IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received September 11, 2020, accepted October 8, 2020, date of publication October 15, 2020, date of current version October 29, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3031438

# A Deep Learning Inspired Belief Rule-Based Expert System

**RAIHAN UL ISLAM [1], MOHAMMAD SHAHADAT HOSSAIN [2], (Senior Member, IEEE), AND KARL ANDERSSON [1], (Senior Member, IEEE)**

[1]Pervasive and Mobile Computing Laboratory, Luleå University of Technology, 931 87 Skellefteå, Sweden
[2]Department of Computer Science and Engineering, University of Chittagong, Chittagong 4331, Bangladesh

Corresponding author: Raihan Ul Islam (raihan.ul.islam@ltu.se)

**ABSTRACT** Recent technological advancements in the area of the Internet of Things (IoT) and cloud services, enable the generation of large amounts of raw data. However, the accurate prediction by using this data is considered as challenging for machine learning methods. Deep Learning (DL) methods are widely used to process large amounts of data because they need less preprocessing than traditional machine learning methods. Various types of uncertainty associated with large amounts of raw data hinder the prediction accuracy. Belief Rule-Based Expert Systems (BRBES) are widely used to handle uncertain data. However, due to their incapability of integrating associative memory within the inference procedures, they demonstrate poor accuracy of prediction when large amounts of data is considered. Therefore, we propose the integration of an associative memory based DL method within the BRBES inference procedures, allowing to discover accurate data patterns and hence, the improvement of prediction under uncertainty. To demonstrate the applicability of the proposed method, which is named BRB-DL, it has been fine tuned against two datasets, one in the area of air pollution and the other in the area of power generation. The reliability of the proposed BRB-DL method, has also been compared with other DL methods such as Long-Short Term Memory and Deep Neural Network, and BRBES by taking into account of the air quality dataset from Beijing city and the power generation dataset of a combined cycle power plant. BRB-DL outperforms the above-mentioned methods in terms of prediction accuracy. For example, the Mean Square Error value of BRB-DL is 4.12 whereas for Long-Short Term Memory, Deep Neural Network, Fuzzy Deep Neural Network, Adaptive Neuro Fuzzy Inference System and BRBES it is 18.66, 28.49, 17.05, 16.37 and 38.15 for combined cycle power plant respectively, which are significantly higher.

**INDEX TERMS** Knowledge based systems, expert systems, multi-layer neural network, learning systems.

## I. INTRODUCTION

The accurate prediction using real-world data is considered as a challenging task for various machine learning methods. It can be observed that nowadays, a large amount of data is generated continuously in different scientific and industrial fields around the world due to the Internet of Things (IoT) and cloud services. These large amounts of data inevitably contain various uncertainties like incompleteness, ignorance, vagueness, imprecision, and ambiguity. These uncertainties pose a significant challenge to the accurate prediction from data.

To address the uncertainties mentioned above, fuzzy-based learning approaches [1] have been widely used. These approaches are used in image processing [2], portfolio management [3], and motor control [4], where uncertainty is a regular phenomenon. Fuzzy learning systems automatically learn the fuzzy membership functions and consequently derive fuzzy rules from a large amount of training data [5]. Using an inference mechanism, the fuzzy values are generated from the fuzzy rules. The fuzzy values are then converted to crisp values using different defuzzification techniques like, the centre of gravity (COG), mean of maximum (MOM), and centre average methods. However, fuzzy learning systems can address uncertainty due to imprecision, ambiguity, and vagueness but not due to incompleteness and ignorance [6].

Belief Rule-Based Expert Systems (BRBESs) represent an improved version of fuzzy learning systems. They facilitate better representation of uncertain knowledge by incorporating a belief structure. Usually, an expert system has two main

The associate editor coordinating the review of this manuscript and approving it for publication was Youqing Wang.

components, one is the knowledge base, and the other is the inference engine. IF-THEN rules are used as a knowledge representation schema in the traditional knowledge base, for example "IF creatinine is present THEN renal failure is definite". The semantic of this rule is that "renal failure" is 100% certain because of the "creatinine is present". However, the rule fails to capture the scenario when it is less than 100% certain that "renal failure" is due to the "creatinine is present". Yang *et al.* [7] proposed a new knowledge representation schema by incorporating distributed assessment called belief structure in the consequent part of the rule, for example "IF the amount of rainfall is Medium and duration of rain is High THEN chance of flooding is (High 60%, Medium 30%, Low 10%)". Due to the new knowledge representation schema with belief structure, the BRBES has been used in different domains such as natural disaster prediction [8], [9], different diseases assessment [10]–[12], and Forex trading forecast [13] where the issue of uncertainty is dominant in making decisions. In general, BRBES can be of two different types. One is Conjunctive BRB, where each antecedent attribute of the rule is connected using the AND logical operator. Another one is Disjunctive, where the OR logical operator is used in the antecedent part of the rule [14]. Conjunctive BRB requires more computational time because it is an example of a combinatorial explosive problem, resulting from the connection of antecedent attributes of a rule by logical AND operator. Consequently, the rule base of conjunctive BRB consists of large numbers of rules [15]. On the contrary, Disjunctive BRB requires less computational time because it uses the logical OR operator in the antecedent part of the rule and hence, this constitutes less number of rules in the rule base.

Recently, Deep Learning is becoming an effective method for solving different pattern recognition and regression problems due to its ability to process raw data directly [16]. However, Deep Learning lacks the capability of addressing different types of uncertainty, since it is based on neural networks, inherently limited in addressing uncertainty [17]. On the other hand, BRBES is capable of addressing various types of uncertainty such as ignorance, incompleteness, ambiguity, vagueness, and imprecision in an integrated framework. However, BRBES lacks the capability of integrating associative memory in its inference procedure since most of the operations are multiplicative, summation, and division based. Since these operators do not have any memorizing capability, they are unable to discover complete patterns from partial information. For example, the use of matching degrees, which will be discussed in Section III, in calculating activation weight of a rule by using multiplicative and division operators are unable to generate accurate activation values of each rule in the BRBES inference framework. These incomplete values of rule activation weights will affect the rule aggregation procedure, which is used to perform the prediction. Therefore, in this study Deep learning-based methods especially Deep Neural Network (DNN) has been considered to calculate the weight of the rule activation by taking into account matching degrees allowing the calculation of more accurate values of the activated rule. The reason for using a DNN based Deep Learning method is that it is based on Artificial Neural Networks (ANN). ANN are associative memory systems and hence, the capability to recall complete situations from partial information as well as the ability to correlate input data with stored information [18]. Thus, our proposed method is based on associative memory, allowing the retrieving of the complete value of rule activation weight by taking into account the matching degrees. This will play an important role in BRBES's inference framework to process especially large amounts of data in a very accurate way. Eventually, this would also contribute to the improvement of the overall prediction accuracy of BRBES as will be demonstrated in Section VII.

The determination of optimal values of the BRBES's learning parameters such as rule weights, attribute weights, and belief degrees also play an important role to increase the prediction accuracy. These optimal values are achieved by using a plethora of learning algorithms [19]–[21]. The integration of associative memory based Deep Learning method with BRBES requires the inclusion of additional learning parameters such as weights of neurons and bias, and they should also be optimized. Hence, the framework of BRBES learning should need to be improved, as will be discussed in Section VI. Eventually, the additional parameters of associative memory would play a role to increase the accuracy of prediction.

In this way, we will advance the BRBES's present methodology as will be demonstrated by using two use cases. One of the use cases is air quality prediction in Beijing city using a dataset containing around 43,824 data points. The other one is the prediction of the electrical energy output of a combined cycle power plant using a dataset which contains 9,568 data points. In addition, the results were compared with different Deep Learning methods such as Deep Neural Network, Long-Short Term Memory (LSTM), and BRBES, where the proposed method in this study named as BRB-DL performed promisingly better than the other methods mentioned.

The remainder of this article is structured as follows. Section II surveys related work on integration of various methods with Deep Learning, and incorporation of different machine learning methods with BRBES. Section III provides the brief overview of the BRBES, while Section IV discusses about Deep Learning methods. Section V describes the Deep Learning inspired BRBES named, BRB-DL. Subsequently, Section VI presents the learning mechanism for BRB-DL, followed by Section VII which presents results and analysis. Lastly, Section VIII concludes the article.

## II. RELATED WORK

This section presents a literature review on integration between 1) Fuzzy and Deep Learning methods; 2) various learning mechanisms with BRBES; and 3) machine learning methods with BRBES.

Deep Learning has been used to solve various problem of prediction from data. However, different algorithms have been used with Deep Learning to improve its accuracy of prediction. Merentitis and Debes [22] have used random forest with Deep Learning to improve classification tasks. In this study, Deep Learning has been used to extract high-level features and passed to the random forest algorithm for performing the classification tasks.

Chen *et al.* [17] presented a novel Fuzzy Deep Learning approach, called Fuzzy Deep Convolutional Network (FDCN), which was proposed for predicting the traffic flow of a city. They combined Fuzzy theory and Deep Residual Network to address the uncertainty. The FDCN contains five modules, namely input, Deep Convolutional Network (DCN), Fuzzy Network (FN), fusion module, and predictor. In the beginning, the input data is passed to both FN and DCN simultaneously. After processing the data, the output is merged in the fusion module and sent to the predictor module. The fusion module uses objective or loss functions for training the parameters of the DCN and FN module. During the training phase, the parameters of DCN and FN are modified to minimize the value of the objective function. After finishing the training phase, the predicted values can be generated by feeding the data into the model. The DCN module is used to capture the pattern of the data, while FN is used to address the uncertainty. However, Fuzzy systems can not address uncertainty due to incompleteness, and ignorance, which will hinder the accuracy of prediction. Furthermore, the traffic flow is complex data because it consists of spatio and temporal information. Therefore, this data will cause incompleteness due to missing information and ignorance due to mismatch of data. Since such types of uncertainties cannot be addressed by fuzzy systems, they will affect the accuracy of FDCN's prediction.

