IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 14 February 2024, accepted 17 February 2024, date of publication 20 February 2024, date of current version 7 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3367829

RESEARCH ARTICLE

Bayesian Neural Network-Based Equipment Operational Trend Prediction Method Using Channel Attention Mechanism

CHANG MING-YU, TIAN LE, AND MAOZU GUO
Beijing Key Laboratory for Research on Intelligent Processing Methods for Construction Big Data, Beijing University of Civil Engineering and Architecture, Beijing 100044, China

Corresponding author: Maozu Guo (guomaozu@bucea.edu.cn)

This work was supported in part by the Ministry of Science and Technology of China under Grant 2021YFF0306303, and in part by the National Natural Science Foundation of China under Grant 62271036.

ABSTRACT This paper proposes a Bayesian neural network method for predicting equipment operational trends based on a channel attention mechanism. Traditional time series prediction methods have limitations in handling complex data and nonlinear relationships. To enhance prediction accuracy and stability, the paper introduces a channel attention mechanism to capture crucial features and contextual information within the data. This mechanism automatically adjusts the weights of feature channels to focus on the influence of key features. By leveraging the advantages of Bayesian neural networks, the model undergoes multiple updates and adjustments while considering uncertainty factors, progressively improving the predictive outcomes. In experiments, the paper utilizes power transformer data from a Kaggle public dataset and a substantial amount of temporary facility equipment data from the Winter Olympics site, comparing the performance against other commonly used prediction methods. Results demonstrate the significant superiority of the Bayesian neural network method with channel attention mechanism in equipment trend prediction, outperforming traditional time series models and other commonly used methods.

INDEX TERMS Equipment operational trend prediction, neural networks, Bayesian neural networks, attention mechanism.

I. INTRODUCTION

With the continuous advancement in technology and societal progress, there's a rising trend towards higher levels of intelligence and automation in equipment. This evolution has led to the generation and accumulation of vast amounts of time-series data. These data streams originate from various devices and sensors, encompassing industrial machinery, medical instruments, transportation systems, energy equipment, and more. They document equipment states, performances, and operational conditions, harboring vital information essential for tasks such as equipment health monitoring, fault prediction, resource optimization, and decision support.

Addressing the prediction of equipment trends often confronts several challenges. Firstly, time-series data commonly contains noise and uncertainty, intensifying the complexity of precise predictions. Secondly, different devices might possess distinct characteristics and operational patterns, making generality and flexibility pivotal requirements for prediction methods. Lastly, comprehending the uncertainty and credibility of predictions is crucial for engineers and decision-makers responsible for equipment maintenance and management.

To tackle these challenges, this paper presents a novel approach for predicting equipment trends based on the channel attention mechanism [1], [22], [23] and Bayesian neural networks(BNN) [2], [25]. The channel attention mechanism enables the model to autonomously select relevant feature channels, concentrating more attention on channels pertinent to the current task. This enhances the model's capability to extract information from equipment data, reducing interference from redundant information, thus

The associate editor coordinating the review of this manuscript and approving it for publication was Mohamed M. A. Moustafa.

enabling better capture of useful data insights. Bayesian neural networks introduce probabilistic modeling [3], facilitating the model to estimate and manage prediction uncertainties more effectively. By introducing probability distributions to represent weights and parameters instead of singular deterministic values, the model can quantify its confidence in predictions, significantly aiding decision-making and risk management. By amalgamating these two methods, the paper aims to provide a predictive framework adaptable to diverse equipment and data characteristics, accurately predicting equipment trends while offering credibility estimation.

The structure of the paper is as follows:Section II reviews relevant work pertaining to equipment trend prediction, encompassing methods in time-series forecasting, applications of attention mechanisms, and the evolution of Bayesian neural networks. Section III elaborates on the proposed methodology, detailing the design of the channel attention mechanism and the modeling of Bayesian neural networks. Section IV presents experimental results and performance evaluations to validate the method's effectiveness. Section V summarizes the primary findings of the research and explores future research directions.

The significance of this paper's work lies in offering a fresh perspective and methodological approaches for the advancement and application of equipment trend prediction. This endeavor aims to propel improvements in equipment performance and reliability across industries such as industrial, medical, and other sectors, providing decision-makers with more reliable data support.

II. RELATED WORK

Mainstream approaches for predicting equipment operational trends include traditional time series forecasting [4], intelligent algorithms [5], and neural networks [6]. Many devices exhibit intricate structures and diverse types, prone to faults and safety issues during operation. Hence, intelligent detection, diagnosis, and prediction of operational trends in devices become imperative.

Over the past few decades, various methods have emerged in the field of time series analysis for capturing and predicting trends. Among these, exponential smoothing [33] stands out as a classical technique with significant success in prior research. By assigning decreasing weights to past observations, exponential smoothing effectively captures trends and seasonal variations in time series data. Previous studies have demonstrated its widespread application in forecasting future trends in areas such as finance, sales, and supply chain management.

On another note, correlation analysis has long been a pivotal issue in data science and statistics. Kendall's Tau [34], as a non-parametric measure, has found extensive application in correlational studies across different domains. In contrast to traditional Pearson correlation, Kendall's Tau is particularly suitable for ordered data and is robust to outliers. In prior research, researchers have successfully employed Kendall's Tau to analyze trends in markets, medical data, and relationships in social sciences.

With the advancement of deep learning, neural networks have garnered substantial attention in time series analysis. The Recurrent Trend Predictive Neural Network (RTPNN) [35], as a specific neural network architecture, possesses the capability to capture complex trends in time series data. Studies indicate that RTPNN not only adapts to nonlinear relationships in the data but also handles long-term dependencies in sequential data, showcasing remarkable performance in tasks such as stock price prediction and weather trend forecasting.

Traditional time series forecasting methods focus on predicting time-based data. Classic methodologies such as Autoregressive Integrated Moving Average (ARIMA) models [7] and exponential smoothing models [8] have been employed for equipment trend prediction. They estimate future trends by analyzing patterns in historical data. However, these methods exhibit limitations in handling nonlinear data and time series with complex dependencies. Recurrent Neural Networks (RNNs) [9] and Long Short-Term Memory networks (LSTMs) [10] represent deep learning methods aimed at capturing long-term dependencies in data, widely applied in equipment trend prediction. Despite their significant performance enhancements in many tasks, they still face challenges dealing with strong nonlinearity and high uncertainty in data.

