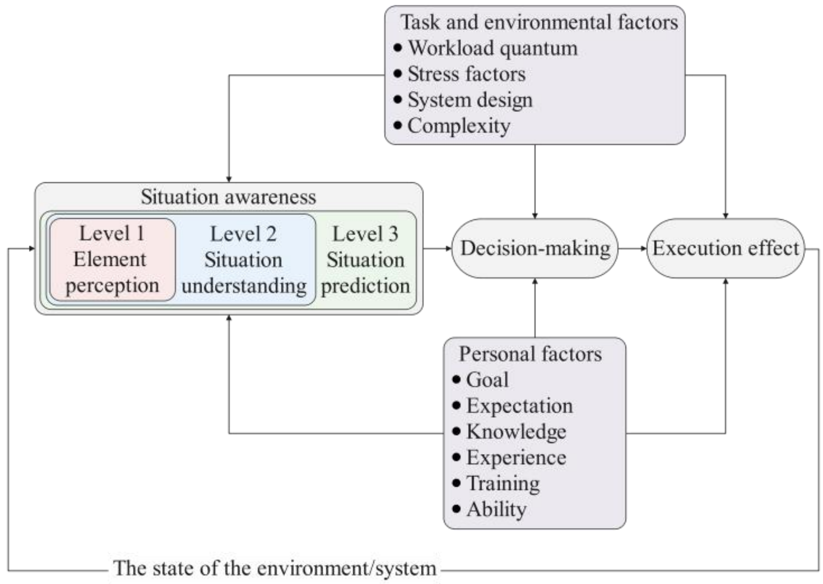


Fig. 1. Situation awareness plays a crucial role in risk assessment.

