

Random vector functional link network: recent developments, applications, and future directions

A. K. Malik^a, Ruobin Gao^b, M.A. Ganaie^c, M. Tanveer^{a,*}, Ponnuthurai
Nagaratnam Suganthan^{d,e,*}

^a*Department of Mathematics, Indian Institute of Technology Indore, Simrol, Indore,
453552, India*

^b*School of Civil & Environmental Engineering, Nanyang Technological University,
Singapore*

^c*Department of Robotics, University of Michigan, Ann Arbor, MI 48109, USA*

^d*KINDI Center for Computing Research, College of Engineering, Qatar University, Doha,
Qatar*

^e*School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore*

Abstract

Neural networks have been successfully employed in various domains such as classification, regression and clustering, etc. Generally, the back propagation (BP) based iterative approaches are used to train the neural networks, however, it results in the issues of local minima, sensitivity to learning rate and slow convergence. To overcome these issues, randomization based neural networks such as random vector functional link (RVFL) network have been proposed. RVFL model has several characteristics such as fast training speed, direct links, simple architecture, and universal approximation capability, that make it a viable randomized neural network. This article presents the first comprehensive review of the evolution of RVFL model, which can serve as the extensive summary for the beginners as well as practitioners. We discuss the shallow RVFLs, ensemble RVFLs, deep RVFLs and ensemble deep RVFL models. The variations, improvements and applications of RVFL models are discussed in detail. Moreover, we discuss the different hyperparameter optimization techniques followed in the literature to improve the generalization performance of the RVFL model. Finally, we give potential future research directions/opportunities that can inspire the researchers to improve the RVFL's architecture and learning algorithm further.

Keywords: Random vector functional link, Ensemble learning, Deep learning, Ensemble deep learning, Randomized neural networks (RNNs).

*Corresponding authors

Email addresses: phd1801241003@iiti.ac.in (A. K. Malik), gaor0009@e.ntu.edu.sg (Ruobin Gao), mudasirg@umich.edu (M.A. Ganaie), mtanveer@iiti.ac.in (M. Tanveer), p.n.suganthan@qu.edu.qa (Ponnuthurai Nagaratnam Suganthan)

1. Introduction

Artificial intelligence (AI) is a rapidly growing field that has the potential to transform many aspects of the world [1]. It refers to the development of computer systems that can perform tasks that typically require human intelligence, such as learning, decision making, and problem solving [2]. AI has the potential to revolutionize industries and improve efficiencies in a wide range of fields, including healthcare [3], transportation [4], finance [5], and energy [6]. Machine learning algorithms are the engine of AI. Hence, developing advanced machine learning algorithms for various tasks is of real value.

Among machine learning algorithms, the artificial neural networks (ANNs) have received considerable attention due to their success in diverse domains such as medicine [7], chemistry [8], robotics [9], control systems [10], industrial applications and function approximation [11, 12], so on. The architecture of ANN is inspired by the topology of biological neurons [13]. The ANN consists of neurons which are simple processing units and these neurons are connected via weighted links. The neurons do mathematical operations that are either linear or nonlinear, and they carry out some task that enables the artificial neural network (ANN) to approximate the unknown function (rule) that generates the data [14]. The training phase of an ANN is an iterative process, and all the parameters are tuned via the back propagation (BP) method [15]. The traditional iterative techniques based on the BP algorithm have some shortcomings, i.e., slow convergence [16], not getting global minima that leads to generates sub-optimal parameters [17], and very sensitive to learning rate [18].

To overcome the above-mentioned issues, randomization based techniques have been proposed [19, 20] with fast convergence and universal approximation properties. The single hidden layer feed-forward neural (SLFN) network architectures have been extensively studied in the last twenty years and employed in various domains, e.g., classification and regression problem, due to their universal approximation capability [21, 14, 22, 23, 24]. In most of the SLFN networks, the learning process is done in output layer while as weights and biases are generated randomly in hidden layer. The output layer weights are calculated either via closed form solution [25] or iterative process [26]. The origin of randomized feedforward networks can be traced in late 20th century [20]. In 1988, Broomhead and Lowe [27] discussed universal approximation property using radial basis function (RBF) network with random centers [28]. There are several other architectures [29] like RBF network, recurrent neural network (RNN) which have randomization based training algorithms [19].

Moreover, Schmidt et al. [30] proposed feed-forward neural network with random weights based on randomization technique. At the same time, connecting the input layer to the output layer via direct link, Pao et al. [31, 32] proposed random vector functional link (RVFL) neural network in 1992, wherein the parameters (weights and biases) from the input layer to the hidden layer are generated randomly from a fixed domain and the output weights are need to be computed analytically. The idea of the direct link can be traced back to the pioneering work in fuzzy systems [33] in 1985. The direct links have shown sig-

nificant improvement in RBF networks’ performance [34] in 2002. The intention behind creating a direct link between inputs and outputs is to capture information about how the first derivative of the output with respect to the inputs. In 1994, Igel and Pao proved that RVFL is a universal approximator [35]. Recently, Needell et al. [36] proposed RVFL networks for function approximation on manifolds that fill the theoretical gap lacking in [35]. In 2016, Zhang and Suganthan [37] conducted a comprehensive evaluation of RVFL for classification problems and concluded some remarkable results about this architecture. After that, the RVFL model got the attention of researchers from diverse domains due to its simple architecture, fast training speed, and universal approximation capability. The shallow RVFL model has been employed successfully in several domains, i.e., forecasting [38, 39, 40], non-linear system identification [41], function approximation [35], classification and regression problem [42, 43, 44], etc.

The RVFL model transforms the original feature space to randomized feature space via random feature map (RFM), and this randomization process makes the RVFL model an unstable classifier. Ensemble learning techniques develop stable, robust, and accurate model integrating the several models known as base models [45]. Combining the diverse and accurate base models [46], the ensemble model performs better than its constitutes models. Broadly speaking, the ensemble learning can be divided into three categories, i.e., bagging [47], boosting [48] and stacking framework [49]. Thus, the RVFL model has been improved (developed) in ensemble frameworks, and the more stable and robust RVFL variants have been proposed and employed in various domains, i.e., crude oil price forecasting [50], medical domain [51], and classification problem [52], etc. Deep learning architectures have the high representation learning capability due to several stacked layers for extracting informative features [53] and have been successfully employed in several domains, i.e., computer vision [54], bioinformatics [55], and visual tracking [56], and speech recognition task [57] etc. On the other hand, Utilising the strength of two individual growing fields, i.e., ensemble learning and deep learning, researchers are developing ensemble deep models [58, 59]. The shallow RVFL model has been extended to deep and ensemble deep architectures that improve its generalization performance. The deep RVFL network [60] has several stacked layers wherein all parameters of hidden layers are generated randomly and kept fixed during the training process, and only output layer parameters are needed to be computed analytically. The deep RVFL model has better representation learning compared to the shallow RVFL model. The deep RVFL model faces memory issues when training data size, the no. of hidden layers, and the feature dimension of the data are considerable. Therefore, to address these issues, an implicit ensemble technique based ensemble deep RVFL network known as edRVFL model has been proposed [60].