In recent years, with rapid advancements in computing technology, deep neural networks have gained prominence in predicting equipment operational trends. For instance, deep neural networks (DNNs) [11], Convolutional Neural Networks (CNNs) [12], recurrent neural networks (RNNs), and composite models, among other neural network architectures, have been applied to tasks such as equipment failure prediction and operational trend prediction. Zhang et al. [13] utilized an LSTM-DNN network for high-resolution short-term precipitation forecasting. Due to their high adaptability and robust feature extraction capabilities, deep neural networks have exhibited excellent performance in prediction tasks across various domains such as engineering, strength prediction, mechanical control, geological disasters, and more. Through extensive data training, neural networks can learn intricate nonlinear relationships, thus making accurate predictions.

The Bayesian neural network [14] is a special type of neural network that introduces uncertainty to the weights of the neural network. This ability to model uncertainty enables Bayesian neural networks to exhibit superior generalization performance when dealing with complex, high-dimensional data. In comparison to other neural network approaches like deep learning, Bayesian neural networks demonstrate significant advantages in prediction accuracy and model robustness. Bayesian methods have showcased strong predictive performance in many practical problems. For instance, in equipment trend prediction, Bayesian neural networks

can effectively handle such problems. For example, they can be employed to predict equipment lifespan, performance degradation, or failure times [28], [29], [30]. By integrating Bayesian methods, predictive models can provide probability distributions for prediction outputs, aiding in further evaluating prediction uncertainties. This means the model can understand the confidence level of prediction results, allowing for more cautious and precise decision-making. In the field of artificial intelligence, Bayesian inference is used to establish models and handle various prediction problems. For instance, in pattern recognition, it's used in areas like image recognition, speech recognition, text classification [31], to establish models and utilize Bayesian updating to calculate the match between input data and the model. In natural language processing, Bayesian networks are used to handle ambiguity and polysemy in natural language [32], as well as determine word meanings based on context. Moreover, Bayesian inference is employed in fields such as data mining, predictive analysis, and machine learning.

Another important characteristic of Bayesian neural networks is their ability to handle noisy data. In real-life scenarios, many data instances come with noise, which might result from measurement errors, environmental interference, or other factors. Bayesian neural networks, through their unique probabilistic modeling capabilities, can effectively suppress the impact of noise on prediction results, thereby offering more accurate and reliable predictions.

The attention mechanism [15] was initially applied in machine translation and reading comprehension and has subsequently found widespread application in other domains, including natural language processing and computer vision. Neural network architectures with attention layers, known as the Transformer architecture [16], [24], exhibit better performance in nearly all language processing tasks. The use of Transformer networks for masked language modeling has resulted in groundbreaking pre-training models, such as Bidirectional Encoder Representations from Transformers (BERT) [17], [27]. The attention mechanism is a crucial component of Transformer networks and has become an indispensable part of natural language processing, significantly impacting language applications. In the context of equipment trend prediction, introducing attention mechanisms can assist the model in focusing on important features and allocating weights to each feature. This can improve predictive performance and enhance modeling capabilities for complex problems.

Through the review of classical time series methods and the exploration of applications of RNNs, LSTMs, attention mechanisms, and Bayesian neural networks, a deeper understanding of equipment trend prediction has been attained. These methodologies provide a robust theoretical foundation and practical applications for equipment trend prediction, aiding in achieving better predictive results in real-world problems.

III. ALGORITHM MODEL

The paper utilizes a Bayesian neural network with an integrated attention mechanism to forecast equipment operational trends. The data is sourced from a Kaggle competition's publicly available dataset, containing various conditions of power transformers along with corresponding health indices. These data were tested on a real dataset collected from the accelerometer sensors on temporary grandstand supports at the Yanqing Alpine Skiing Center during the Beijing Winter Olympics, capturing a type of non-linear and non-stationary signal.

The model introduced in this paper employs a Bayesian neural network that incorporates an attention mechanism, allowing for modeling the uncertainty of prediction outcomes. By integrating a channel attention mechanism into the Bayesian Neural Network (BNN), the model automatically adjusts the importance of each channel, thereby enhancing its performance. This mechanism aids in capturing the correlations and significance between features, thereby improving the accuracy of regression tasks. In certain datasets, different channels' features might hold varying degrees of importance. The channel attention mechanism dynamically adjusts channel weights based on data distribution, enabling the model to effectively handle imbalanced features.

This model comprises three components: a preprocessing layer, a Bayesian network layer integrated with the channel attention mechanism, and a prediction layer. The data undergoes preprocessing operations in the initial layer, then enters the Bayesian network layer with the integrated channel attention mechanism for feature extraction. Subsequently, a one-dimensional convolutional layer operates on the data's 14 channels, transferring features to the Bayesian linear layer to produce the final output. The overall framework of the model is illustrated in Figure 1.

A. PREPROCESSING LAYER

The preprocessing layer comprises a one-dimensional convolutional layer. Let's consider the initial input tensor as X, where $X \in R^{B\times L}$, with B representing the batch size, C representing the number of channels (typically denoting different features or filters in convolutional neural networks), and L representing the input sequence length. To ensure proper data preprocessing using the convolutional neural network, an unsqueeze operation might be necessary to alter the dimension of the data input. Therefore, before preprocessing, it might be required to insert a new dimension into the input X, modifying its dimensions to $X \in R^{B\times C\times L}$

Therefore, before preprocessing, an additional dimension is inserted into the input X's dimension, transforming its shape into $(B, 1, L)$.

$$y_{i,j,k} = x_{i,k} \tag{1}$$

Following the aforementioned equation, where the shape of X becomes $(B, 1, L)$, the input X undergoes a