Deng *et al.* [23] proposed a hierarchical fused Fuzzy Deep Neural Network (FDNN) for data classification. The data is passed to the fuzzy system and DNN module simultaneously, and then the output of these two modules are fused to transfer to the task-driven layer for generating the classification result. According to the authors, the fuzzification of the fuzzy module helps in addressing uncertainty, and Deep Learning reduces the noise of the data. The FDNN has been evaluated using the classification of brain tissues from MRI images as well as predicting stock prices, where it showed promising results. However, FDNN suffers from the inherent limitation of the fuzzy systems, which is the lack of addressing uncertainty due to incompleteness, and ignorance. According to Deng *et al.* [23], the data contains various kinds of noise, which causes incompleteness and ignorance. Since fuzzy systems can handle uncertainty due to imprecision, ambiguity, and vagueness but not due to incompleteness and ignorance, the accuracy of the classification of brain tissues from the MRI images could not be improved by using FDNN.

From the above discussion, it can be concluded that fuzzy systems have been used as a separate box while integrating with Deep Learning to address the uncertainty. Fuzzy systems can address uncertainty, due to imprecision, ambiguity, and vagueness [8]. Therefore, the accuracy of these integrated methods will be hindered in the cases where there is presence of uncertainty due to ignorance and incompleteness.

Various learning algorithm such as Genetic Algorithm (GA), Differential Evolution (DE), and Particle Swarm Optimization (PSO) have been used to support learning in BRBES. They have been used as a separate boxes with BRBES. Yang *et al.* [24] proposed a learning algorithm for BRBES to find the optimal values of the learning parameters such as attribute weight, rule weight, and belief degrees to improve the accuracy of prediction. Chang *et al.* [25] proposed a joint optimisation method for BRBES by taking into account of both structure and learning parameters. In this method, the Akaike Information Criterion (AIC) was used as an objective function to increase the accuracy of BRBES' prediction. They used this method to predict pipeline leak detection. Yang *et al.* [26] also proposed a joint parameter and structure optimisation model to support learning in BRBES but they have not use AIC as the objective function. However, they used a heuristic algorithm for structure optimisation and a Differential Evolution (DE) algorithm for parameter optimisation. Islam *et al.* [27] proposed a joint optimisation method using an enhanced Belief Rule-Based Adaptive Differential Evolution (eBRBaDE) to improve the prediction accuracy of BRBES. This method helps to optimise the parameters of BRBES by using eBRBaDE. They have used this method to predict Power Usage Effectiveness (PUE) of a datacentre, where BRBES showed higher prediction accuracy compared to other evolutionary algorithms, like PSO, GA and DE. The aforementioned research helped BRBES to improve its prediction accuracy.

Li *et al.* [28] proposed integration of Conditional Generalized Minimum Variance (CGMV) and BRBES for the safety assessment of a complex system like WD615 model diesel engine. In this integrative method, CGMV was used for feature selection, while BRBES was used for safety assessment. Both CGMV and BRBES were used as separate boxes. However, they did not modify the BRBES inference procedure, which would decrease the prediction accuracy for large amounts of data.

Chang *et al.* [29] proposed an integrated Principal Component Analysis (PCA) and BRBES method, named PCA-BRB, for monitoring the health of the running gear of a high speed train. They used PCA for feature selection and BRBES for making decisions on health tasks of the high speed train. In this integration the PCA and BRBES were used as separate boxes and no changes were made in the BRBES inference procedure. Furthermore, PCA-BRB does not provide any features to handle large amounts of data, which might hinder the accuracy of prediction.

Kabir *et al.* [30] proposed integration of a Deep Learning method with BRBES to predict air pollution using outdoor images of Beijing city. They have used Convolutional Neural Networks (CNN) to predict PM2.5 values from outdoor images. The predicted PM2.5 values from CNN and sensor readings of PM2.5 values were used to predict air quality level using BRBES. In their method of integration, CNN and BRBES were used as separate boxes. This integration did not make any modification of the BRBES inference procedure to incorporate the associative memory. Therefore, the proposed integration will suffer in processing large amounts of data as BRBES does not have the capability to use associative memory.

In summary, Fuzzy systems have been integrated with Deep Learning methods in various ways to address the uncertainty of large amounts of data [17], [23]. However, these Fuzzy and Deep Learning integrations lack in prediction accuracy as they are not able to address all types of uncertainty due to limitations of the Fuzzy system. Different learning methodologies like *fmincon* [24], DE [26], and eBRBaDE [27] have been used to improve the prediction accuracy of BRBES. On the other hand, various methods like CGMV [28], PCA [29], and CNN [30] have been integrated as a separate box to improve the performance of BRBES. However, there have been no attempts to incorporate associative memory with the BRBES inference procedure to improve the BRBES's accuracy of prediction for large amounts of data. Therefore, this study focuses on integration of associative memory based Deep Learning method with BRBES inference procedure to improve its accuracy for prediction. Therefore, in the following sections, BRBES and Deep Learning will be presented in details. Afterwards, the integration of associative memory based Deep Learning methods with BRBES will be presented.

## III. BELIEF RULE-BASED EXPERT SYSTEM

A belief rule has two parts: one is antecedent or premise part, which consists of antecedent attributes; while the other is consequent or conclusion part which contains the consequent attribute. The antecedent attributes use referential values, and the belief degrees are associated with the consequent attribute of the belief rule, which is shown in Eq. (1). Each belief rule is assigned with a rule weight to show its importance.

$$R_k : \begin{cases} \text{IF } (A_1 \text{ is } V_1^k) \text{ AND / OR } (A_2 \text{ is } V_2^k) \text{ AND / OR} \\ \ldots \text{ AND / OR } (A_{T_k} \text{ is } V_{T_k}^k) \\ \text{THEN } (C_1, \beta_{1k}), (C_2, \beta_{2k}), \ldots, (C_N, \beta_{Nk}) \end{cases} \tag{1}$$

where $\beta_{jk} \geq 0, \sum_{j=1}^{N} \beta_{jk} \leq 1$ with rule weight $\theta_k$, and attribute weights $\delta_{k1}, \delta_{k2}, \ldots \delta_{kTk}, k \in 1, \ldots, L$

where $A_1, A_2, \ldots, A_{T_k}$ are the antecedent attributes of the $k^{\text{th}}$ rule. $V_i^k (i = 1, \ldots, T_k,\ k = 1, \ldots, L)$ is the referential value of the $i^{\text{th}}$ antecedent attribute. $C_j$ is the $j^{\text{th}}$ referential value of the consequent attribute. $\beta_{jk} (j = 1, \ldots, N, k = 1, \ldots, L)$ is the degree of belief for the consequent reference value $C_j$. If $\sum_{j=1}^{N} \beta_{jk} \leq 1$, then the $k^{\text{th}}$ rule is considered as complete; otherwise, it is incomplete.

Usually, the collection of belief rules is called the Belief Rule Base (BRB). The logical connectives of the antecedent attributes in a belief rule can be either AND or OR. A belief rule is considered as conjunctive if the antecedent attributes are connected using AND. Similarly, if the antecedent attributes of the belief rule are connected with OR, then it is called a disjunctive rule. Based on the logical connectivity of the BRB, a BRBES can be named as conjunctive or disjunctive BRBES.

After constructing the BRB, the inference procedure is used to generate the output. The inference procedure consists of various steps which are illustrated in Fig. 1. These are input transformation, rule activation, weight calculation, belief degree update, and rule aggregation using the evidential reasoning approach. The input data is distributed over the referential values of the antecedent attributes, which is called the matching degree, achieved through the inference process of input transformation. Then the belief rules are called packet antecedent. Subsequently, activation weights of the rules are calculated using matching degrees.

The activation weight $w_k$ for the $k^{\text{th}}$ rule for conjunctive assumption is calculated by the following expression:

$$w_k = \frac{\theta_k \prod_{i=1}^{T_k} \alpha_i^k}{\sum_{l=1}^{L} (\theta_k \prod_{i=1}^{T_k} \alpha_i^l)} \tag{2}$$

Here, $\theta_k$ is the rule weight and $\alpha_k$ is the matching degree of the $k^{\text{th}}$ rule. As, in the conjunctive assumption all matching degrees are multiplied to address the AND operation of the belief rule.

However, for disjunctive assumption the activation weight $w_k$ for the $k^{\text{th}}$ rule is calculated by the following expression:

$$w_k = \frac{\theta_k \sum_{i=1}^{T_k} \alpha_i^k}{\sum_{l=1}^{L} (\theta_k \sum_{i=1}^{T_k} \alpha_i^l)} \tag{3}$$

Here, $\theta_k$ is the rule weight and $\alpha_k$ is the matching degree of the $k^{\text{th}}$ rule. In the disjunctive assumption all matching degrees are summed to address the behaviour of OR operator of the belief rule. It is also necessary to mention why the complete value of the rule activation weight is difficult to calculate using multiplicative, summation and division operator and why this is possible in associative memory.

However, Eqs. (2) and (3) are multiplicative and summation in nature, which does not have component of storing patterns like associative memory.

If any of the antecedent attributes are ignored, the belief degree associated with each belief rule needs to be updated.