RVFL has been improved in multiple aspects both in shallow and deep frameworks and has been applied in diverse domains. In this paper, we present journey of shallow and deep RVFL along with its applications. We conclude this article with potential future research directions that might inspire researchers to develop this architecture further.

The rest part of this paper is organised as follows. In Section 2, we present the formulation of the standard RVFL model. Section 3 discusses the research methodology and objective of this article. The improvements in shallow RVFL and their applications are discussed in the Section 4. In Section 5 and 6, we discuss the semi-supervised methods and clustering based methods that have been developed based on the RVFL model, respectively. We present the ensemble learning based RVFL model in Section 7 and Section 8 discusses the deep architectures based on the RVFL model. Section 9 discusses the hyper-parameters optimization and experimental setup details and Section 10 discusses the applications of the RVFL model. In Section 11, comparison of RVFL with other machine learning models are given. Finally, the potential future directions with conclusions are given in Section 12.

2. The standard RVFL architecture and it's mathematical foundation

In this section, we discuss formulation of the standard RVFL model. Let $X = [x_1, x_2, \dots, x_N]^T$, $x_i \in \mathbb{R}^d$ be the training dataset and $Y = [y_1, y_2, \dots, y_N]^T$, $y_i \in \mathbb{R}^c$, be the target matrix. Here, d represents the number of features in each sample (x_i) and c denotes the number of classes. Fig. 3 shows the layout of the paper and Fig. 1(b) shows the different types of architectures of the RVFL model.

2.1. Random vector functional link (RVFL) network

RVFL is a randomized version of a single hidden layer feedforward neural (SLFN) network, with three layers known as the input, hidden, and output layers. All three layers consist of neurons are connected via weights. To avoid the implementation of the back propagation algorithm, the weights from the input layer to the hidden layer are generated randomly from a domain and kept fixed during the training process. Only the output weights are analytically computed by the least square method. In RVFL, original features are also used to link the input and output layers. The direct links improve the generalization performance of RVFL [37]. The architecture of RVFL is given in Fig.1 (a). Mathematically, the RVFL model, i.e., $\mathbf{f} : \mathbb{R}^d \rightarrow \mathbb{R}^c$, can be written as:

$$\mathbf{f}(x_i) = \sum_{k=1}^d \beta_k x_{ik} + \sum_{k=d+1}^L \beta_k \theta(\langle \mu_k, x_i \rangle + \sigma_k), \quad i = 1, 2, \dots, N. \quad (1)$$

In particular, $\langle \mu_k, x_i \rangle = \mu_k \cdot x_i$ is the standard inner product defined on Euclidean space ($\mathbb{R}^d$). The objective function of standard RVFL model with L hidden nodes can be written as:

$$\begin{aligned} \min \quad & \frac{1}{2} \|\beta\|^2 + \frac{1}{2} C \|\xi\|^2 \\ \text{subject to} \quad & H\beta - Y = \xi. \end{aligned} \quad (2)$$

where $\|\cdot\|$ represents the Frobenius norm and $\xi = [\xi_1, \xi_2, \dots, \xi_N]^T$ is the error term corresponding to N samples. This is the quadratic optimization problem with linear constraints. β and H are the output weight matrix and concatenation matrix consist of input data and outputs from the hidden layer, respectively and Y is the target matrix.

The optimization problem (2) can be rewritten as:

$$\min_{\beta \in \mathbb{R}^{(d+L) \times c}} \frac{1}{2} \|\beta\|^2 + \frac{1}{2} C \|H\beta - Y\|^2, \quad (3)$$

here,

$$H = [H_1 \ H_2]_{N \times (d+L)},$$

where

$$H_1 = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{Nd} \end{bmatrix}_{N \times d}, \quad (4)$$

and

$$H_2 = \begin{bmatrix} \theta(\mu_1 \cdot x_1 + \sigma_1) & \theta(\mu_2 \cdot x_1 + \sigma_2) & \cdots & \theta(\mu_L \cdot x_1 + \sigma_L) \\ \vdots & \ddots & \vdots & \vdots \\ \theta(\mu_1 \cdot x_N + \sigma_1) & \theta(\mu_2 \cdot x_N + \sigma_2) & \cdots & \theta(\mu_L \cdot x_N + \sigma_L) \end{bmatrix}_{N \times L}, \quad (5)$$

$$\beta = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{(d+L)} \end{bmatrix}_{(d+L) \times c} \quad \text{and} \quad Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}_{N \times c}.$$

Here, $\beta_k = [\beta_{k1}, \beta_{k2}, \dots, \beta_{kc}]$ is the output weight vector connecting the k^{th} input (hidden) node and the output nodes, where $1 \leq k \leq d + L$, and $\mu_j = [\mu_{j1}, \mu_{j2}, \dots, \mu_{jd}]$ is the weight vector connecting the j^{th} hidden node and the input nodes, $1 \leq j \leq L$. Also, $x_i = [x_{i1}, x_{i2}, \dots, x_{id}]$ is the i^{th} sample. For the target matrix, $y_i = [y_{i1}, y_{i2}, \dots, y_{ic}]$, $1 \leq i \leq N$. Moreover, $\theta(\cdot)$ and σ_i are the non-constant activation function and the bias term of i^{th} hidden node, respectively.

The optimal solution of the problem (2) when $\delta = \frac{1}{C} = 0$ is as follows:

$$\beta = H^+ Y, \quad (6)$$

where H^+ represents the Moore-penrose generalized inverse of the matrix H [61]. The regularization term is employed to avoid the over-fitting issue.