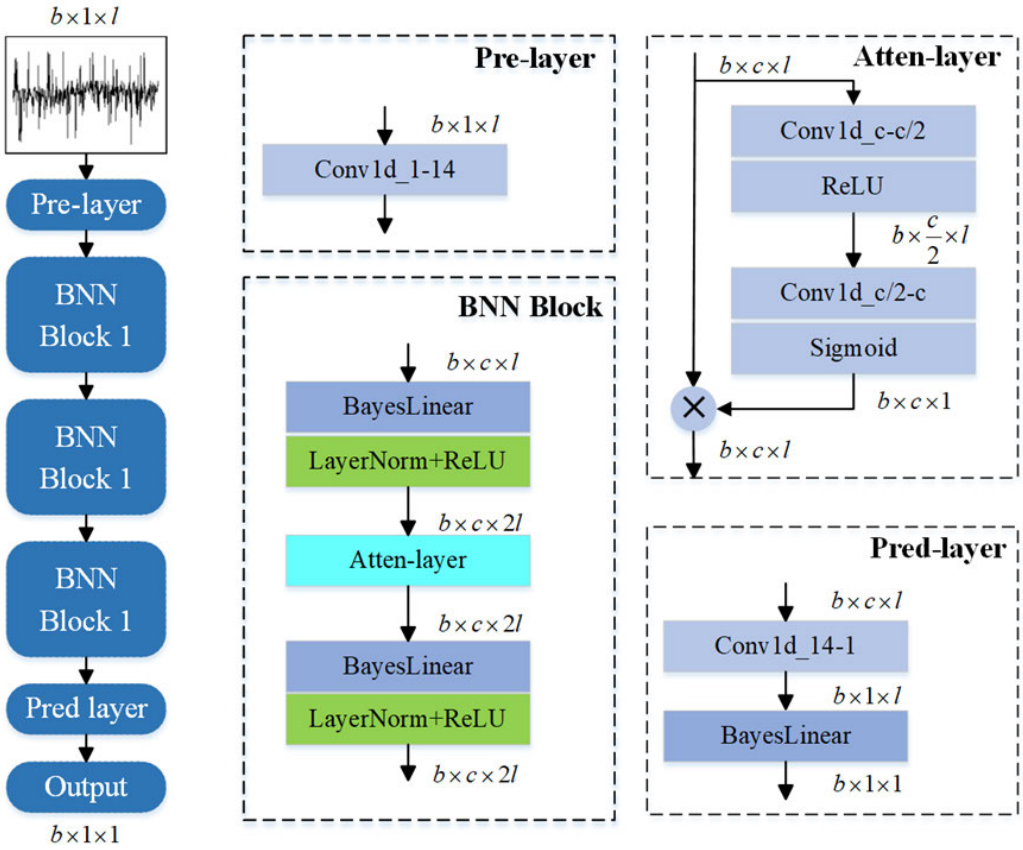

FIGURE 1. Bayesian neural network framework with channel attention mechanism.

one-dimensional convolution operation, generating an output y according to the following formula:

$$y_i = \sum_{j=0}^{K-1} \left(x_{c,i+j} * \omega_j\right) + b \tag{2}$$

where y_i represents the i-th element of the output feature map; c denotes the channel index, indicating the current channel being processed; i denotes the position index in the output feature map; j signifies the element index within the convolutional kernel, ranging from 0 to $K-1$, where K is the size of the convolutional kernel; $x_{c,i+j}$ denotes the element in the input data, where c represents the channel index, and $i+j$ denotes the position within the input sequence; ω_j represents the weight in the convolutional kernel, where j denotes the weight index; and b represents the bias term used to adjust the output.

The dataset contains a total of fourteen columns of data characterizing various conditions (e.g., hydrogen, oxygen, etc.) and corresponding health indices of power transformers.After the aforementioned preprocessing, the original data was transformed from its $2D$ shape of $(B, 14)$ to a three-dimensional feature representation of $(B, 14, 14)$, effectively mapping the data into a higher-dimensional feature space.

B. INCORPORATING CHANNEL ATTENTION MECHANISM INTO BAYESIAN NEURAL NETWORK

The Bayesian neural network, designed based on the channel attention mechanism in this paper, comprises three parts: the preprocessing layer, the attention layer, and the postprocessing layer.

1) PREPROCESSING LAYER

A Bayesian neural network introduces probability distributions into the model parameters to reflect their uncertainty. In traditional neural networks, weight parameters are typically considered deterministic values. However, in Bayesian neural networks, these parameters are treated as random variables and described using probability distributions to capture their uncertainty. This capability enables Bayesian inference, offering a more comprehensive estimation of uncertainty. Sending preprocessed data into a Bayesian neural network equipped with a channel attention mechanism can enhance the model's focus on specific channels, thereby improving its capacity to learn and represent data features. This fusion effectively utilizes the Bayesian neural network to model parameter uncertainty while dynamically adjusting the weights of different channels through the channel attention mechanism, thus better capturing critical information and patterns within the data.

Firstly, a neural network model is defined:

$$y = f(x; W) \tag{3}$$

where x is the input, y is the output, and W represents the network weights.

In this process, the data initially passes through a linear layer with Bayesian weights, then proceeds through batch normalization and the ReLU activation function, mapping input features to output features. This series of steps effectively handles data, models the uncertainty of parameters using Bayesian weights, and processes data through batch normalization and activation functions to obtain a more representative and robust feature representation. Bayesian weights imply uncertainty in the layer's weights, introducing a prior distribution for each weight before training. Typically, a normal distribution is used as the prior distribution, expressed as follows:

$$P(W) \sim N\left(\theta \mid \mu, \sigma^2\right) \tag{4}$$

where θ is the random variable representing the weights; μ is the prior mean of the weights; σ is the prior variance of the weights.

By observing the training data, updating the distribution of weights leads to the posterior distribution. According to Bayes' theorem, the posterior distribution can be represented as follows:

$$P(\theta \mid D) = \frac{P(D \mid \theta)P(\theta)}{P(D)} \tag{5}$$

where D represents the observed data, $P(\theta \mid D)$ is the posterior distribution of the weights, $P(D \mid \theta)$ is the likelihood of the data given the weights, and $P(D)$ is the marginal probability of the data.

The tensor output with uncertainty from the Bayesian linear layer is normalized, scaling the features to a distribution with a mean of 0 and a standard deviation of 1. Then, the ReLU activation function sets negative values to zero while preserving positive values from the normalized data, introducing non-linearity to enable the network to learn complex functions.

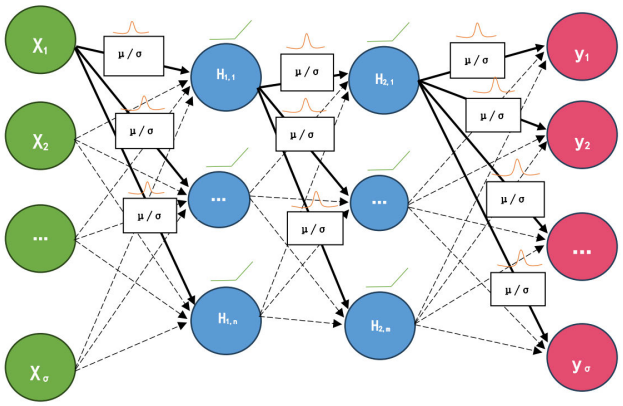

FIGURE 2. Bayesian neural network.

2) CHANNEL ATTENTION MECHANISM

After a series of preprocessing steps, the data is fed into the proposed channel attention module of this study. Comprising one-dimensional convolution, ReLU activation function, and Sigmoid activation function, this module generates attention weights through convolutional operations. These weights are utilized to weight information from different positions within the input data, allowing the model to better focus on specific parts of the input during both pre- and post-processing stages. This enhances features related to fault information, suppresses interfering features, and improves network performance. Overall, this mechanism initially reduces the number of channels in the input data using a one-dimensional convolutional layer, then restores the channel count to the original number through another one-dimensional convolutional operation.