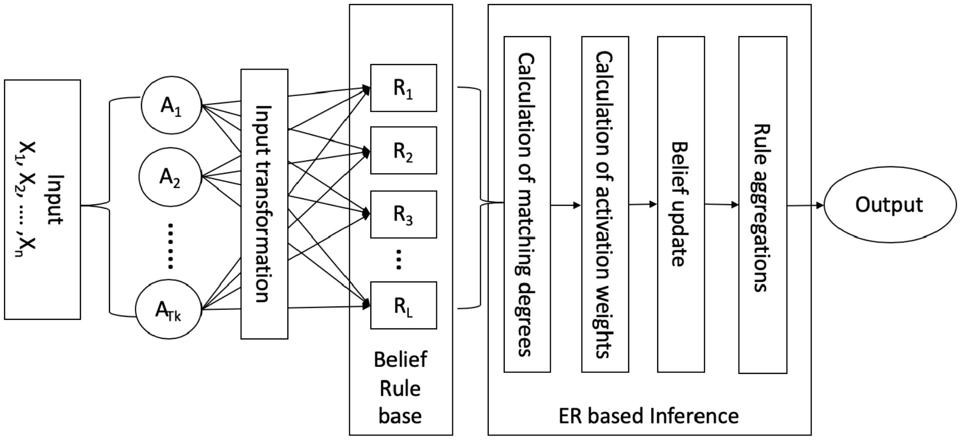



**FIGURE 1.** **Working process of BRBES.**

The belief degree update is calculated using Eq. (4) [7].

$$\beta_{jk} = \bar{\beta}_{jk} \frac{\sum_{t=1}^{T_k} (\lambda(t,k) \sum_{i=1}^{I_t} (\alpha_{ti}))}{\sum_{t=1}^{T_k} \lambda(t,k)} \tag{4}$$

where

$$\lambda(t,k) = \begin{cases} 1 & \text{if the } t^{th} \text{ attribute is used in} \\ & \text{defining rule } R_k (k = 1, \ldots, T_k) \\ 0 & \text{otherwise} \end{cases}$$

Here, $\bar{\beta}_{jk}$ represents the original belief degree, while the updated belief degree is $\beta_{jk}$ of the $k$ $^{\text{th}}$ rule. $\alpha_{ti}$ represents the degree to which the input value belongs to an attribute.

Afterwards, the rule aggregation is performed using the recursive reasoning algorithm as shown in Eq. (5) [31]

$$\beta_j = \frac{\mu \times [X - \prod_{k=1}^{L}(1 - \omega_k \sum_{j=1}^{N} \beta_{jk})]}{1 - \mu \times [\prod_{k=1}^{L} 1 - \omega_k]} \tag{5}$$

where

$$\mu = \left[ \sum_{j=1}^{N} \prod_{k=1}^{L} (\omega_k \beta_{jk} + 1 - \omega_k \sum_{j=1}^{N} \beta_{jk}) - (N-1) \times \prod_{k=1}^{L} (1 - \omega_k \sum_{j=1}^{N} \beta_{jk}) \right]^{-1}$$

Here, $\omega_k$ is the activation weight of the $k^{\text{th}}$ rule, while $\beta_j$ denotes the belief degree related to one of the consequent reference values.

The fuzzy output of the rule aggregation procedure is converted to a crisp value using the utility values of the consequent attribute, which is considered as the final result, as shown in Eq. (6).

$$z_i = \sum_{j=1}^{N} u(O_j)\beta_j \tag{6}$$

where $z_i$ is the expected numerical value and $u(Oj)$ is the utility score of each referential value.

In summary, the input data $x_1, x_2, \ldots, x_n$ is mapped to the matching degree of the referential values $A_1, A_2, \ldots, A_{tk}$ of the antecedent attributes as shown in Fig. 1. Firstly, input transformation is performed, which generates matching degrees. Using the matching degrees, activation weights of the rules are calculated. Then, belief degrees associated with rules are modified during the belief update step. Subsequently, using the recursive reasoning algorithm in the rule aggregation step, the fuzzy output is generated, which is converted into a crispy value using the utility function. The uncertainty due to vagueness, imprecision, ambiguity, incompleteness, and ignorance are addressed by the belief schema, belief degree update step, and evidential reasoning inference procedure. By following the above-mentioned steps BRBES addresses various types of uncertainty.

The BRBES is presented in terms pseudocode in Algorithm 1, where $X_{i,j}(i = 1, \ldots, N; j = 1, \ldots, TR)$ denotes input data, $N$ denotes total number of data, $r_{p,q}$ denotes the $q^{\text{th}}$ referential value of $p^{\text{th}}$ attribute, $BRB$ denotes the belief rule base, and $Y_i(i = 1, \ldots, N)$ denotes the predicted output. Lines 2 to 6 handles the input transformation step for each data. The matching degree is computed using the referential values of the antecedent attributes in the input transformation step. Line 8 calculates the activation weight using the matching degrees. Line 9 executes the calculation of belief update. Line 10 shows the rule aggregation operation using Eq. (5), which generates a fuzzy output value. Finally, the fuzzy output value is converted to a crisp value using Eq. (6) as shown in line 11. Lines 8 to 11 are invoked for each input value of the dataset to calculate the output.

## IV. DEEP LEARNING APPROACH

Deep Learning is a method, which automatically discovers necessary representation from data to calculate prediction or classification [16]. The simplest form of Deep Learning consists of input, hidden, and output layers. Usually, the data

**Algorithm 1** BRBES Algorithm

Let $X_{i,j}(i = 1, \ldots, N; j = 1, \ldots, TR)$ denotes input data, $N$ denotes total number of data, $TR$ denotes the total number of attribute. $r_{p,q}$ denotes the $q^{\text{th}}$ referential value of $p^{\text{th}}$ attribute, $BRB$ denotes the belief rule base, and $Y_i(i = 1, .., N)$ denotes the predicted output

**Input** $X_{i,j}$, $r$, $BRB$, $N$

**Output** $Y_i$

```
procedure BRBES(X_{i,j}, r_{i,j}, BRB, N)
    for each i ∈ N do
        for each j ∈ M do
            X_{i,j} is transformed to matching degree, md_{i,k}(k = 1, ..., L) based on the referential value r
        end for
    end for
    for each i ∈ N do
        Calculate activation weight using Eq. (2) for conjunctive BRB and Eq. (3) for disjunctive BRB
        Calculate belief update using Eq. (4)
        Calculate rule aggregation using Eq. (5)
        Convert crisp value Y_i from fuzzy value generated from rule aggregation using Eq. (6)
    end for
end procedure
```

is fed to the input layer and passed to the hidden layer. The hidden layer can contain multiple layers. The multiple hidden layers signify the word 'Deep'. Each layer has multiple numbers of neurons. Each neuron has an activation function which creates a non-linear representation of data. The neuron helps to capture features of the data. A neuron is mathematically represented by Eqs. (7) and (8).

$$z_i^l = w_i^l * x^l + b_i^l \tag{7}$$

Here, $w$ is the weight, $x$ is input, and $b$ is the bias of $i^{th}$neuron of $l^{th}$ layer.

$$y = g(z_i^l) \tag{8}$$

Here, $g$ is the activation function.

There are several methods in Deep Learning such as Deep Neural Networks (DNN) [32], Convolutional Neural Networks (CNN) [33], Long Short-Term Memory (LSTM) [34], and Recurrent Neural Networks (RNN) [35]. To learn from the data various optimization methods are used. The gradient descent approach is used for neural networks. However, it is not suitable for Deep Learning due to its long computational time. The Stochastic Gradient Descent (SGD) approach is used for training the Deep Learning method with back propagation.

## V. DEEP LEARNING INSPIRED BELIEF RULE-BASED EXPERT SYSTEM (BRB-DL)

This section presents the integration of Deep Learning with BRBES, named BRB-DL. The BRB-DL consists of a BRB and the inference procedure. Fig. 2 illustrates the workflow of a Deep Learning integrated BRBES method. The initial BRB can be created using experts' opinion or based on the method described in [36].

The BRB-DL inference procedure consists of four steps: 1) Input transformation; 2) Deep learning processing; 3) Belief update; and 4) Rule aggregation. A detailed description of these steps are provided below.

### A. INPUT TRANSFORMATION

First, the input data is mapped to matching degrees of the referential values of the antecedent attributes during input transformation as described in [36]. Input transformation is marked as 'a' in Fig. 2.

### B. DEEP LEARNING PROCESSING

The second step is the deep learning processing, which is marked as 'b' in Fig. 2. In this step, various Deep Learning methods like DNN, LSTM, and CNN can be used. In this instance, we have used DNN. During this step, a deep learning multi-layer is constructed which has one input layer, multiple hidden layers, and one output layer. The matching degree from the input transformation step is passed to the input layer. The number of neurons in the input layer is usually equal to the total number of referential values. The hidden layer can have $n_1$ layers, where each layer has $n_2$ neurons in each layer. All the neurons are fully connected. Eqs. (9) and (10) represent each neuron. Eq. (10) is also known as the activation function. The output layer contains the same number of neurons as the number of belief rules. The neurons of the output layer uses Eqs. (9) and (11). The output layer produces the activation weight for the belief rules of the BRB, which activate the belief rules. The weights $c$ of neurons are initialized with random values and bias $b$ of neurons are initialized with zero [23].

$$z_i = (c_i * \alpha_i) + b_i \tag{9}$$

Here $c_i$ is the weight and $\alpha_i$ is the matching degree, and $b_i$ is the bias.

$$\omega_i = max(0, z_i) \tag{10}$$

Here, $\omega_i$ is the activation weight.

$$\omega_i = \frac{\exp(z_i)}{\sum(\exp(z_i))} \tag{11}$$

Here, $\omega_i$ is the activation weight.