Therefore, the optimization problem with regularization term is solved. Let the Lagrangian be

$$L(\beta, \xi, \alpha) = C \frac{1}{2} \|\xi\|^2 + \frac{1}{2} \|\beta\|^2 - \alpha^T (H\beta - Y - \xi), \quad (7)$$

and obtain the partial derivatives of L w.r.t β , ξ and α and set them to zero.

$$\frac{\partial L}{\partial \beta} = 0 \implies \beta = H^T \alpha, \quad (8)$$

$$\frac{\partial L}{\partial \xi} = 0 \implies \alpha = -C\xi, \quad (9)$$

$$\frac{\partial L}{\partial \alpha} = 0 \implies H\beta - Y - \xi = 0. \quad (10)$$

When the number of features is less than number of samples, from (9) and (10), we obtain $\alpha = -C(H\beta - Y)$. By substituting the value of α in (8):

$$\beta = (H^T H + \frac{1}{C} I)^{-1} H^T Y \quad (11)$$

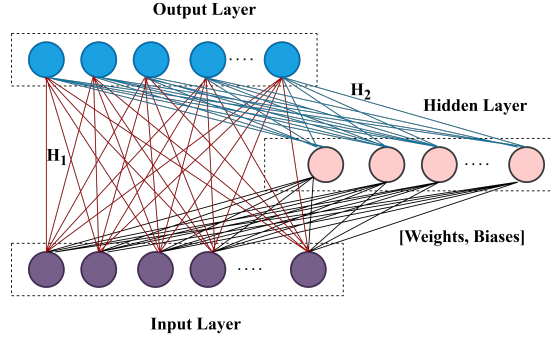
When the number of samples is less than the number of features, after substituting (8) and (9) into (10), we obtain $\alpha = (HH^T + \frac{1}{C} I)^{-1} Y$, By substituting the value of α in (8):

$$\beta = H^T (HH^T + \frac{1}{C} I)^{-1} Y. \quad (12)$$

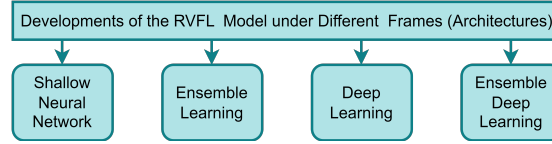
Therefore, in this case the optimal solution of (2) is given by,

$$\beta = \begin{cases} (H^T H + \frac{1}{C} I)^{-1} H^T Y, & (d + L) \leq N \\ H^T (HH^T + \frac{1}{C} I)^{-1} Y, & N < (d + L), \end{cases} \quad (13)$$

where C is the regularization parameter to be tuned and I is an identity matrix of appropriate dimension. Both matrices $H^T H$ and HH^T are symmetric positive semidefinite matrix and $C > 0$, so both matrices in (13) is positive definite, therefore, $(HH^T + \frac{1}{C} I)$ and $(H^T H + \frac{1}{C} I)$ are non singular matrix.



(a) Shallow RVFL: Red lines show the direct links between input layer to output layer, black lines show the connection between input layer to hidden layer and blue lines show the links between hidden layer to output layer.



(b) Different types of architectures of RVFL model.

Figure 1: The architectures of RVFL model.

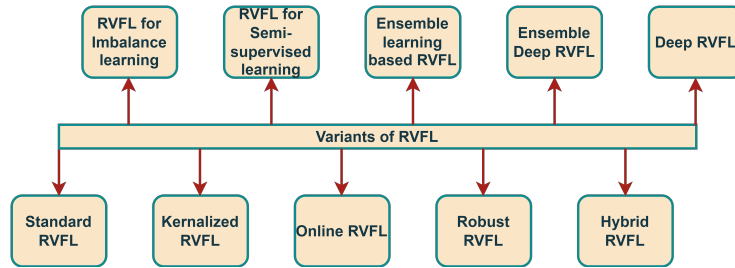


Figure 2: Architectural/Algorithmic Variants of the RVFL model.

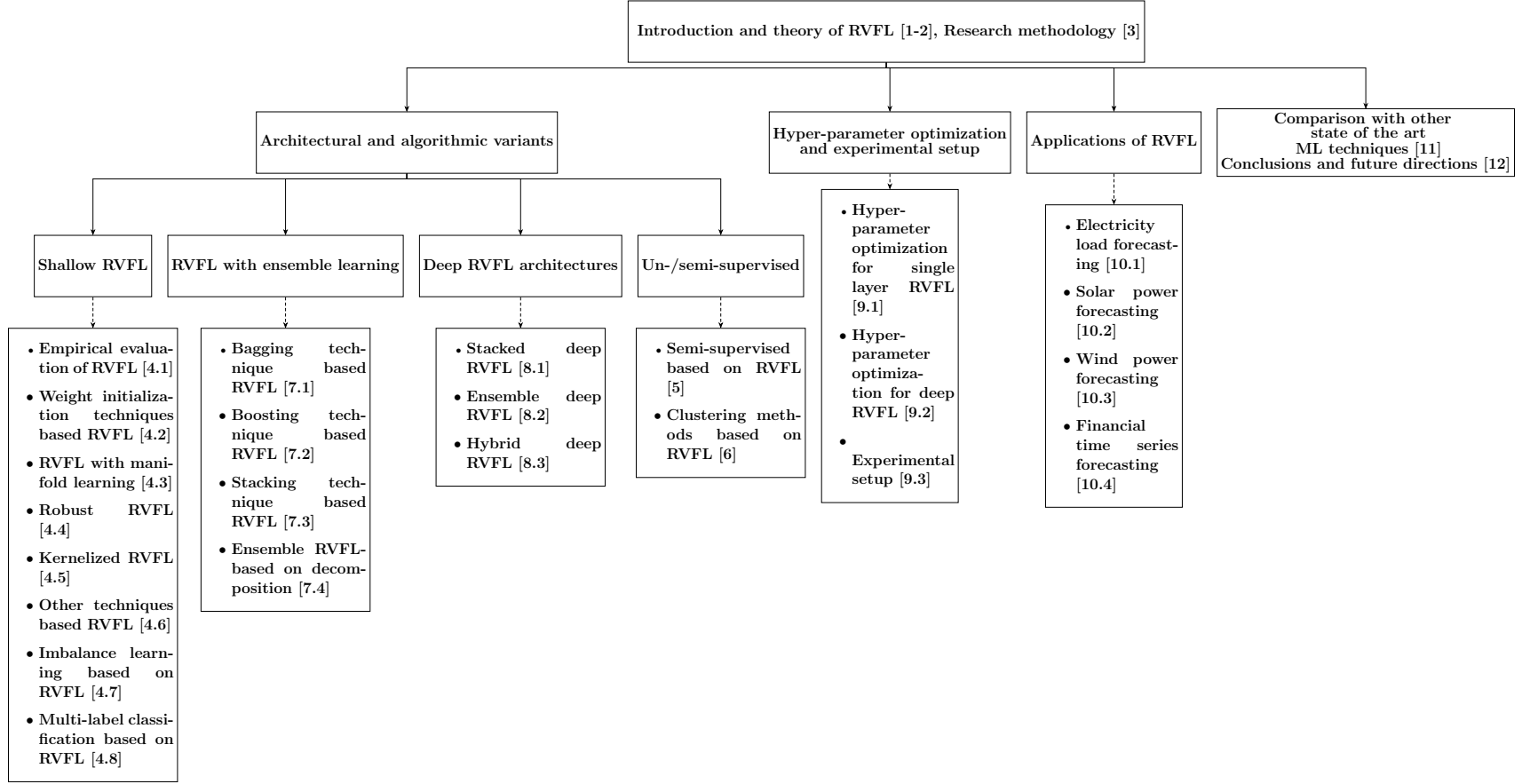


Figure 3: Layout of the paper.