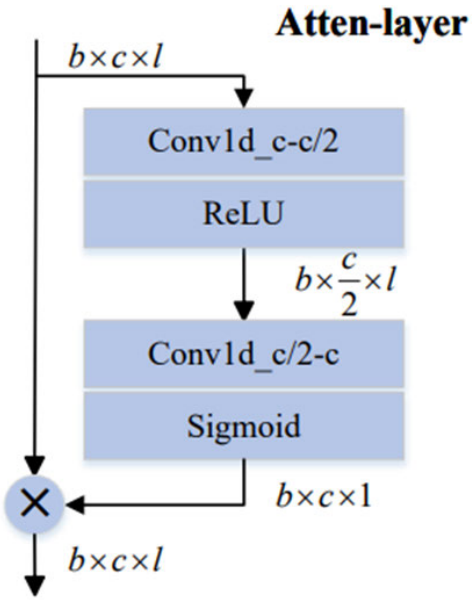

FIGURE 3. Framework of the channel attention mechanism.

In this process, the ReLU activation function is used to introduce non-linearity, while the Sigmoid activation function is utilized to confine the output within the range of [1, 0], representing attention weights. The formulas are represented as follows

$$x_1 = Conv1D\,(x,\ W_1,\ b_1) \tag{6}$$

$$x_2 = ReLU\ (x_1) = \max\,(0, x_1) \tag{7}$$

$$x_3 = Conv1D\,(x_2, W_2, b_2) \tag{8}$$

$$x_4 = Sigmoid\ (x_3) = \frac{1}{1 + \exp\,(-x_3)} \tag{9}$$

3) POST-PROCESSING LAYER

Finally, the data is passed through a post-processing layer to further refine the feature representations obtained from the preceding processing layers, thereby generating the final

output of the model. The post-processing layer introduces additional non-linearity, enhancing the model's representational capacity.The post-processing layer comprises a Bayesian Linear layer, a normalization layer, and a ReLU activation function. The Bayesian Linear layer maps the output features from the preceding processing layers to an equal number of output features, further refining the features from the previous layers to better align with the model's target tasks.Subsequently, the normalization layer standardizes the output from the post-processing layer, ensuring that each feature dimension of the output possesses similar statistical properties.Finally, the ReLU activation function is applied to the output data, utilizing rectified linear units to set all negative values to zero while retaining positive values.

C. PREDICTION LAYER

The feature tensor, processed through multiple layers, is passed on to the final prediction layer to generate the ultimate prediction results. This step utilizes the feature data processed through various neural network layers for the final output, obtaining the model's prediction for the input data.

The prediction layer comprises a one-dimensional convolutional layer and a Bayesian Linear layer. The one-dimensional convolutional layer operates on the input's 14 channels to derive a feature map for a single channel. Assuming the output of this layer is denoted as y_1, where *Conv* represents the convolutional operation, W_1 represents the weights of the convolutional kernel, and b_1 represents the bias. The formula is represented as follows:

$$y_1 = Conv\,(x, W_1) + b_1 \tag{10}$$

The one-dimensional feature map is passed through a Bayesian Linear layer, which incorporates prior parameters, to generate the final prediction output. Assuming the output of this layer is denoted as y_2, W_2 where represents the weights of the linear layer and b_2 represents the bias of the linear layer. The formula is expressed as follows:

$$y_2 = W_2 * y_1 + b_2 \tag{11}$$

The final output is a scalar value representing the model's prediction for the input data, while also considering the uncertainty within Bayesian inference. This output can be used for regression tasks or other applications that require predictions on data.

D. LOSS FUNCTIONS

1) MEAN SQUARED ERROR LOSS FUNCTION

When there's a need to assess the disparity between predicted outcomes and real values across multiple samples, a straightforward subtraction of predicted values from actual ones might result in the offsetting of positive and negative errors, thereby diminishing the meaningfulness of the overall error. To tackle this, a prevalent method involves utilizing the absolute value loss function to measure the absolute differences between predicted and actual values.

The absolute value loss function insensitive enough to samples that exhibit significant deviations. This means it doesn't particularly penalize those samples that deviate far from the predicted values, thereby failing to effectively guide supervised training, especially when outliers are present. Additionally, during the actual training process, it's often unnecessary to explicitly calculate the loss function's value at every step because the loss function updates automatically as the backpropagation progresses. In such scenarios, the Mean Squared Error (MSE) [18] comes into play, encapsulating the core idea: by minimizing the sum of squared differences between each training point and the best-fitting line, it measures the Euclidean distance between predicted and actual values. The closer the predicted values are to the actual values, the smaller the mean squared error becomes. The specific computational process for the mean squared error loss function is represented by the following formula:

$$\text{LoSS } = MSE = \frac{1}{N}\left(y' - y\right)^2 \tag{12}$$

where N represents the number of samples, y' and y correspond to the predicted and actual values, respectively.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. DATA

The data used in the experiment was obtained from a publicly available dataset on Kaggle, which includes various conditions of power transformers (such as hydrogen, oxygen, etc.) along with their respective health indices. This dataset is utilized for fault analysis of power transformers. This paper employs the health index to analyze and predict the operational trends of the equipment.

Additionally, this paper also utilized acceleration sensors on the temporary stands of the Beijing Winter Olympics Yanqing Alpine Skiing Center, generating real multi-dimensional time-series data. The dataset was divided for experimentation into training, validation, and testing sets in a ratio of 3:1:1.

B. EXPERIMENTAL SETUP

The experimental dataset was divided into training, validation, and testing sets in a ratio of 3:1:1. The training model employed the Adam optimizer with basic parameter settings as follows: lr = 0.0005, weight_decay = 1e-5, and 200 training epochs were conducted using the training set input to the model. The experiments were conducted on an Ubuntu system, utilizing a Tesla V100 GPU with 128GB of memory. The programming language used was Python 3.6, employing PyTorch as the neural network framework. Data handling and computations were performed using the NumPy and Pandas libraries.''

C. RESULTS AND ANALYSIS

This paper utilized two datasets: one from the publicly available dataset in the Kaggle competition and another collected from the Winter Olympics site. To better assess

the model's performance and generalization ability, a five-fold cross-validation method [19], [26] was employed for model evaluation.To comprehensively assess the model's performance and generalization ability, a five-fold cross-validation method was employed for model evaluation. In the five-fold cross-validation, the dataset was randomly divided into five equally sized parts, with one part used for testing and the remaining four parts used for training and validation. This process was repeated five times, ensuring that each subset served as the test set once. At the beginning of each training iteration, the model should be reinitialized. Finally, the average of the five test results was calculated as the model's performance evaluation metric.This method maximizes the use of data for training the model and utilizes all data points during testing, thereby better evaluating the model's performance and generalization ability. Additionally, it helps alleviate the impact of randomness in dataset partitioning. In practical applications, five-fold cross-validation is commonly used for selecting optimal model parameters or comparing the performance of different models. When employing five-fold cross-validation, it is important to ensure that the dataset partitioning is random, and the data distribution in each part should be as similar as possible.