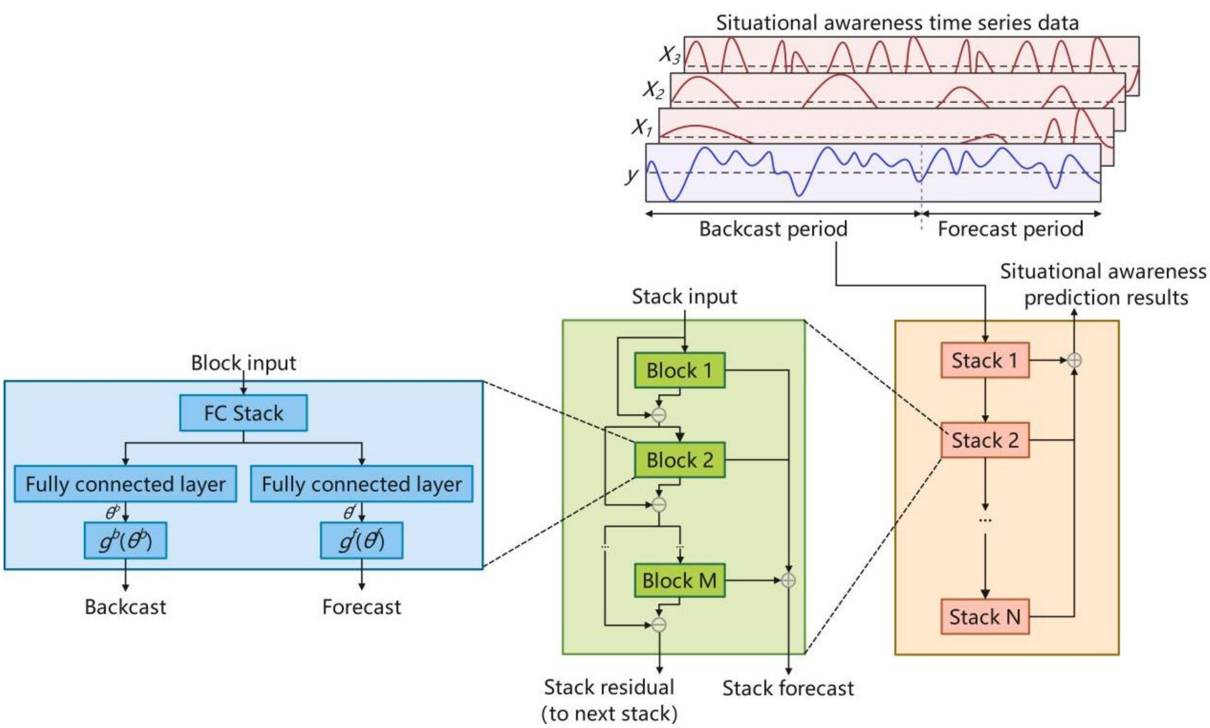


Fig. 2. N-BEATSx Overall Framework Diagram

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2025.3581958

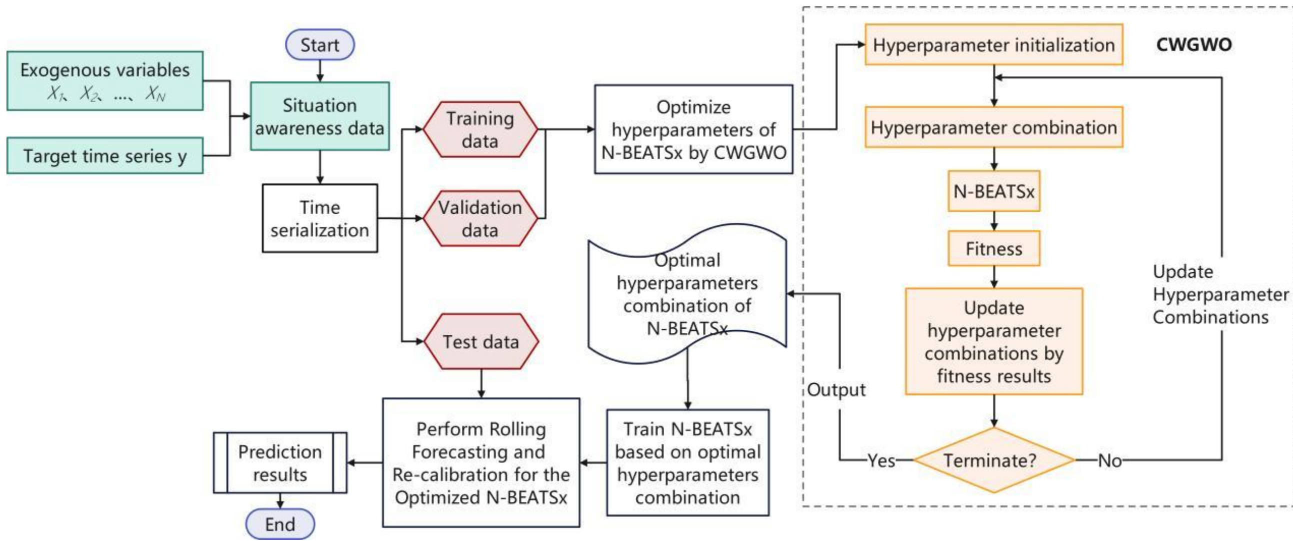