3. Research methodology

The studies included in this review paper were obtained by searching the Google Scholar and Scopus search engines. The papers are the result of keywords random vector functional link and deep RVFL. The articles are screened based on the title and abstract, followed by the screening of full-text version. The focus of this article is to represent the developments of shallow RVFL, ensembles of RVFL models, deep RVFL, ensembles of deep RVFL-based models and their applications. Having these search strategy, we included fundamental papers on RVFL model proposed in the late 20th century and after that we included all RVFL papers from 2016 onward. This paper explains the following issues:

1. How the weights are initialized in RVFL-based models.
2. Techniques used to improve the robustness of the models in presence of noise/outliers and class imbalance problems.
3. Techniques followed for selection of hidden nodes like kernelized approach for RVFL.
4. How different approaches of ensemble learning like bagging, boosting and stacking improve the performance of the RVFL-based ensemble models.
5. Techniques followed for the development of deep RVFL-based models.
6. Development of RVFL-based models for different scenarios like semi-supervised learning, unsupervised learning and regression problems.
7. How to select the different parameters of the RVFL-based models, followed by the techniques for the optimization of these models.
8. Approaches followed for analyzing the effect of hyperparameters on the generalization performance of the RVFL-based models.
9. The loop holes in the current literature of the RVFL-based models and possible future research directions.

4. Developments and applications of RVFL model

In this section, we discuss the journey of improvements in the shallow RVFL architecture.

4.1. Empirical evaluation of RVFL for classification and regression problems

Few scientific questions regarding the RVFL model have been addressed rigorously in the literature. Zhang and Suganthan [37] conducted a comprehensive evaluation of RVFL model to answer the following questions, i.e., the impact of direct links, the effect of bias term in the output layer, and different types of activation functions in the hidden layer, domain from where random parameters are generated and the output weights calculation techniques over 121 UCI classification datasets. The experimental results demonstrate that RVFL with direct links from the input layer to the output layer has better generalization performance than RVFL without direct links, and output weights calculated via ridge regression method are better than Moore-Penrose pseudoinverse based

output weights. Moreover, generating the random weights and biases within a suitable domain from the input layer to the hidden layer has a significant impact on the performance of RVFL, and biases term in the output layer may or may not have any impact on the performance of RVFL. Similarly, a comprehensive evaluation of orthogonal polynomial expanded RVFL (OPE-RVFL) [44] method was conducted in which the input patterns are expanded non-linearly using four orthogonal polynomials, i.e., Chebyshev, Hermite, Laguerre, and Legendre with three activation functions tansig, logsig, and tribas. The results demonstrate that direct links in OPE-RVFL model also have a significant impact on regression problem, and ridge regression based approach is better than Moore-Penrose pseudoinverse based approach. Basically, this study also supports direct links in RVFL (as in [37]). Moreover, the results demonstrate that the OPE-RVFL model with Chebyshev polynomial performs better than other compared polynomials, and tansig is the best performing activation function followed by tribas and logsig. Table 1 shows the commonly used activation functions with RVFL model in the literature.

The random feature mapping (RFM) mechanism has a significant impact on the performance of RVFL and plays a vital role in the success of RVFL, but there is very little research to explore this topic. Cao et al. [62] conducted an experiment to study the relationship between the rank of input data and the performance of RVFL and introduced a concept to measure the quality of RFM via dispersion degree of matrix information distribution (DDMID). Moreover, in [62], several scientific questions were addressed, such as the relationship between the performance of the RVFL model and the rank of the input dataset, the sensitivity of this relationship with the different types of activation function, and the number of hidden nodes, respectively and many more. Rasheed et al. [63] implemented standard RVFL with different activation functions into the respiratory motion prediction and compared RVFL with direct and without direct links. The authors find out that the results with hardlim activation function are better than results with sigmoid, sine, tribas, radbas and sign functions, and direct links prevent RVFL from overfitting. Also, RVFL with direct links has better generalization performance than RVFL without direct links. Dudek [64] also conducted an experiment to answer a few scientific questions related to the RVFL model, such as whether direct links and bias terms in the output layer are necessary for modeling the data in regression problem and what should be the optimal domain (or optimal process) for choosing the random parameters for the hidden layer. In this study, three methods for selecting the random parameters are considered, and the author concludes that direct links seem helpful in modeling the target function with linear regions. The target function with non-linear nature can be modeled with RVFL having no direct connections and bias terms.

4.2. RVFL with different weight initialization techniques

The RVFL model has a random initialization process from the input layer to the hidden layer, wherein original feature space is transformed into randomized feature space, and hence, RVFL has breakneck training speed and less tunable

parameters. It is also capable to model complex (linear/non-linear) data and has universal approximation property [35]. On the other hand, the randomization mechanism makes RVFL an unstable model. The quality of random weights and hidden biases significantly affect the RVFL model’s performance. Traditionally, the random weights and hidden biases are randomly generated with some probability distribution from a fixed domain, i.e. $[-1, 1]$ and $[0, 1]$, respectively. Study [37] reveal that the aforementioned range is not the optimal domain to choose the random parameters. Zhang et al. [42] proposed a sparse pre-trained random vector functional link (SP-RVFL) network in which sparse autoencoder with L_1 norm is employed to generate weights and hidden biases from input layer to hidden layer. Sparse auto-encoder learned superior network parameters than the randomization process. The experimental evaluation on different datasets demonstrate that SP-RVFL outperforms standard RVFL. SP-RVFL model generates sparse weights between input layer to hidden layer that is better as compared to randomly generated weights.