The data was initially fed into a preprocessing layer, utilizing one-dimensional convolution to alter the channel dimensions and map the data to a high-dimensional feature representation to adapt to subsequent operations. The preprocessed data was then fed into a Bayesian neural network that incorporated channel attention mechanisms. This network processed the data through three parts: pre-processing layers, attention mechanism layers, and post-processing layers, outputting multi-layer processed feature tensors that were fed into the final prediction layer for forecasting. Figure 4 illustrates the comparison between predicted values obtained using the BNN and actual values. It can be observed from the overlapping curves that the prediction performance is quite satisfactory, except for the initial experiment in the first fold, while subsequent folds demonstrate good performance.

Simultaneously, from a quantitative perspective, this paper conducted an analysis by observing the results of the five-fold cross-validation. Taking the average of the five experiments as the performance evaluation for the BNN model, as shown in Table 1, the overall observation suggests that the performance of the BNN on this task is not very consistent. Although the prediction accuracy is relatively good when averaging the results of the five tests, analyzing the results of each cross-validation fold indicates that due to either the small size of the training data or the presence of noise, the BNN might be excessively sensitive to the data or unable to capture the true data distribution, leading to inconsistent and significant variations in performance.

The incorporation of channel attention mechanisms in this paper aimed to enhance the model's stability by improving its focus on data features. This mechanism assists in adjusting the model's emphasis on specific channels, thereby enabling the model to concentrate more on crucial features and consequently improving prediction stability.

TABLE 1. BNN 50 fold cross validation.

Fold index	MSE	MAE	RMSE	MAPE
1	1.06	0.91	1.03	1.00
2	0.52	0.52	0.72	1.07
3	0.48	0.50	0.69	0.73
4	0.44	0.50	0.66	0.88
5	0.36	0.45	0.60	0.99
Average	0.57	0.58	0.74	0.94

The Figure 5 displays the predictions made using the BNN+Attention model. In comparison to the BNN approach, although the first fold's performance remains suboptimal, resulting in a similar underperformance of both models in the initial fold, this might be due to a couple of reasons: (1) The dataset itself might contain a specific distribution, and the first fold might include a less representative or anomalous part of the dataset, leading to poor performance in the initial fold. (2) During the initial model training, random weight initialization or initial parameter settings might not have been optimal, resulting in poorer performance in the first fold. However, this issue does not affect the algorithm's effectiveness and accuracy.

From the subsequent folds, it can be observed that many parts are more closely aligned than those in the BNN model. The overlap between predicted values and actual values in the trends is significantly improved, indicating a notable enhancement in performance.

TABLE 2. BNN+Attention 50 fold cross validation.

Fold index	MSE	MAE	RMSE	MAPE
1	0.46	0.48	0.68	0.92
2	0.41	0.48	0.64	0.94
3	0.54	0.53	0.73	0.76
4	0.48	0.49	0.69	1.31
5	0.41	0.47	0.64	0.63
Average	0.46	0.49	0.68	0.91

Additionally, this paper compared the proposed model, BNN+Attention, with several classical forecasting models such as Recurrent Neural Network (RNN), Long Short-term Memory (LSTM), Gated Recurrent Neural Network (GRU), Bi-directional Long Short-term Memory (Bi-LSTM), Informer, and others as competitive models. The comparison results, as shown in Table 3, further substantiate that the proposed model outperforms these models as well as the BNN in the forecasting task.

Compared to competitive models like RNN, LSTM, GRU, and Bi-LSTM, the results from the four evaluation metrics show a substantial decrease in MSE by 5.39, 4.46, 4.44, and 4.01, respectively. Similarly, the MAE decreased by 28.72, 13.65, 13.5, and 12.37, while RMSE decreased by 33.57, 23.48, 23.33, and 19.27. MAPE stands for Mean

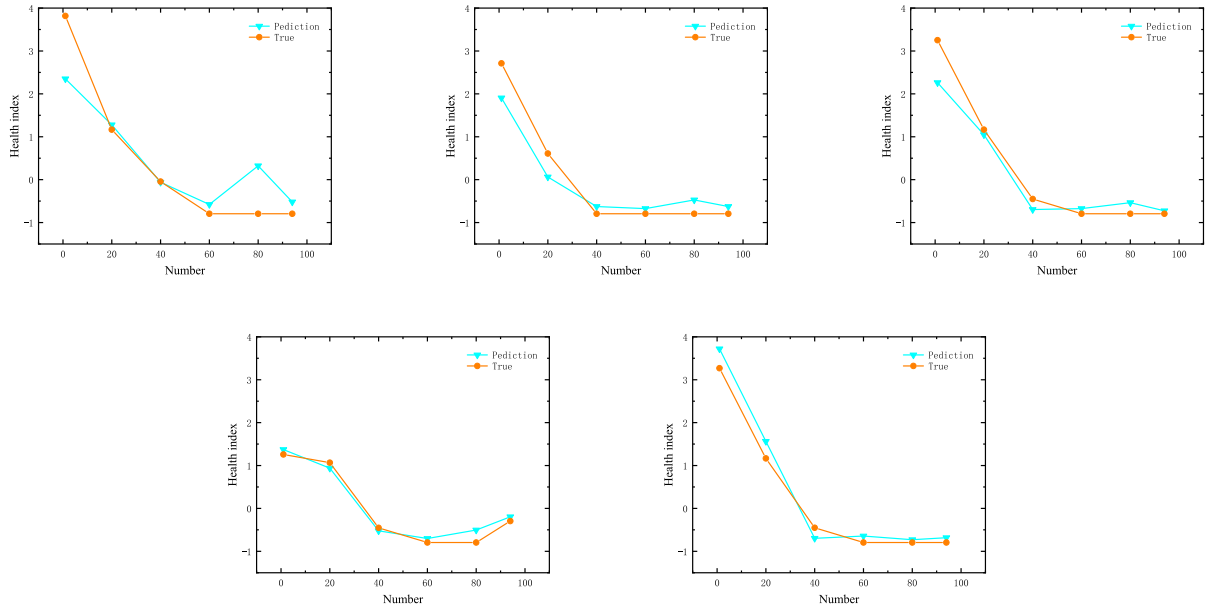

FIGURE 4. The results of the five-fold cross-validation using BNN.