The simplest associative memory can be presented using Eq. (12). Here, $a$ is the input pattern, $b$ is the output pattern, and $M$ is the memory matrix [37].

$$b = f(M, a) \tag{12}$$

Eq. (9) represents the associative memory part, where $\alpha_i$ is the input, $z_i$ is the output, and $c_i$ is the memory. Therefore, it can be concluded that BRB-DL contains the associative memory feature to calculate the activation weight. This helps BRB-DL to discover various patterns more accurately from the data

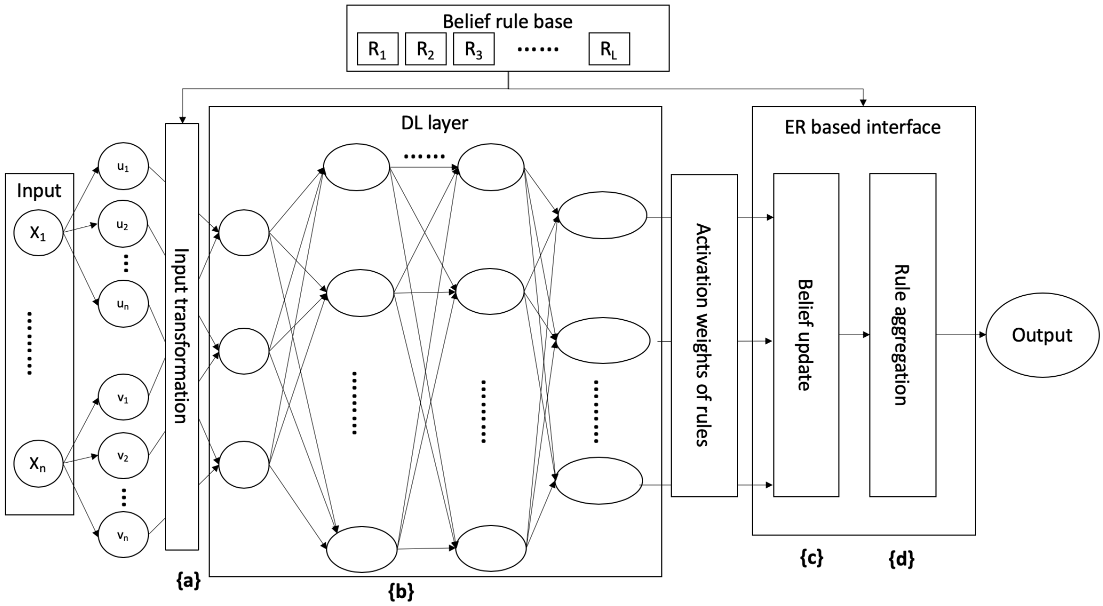



**FIGURE 2.** Working process of BRB-DL.

compared to the previous activation weight calculation using Eqs. (2) and (3). Eq. (2) calculates the activation weight using multiplication and division operations between the matching degree $\alpha_i$ and the rule weight $\theta_k$. Similarly, Eq. (3) uses summation and division operations between the matching degree $\alpha_i$ and the rule weight $\theta_k$. These equations do not have any variables for memorizing like $M$ in Eq. (12). Hence, Eqs. (2) and (3) are not able to memorize any pattern from the data. Therefore, the new approach will be able to discover different patterns and their corresponding activation weights, which will help to produce more accurate prediction of the output.

The memory in the DNN is an associative memory allowing retrieving complete information from partial information. Eq. (11) calculates the complete information by using associate memory which can be derived by Eq. (9). Therefore, it can be concluded that BRBES contains the associative memory to compute the accurate or complete activation weight of a rule of the BRB. An integration of DNN with BRBES will advance the capability of BRBES because of the integration of associative memory. Hence, the inclusion of DNN inside the inference process of BRBES will remove more uncertainty when calculating the rule activation weight. This will be demonstrated using in our case study where BRB-DL will turn out to perform better than the BRBES, which will be presented in Section VII.

### C. BELIEF UPDATE

The third step is the belief update, which is calculated using Eq. (4). The belief update helps to address uncertainty due to ignorance which is caused due to the missing of the antecedent attribute. This step is marked as 'c' in Fig. 2.

### D. RULE AGGREGATION

The fourth step is the rule aggregation. This step is performed using Eq. (6) which generates a fuzzy value. The fuzzy value is later converted into a crisp value, which is the predicted output. This step is marked as 'd' in Fig. 2. The belief schema, belief degree update, and evidential reasoning inference address the uncertainty due to vagueness, imprecision, ambiguity, and incompleteness and ignorance.

The BRB-DL method is presented as pseudocode in Algorithm 2, where $X_{i,j}(i = 1, \ldots, N; j = 1, \ldots, TR)$ denotes input data, $N$ denotes total number of data, $TR$ denotes the total number of attribute, $r_{s,t}$ denote the $t^{\text{th}}$ referential value of $s^{\text{th}}$ attribute, $BRB$ denotes the belief rule base, $Y_i(i = 1, .., N)$ denotes the predicted output, $NumOfHiddenLayers$ denotes the number of hidden layers, and $nh$ denotes the number of neurons in each hidden layer. Lines 2 to 6 represent the input transformation step where the matching degree $md$ is generated. Associative memory based deep learning processing is performed in lines 8 to 14. Afterwards, belief update and rule activation are performed in lines 16 and 17. Finally, the fuzzy value generated from rule aggregation is transformed to a crisp value in line 18. The main difference between Algorithm 1 and 2 appears on line 8 of Algorithm 1 and lines 10 to 12 of Algorithm 2. The activation weight calculation is replaced by the deep learning processing step to incorporate the associate memory with BRBES.

In summary, the proposed BRB-DL method is able to handle all types of uncertainty due to the input transformation, belief update, and rule aggregation steps. The deep learning processing helps to integrate the associative memory with BRBES to discover accurate patterns from data. Therefore, it can be concluded that the proposed integration of Deep

**Algorithm 2** BRB-DL Algorithm

Let $X_{i,j}(i = 1, \ldots, N; j = 1, \ldots, TR)$ denotes input data, $N$ denotes total number of data, $TR$ denotes the total number of attribute. $r_{s,t}$ denotes the $t^{\text{th}}$ referential value of $s^{\text{th}}$ attribute, $BRB$ denotes the belief rule base, $Y_i(i = 1, .., N)$ denotes the predicted output, *NumOfHiddenLayers* denotes the number of layers in hidden layer, and $nh$ denotes the number of neuron in each layer of hidden layer

**Input** $X_{i,j}$, $r_{i,j}$, $BRB$, $nh$, $N$

**Output** $Y_i$

```
procedure BRB_DL(X_{i,j}, r, BRB, N, NumOfHiddenLayers, nh)
    for each i ∈ N do
        for each j ∈ M do
            X_{i,j} is transformed to matching degree, md_{i,k}(k = 1, ..., L) based on the referential value r
        end for
    end for
    for each i ∈ N do
        ni=total number of referential values
        md_{i,k}(k = 1, ..., L) is used in input layer for ni neurons using Eqs. (9) and (10)
        for each p ∈ NumOfHiddenLayers do
            α_{p,q}(q = 1, ..., nh) computed using Eqs. (9) and (10) for nh number of neurons
        end for
        nl= number of belief rules in BRB
        Calculate α_{i,q}(q = 1, ..., nl) using Eq. (9) and (11) for nl neurons
        ω_{i,q} = α_{i,q} where (q = 1, ..., nl)
        Perform belief update using Eqs. (4)
        Perform rule aggregation using Eq. (5)
        Convert crisp value Y_i from fuzzy value generated from rule aggregation using Eq. (6)
    end for
end procedure
```

Learning and BRBES will help to improve the prediction accuracy of BRB-DL due to the accurate pattern discovery from data while addressing uncertainty. Furthermore, the learning parameters of the BRB-DL need to be optimised to improve the prediction accuracy. Learning for the BRB-DL method will be described in the next section.

## VI. LEARNING FOR BRB-DL

The learning procedure plays an important roles for the BRB-DL to learn about the learning parameters from the training dataset. These parameters are generally assigned by domain experts, or randomly selected. For BRBES the common learning parameters are attribute weights ($\theta_k$), rule weights ($\delta_i$), and belief degrees ($\beta_k$). Additional parameters such as weights of neuron ($c_i$) and bias ($b_i$) are included as required by the DNN method of the deep learning process. The antecedent attributes and belief rules are prioritized using the attribute weights and rule weights consecutively. Belief degrees of the consequent attribute is used to present the uncertainty of the output. Hence, the learning parameters are essential for BRB-DL. Therefore, a suitable method is needed to find the optimal values of the learning parameters. By training the BRB-DL with data, the optimal values of the learning parameters can be discovered [24].

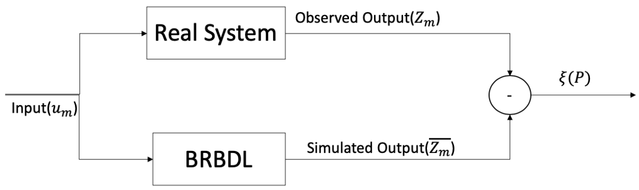



**FIGURE 3. The learning process of the BRBDL.**

The learning parameters need to be trained to determine the optimal values by an objective function considering the linear equality and inequality constraints. The output from BRB-DL is considered as simulated output ($z_m$) and output from the system is named observed output ($\bar{z}_m$). The difference $\xi(p)$ between simulated and observed output needs to be minimized by the optimization process, as shown in Fig. 3. The training sample contains $M$ data points, where the input for BRB-DL is $u_m$, the observed output is $\bar{z}_m$, and the simulated output is $z_m$ $(m = 1, \ldots, M)$. The error $\xi(p)$ is measured by using Eq. (13).

$$\xi_{(p)} = \frac{1}{M}\sum_{m=1}^{M}(z_m - \bar{z}_m)^2 \tag{13}$$

The optimisation of the learning parameters is defined as:

$$\min_P \xi(p) \quad P = P(\mu(O_j), \theta_k, \delta_k, \beta_{jk}, c_i, b_i) \tag{14}$$

The objective function for training the BRB-DL consists of Eqs. (5) and (6). Furthermore, to ensure the completeness of the belief rules, the summations of the belief degree for each rule should be one. Additionally, the values of attribute weights, rule weights, and belief degrees should be between zero (0) and one (1). Henceforth, to enforce the above-mentioned criteria the following constraints are considered:

- Utility values of the consequent and antecedent attributes $\mu(O_j)(j = 1, \ldots, n)$:

$$\mu(O_i) < \mu(O_j); \quad \text{If } i < j$$

- Rule weights $\theta_k(k = 1, \ldots, K)$:

$$1 \geq \theta_k \geq 0;$$

- Antecedent attribute weights $\delta_k$, $(k = 1, \ldots, K)$:

$$1 \geq \delta_k \geq 0;$$

- Consequent belief degrees for the $k$th rule $\beta_{jk}$, $(j = 1, \ldots, n, k = 1, \ldots, L)$:

$$1 \geq \beta_{jk} \geq 0;$$

$$\sum_{j=1}^{n} \beta_{jk} = 1;$$

- Weights of the neurons $c_i (i = 1, \ldots, N)$:

$$1 \geq c_i \geq 0;$$

- Bias of the neurons $b_i (i = 1, \ldots, N)$:

$$1 \geq b_i \geq 0;$$

We have used the *fmincon* function, available in Matlab as an optimal training procedure to determine optimal values of learning parameters, weights of neurons, and bias values.