Cao et al. [65] studied the relationship between the probability distribution of the features (variables) in the datasets and probability distribution of input weights and hidden biases in a neural network with random weights (NNRW) (in particular with RVFL and ELM). This study generated seven regression datasets with known distributions (Gaussian, Gamma, and Uniform). The input weights and hidden biases are initialized with different distributions, and the models, i.e., RVFL and ELM, are trained, respectively. The experimental results conclude several phenomena, such as the model having input weights and hidden biases from Gaussian distribution can have a faster convergence rate than ones with the Gamma and Uniform distribution. Suppose one or more features follow Gamma distribution, and input weights and hidden biases follow the same distribution. In that case, the corresponding model has a slow convergence rate and faces an over-fitting issue, etc. In another similar study [43], experiments were conducted to study the impact of Gaussian, Uniform, and Gamma distributions on the performance of the RVFL model, and the results suggest that input weights and hidden biases’ probability distribution has a significant impact on the performance of RVFL model. This study also confirms that RVFL with direct links performs better than RVFL without direct links.

Tanveer et al. [66] proposed several ways to generate input weights and hidden biases in RVFL, wherein twin bounded support vector machine (SVM), least-square twin SVM, twin k-class SVM, least-square twin k-class SVM, and robust energy-based least square twin SVM models were employed to initialize input weights and hidden biases. The experiments illustrate that the twin bounded SVM-based approach has better generalization performance with a lower model rank (Friedman rank) than other proposed approaches.

In [67], the random subspace Fisher linear discriminant (FLD) method generates the random weights and hidden biases wherein important features are assigned higher weights than weights assigned to more minor essential features. Therefore, this approach improves the performance of RVFL. Pan et al. [68] proposed a novel method, Jaya-RVFL, where a new emerging intuitive optimization

technique- Jaya algorithm [69] is employed to optimize the randomization range of input weights and employed to transient stability assessment in power system. Lu et al. [70] proposed three classification methods, i.e., MobileNet-RVFL-CBA, MobileNet-ELM-CBA, and MobileNet-SNN-CBA, respectively, to classify the brain MRI image. In each scenario, first, the MobileNetV2 [71] trained over ImageNet data is employed to extract the features from brain MRI images and then to improve the generalization performance of these randomized neural networks (RVFL, ELM, and SNN [30]), chaotic bat algorithm (CBA) is utilized to optimize the random weights and biases. The empirical evaluation of these models for classifying brain MRI data demonstrates that MobileNet-RVFL-CBA performs better than MobileNet-ELM-CBA and MobileNet-SNN-CBA in terms of sensitivity and overall accuracy. In the literature, it can be seen that several approaches such as different distributions [65], auto encoder [42] and SVM models [66] have been used to calculate the weights from input layer to hidden layer. There is no such method which is always adaptable. Therefore, a further research is required to develop the efficient techniques to initialize the weights from input layer to the hidden layer in RVFL.

4.3. RVFL with manifold learning theory

Maintaining the data's global and local geometric structure while data is proceed via several randomization processes is a challenging task. Manifold learning theory helps to overcome this issue. The standard RVFL model does not consider the geometrical aspect of the data and hence, lose information that leads to lower generalization performance. Li et al. [72] proposed discriminative manifold RVFL neural network (DM-RVFLNN), wherein a soft label matrix is used to enlarge the margin between inter-class samples that makes the DM-RVFLNN model more discriminative, and manifold learning based with-in class similarity graph is used to enhance the compactness and the similarity with-in class samples. Experiments on the rolling bearing fault diagnosis process demonstrate that the DM-RVFLNN model is superior and effective compared to standard RVFL. Considering the topological relationship of samples and to improve the robustness of the RVFL model, sparse Laplacian regularized RVFL neural network with $L_{2,1}$ -norm (SLapRVFL) [73] was proposed to assess the dry weight of hemodialysis patients and the experiments demonstrate that SLapRVFL is more robust than standard RVFL. Here, $L_{2,1}$ -norm was used with regularization term to obtain sparse output parameters. The standard RVFL model doesn't consider the within-class/total variance of training data while obtaining the final output parameters; therefore, Ganaie et al. [74] proposed two variants of the RVFL model known as total variance minimization based RVFL (Total-Var-RVFL) that employs the complete variance information of the training data in the objective function of standard RVFL, and intraclass variance minimization based RVFL (Class-Var-RVFL) wherein the variance of each sample from its respective class is considered, and hence, both models show better generalization performance than standard RVFL model.

Parija et al. [75] proposed minimum variance-based kernel RVFL (MVKRVFL) to identify the seizure and non-seizure epileptic EEG signal. In MVKRVFL,

both total variance of the training data and within class variance are minimized to improve the generalization performance of RVFL model. Kernel trick is also employed to avoid the hidden layer nodes and activation function (as these parameters are chosen in RVFL). In MVKRVFL, deep long short-term memory (DLSTM) [76] is employed to extract the features from epileptic EEG signals. The empirical evaluation of the DLSTM-MVKRVFL model over EEG data demonstrates that it efficiently classifies seizure and non-seizure movement. In [77], co-trained RVFL (coRVFL) model is proposed in which two feature spaces i.e., randomly projected features and sparse- l_1 norm autoencoder based features are employed. The coRVFL model utilizes the strength of different features spaces and improve the generalization performance of RVFL model. Alzheimer’s disease (AD) typically affects the brain’s cognitive functions, damages the cells and memory in the brain. Having heterogeneous medical data, it is a challenging task to diagnose it at an early stage. Dai et al. [78] proposed a hybrid model combining features extracted from different modalities and introduced the manifold concept in the RVFL model to enhance the diagnosis process of AD. Adopting the manifold theory in its optimization process, RVFL model has the capability in maintaining the geometrical properties of the data while calculating the final output parameters. Literature indicates that maintaining the local and global geometrical (or statistical) properties of the data, manifold leaning based RVFL models have better generalization performance than standard RVFL.

4.4. Robust RVFL models

The standard RVFL model employs L_2 norm-based loss function that is sensitive to outliers and hence, affects the generalization performance of the model. The standard RVFL doesn’t perform well over noisy datasets, and therefore, one needs to handle the noisy datasets with extra attention. Managing noisy data is a challenging task. Thus, several approaches have been introduced to address such problems. To deal with datasets having noise or outliers and to reduce the complexity of the model so that the generalization performance of the model can be improved, Cui et al. [79] proposed RVFL-based approach, wherein a novel feature selection method is introduced to make the RVFL model more efficient and robust based on the augmented Lagrangian method. The proposed RVFL-FS method can be fitted into a parallel or distributed computing environment. The RVFL-FS method is employed for the indoor positioning system (IPS) as a regression problem to illustrate the proposed idea. RVFL-FS model works on the idea that instead of using the all hidden nodes, one should select the hidden nodes to generate the robust features. Therefore, RVFL-FS model is computationally efficient and robust as compared to the standard RVFL model. Samal et al. [80] proposed a robust non-iterative RVFL, i.e., Added activation function based exponentially enhanced robust RVFL (AAERVFL), wherein trigonometric function based exponentially expanded input vector is connected by a weighted direct link from the input layer to the output layer and a new activation function using a weighted linear combination of two activation functions, i.e., the local sigmoid and global Morlet wavelet function, is introduced.