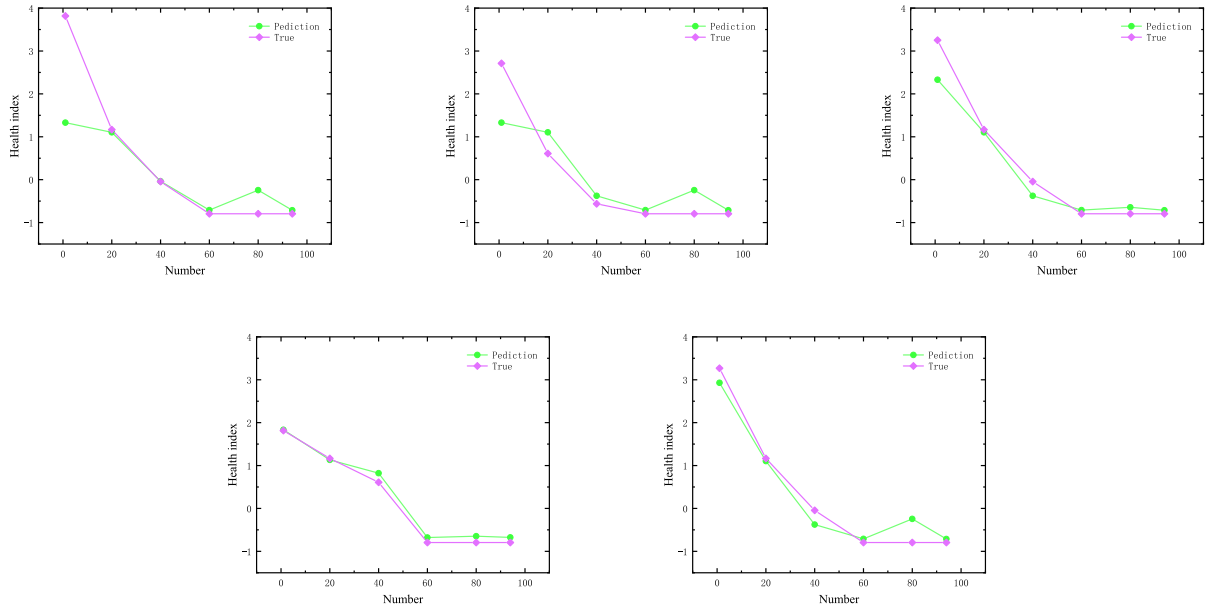

FIGURE 5. BNN+Attention five fold cross validation result chart.

TABLE 3. Comparison experiment of power transformer dataset.

Methods	MSE	MAE	RMSE	MAPE (%)
RNN	5.85	29.21	34.25	3.94
LSTM	4.92	14.14	24.16	0.61
GRU	4.9	13.99	24.01	0.58
Bi-LSTM	4.47	12.86	19.95	1.37
Informer [20]	0.57	0.61	0.69	0.31
BNN [21]	0.57	0.58	0.74	0.94
BNN+Attention	0.46	0.49	0.68	0.91

Absolute Percentage Error, which is a metric used to measure the difference between predicted and actual values as a percentage of the actual values. It is commonly used to evaluate the accuracy of predictive models.The errors are much higher than those of the model proposed in this paper. Although the MAPE of the proposed model may not be optimal, this metric alone does not necessarily reflect the model's overall performance. Considering the other three metrics, this model performs exceptionally well in this task.

The reasons for the poor performance of the cyclic model are as follows: (1) The performance of cyclic models usually depends on the characteristics of the data. If the data has

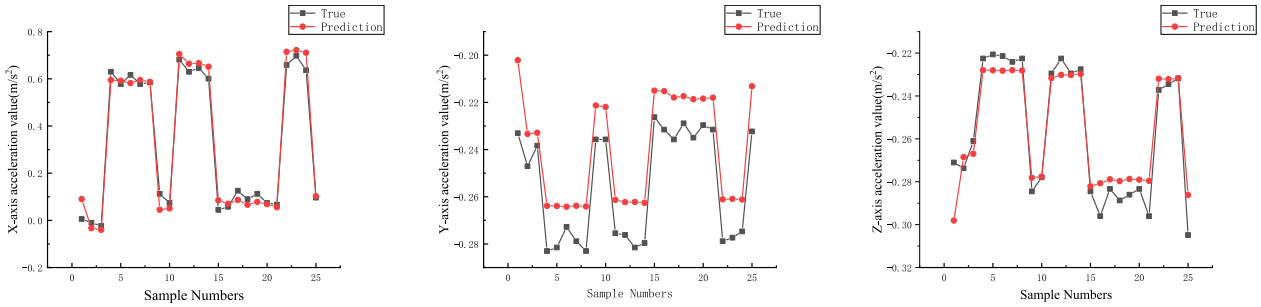

FIGURE 6. Winter olympics real dataset prediction results.

a more complex, nonlinear pattern and weak long-term dependencies, the recurrent model may not perform well. (2) Recurrent models may be prone to overfitting on small datasets. (3) Since the dataset input has 14 features, this task may be beyond the capabilities of the recurrent model, especially if the task needs to deal with a large amount of contextual information or global relationships.

Although BNN+Attention don't produce as good a MAE and MSE as other state-of-the-art methods, the values produced still seem to be high. This may be due to the following reasons: (1) MSE is more sensitive to outliers (squared prediction error). If there are some outliers in your dataset, they may have a significant impact on the calculation of MSE, resulting in a smaller MSE. Meanwhile, MAE has relatively little effect on outliers because it only calculates the absolute value of the error. For this task it is allowed to have outliers, which will be more realistic in predicting the operating trend of the equipment. (2) The distributional characteristics of the data set may also lead to differences in MSE and MAE. For example, if there are some relatively large errors in the data, the squares of these errors may dominate the MSE, resulting in a smaller MSE. The MAE, on the other hand, will treat all errors more evenly, resulting in a larger MAE.

The high MAPE can be attributed to a few reasons: The dataset might contain extreme values that lead to an increase in MAPE. However, these values might represent specific cases in the dataset and may not be crucial for general situations. In such cases, a high MAPE does not affect the model's effectiveness on the majority of data.

A high MAPE could be due to errors within a specific range rather than an overall high error. If the model performs well within a specific data range but has higher errors in other ranges, a high MAPE does not affect the model's effectiveness on the majority of data.

Additionally, this paper conducted experiments comparing the proposed model with Informer. While Informer is a popular time-series prediction model, it did not perform well in this experimental prediction task. Conversely, the proposed model in this paper demonstrated outstanding performance across all evaluation metrics, achieving high prediction accuracy.

Moreover, by introducing channel attention mechanisms into the BNN network, it's evident from both the visualization results and the evaluation metrics that the model outperforms the BNN network.