Fig. 3. CWGWO-N-BEATSx Overall Flowchart

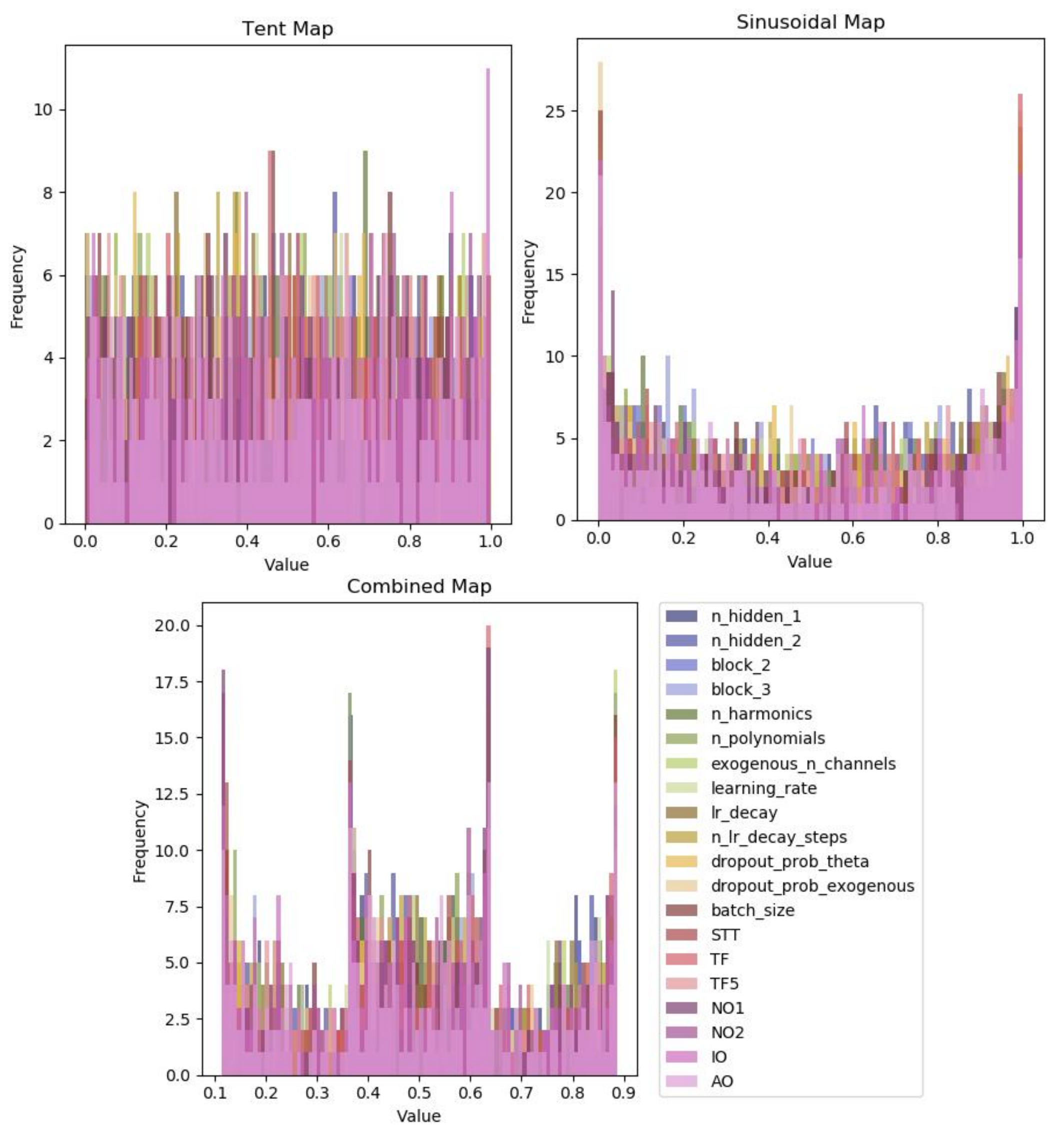


Fig. 4. Tent mapping, sinusoidal mapping, and "Tent-sinusoidal" mapping of the initialised particle distributions of the populations