## VII. RESULTS

In this section, the performance of the proposed new method is evaluated in detail. Evaluation techniques play a significant role in measuring the performance of a method. Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) are some of the standard techniques used for comparing the performance among different methods [38]. Mean Square Error (MSE) has been used to evaluate the performance of the proposed BRB-DL method and others, which is very commonly used for performance measurement. Our proposed BRB-DL method has been compared with various Deep Learning methods such as DNN [34], LSTM [34], FDNN [23], Adaptive-network-based fuzzy inference system (ANFIS) [39], BRBES with *fmincon* [24], and eBRBaDE [27]. In this study, PM2.5 values from Beijing and electrical energy output (EP) of a combined cycle power plant (CCPP) have been used for comparison among different methods.

### A. USE CASE SCENARIO

We have used two datasets for measuring the accuracy of the BRB-DL method. The first dataset was gathered from a combined cycle power plant. The data was collected over six years, from 2006 to 2011. The dataset contains 9,568 data points, which comprises hourly average ambient variables temperature (T), ambient pressure (AP), relative humidity (RH), exhaust vacuum (V), and net hourly electrical energy output (EP) of the plant [40]. The experiments were conducted using a MacBook Pro with an Intel Core i7 processor (2.2 GHz) and 16 GB RAM.

The second dataset contains air quality data from Beijing [41]. The air quality data (PM2.5) were collected from twelve air quality monitoring sites in Beijing. The meteorological data of the dataset were collected from the nearest weather station of the China Meteorological Administration. The dataset contains PM2.5 values, dew point, temperature, pressure, combined wind direction, cumulated wind speed, cumulated hours of snow, and cumulated hours of rain. The data was gathered over four years from 1 March 2013 to 28 February 2017 and contained 43,824 data points.

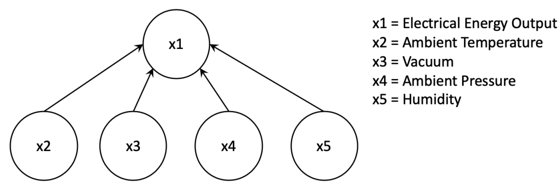



**FIGURE 4.** BRB tree for EP prediction.

**TABLE 1.** Configuration of LSTM, DNN, FDNN, BRB-DL for EP prediction.

| | No of hidden layers | Neurons in each hidden layer |
|---|---|---|
| LSTM | 3 | 100 |
| DNN | 2 | 400 |
| FDNN | 1 | 100 |
| BRB-DL | 3 | 100 |

### B. PERFORMANCE OF BRB-DL FOR EP PREDICTION

The dataset of a combined cycle power plant has been divided into a 80:20 ratio for training and test datasets. We have used $k$-fold cross-validation for our experiments. Usually, 5 or 10 folds are commonly used for cross-validation [38]. Considering the execution time, fivefold cross-validation is used in these experiments. The hourly average ambient variables temperature (x2), exhaust vacuum (x3), ambient pressure (x4), and relative humidity (x5) have been used as antecedent attributes and electric power output (x1) as the consequent attribute. The BRB tree is shown in Fig. 4, which illustrates the antecedent (x2, x3, x4, and x5) and consequent (x1) attributes for the initial rule base. Each attribute has three referential values.

The DNN of BRB-DL contains five layers. The input layer consists of twelve neurons, while three hidden layers each have twelve neurons, and the output layer has three neurons. For predicting EP, we have considered the disjunctive BRB as it require less computational time. The interior-point algorithm of the Matlab *fmincon* tool has been used as the learning mechanism for BRB-DL. Furthermore, for LSTM, we have considered five layers, and four layers for DNN based on the empirical analysis. Adam was used as learning mechanism for LSTM and DNN. For FDNN we have used one layer of DNN. ANFIS configured with four Gaussian membership function. The detail of the LSTM, DNN, FDNN and BRB-DL are presented in Table 1. The configurations have been selected based on empirical analysis

The second, third, fourth, fifth, sixth, seventh, and eighth columns of Table 2 presents the MSE values of predicted EP of the five folds during the training of LSTM, DNN, FDNN, ANFIS, BRB-DL, and BRBES with learning mechanisms eBRBaDE and *fmincon* respectively. The average MSE values of predicted EP by LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON have been illustrated by Fig. 5. BRBES-eBRBaDE and BRBES-FMINCON predicted EP with average MSE of 27.17 and 28.65, which shows that BRBES-eBRBaDE performs better than the

**TABLE 2.** MSE values of EP prediction by LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON of training dataset.

| | LSTM | DNN | FDNN | ANFIS | BRB-DL | BRBES-eBRBaDE | BRBES-FMINCON |
|---|---|---|---|---|---|---|---|
| 1st Fold | 17.89 | 31.25 | 15.23 | 16.65 | 16.29 | 32.87 | 29.79 |
| 2nd Fold | 16.74 | 32.67 | 16.54 | 16.87 | 17.04 | 34.65 | 29.22 |
| 3rd Fold | 16.74 | 23.34 | 16.34 | 16.01 | 15.62 | 22.77 | 28.09 |
| 4th Fold | 14.76 | 34.93 | 16.24 | 15.99 | 15.83 | 22.13 | 29.19 |
| 5th Fold | 17.04 | 22.21 | 16.19 | 16.20 | 15.60 | 23.43 | 26.95 |
| Average | 16.64 | 28.88 | 16.11 | 16.34 | 16.08 | 27.17 | 28.65 |

**TABLE 3.** MSE values of EP prediction by LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON of the test dataset.

| | LSTM | DNN | FDNN | ANFIS | BRB-DL | BRBES-eBRBaDE | BRBES-FMINCON |
|---|---|---|---|---|---|---|---|
| 1st Fold | 16.74 | 28.93 | 17.23 | 15.13 | 3.86 | 46.77 | 22.16 |
| 2nd Fold | 16.85 | 30.11 | 17.65 | 14.32 | 3.90 | 56.47 | 25.65 |
| 3rd Fold | 19.35 | 24.20 | 16.91 | 17.69 | 4.27 | 38.58 | 29.15 |
| 4th Fold | 21.13 | 36.63 | 17.51 | 17.77 | 4.38 | 24.38 | 32.65 |
| 5th Fold | 19.23 | 22.60 | 15.97 | 16.92 | 4.21 | 24.57 | 36.14 |
| Average | 18.66 | 28.49 | 17.05 | 16.37 | 4.12 | 38.15 | 29.15 |

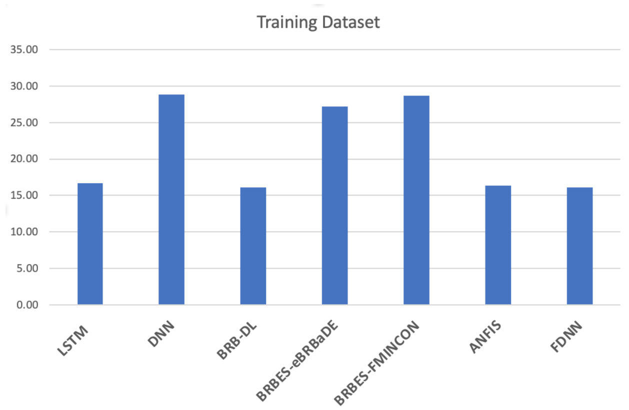



**FIGURE 5.** Comparison of MSE for LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, BRBES-FMINCON, ANFIS, and FDNN for training dataset.

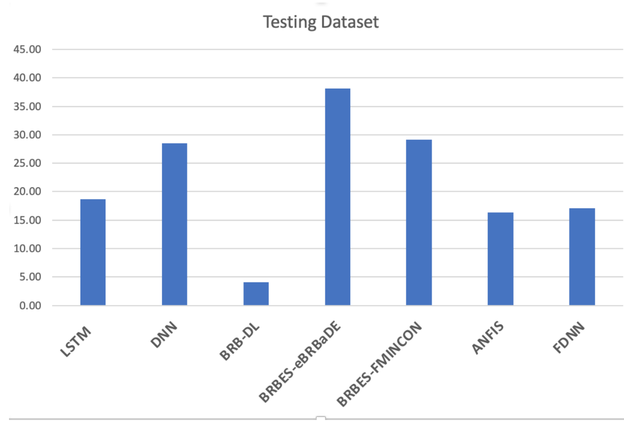



**FIGURE 6.** Comparison of MSE for LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, BRBES-FMINCON, ANFIS, and FDNN for the test dataset.

BRBES-FMINCON during training due to the eBRBaDE training procedure. LSTM and DNN predicted EP with average MSE of 16.64 and 28.88, whereas BRB-DL predicted EP with average MSE of 16.08. FDNN and ANFIS predicted EP with average MSE of 16.11 and 16.08 Therefore, BRB-DL performed better than LSTM and DNN due to the associative memory based Deep Learning method and addressing the uncertainty of data. Furthermore, BRB-DL has the lowest MSE values among LSTM, DNN, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON, which is also visible in Fig. 5.

Table 3 presents the MSE values of predicted EP of the five folds during testing of LSTM, DNN, FDNN, ANFIS, BRB-DL, and BRBES with learning mechanisms eBRBaDE and *fmincon* respectively. The average MSE values of five folds of LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON are 18.66, 28.49, 17.05, 16.37, 4.12, 38.15, and 29.15 respectively. Therefore, it can be concluded that BRB-DL is performing better than LSTM, DNN, FDNN, ANFIS, BRBES-eBRBaDE, and BRBES-FMINCON. Furthermore, LSTM is performing better than DNN as LSTM can memorize the historical context of the data. However, BRBES-FMINCON is doing better than BRBES-eBRBaDE. Fig. 6 illustrates the average MSE values of LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON, where it can also be seen that BRB-DL is performing better than other methods due to the incorporation of the association memory based Deep Learning method.