The AAERVFL model is examined with five different non-linear systems and three real-world datasets like electricity price prediction, currency exchange rate prediction, and stock price prediction. The experiments demonstrate the superiority of the AAERVFL model over the standard RVFL.

In industrial processes like the mineral grinding process, noisy data with outliers is acquired due to unavoidable circumstances. Dai et al. [81] proposed robust regularised RVFL (RR-RVFL) and its online version as well. Weight calculated from a non-parametric kernel density estimation method is assigned to the empirical error corresponding to each sample. Therefore, the weighting mechanism reduces the negative impact of the outliers over the RR-RVFL model. The experiments over the mineral grinding process demonstrate that the RR-RVFL model is more robust and has better generalization performance than standard RVFL. In RR-RVFL model, weights are assigned using Gaussian kernel function which can be replaced via other efficient functions such as piece-wise continuous function to develop more robust model. Predicting the stock market movement is a significant task for future investment. Chen et al. [82] fused the two different algorithms, i.e., RVFL and group method of data handling (GMDH) [83], and proposed RVFL-GMDH model consist of many nice properties such as resist noise/outlier effectively, avoiding the over-fitting problem and has better generalization performance compared to standard RVFL model. The RVFL-GMDH model effectively predicts the turning point of the stock price. Iron and steel-making-based industries are the famous industries in the modern world, and now data-driven models are being employed for the estimation of molten iron quality (MIQ) in these industries.

Cauchy distribution weighted M-estimation-based robust RVFL [84] model was developed to estimate the molten iron quality. The training data having outliers affect the modeling capability of standard RVFL, so extra care is needed to handle such data. Therefore, weights are assigned to outliers in the data using Cauchy distribution so that their (outliers) contribution in the modeling process can be identified and the negative influence of these outliers can be reduced. Several techniques such as weighting method [81], different loss function [84], have been used with RVFL to improve its robustness and generalization performance. The standard RVFL model with the assumption that all the samples are equally important gives equal weights to each sample for calculating the final parameters but in real world, noisy datasets with outliers are acquired that might have negative impact on the model performance. Therefore, one need to take care of these datasets. In [85], intuitionistic fuzzy theory is employed that define membership and non-membership function to address above issues with RVFL model and proposed intuitionistic fuzzy RVFL (IFRVFL) and to check applicability of IFRVFL, it has been employed for diagnosis of Alzheimer’s disease. IFRVFL is a robust and binary classifier. Similar works can be seen in [86, 87]. Therefore, it should be extended to multiclass problem. Hazarika and Gupta [88] proposed robust 1 norm RVFL (1N RVFL) model wherein the optimization problem is solved via newton technique. The 1N RVFL model produce spare outputs and hence, has less number of hidden neurons as compared to standard RVFL model.

Table 1: The summary of activation functions commonly used in RVFL model

S. No.	Activation function	Mathematical formulation
1	Sigmoid	$\theta(x) = \frac{1}{1+e^{-x}}$
2	Sign (Signum)	$\theta(x) = \begin{cases} -1, & x < 0, \\ 0, & x = 0, \\ 1, & x > 0 \end{cases}$
3	Rectified linear unit (Relu)	$\theta(x) = \max(0, x)$
4	Sine	$\theta(x) = \sin(x)$
5	Radbas	$\theta(x) = e^{-x^2}$
6	Hard limit (Hardlim)	$\theta(x) = \begin{cases} 1, & x \leq 0 \\ 0, & \text{otherwise,} \end{cases}$
8	Tribas	$\theta(x) = \max(1 - x , 0)$
9	Hyperbolic tangent (Tanh)	$\theta(x) = \frac{1-e^{-x}}{1+e^{-x}}$
10	Radial basis function (RBF)	$\theta(\mu, \sigma, x) = e^{-\sigma\ x-\mu\ ^2}$
11	Multiquadratic	$\theta(\mu, \sigma, x) = \sqrt{\ x - \mu\ ^2 + \sigma^2}$
12	Scaled exponential linear units (Selu)	$\theta(x) = \gamma(\max(0, x) + \min(0, \alpha(e^x - 1)))$

4.5. Kernelized RVFL models

For training the standard RVFL, one needs to determine the number of enhancement nodes and activation function in advance. Manually determining the optimal range of hidden nodes and the optimal activation function is challenging task. Kernel methods can be used to address aforementioned issues. Chakravarti et al. [41] proposed kernel exponentially extended random vector functional link network (KERVFLN) for non-linear system identification. Expanding the dimension of input vector via trigonometric function and utilizing the expended vector into learning process increase the generalization performance of the RVFL model. Here, the kernel function is used to increase the stability of standard RVFL, and the inputs are extended using a trigonometric function that improves the generalization performance of the KERVFLN model. Based on the analogy of teacher-student interaction mechanism, Zhang and Yang [89] proposed RVFL+ and kernel RVFL+ (KRVFL+) model that utilizes the learning using privileged information (LUPI) paradigm into the training process of standard RVFL, and the experiments demonstrate that RVFL+ has better generalization performance compared to standard RVFL model. Moreover, a tight generalization error bound based on the Rademacher complexity is derived for the RVFL+ model and proved the efficiency and effectiveness of the RVFL+ and KRVFL+ models. Several machine learning approaches have been developed to classify brain images according to brain abnormalities in the medical domain. Machine learning tools help physicians to make decisions. Nayak et al. [90] proposed kernel RVFL (KRVFL) model with a new feature descriptor based on Tsallis entropy and fast curvelet transform to classify brain abnormalities such as brain stroke, degenerative disease, infectious, brain tumor, and normal brain. An efficient hybrid model [91] consists of weighted multi-kernel RVFL network (WMKRVFLN), empirical mode decomposition (EMD) based

features, and water cycle algorithm (WCA) was proposed to diagnose and classify the epileptic electroencephalogram (EEG) signals. When number of samples are large enough then Kernel based RVFL models are not applicable over large scale datasets.