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2025.3581958

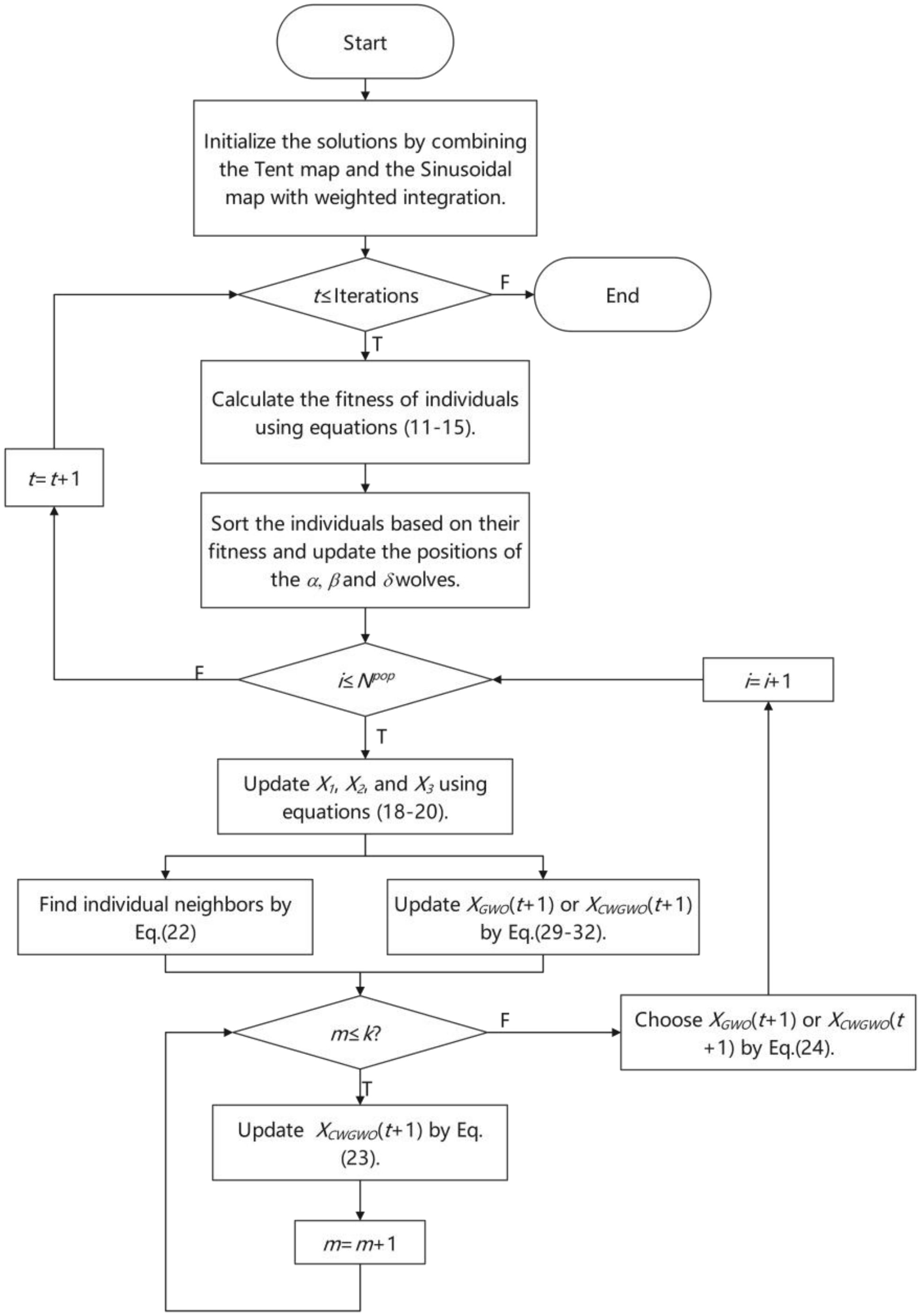


Fig. 5. Flowchart of CWGWO algorithm

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2025.3581958

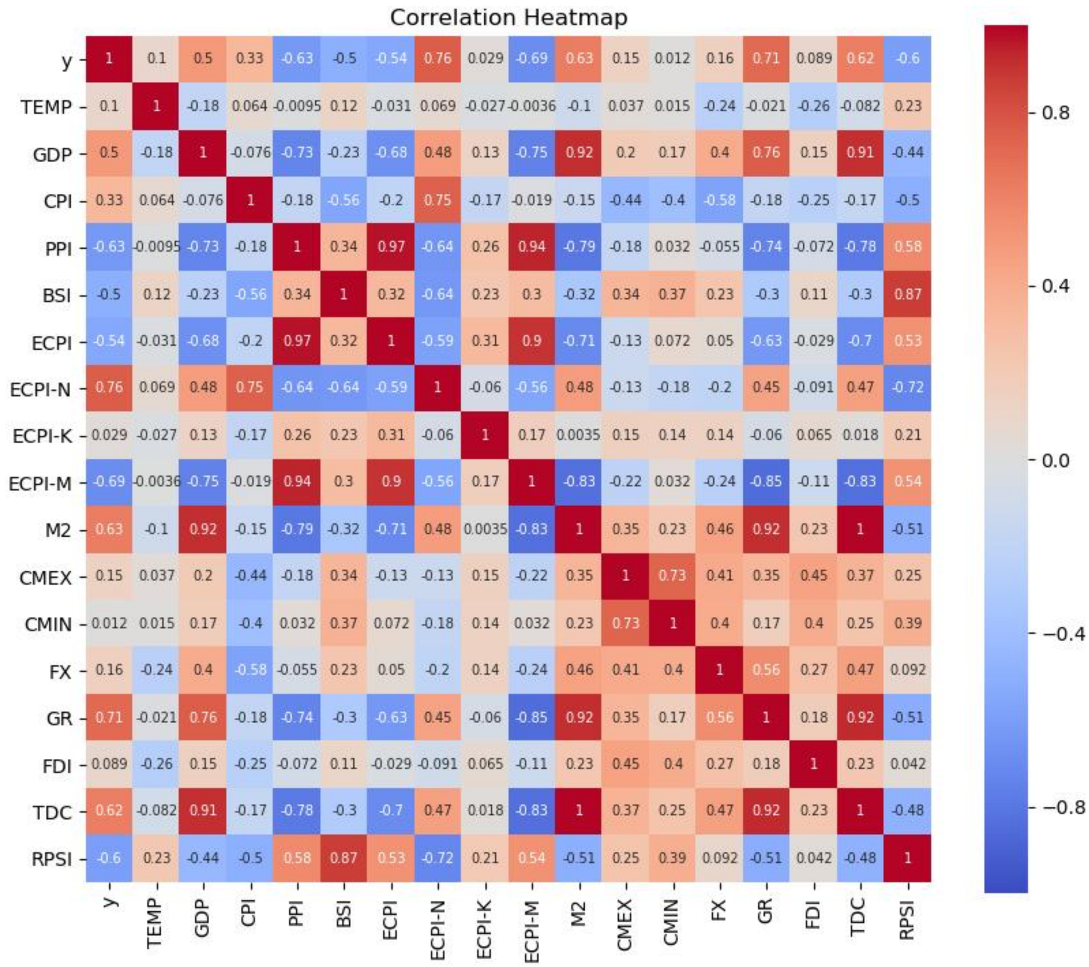