Fig. 7 presents a comparison between actual EP values and the predicated EP values by LSTM, FDNN, ANFIS, DNN, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON of the test dataset. For better visualization of predicted output of BRB-DL, a comparison of actual EP values and predicted EP values by BRB-DL is illustrated in Fig. 8.

From the results of training and testing, it can be observed that MSE values of predicted EP by BRB-DL are better than the LSTM, DNN, FDNN, ANFIS, BRBES with learning mechanism eBRBaDE and BRBES with *fmincon* respectively. It can also be observed that BRB-DL is performing better than the BRBES with eBRBaDE and BRBES with *fmincon* respectively. The incorporation of the associative memory with BRBES helped BRB-DL to predict EP with higher accuracy than BRBES as there is no associative memory in BRBES. Besides, BRB-DL due to its capability of addressing uncertainty of data helped to predict EP with higher accuracy than LSTM, DNN. As fuzzy system do not address all type of uncertainty, FDNN and ANFIS is performing worst than BRB-DL. Therefore, it can be summarized that the incorporation of the associative memory based deep learning processing step with BRBES helped BRB-DL to discover accurate patterns from data with their associated uncertainty.

### *C. PERFORMANCE OF BRB-DL FOR PM 2.5 PREDICTION*

To further evaluate the performance of our proposed BRB-DL method, a dataset with air quality data from Beijing has been used. The dataset was divided into a 80:20 ratio for training and test datasets. We have used five-fold cross-validation for evaluating the performance. The dew (x2), wind direction (x3), and wind speed (x4) have been used as antecedent attributes and PM2.5 (x1) as the consequent attribute.

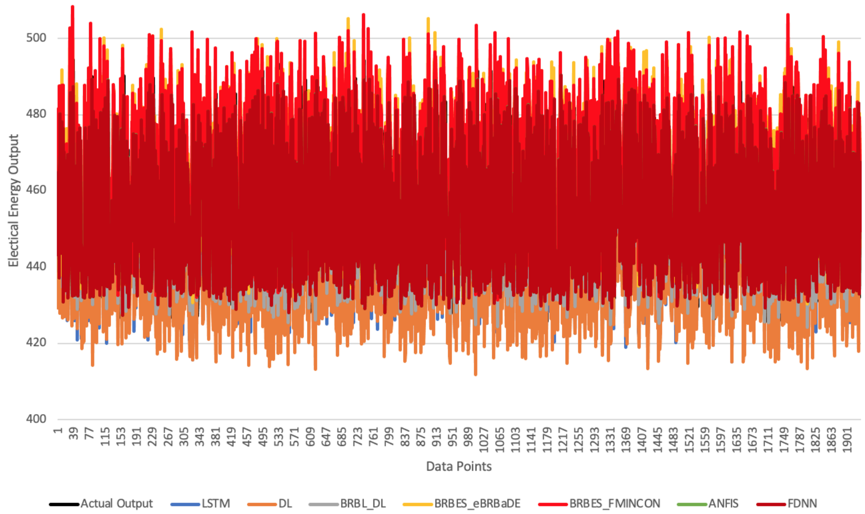



**FIGURE 7.** Comparison of actual EP values and predicted EP values by actual EP, LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON for the test dataset.

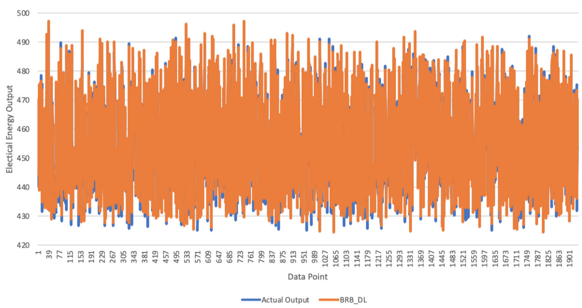



**FIGURE 8.** Comparison of actual EP values and predicted EP values by BRB-DL for the test dataset.

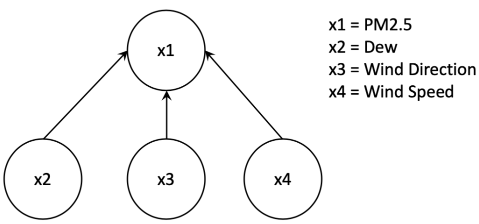



**FIGURE 9.** BRB tree for PM2.5 prediction.

The BRB tree is shown in Fig. 9, which illustrates the antecedent and consequent attributes for the initial rule base, where each attribute has three referential values.

The BRB-DL has five Deep Learning layers. These are one input layer, three hidden layers, and one output layer. The input, hidden, and output layers have nine, twelve, and three neurons, respectively. For predicting PM2.5 values, we have considered the disjunctive BRB due to its less computational time. The interior-point algorithm of the Matlab *fmincon* tool has been used. The configuration of the deep learning layers has been decided based on empirical studies. Furthermore, for LSTM and DNN we have considered five layers based on the empirical analysis. Adam was used as learning mechanism for LSTM and DNN based on empirical analysis. For FDNN we have used one layer of DNN. ANFIS configured with four Gaussian membership functions. The detail of the LSTM, DNN, FDNN and BRB-DL are presented in Table 1. The configurations have been selected based on empirical analysis.

**TABLE 4.** Configuration of LSTM, DNN, FDNN, BRB-DL for PM 2.5 prediction.

| | No of hidden layers | Neurons in each hidden layer |
|---|---|---|
| LSTM | 5 | 100 |
| DNN | 5 | 400 |
| FDNN | 1 | 100 |
| BRB-DL | 3 | 100 |

Table 5 presents the MSE values of PM2.5 of the five folds during the training of LSTM, DNN, FDNN, ANFIS, BRB-DL, and BRBES with learning mechanisms eBRBaDe and *fmincon* on second, third, forth, fifth, sixth, seventh, and eighth columns respectively. The average MSE values of five folds of LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON are 0.00337, 0.00537, 0.00367, 0.00701, 0.00317, 0.00479, and 0.00757 respectively, which shows that BRB-DL is performing better than other methods. Furthermore, it can be observed from Table 5 that average MSE value for LSTM is 0.00337 and DNN is 0.00537, which shows that LSTM is performing better than DNN. From Table 5, it can also be concluded that BRBES-eBRBaDE is performing better than BRBES-FMINCON since the average MSE values for BRBES-eBRBaDE and BRBES-FMINCON are 0.00479 and 0.00757 respectively. The average MSE values of LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON are shown in Fig. 10. From Fig. 10, it can be concluded that BRB-DL is predicting PM2.5 with the lowest average MSE values among LSTM, DNN, FDNN, ANFIS, BRBES-eBRBaDE, and BRBES-FMINCON.

**TABLE 5.** MSE values of PM2.5 prediction by LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON of the training dataset.

| | LSTM | DNN | FDNN | ANFIS | BRB-DL | BRBES-eBRBaDE | BRBES-FMINCON |
|---|---|---|---|---|---|---|---|
| 1st Fold | 0.00451 | 0.00651 | 0.0039 | 0.00696477 | 0.0039 | 0.0056 | 0.00810 |
| 2nd Fold | 0.00145 | 0.00345 | 0.00287 | 0.00694875 | 0.00187 | 0.00317 | 0.00784 |
| 3rd Fold | 0.00224 | 0.00424 | 0.00351 | 0.00738465 | 0.00288 | 0.00424 | 0.00767 |
| 4th Fold | 0.0022 | 0.0042 | 0.00376 | 0.00683483 | 0.00286 | 0.00461 | 0.00711 |
| 5th Fold | 0.00644 | 0.00844 | 0.00429 | 0.0069574 | 0.00435 | 0.00635 | 0.00714 |
| Average | 0.00337 | 0.00537 | 0.003666 | 0.00701808 | 0.00317 | 0.00479 | 0.00757 |

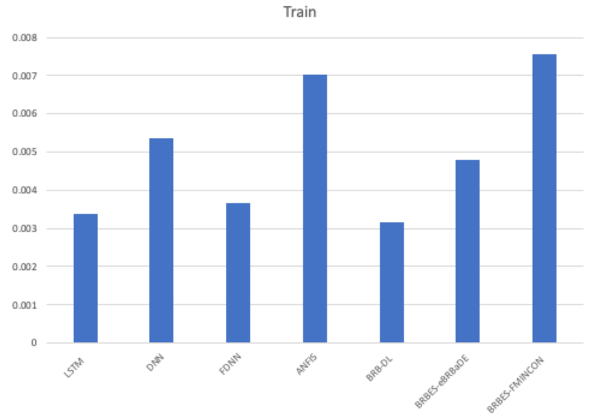



**FIGURE 10.** Comparison of MSE for LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON for the training dataset.

**TABLE 6.** MSE values of PM2.5 prediction by LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON for the test dataset.

| | LSTM | DNN | FDNN | ANFIS | BRB-DL | BRBES-eBRBaDE | BRBES-FMINCON |
|---|---|---|---|---|---|---|---|
| 1st Fold | 0.00645 | 0.00655 | 0.00351 | 0.00724 | 0.00371 | 0.00375 | 0.00366 |
| 2nd Fold | 0.00272 | 0.00282 | 0.00263 | 0.00724 | 0.00173 | 0.00143 | 0.00450 |
| 3rd Fold | 0.00238 | 0.00248 | 0.00245 | 0.00556 | 0.00275 | 0.00358 | 0.00484 |
| 4th Fold | 0.00184 | 0.00194 | 0.00267 | 0.00776 | 0.00177 | 0.00384 | 0.00522 |
| 5th Fold | 0.00076 | 0.00086 | 0.00231 | 0.00730 | 0.00280 | 0.00276 | 0.00453 |
| Average | 0.00283 | 0.00293 | 0.00271 | 0.00702 | 0.00255 | 0.00307 | 0.00455 |

The second, third, forth, fifth, sixth seventh, and eighth columns of Table 6 presents the MSE values of predicted PM2.5 of the five folds during testing of LSTM, DNN, FDNN, ANFIS, BRB-DL, and BRBES with learning mechanisms eBRBaDE and *fmincon* respectively. From Table 6, it can be observed that BRBES-eBRBaDE and BRBES-FMINCON have average MSE values of 0.00307 and 0.00455, which shows that BRBES-eBRBaDE is performing better in predicting PM2.5 than BRBES-FMINCON for the test dataset. According to Table 6, LSTM and DNN have average MSE values of 0.00283 and 0.00293 respectively. This shows that LSTM is performing better than DNN for the test dataset. Furthermore, from Table 6, it can be observed that BRB-DL is performing better than all the methods as the average MSE value of BRB-DL is 0.0025. The average MSE values of five folds of LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON are shown in Fig. 11.