4.6. RVFL with bayesian inference (BI) and other techniques

Scardapane and Wang [92] proposed several alternatives to train standard RVFL by exploiting the Bayesian Inference (BI) framework. In the standard RVFL model, the optimal output weights are generally calculated via (regularized) least square method but here (in [92]), probability distribution of the output weights is derived. The Bayesian approach has several advantages, i.e., additional prior information can be introduced in the training process of standard RVFL and the capability of automatically inferring hyper-parameters from given data, etc. Experimental results demonstrate that the Bayesian Inference (BI) based approaches are better than least square approaches (as in standard RVFL). The BI approach to train RVFL model gives a new area of research to develop robust RVFL model. Introducing hybrid regularization term with L_2 and L_1 norm into standard RVFL, Ye et al. [93] proposed $L_2 - L_1$ RVFL model that overcomes the stability and sparsity issue of the standard RVFL and gives an iterative algorithm to train $L_2 - L_1$ RVFL model. Alalimi et al. [94] employed the Spherical Search Optimizer (SSO) algorithm to optimize the RVFL model and named it the SSO-RVFL model. The SSO algorithm improves the parameters of the standard RVFL model. Dai et al. [95] incorporated LUP paradigm into incremental RVFL (IRVFL) model and proposed IRVFL+ that has strong theoretical foundation. IRVFL+ model has been trained via two approaches. The first one is named IRVFL-I+ that focused on speed of the model and another one is IRVFL-II+ that focused on accuracy. Incremental learning based RVFL model solve the problem of constructing an appropriate RVFL model. Here, IRVFL-II+ is computationally expensive as compared to standard RVFL and IRVFL-I+ model. In [96], a model for artificially intelligent diagnosis that uses privileged information to learn was proposed to help with ELN differential diagnosis when dealing with single- or dual-modal picture data. In order to create a more effective unmanned aerial vehicle automatic target recognition system, [97] research suggests two unique machine learning methods, namely Random Vector Functional Link Forests and Extreme Learning Forests. For EEG-based driving fatigue detection, in [98], an auto-weighting incremental random vector functional link (AWIRVFL) network model that combines incremental learning and online regression prediction. Although AWIRVFL outperformed several deep learning models, its network topology is still shallow, which restricts its feature learning capacity in describing the underlying characteristics of EEG data.

In another approach, the RVFL-MO method [99] optimize the RVFL model via the mayfly optimization (MO) algorithm to predict the performance of solar photovoltaic thermal collector combined with the electrolytic hydrogen production system. Experiments demonstrate that the RVFL-MO model performs better than standard RVFL. Elsheikh et al. [100] proposed an enhanced RVFL

model with equilibrium optimizer (EO), i.e., RVFL-EO, to predict kerf quality indices during CO_2 laser cutting of polymethylmethacrylate (PMMA) sheets. The equilibrium optimizer (EO) algorithm is employed to obtain the parameters of RVFL model that enhance its generalization performance. Several statistical tests were used to compare RVFL-EO with the standard RVFL model, and results indicate the superior performance of RVFL-EO model. A novel classification method [70] based on MobileNet and three feed-forward neural networks with random weights, i.e., extreme learning machine (ELM), Schmidt neural network (SNN), and RVFL network, to classify brain magnetic resonance image (MRI) image was proposed. Here, MobileNetv2 is employed to extract features from input brain image, and then the classification task is executed via ELM, SNN, and RVFL models, respectively. The experimental results reveal that the MobileNet-RVFL-CBA method performs better than other proposed MobileNet-ELM-CBA, MobileNet-SNN-CBA methods and compared state-of-the-art methods.

Naive bayes classifier has the capability to handle the mixed data containing categorical and numerical attributes. On the other hand, the one-hot encoding technique is used to deal with categorical features in neural networks. To avoid the use of one-hot encoding with neural network, Ruz and Henriquez [101] proposed a two-stage learning approach based on the RVFL model and Naive Bayes classifier, i.e., RVFL-NB, to handle the mixed data. In the first stage, the Naive Bayes classifier is employed to compute the posterior probabilities for each class, and in the second stage, the RVFL model is trained using as inputs the continuous features and including as additional hidden units the posterior probabilities obtained in the previous step. Single hidden layer feed-forward neural networks face a challenge, i.e., how to choose a number of hidden neurons in the hidden layer because this choice leads to underfitting and overfitting phenomena. To address this issue, a non-iterative method [102] for pruning hidden neurons with random weights was proposed. The pruning method is based on Garson's algorithm and was employed on three neural networks, i.e., single hidden layer neural network with random weights (RWSLFN), RVFL, and ELM, to increase their generalization performance.

Parsimonious RVFL (pRVFL) [103] model was proposed for the data stream and hence, perform better than the standard RVFL model. pRVFL model has flexible and adaptive working principle wherein its structure is automatically generated and pruned. In the optical fiber pre-warning system (OFPS), most of the feature extraction methods are quested from the view of the time domain. To address this issue, using multi-level wavelet decomposition, Wang et al. [104] extract intrusion signal features of the running, digging, and pick mattock in the frequency domain and then for considering the feature of each intrusion type, the average energy ratio of different frequency bands is obtained. Finally, the RVFL model is trained for the classification and identification of the signal. The results demonstrate that RVFL correctly classifies the different intrusion signals. El-Said et al. [105] conducted experiments with four machine learning algorithms, i.e., support vector machine (SVM) [106], K-nearest neighbor (K-NN) [107], sequential minimal optimization regression [108, 109] and RVFL

model, to predict the air injection effect on the thermohydraulic performance of shell and tube heat exchanger. The experimental analysis reveals that the RVFL model outperforms compared models with excellent accuracy and better generalization performance.