Fig. 6. Correlation of raw exogenous variables in the Cncert dataset

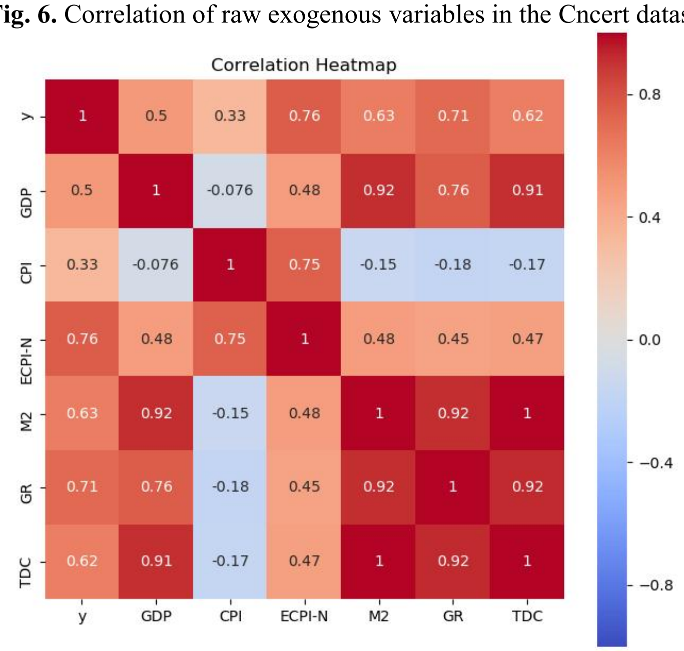
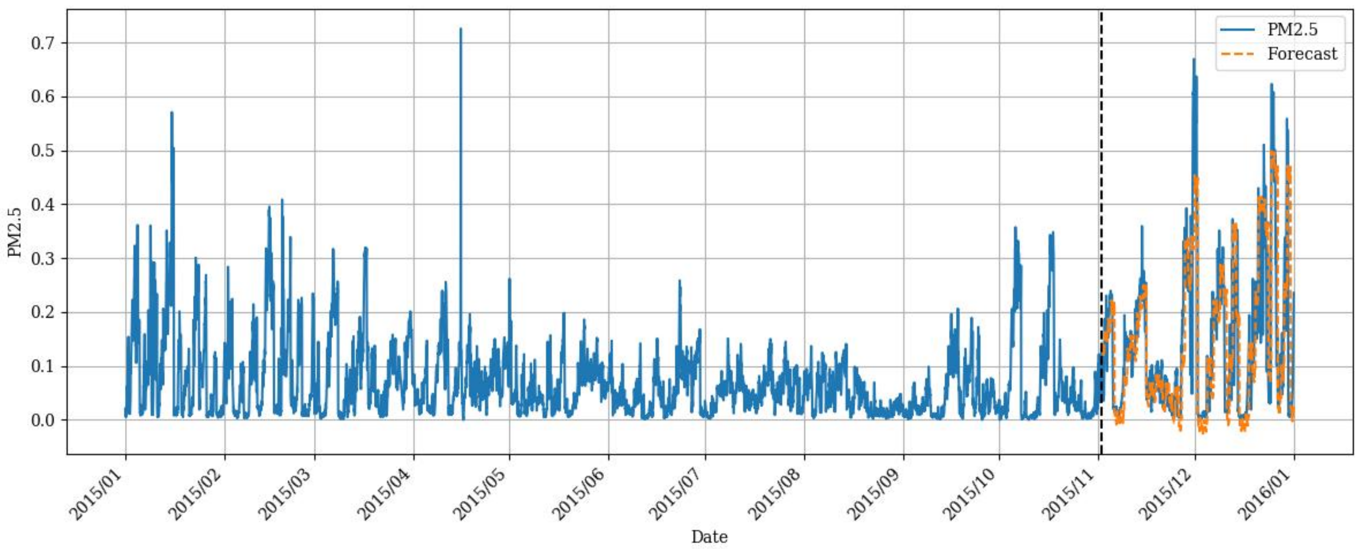
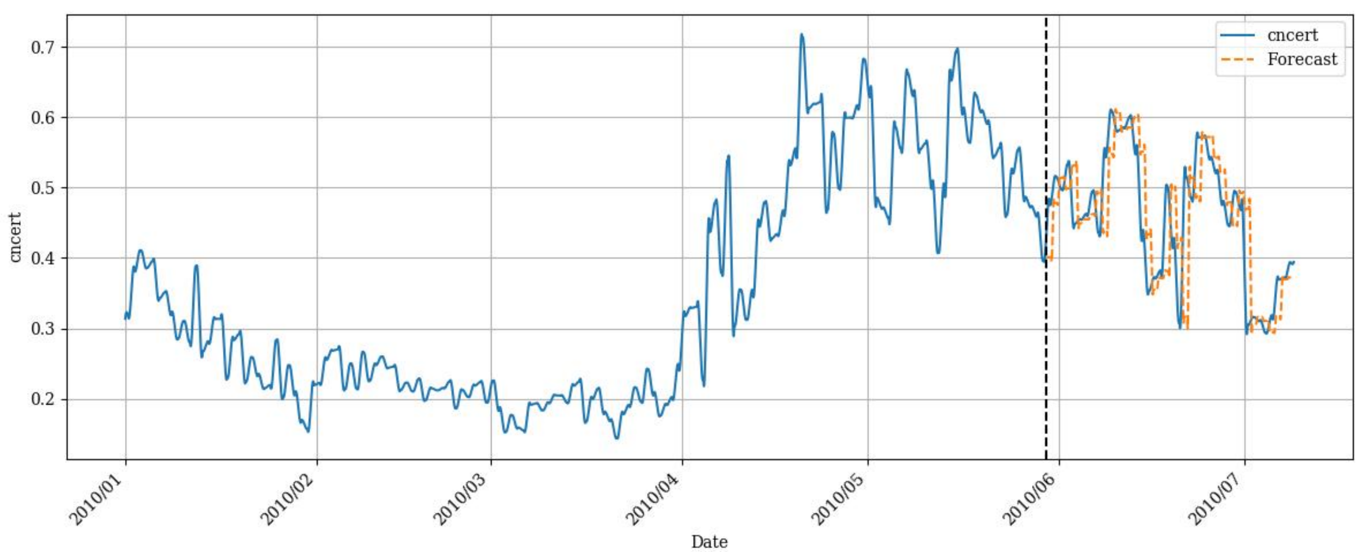


Fig. 7. Correlations after screening for exogenous variables in the Cncert dataset



a. BJ_2015



b. Cncert

Fig. 8. Visualization of CWGWO-N-BEATSx Model Performance in Situation Prediction

TABLE I

Provides an overview of the various prediction techniques utilized in situation prediction.