Our proposed BRB-DL method has predicted PM2.5 values with less error than BRBES with learning mechanism eBRBaDE and *fmincon* for training and testing. BRB-DL, with its additional capability of associative memory-based DL method, can discover more accurate patterns form data

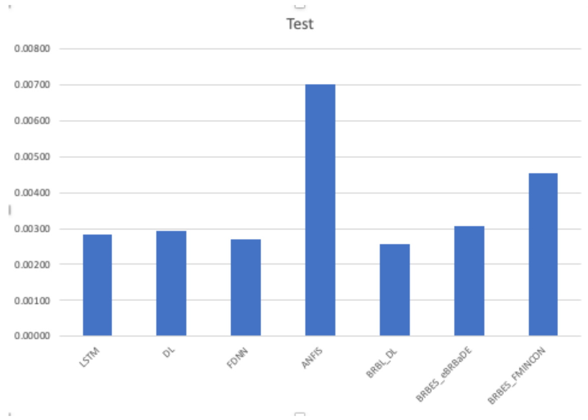



**FIGURE 11.** Comparison of MSE for LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON for the test dataset.

than BRBES since BRBES does not have any associative memory. It can also be observed that BRBES with eBRBaDE performed better than BRBES with *fmincon* due to eBRBaDE's capability of optimal exploration and exploitation of the search space. From training and testing, it can be observed that LSTM is predicting PM2.5 values with lower MSE than DNN due to LSTM's capability of keeping a historical context of the input data. BRB-DL has predicted PM2.5 values with lower MSE than LSTM and DNN during training and testing. Due to the associative memory-based DL method, BRB-DL was able to discover patterns of the data and address uncertainty of data using the ER based inference mechanism.

Fig. 12 presents a comparison between actual PM2.5 values and the predicated PM2.5 values by LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON of the test dataset. For better visualization of predicted output of BRB-DL, a comparison of actual PM2.5 values and predicted PM2.5 values by BRB-DL is illustrated in Fig. 13.

In summary, it can be observed that our proposed BRB-DL method is able to better predict with higher accuracy for two different datasets than other Deep Learning methods such as LSTM, DNN, and BRBES with learning mechanism eBRBaDE and *fmincon*. Deep Learning methods like DNN and LSTM are not able to address uncertainty of data as it is based on Neural Networks, inherently limited in addressing uncertainty. BRBES is able to address uncertainty due to its ER based inference mechanism and incorporation of belief structure with the rule base. However, BRBES lacks a mechanism for discovering patterns of data. The incorporation of the DL method with BRBES enables BRB-DL to discover accurate patterns of data due to the associative memory. The BRB-DL method is able to address uncertainty due to the inherent capability of BRBES. For the aforementioned features, BRB-DL predicted values with the lowest MSE compared to other DL methods such as LSTM, DNN, and BRBES with learning mechanism eBRBaDE and *fmincon* respectively. Furthermore, BRB-DL has been compared with fuzzy based systems like FDNN and ANFIS. As fuzzy systems are not able to address all types of uncertainty, their prediction

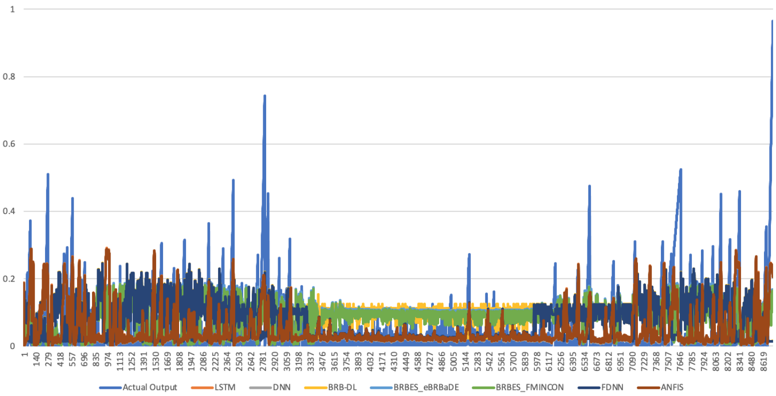



**FIGURE 12.** Comparison of actual PM2.5 values and predicted PM2.5 values by LSTM, DNN, FDNN, ANFIS, BRB-DL, BRBES-eBRBaDE, and BRBES-FMINCON for the test dataset.

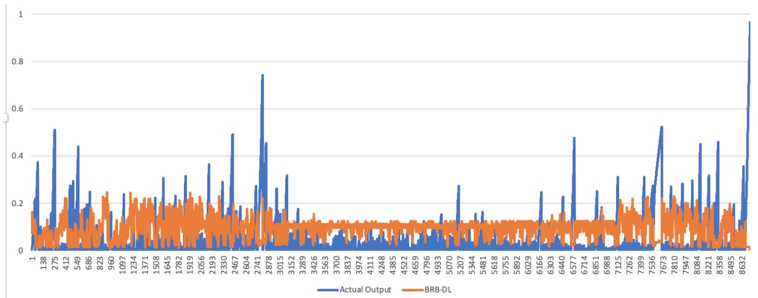



**FIGURE 13.** Comparison of actual PM2.5 values and predicted PM2.5 values for the test dataset by BRB-DL.

accuracy was not better than BRB-DL. Therefore, it can be argued the integration of Deep Learning with BRBES helps improving the prediction accuracy by BRB-DL.

## VIII. CONCLUSION

In our proposed BRB-DL method, a deep processing layer has been added replacing the activation weight calculation step of BRBES. DNN, which is a multi-layered neural network, is used as the Deep Learning method. The deep processing layer of BRB-DL contains the associative memory. The activation weight calculation step of BRBES uses multiplication, summation, and division operators using matching degrees and rule weights as shown in Eq. (2) and (3). Therefore, this step lacks any component of associative memory, resulting in incomplete calculation of rule activation weights. However, the deep processing layer of BRB-DL enables the calculation of complete values of the rule activation weights as described in Section V. Therefore, BRB-DL with the integration of associative memory based Deep Learning method within BRBES inference procedures allows more accurate prediction under uncertainty. Furthermore, during the learning process of BRB-DL, the inclusion of additional parameters such as weights and bias of neurons played an important role to increase prediction accuracy as discussed in Section VI. In this way, the present BRBES inference framework has been advanced due to the inclusion of DNN. This novel BRB-DL method has been applied to predict electric energy output of a combined cycle power plant and air quality (PM2.5) values in Beijing city. The results of BRB-DL have been compared with other Deep Learning methods including LSTM, DNN ans fuzzy based systems like FDNN and ANFIS as well as with BRBES. In case of LSTM and DNN, the Adam learning mechanism was used, while for BRBES, eBRBaDE and *fmincon* learning mechanism were used. However, in case of BRB-DL, *fmincon* was used in the light of the learning mechanism as described in Section VI, where additional learning parameters such as weight and bias of neurons are considered. The MSE values of LSTM and DNN with the Adam learning mechanism were 18.66 and 28.49 respectively. The MSE values of FDNN and ANFIS were 17.05 and 16.37 respectively. The MSE value of BRBES with *fmincon* was 29.15, while it was 38.15 for eBRBaDE. However, the MSE value of BRB-DL was found to be 4.12, which is better than the other methods. The reason for this is that the inclusion of DNN within the BRBES inference framework played an important role to increase the accuracy of prediction. From these results, it can be argued that a significant advancement of the accurate prediction capability has been achieved with the novel method proposed in this study. For the air quality (PM2.5) prediction in Beijing city, the MSE values of BRB-DL, LSTM, DNN, FDNN, ANFIS and BRBES with learning mechanism eBRBaDE and *fmincon* are 0.00317, 0.00337, 0.00367, 0.00701, 0.00537, 0.00479, and 0.00757 respectively. The results show that BRB-DL performs better than Deep learning methods such as LSTM, DNN, and BRBES with learning mechanism eBRBaDE and *fmincon* respectively. However, the performance of the BRB-DL needs to be evaluated with datasets from diverse domains to ensure its efficiency and robustness further. The proposed BRB-DL method has been evaluated using numerical data mostly. Therefore, BRB-DL might have limitation for visual and acoustic data like, images and audio. BRB-DL have been presented for solving regression problem which might not be suitable for classification and cluster-based methods. Furthermore, BRB-DL can be used for addressing supervised learning.

## REFERENCES

[1] C.-T. Lin and C. S. G. Lee, "Neural-network-based fuzzy logic control and decision system," *IEEE Trans. Comput.*, vol. 40, no. 12, pp. 1320–1336, Dec. 1991.

[2] H. Keung Kwan and Y. Cai, "A fuzzy neural network and its application to pattern recognition," *IEEE Trans. Fuzzy Syst.*, vol. 2, no. 3, pp. 185–193, Aug. 1994.

[3] M. K. Mehlawat and P. Gupta, "Fuzzy chance-constrained multiobjective portfolio selection model," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 3, pp. 653–671, Jun. 2014.