The experimental results indicate that the channel attention mechanism allows the Bayesian Neural Network to automatically adjust the importance of each channel, thereby enhancing the model's performance. It assists the model in capturing correlations and the importance of features, improving prediction accuracy. Not only can it mitigate the impact of irrelevant or noisy channels, but it also enhances the model's generalization ability on unseen data. As depicted in Figure 6 and Table 4, the paper conducted tests on the model's generalization ability based on real data collected from the Winter Olympics site, showcasing both visualization results and experimental evaluation metrics comparing actual values with predicted values.

TABLE 4. Real winter olympics dataset.

Methods	RMSE	MAE	MAPE (%)
RNN	0.005	0.0032	77.8
LSTM	0.0037	0.0018	26.6
GRU	0.0051	0.0038	50.6
Bi-LSTM	0.0086	0.0074	176.1
Informer	0.1843	0.2477	27.2
BNN	0.008	0.0698	0.73
BNN+Attention	0.007	0.0405	0.82

By observing Figure 6 and Table 4, we can assess the model's performance on unknown data. In Table 4,we compared with the competing models such as RNN, LSTM, GRU and Bi-LSTM, the model in this paper shows that it exhibits good stability for real data, and in terms of all the metrics, our model performs smoothly and with low error. This comparative table clearly illustrates the model's ability to generalize when encountering new data. On the test set, the small discrepancies between predicted and actual values demonstrate stability across different samples. This suggests that the model exhibits strong generalization capabilities and accurately extends to unknown data. In addition, from the observations in Figure 6, which indicates that the horizontal coordinates represent the number of data sample points collected in a time point, and the vertical coordinates are the

acceleration values of the corresponding samples, it can be seen from the curves in the figure that there is no obvious prediction error or unstable results, indicating that there is no overfitting or underfitting phenomenon. This indicates that the model has excellent generalization ability.

V. CONCLUSION

This study explored the integration of channel attention mechanisms into Bayesian Neural Networks to enhance the accuracy and interpretability of equipment trend predictions. The research demonstrates that this approach, combining uncertainty modeling and feature selection, holds extensive potential applications in equipment management. By introducing Bayesian Neural Networks, this paper achieved better estimations of model parameter uncertainties, thereby enhancing the credibility of predictive results. Simultaneously, the channel attention mechanism aided the model in dynamically adjusting the weights of each feature channel, allowing better capture of essential features within the data. This not only improved predictive performance but also heightened the model's interpretability, enabling decision-makers to better comprehend the model's operational principles.

While this research has made significant strides in equipment trend prediction, several future research directions remain. Firstly, further exploration can be done on various types of channel attention mechanisms to adapt to diverse data distributions and tasks. Secondly, consideration can be given to integrating multimodal data (e.g., sensor data and image data) into the proposed model to enhance comprehensive performance. Lastly, investigating methods to handle larger-scale datasets can further validate the scalability of our approach. Additionally, applying this methodology to other domains such as industrial automation, medical equipment management, and financial forecasting holds promise for future research endeavors.

REFERENCES

[1] R. Liu and Z. Wang, "Assigning channel weights using an attention mechanism: An EEG interpolation algorithm," *Frontiers Neurosci.*, vol. 17, 2023, Art. no. 1251677.

[2] M. Magris and A. Iosifidis, "Bayesian learning for neural networks: An algorithmic survey," *Artif. Intell. Rev.*, vol. 56, no. 10, pp. 11773–11823, Oct. 2023.

[3] R. Goyal, V. De Gruttola, and J. P. Onnela, "Framework for converting mechanistic network models to probabilistic models," *J. Complex Netw.*, vol. 11, no. 5, 2023, Art. no. cnad034.

[4] N. Wang and X. Zhao, "Time series forecasting based on convolution transformer," *IEICE Trans. Inf. Syst.*, vol. 106, no. 5, pp. 976–985, 2023.

[5] M. Zhang, F. Tao, Y. Zuo, F. Xiang, L. Wang, and A. Y. C. Nee, "Top ten intelligent algorithms towards smart manufacturing," *J. Manuf. Syst.*, vol. 71, pp. 158–171, Dec. 2023.

[6] D. F. Anderson, B. Joshi, and A. Deshpande, "On reaction network implementations of neural networks," *J. Roy. Soc. Interface*, vol. 18, no. 177, Apr. 2021, Art. no. 20210031.

[7] S. Carta, A. Medda, A. Pili, D. R. Recupero, and R. Saia, "Forecasting e-commerce products prices by combining an autoregressive integrated moving average (ARIMA) model and Google Trends data," *Future Internet*, vol. 11, no. 1, p. 5, Dec. 2018.

[8] L. Rubio, A. J. Gutiérrez-Rodríguez, and M. G. Forero, "EBITDA index prediction using exponential smoothing and ARIMA model," *Mathematics*, vol. 9, no. 20, p. 2538, Oct. 2021.

[9] Z. Ma, H. Zhang, and J. Liu, "MM-RNN: A multimodal RNN for precipitation nowcasting," *IEEE Trans. Geosci. Remote Sens.*, 2023.

[10] X. Wang and Y. Zhang, "Multi-step-ahead time series prediction method with stacking LSTM neural network," in *Proc. 3rd Int. Conf. Artif. Intell. Big Data (ICAIBD)*, May 2020, pp. 51–55, doi: 10.1109/ICAIBD49809.2020.9137492.

[11] Z. Liu, C. Tan, Y. Liu, H. Li, B. Cui, and X. Zhang, "A study of a domain-adaptive LSTM-DNN-based method for remaining useful life prediction of planetary gearbox," *Processes*, vol. 11, no. 7, p. 2002, Jul. 2023, doi: 10.3390/pr11072002.

[12] J. Zhang, L. Ye, and Y. Lai, "Stock price prediction using CNN-BiLSTM-attention model," *Mathematics*, vol. 11, no. 9, p. 1985, Apr. 2023.

[13] X. Zhang, X. Lu, W. Li, and S. Wang, "Prediction of the remaining useful life of cutting tool using the Hurst exponent and CNN-LSTM," *Int. J. Adv. Manuf. Technol.*, vol. 112, nos. 7–8, pp. 2277–2299, Feb. 2021.

[14] M. Joshaghani, A. Davari, F. N. Hatamian, A. Maier, and C. Riess, "Bayesian convolutional neural networks for limited data hyperspectral remote sensing image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, pp. 1–5, 2023.

[15] C. Fang, D. He, K. Li, Y. Liu, and F. Wang, "Image-based thickener mud layer height prediction with attention mechanism-based CNN," *ISA Trans.*, vol. 128, pp. 677–689, Sep. 2022.

[16] L. Shen and Y. Wang, "TCCT: Tightly-coupled convolutional transformer on time series forecasting," *Neurocomputing*, vol. 480, pp. 131–145, Apr. 2022.