Method type	Reference	Year	Field	Prediction Method
Neural Network	[13]	2022	Network Security situation awareness	SA+HHGA+RBF
	[16]	2022	Network Security Prediction	SSA+ELM
	[18]	2023	Air Combat Situation Assessment	AE+ELM
	[19]	2024	Network Security Prediction	PSO+ABiLSTM
Hybrid Architecture	[11]	2020	Traffic Volume Forecasting	ToGCN+Seq2Seq
	[12]	2021	Flood Forecasting	CNN+LSTM
	[14]	2022	The Spread of COVID-19	ARIMA+ LSTM
	[15]	2022	Traffic Situation in Terminal Area	CNN+GRU
	[17]	2023	Cybersecurity Trend Forecasting	BiLSTM+VAR+MSCNN
	[20]	2024	Travel Demand Forecasting	GCN+GRU

TABLE II
Overview of the key hyperparameters that have a substantial impact on the performance of N-BEATSx.

Number of first hidden layer neurons in each block: size2	size2 ∈ [10,512], size2 ∈ Z+
Number of second hidden layer neurons in each block: size3	size3 ∈ [10,512], size3 ∈ Z+
Number of blocks in the second stack: B2	B2 ∈ [1,2], B2 ∈ Z+
Number of blocks in the third stack: B3	B3 ∈ [1,2], B3 ∈ Z+
stack type: STT	STT ∈ [1,2,3,4,5]
Normalisation method for the target variable y: NO1	NO1 ∈ [1,2,3]
Normalisation methods for exogenous variables: NO2	NO2 ∈ [1,2,3]
Batch size: BS(batch_size)	BS ∈ [256,512], BS ∈ Z+
Weight initialisation method: IO	IO ∈ [1,2,3]
Activation function of the block: AO	AO ∈ [1,2,3,4]
Whether batch normalisation is used: TF	TF ∈ [0,1], TF ∈ Z
Number of harmonic terms per stack: Nhr	Nhr ∈ [0,1]
Number of polynomial terms per stack: Npol	Npol ∈ [0,1]
Number of channels for exogenous variables: ENC	ENC ∈ [1,10], ENC ∈ Z+
Learning rate: LR	LR ∈ [0.00001,1]
Learning rate decay factor: LD	LD ∈ [0,1]
Learning rate decay steps: NLDS	NLDS ∈ [1,2,3,4,5]
N-BEATS base dropout probability: DPT	DPT ∈ [0,1]
Exogenous variable dropout probability: DPE	DPE ∈ [0,0.5]
Whether the input contains data from the first 3 time steps of y: TF1	TF1 ∈ [0,1], TF1 ∈ Z
Whether the input contains data for the first 4 time steps of y: TF2	TF2 ∈ [0,1], TF2 ∈ Z
Whether the input contains data for the first 8 time steps of y: TF3	TF3 ∈ [0,1], TF3 ∈ Z
Whether the input contains data from the first 1 time step of Ex1: TF4	TF4 ∈ [0,1], TF4 ∈ Z
Whether the input contains data from the first 2 time steps of Ex1: TF5	TF5 ∈ [0,1], TF5 ∈ Z
Whether the input contains data from the first 8 time steps of Ex1: TF6	TF6 ∈ [0,1], TF6 ∈ Z

TABLE III
Dataset description

Dataset	Description	Number of sample	Number of exogenous	Time interval tΔ
Cncert	the number of five security threats	190	6	week
BJ	PM2.5 index	52560	6	hour

TABLE IV
Overview of the key hyperparameters that have a substantial impact on the performance of N-BEATSx.