[4] F.-J. Lin, C.-H. Lin, and P.-H. Shen, "Self-constructing fuzzy neural network speed controller for permanent-magnet synchronous motor drive," *IEEE Trans. Fuzzy Syst.*, vol. 9, no. 5, pp. 751–759, Oct. 2001.

[5] C.-F. Juang and C.-T. Lin, "An online self-constructing neural fuzzy inference network and its applications," *IEEE Trans. Fuzzy Syst.*, vol. 6, no. 1, pp. 12–32, 1998.

[6] R. Ul Islam, M. S. Hossain, and K. Andersson, "A novel anomaly detection algorithm for sensor data under uncertainty," *Soft Comput.*, vol. 22, no. 5, pp. 1623–1639, Mar. 2018.

[7] J.-B. Yang, J. Liu, J. Wang, H.-S. Sii, and H.-W. Wang, "Belief rule-base inference methodology using the evidential reasoning approach-RIMER," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 36, no. 2, pp. 266–285, Mar. 2006.

[8] R. Ul Islam, K. Andersson, and M. S. Hossain, "A Web based belief rule based expert system to predict flood," in *Proc. 17th Int. Conf. Inf. Integr. Web Appl. Services (iiWAS)*, 2015, pp. 19–26.

[9] M. R. Sabbir Hridoy, M. S. Hossain, R. U. Islam, and K. Andersson, "A Web based belief rule based expert system for assessing flood risk," in *Proc. 19th Int. Conf. Inf. Integr. Web Appl. Services (iiWAS)*, 2017, pp. 434–440.

[10] M. S. Hossain, I. B. Habib, and K. Andersson, "A belief rule based expert system to diagnose dengue fever under uncertainty," in *Proc. Comput. Conf.*, Jul. 2017, pp. 17–23.

[11] R. Karim, K. Andersson, M. S. Hossain, M. J. Uddin, and M. P. Meah, "A belief rule based expert system to assess clinical bronchopneumonia suspicion," in *Proc. Future Technol. Conf. (FTC)*, Dec. 2016, pp. 655–660.

[12] M. S. Hossain, F. Ahmed, Fatema-Tuj-Johora, and K. Andersson, "A belief rule based expert system to assess tuberculosis under uncertainty," *J. Med. Syst.*, vol. 41, no. 3, p. 43, Mar. 2017.

[13] L. Dymova, P. Sevastjanov, and K. Kaczmarek, "A forex trading expert system based on a new approach to the rule-base evidential reasoning," *Expert Syst. Appl.*, vol. 51, pp. 1–13, Jun. 2016.

[14] L. Chang, X. Ma, L. Wang, and X. Ling, "Comparative analysis on the conjunctive and disjunctive assumptions for the belief rule base," in *Proc. Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discovery (CyberC)*, Oct. 2016, pp. 153–156.

[15] L. Chang, Z. Zhou, Y. You, L. Yang, and Z. Zhou, "Belief rule based expert system for classification problems with new rule activation and weight calculation procedures," *Inf. Sci.*, vol. 336, pp. 75–91, Apr. 2016.

[16] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.

[17] W. Chen, J. An, R. Li, L. Fu, G. Xie, M. Z. A. Bhuiyan, and K. Li, "A novel fuzzy deep-learning approach to traffic flow prediction with uncertain spatial–temporal data features," *Future Gener. Comput. Syst.*, vol. 89, pp. 78–88, Dec. 2018.

[18] M. P. Singh and V. K. Saraswat, "Multilayer feed forward neural networks for non-linear continuous bidirectional associative memory," *Appl. Soft Comput.*, vol. 61, pp. 700–713, Dec. 2017.

[19] Z.-J. Zhou, G.-Y. Hu, C.-H. Hu, C.-L. Wen, and L.-L. Chang, "A survey of belief rule-base expert system," *IEEE Trans. Syst., Man, Cybern., Syst.*, early access, Nov. 7, 2019, doi: 10.1109/TSMC.2019.2944893.

[20] Y.-W. Chen, J.-B. Yang, D.-L. Xu, Z.-J. Zhou, and D.-W. Tang, "Inference analysis and adaptive training for belief rule based systems," *Expert Syst. Appl.*, vol. 38, no. 10, pp. 12845–12860, Sep. 2011.

[21] B. Qian, Q.-Q. Wang, R. Hu, Z.-J. Zhou, C.-Q. Yu, and Z.-G. Zhou, "An effective soft computing technology based on belief-rule-base and particle swarm optimization for tipping paper permeability measurement," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 3, pp. 841–850, Mar. 2019.

[22] A. Merentitis and C. Debes, "Automatic fusion and classification using random forests and features extracted with deep learning," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2015, pp. 2943–2946.

[23] Y. Deng, Z. Ren, Y. Kong, F. Bao, and Q. Dai, "A hierarchical fused fuzzy deep neural network for data classification," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 4, pp. 1006–1012, Aug. 2017.

[24] J.-B. Yang, J. Liu, D.-L. Xu, J. Wang, and H. Wang, "Optimization models for training belief-rule-based systems," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 37, no. 4, pp. 569–585, Jul. 2007.

[25] L.-L. Chang, Z.-J. Zhou, Y.-W. Chen, T.-J. Liao, Y. Hu, and L.-H. Yang, "Belief rule base structure and parameter joint optimization under disjunctive assumption for nonlinear complex system modeling," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 9, pp. 1542–1554, Sep. 2018.

[26] L.-H. Yang, Y.-M. Wang, J. Liu, and L. Martínez, "A joint optimization method on parameter and structure for belief-rule-based systems," *Knowl.-Based Syst.*, vol. 142, pp. 220–240, Feb. 2018.

[27] R. U. Islam, M. S. Hossain, and K. Andersson, "A learning mechanism for BRBES using enhanced belief rule-based adaptive differential evolution," in *Proc. 9th Int. Conf. Inform., Electron. Vis. (ICIEV)*, Aug. 2020, pp. 1–10.

[28] G. Li, Z. Zhou, C. Hu, L. Chang, Z. Zhou, and F. Zhao, "A new safety assessment model for complex system based on the conditional generalized minimum variance and the belief rule base," *Saf. Sci.*, vol. 93, pp. 108–120, Mar. 2017.

[29] C. Cheng, X. Qiao, W. Teng, M. Gao, B. Zhang, X. Yin, and H. Luo, "Principal component analysis and belief-rule-base aided health monitoring method for running gears of high-speed train," *Inf. Sci.*, vol. 63, pp. 199202-1–199202-3, 2020.

[30] S. Kabir, R. U. Islam, M. S. Hossain, and K. Andersson, "An integrated approach of belief rule base and deep learning to predict air pollution," *Sensors*, vol. 20, no. 7, p. 1956, Mar. 2020.

[31] M. S. Hossain, S. Rahaman, A.-L. Kor, K. Andersson, and C. Pattinson, "A belief rule based expert system for datacenter PUE prediction under uncertainty," *IEEE Trans. Sustain. Comput.*, vol. 2, no. 2, pp. 140–153, Apr. 2017.

[32] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.

[33] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[34] N. Ketkar, *Deep Learning With Python: A Hands-on Introduction*, vol. 1. Berkeley, CA, USA: Apress, 2017.

[35] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.

[36] M. S. Hossain, M. A. Hasan, M. Uddin, M. M. Islam, and R. Mustafa, "A belief rule based expert system to assess lung cancer under uncertainty," in *Proc. 18th Int. Conf. Comput. Inf. Technol. (ICCIT)*, Dec. 2015, pp. 413–418.

[37] J. Liu, M. Gong, and H. He, "Deep associative neural network for associative memory based on unsupervised representation learning," *Neural Netw.*, vol. 113, pp. 41–53, May 2019.

[38] M. Kuhn and K. Johnson, *Applied Predictive Modeling*. New York, NY, USA: Springer, 2016.

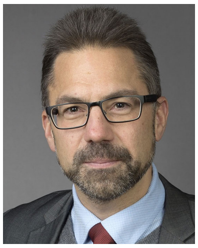
[39] J.-S. R. Jang, "ANFIS: Adaptive-network-based fuzzy inference system," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 3, pp. 665–685, Jun. 1993.

[40] P. Tüfekci, "Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods," *Int. J. Electr. Power Energy Syst.*, vol. 60, pp. 126–140, Sep. 2014.

[41] S. Zhang, B. Guo, A. Dong, J. He, Z. Xu, and S. X. Chen, "Cautionary tales on air-quality improvement in Beijing," in *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 473, no. 2205, pp. 1–14, 2017.

**RAIHAN UL ISLAM** received the M.Sc. degree in mobile systems and the Ph.D. degree in pervasive and mobile computing from the Luleå University of Technology, Sweden. He was a Software Engineer with the NEC Laboratories Europe, Context-Aware Services (CAS) and Smart Environments Technologies Group. His main research interests include expert systems, machine learning, smart cities, M2M communication, mobile systems, and pervasive and ubiquitous computing.

**MOHAMMAD SHAHADAT HOSSAIN** (Senior Member, IEEE) received the M.Phil. and Ph.D. degrees in computation from the University of Manchester Institute of Science and Technology (UMIST), U.K., in 1999 and 2002, respectively. He is currently a Professor of computer science and engineering with the University of Chittagong (CU), Bangladesh. His current research interests include e-government, modeling of risks, uncertainties using evolutionary computing techniques, investigation of pragmatic software development tools, and methods for information systems in general and expert systems in particular.



**KARL ANDERSSON** (Senior Member, IEEE) received the M.Sc. degree in computer science and technology from the Royal Institute of Technology, Stockholm, Sweden, and the Ph.D. degree in mobile systems from the Luleå University of Technology, Sweden. He was with the Internet Real-time Laboratory, Columbia University, New York, NY, USA, and the National Institute of Information and Communications Technology, Tokyo, Japan. He is currently an Associate Professor of pervasive and mobile computing with the Luleå University of Technology. His research interests include green and mobile computing, the Internet of Things, cloud technologies, and information security.