[17] C. Feng, Z. Wang, G. Li, X. Yang, N. Wu, and L. Wang, "BERT-PPII: The polyproline type II helix structure prediction model based on BERT and multichannel CNN," *BioMed Res. Int.*, vol. 2022, pp. 1–14, Aug. 2022.

[18] K.-M. Lee, P.-J. Lee, and T.-A. Bui, "Edge enhancement loss function for target object IR image super resolution," in *Proc. IEEE 10th Global Conf. Consum. Electron. (GCCE)*, Oct. 2021, pp. 462–463.

[19] T.-M. Dutschmann, L. Kinzel, A. ter Laak, and K. Baumann, "Large-scale evaluation of k-fold cross-validation ensembles for uncertainty estimation," *J. Cheminform.*, vol. 15, no. 1, p. 49, Apr. 2023.

[20] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 12, 2021, pp. 11106–11115.

[21] Y. Wang, S. Ke, C. An, Z. Lu, and J. Xia, "A hybrid framework combining LSTM NN and BNN for short-term traffic flow prediction and uncertainty quantification," *KSCE J. Civil Eng.*, vol. 28, no. 1, pp. 363–374, Jan. 2024.

[22] X. Wang, G. Xue, S. Huang, and Y. Liu, "Underwater object detection algorithm based on adding channel and spatial fusion attention mechanism," *J. Mar. Sci. Eng.*, vol. 11, no. 6, p. 1116, May 2023.

[23] G. Huang, J. Zhu, J. Li, Z. Wang, L. Cheng, L. Liu, H. Li, and J. Zhou, "Channel-attention U-Net: Channel attention mechanism for semantic segmentation of esophagus and esophageal cancer," *IEEE Access*, vol. 8, pp. 122798–122810, 2020.

[24] D. Yao and Y. Shao, "A data efficient transformer based on Swin Transformer," *Vis. Comput.*, vol. 2023, pp. 1–10, Jul. 2023.

[25] J. Morales and W. Yu, "Improving neural network's performance using Bayesian inference," *Neurocomputing*, vol. 461, pp. 319–326, Oct. 2021.

[26] S. Vosgerau, N. Krattenmacher, C. Falker-Gieske, A. Seidel, J. Tetens, K. F. Stock, W. Nolte, M. Wobbe, R. Reents, C. Kühn, M. V. D. Prondzinski, E. Kalm, and G. Thaller, "Analyses on the influence of the structure of the joint reference population in the German warmblood horse using different cross-validation approaches for the trait withers height," *Züchtungskunde*, vol. 94, no. 5, pp. 347–362, 2022.

[27] L. Zhang, J. Pan, X. Ma, and C. Yang, "AFS-BERT: Information entropy-based adaptive fusion sampling and BERT embedding model for link prediction," *Int. J. Modern Phys. B*, vol. 37, no. 24, Sep. 2023, Art. no. 2350231.

[28] Z. Li, P. Cheng, and J. Zheng, "Prediction of time to slope failure based on a new model," *Bull. Eng. Geol. Environ.*, vol. 80, no. 7, pp. 5279–5291, Jul. 2021.

[29] F. Lin, C. Wang, M. H. Nazari, and W. Li, "Supervisory control to maximize mean time to failure in discrete event systems," *Discrete Event Dyn. Syst.*, vol. 33, no. 2, pp. 105–127, Jun. 2023.

[30] M. O. Adeleke, G. Baio, and A. G. O'Keeffe, "Regression discontinuity designs for time-to-event outcomes: An approach using accelerated failure time models," *J. Roy. Stat. Soc. Ser. A, Statist. Soc.*, vol. 185, no. 3, pp. 1216–1246, Jul. 2022.
[31] I. Koch, K. Naito, and H. Tanaka, "Kernel naive Bayes discrimination for high-dimensional pattern recognition," *Austral. New Zealand J. Statist.*, vol. 61, no. 4, pp. 401–428, Dec. 2019.
[32] D. Valcamonico, P. Baraldi, E. Zio, L. Decarli, A. Crivellari, and L. L. Rosa, "Combining natural language processing and Bayesian networks for the probabilistic estimation of the severity of process safety events in hydrocarbon production assets," *Rel. Eng. Syst. Saf.*, vol. 241, Jan. 2024, Art. no. 109638.
[33] P. R. Winters, "Forecasting sales by exponentially weighted moving averages," *Manag. Sci.*, vol. 6, no. 3, pp. 324–342, 1976, doi: 10.1287/mnsc.6.3.324.
[34] D. K. Bukovšek and N. Stopar, "On the exact regions determined by Kendall's tau and other concordance measures," *Medit. J. Math.*, vol. 20, no. 3, p. 147, 2023.
[35] M. Nakip, C. Güzelís, and O. Yildiz, "Recurrent trend predictive neural network for multi-sensor fire detection," *IEEE Access*, vol. 9, pp. 84204–84216, 2021, doi: 10.1109/ACCESS.2021.3087736.

CHANG MING-YU received the bachelor's degree from Beijing University of Civil Engineering Architecture, Beijing, China, where he is currently pursuing the master's degree in mechanics. He has published one paper and authored one patent. His main research interests include artificial intelligence, computer networks, and big data processing.

TIAN LE received the bachelor's degree from Xidian University and the Ph.D. degree from the School of Computer Sciences, Beijing University of Posts and Telecommunications. He is currently a Vice Professor and a Senior Researcher with the School of Electrical and Information Engineering, Beijing University of Civil Engineering Architecture. He has published more than 20 papers and authored more than seven patents. His research interests include the Internet of Things, cloud computing, and wireless communications.

MAOZU GUO received the bachelor's and master's degrees from the Department of Computer Sciences, Harbin Engineering University, in 1988 and 1991, respectively, and the Ph.D. degree from the Department of Computer Sciences, Harbin Institute of Technology, in 1997. He is currently the Dean of the School of Electrical and Information Engineering, Beijing University of Civil Engineering Architecture. He has implemented several projects from the Natural Science Foundation in China (NSFC), National 863 Hi-tech Projects, the Science Fund for Distinguished Young Scholars of Heilongjiang Province, and the International Cooperative Project. He has authored or coauthored more than 200 refereed papers in journals and conferences. His research interests include machine learning and data mining, computational biology and bioinformatics, advanced computational models, image process, computer vision, and computational architecture. He is a Program Examining Expert of the Information Science Division, NSFC, a Senior Member of China Computer Federation (CCF), a member of the CCF Artificial Intelligence and Pattern Recognition Society and Chinese Association for Artificial Intelligence (CAAI), and a Standing Committee Member of the Machine Learning Society of CAAI. He was a recipient of the Second Prize of the Province Science and Technology Progress and the Third Prize of the Province Natural Science.