	BJ					Cncert				
	MAE	SMAPE	MSE	Total_P	fitness	MAE	SMAPE	MSE	Total_P	fitness
PSO	3.633	50.7029	0.0043	0.015	3.649	5.125	12.3639	0.0054	0.0113	5.136

GSA	3.637	50.7804	0.0043	0.017	3.654	5.154	12.3099	0.0054	0.0313	5.185
GA	3.679	51.0736	0.0044	0.086	3.765	5.173	12.3284	0.0054	0.0114	5.184
SCA	3.679	51.0736	0.0044	0.086	3.765	5.173	12.3284	0.0054	0.0114	5.184
MVO	3.658	50.9323	0.0043	0.051	3.708	5.162	12.3174	0.0054	0.0221	5.184
SSA	2.814	49.3765	0.0020	0.010	2.913	5.157	12.3584	0.0054	0.0334	5.190
BBO	3.153	50.0275	0.0026	0.038	3.191	5.136	12.2937	0.0054	0.0152	5.151
GWO	3.274	50.2120	0.0038	0.023	3.297	5.096	12.2083	0.0052	0.0186	5.115
CWGWO	2.5356	48.7751	0.0017	0.017	2.5528	4.6901	11.1479	0.0048	0.0122	4.7023

Table V

Values of hyperparameters and dictionary lengths of exogenous variables after CWGWO optimisation

	BJ	Cncert
size2	11	10
size3	10	10
B2	1	1
B3	1	2
stack's type	['identity']	['identity']
lenex	10	7

Table VI

Ablation experiment results for DIGWO

	BJ				Cncert			
	MAE	SMAPE	MSE	fitness	MAE	SMAPE	MSE	fitness
iGWO	2.651	49.9962	0.0019	2.712	4.915	11.4332	0.0050	4.940
iGWO_W	2.627	49.3268	0.0018	2.652	4.857	11.3049	0.0049	4.892
iGWO_T	2.596	48.9587	0.0018	2.613	4.705	11.2248	0.0049	7.758
iGWO_T_W	2.5356	48.7751	0.0017	2.5528	4.6901	11.1479	0.0048	4.7023

Table VII

The contrast between the state-of-the-art research and our CWGWO-N-BEATSx.

		iTransformer	CNN+BiLSTM+Attention	iGWO-N-BEATS	N-BEATSx	SSA-N-BEATSx	CWGWO-N-BEATSx
BJ	MAE	5.220	3.029	2.814	2.693	2.614	2.5356
	SMAPE	68.5344	53.1258	51.3103	50.2637	49.3765	48.7751
	MSE	0.0130	0.0032	0.0029	0.0024	0.0020	0.0017
Cncert	MAE	7.728	6.728	5.296	5.215	5.157	4.6901
	SMAPE	15.6306	13.3894	13.012	12.6359	12.3584	11.1479
	MSE	0.0092	0.0065	0.0059	0.0056	0.0054	0.0048

Table VIII
Performance Comparison with Mean ± Std (on BJ and Cncert Datasets)

Model	Dataset	MAE (mean ± std)	SMAPE (mean ± std)	MSE (mean ± std)
iTransformer	BJ	5.220 ± 0.045	68.534 ± 0.250	0.0130 ± 0.0004
CNN+BiLSTM+Attention	BJ	3.029 ± 0.031	53.126 ± 0.201	0.0032 ± 0.0001
iGWO-N-BEATS	BJ	2.814 ± 0.031	51.310 ± 0.231	0.0029 ± 0.0002
N-BEATSx	BJ	2.693 ± 0.035	50.264 ± 0.226	0.0024 ± 0.0001
SSA-N-BEATSx	BJ	2.614 ± 0.030	49.377 ± 0.210	0.0020 ± 0.0001
CWGWO-N-BEATSx	BJ	2.536 ± 0.021	48.775 ± 0.190	0.0017 ± 0.0001
iTransformer	Cncert	7.728 ± 0.060	15.631 ± 0.210	0.0092 ± 0.0004
CNN+BiLSTM+Attention	Cncert	6.728 ± 0.038	13.389 ± 0.172	0.0065 ± 0.0001
iGWO-N-BEATS	Cncert	5.296 ± 0.052	13.012 ± 0.163	0.0059 ± 0.0001
N-BEATSx	Cncert	5.215 ± 0.034	12.636 ± 0.174	0.0056 ± 0.0002
SSA-N-BEATSx	Cncert	5.157 ± 0.028	12.358 ± 0.168	0.0054 ± 0.0002
CWGWO-N-BEATSx	Cncert	4.690 ± 0.019	11.148 ± 0.160	0.0048 ± 0.0001