Borah and Gupta [110] proposed unconstrained convex minimization based implicit Lagrangian twin RVFL for binary classification (ULTRVFLC) for addressing the overfitting issue in the standard RVFL and hence, has better generalization capability as compared to standard RVFL model. Unlike TWSVM and twin ELM (TELM), in ULTRVFLC, three iterative convergent schemes are employed to make the model computationally efficient. The least-square twin SVM (LSTSVM) [111] has been a successful classifier, and it works on original feature space. On the other hand, the RVFL model works on both original and randomized features. Ganaie and Tanveer [112] proposed a novel LSTSVM model with enhanced features obtained from the pre-trained RVFL model and hence, improved the generalization performance of the base line model. Prediction of international oil prices has become a hot topic in the field of energy system modeling and analysis. Tang et al. [113] proposed a new technique introducing a multi-scale forecasting methodology with multi-factor search engine data (SED). Incorporating the informative SED, the multi-scale relationship with oil price is explored, and four machine learning models, i.e., ELM, RVFL, linear regression (LR), and backpropagation neural network (BPNN), are employed in this task. Mary et al. [114] employed standard RVFL in the image retrieval (IR) framework for better performance. To address the instability issue in the sliding mode control system, Zhou and Wu [115] proposed an adaptive fuzzy RVFL (FRVFL), wherein self-mapping between fuzzy rules and hidden layers is employed and adaptive rules are also employed to achieve self-adjustment for the output weights. FRVFL model combine RVFL with dynamic fuzzy system to improve its generalization performance. To address the threats issues in android malware detection tools, in [116], a novel technique using RVFL model with artificial jellyfish search (JS) optimizer algorithm for selecting the optimal features of android malware datasets, i.e., RVFL+JS, has been proposed. The JS algorithm reduces the redundant and irrelevant features from the data that handle the storage and time complexity issue and hence, improves the generalization performance of the RVFL+JS model. In [117], ResNet18 was used to extract wrinkle image features, and an RVFL algorithm optimised by the TSA algorithm based on logistic maps and opposition based learning was proposed for evaluating fabric wrinkle level. This was done to address the issues of low accuracy and low efficiency in evaluating the wrinkle degree by visual perception and the shortcomings of the current artificial neural networks in evaluating the wrinkle level. However, the amount of fabric samples at some levels was insufficient, and just a few different fabric types were included in the study's fabric samples. Label distribution learning (LDL), as opposed to multi-label learning (MLL), can reflect the importance of pertinent labels in samples, which is why many LDL studies have lately been appearing. In [118], an unique LDL framework based on RVFL is proposed, which can efficiently and precisely handle the live data stream.

4.7. Imbalance learning based on RVFL model

The class imbalance problem occurs when one class has small samples compared to other classes. Standard RVFL is not capable of handling imbalance data. Cao et al. [119] proposed improved fuzziness based RVFL (IF-RVFL) where synthetic minority over-sampling technique (SMOTE) [120] is combined with fuzziness based RVFL model [121]. The experiments on real-life liver disease dataset demonstrate that the IF-RVFL model performs better than standard RVFL and F-RVFL model. To diagnose the power quality disturbance (PQD), Sahani and Dash [122] proposed class-specific weighted RVFL (CSWRVFL). Here, a novel signal decomposition technique- reduced sample empirical mode decomposition- is proposed to extract the highly correlated monocomponent mode of oscillations. Hilbert transforms (HT) extracted the two effective power quality indices are extracted from Hilbert transforms (HT). Finally, the combined framework RSHHT-CSWRVFL is applied for online monitoring the power quality disturbances (PQDs) with better classification accuracy.

4.8. Multi-label classification based on RVFL model

In a traditional classification problem, each sample is associated with only one target label from a set of labels. However, each sample can be related to more than one label in multi-label classification problems. There are several fields, i.e., medical diagnosis, music categorization, etc., wherein multi-label data is produced. Chauhan and Tiwari [123] extended the standard RVFL for the multi-label task. In [123], randomization based neural networks, i.e., multi-label RVFL (ML-RVFL), multi-label kernelized RVFL (ML-KRVFL), multi-label broad learning system (ML-BLS), and multi-label fuzzy BLS (ML-FBLS) were proposed to handle the multi-label classification problems. Here, the ML-KRVFL model performs better than other compared models.

Table 2 shows the summary of the shallow RVFL model and its variants. The table gives a highlight of the journey of shallow RVFL in tabular form with variants name, activation functions, hyper-parameter optimization method/solution and finally their applications. In summary, researchers have developed several variants of RVFL model using various methods such as different kind of initialization techniques [37, 42], kernel methods [89, 41], manifold learning [73, 74], fuzzy theory [85] and so on. Kernel based RVFL models [91, 90] are robust, stable and have better generalization performance than standard RVFL model. There is no need to choose activation functions and hidden nodes in kernel based RVFL models, however, these models are not suitable for large scale datasets (when N is large enough). Standard RVFL doesn't consider geometric informations of the data while calculating the output weights whereas, using these kind of informations, RVFL variants with manifold learning theory have better generalization than standard RVFL.

Table 2: The summary of shallow RVFL models

Year	Literature	Model description	Activation function	Hyper-parameter optimization or Solution	Application
2022	Ganaie et al. [124]	MVRVFL+	Relu	Closed form	Classification
2022	Malik et al. [85]	Intuitionistic fuzzy RVFL (IFRVFL)	Selu, relu, sigmoid, sin, hardlim, tribas, radbas, sign function	Closed form	Alzheimer's disease diagnosis
2022	Hazarika and Gupta [88]	1-norm RVFL (1N RVFL)	Sine, relu	Newton technique	Classification problem
2021	Samal et al. [80]	A non-iterative robust AAERVFL	Sigmoid, global morlet wavelet functions	Closed form	Nonlinear system identification
2021	Parija et al. [75]	Minimum variance based kernel RVFL (MVKRVFL)	-	Closed form	Epileptic EEG signal classification
2021	Chen et al. [82]	Group method of data handling (GMDH) based RVFL (RVFL-GMDH)	Sigmoid	Iterative method	Stock price prediction
2021	Zhou and Wu [115]	Adaptive fuzzy RVFL (FRVFL)	-	Iterative method	Slide mode control for manipulators
2021	Zhang et al. [125]	Reinforced fuzzy clustering-based rule model (RFCRM)	-	Iterative method	Regression problem
2021	Elkabbash et al. [116]	RVFL+JS	-	Jellyfish search optimizer	Android malware classification
2021	Guo et al. [73]	Sparse Laplacian regularized RVFL (SLapRVFL)	-	Iterative method	Assessing dry weight of hemodialysis patients
2021	Alalimi et al. [94]	Spherical search optimizer based RVFL (SSO-RVFL)	Tribas, sign, hardlim, radbas, sin, sig	Spherical search optimization algorithm	Prediction of oil production in China
2021	Gao et al. [126]	Walk-forward EWT based RVFL (WFEWT-RVFL)	-	Closed form	Time series forecasting
2021	Tanveer et al. [66]	TBSVM-FL, TWKSVC-FL, LSTWKSVC-FL, RELSTSVM-FL, LSTSVM-FL	Radbas	Closed form (LS)	Classification problem
2021	Abd Elaziz et al. [99]	Mayfly optimization (MO) algorithm based RVFL (RVFL-MO)	Sign, hardlim, sig, tribas, radbas	Mayfly optimization algorithm	solar photovoltaic thermal collector combined with electrolytic hydrogen production system
2021	Elsheikh et al. [100]	Equilibrium optimizer based RVFL (RVFL-EO)	Sign, hardlim, sig, tribas, radbas	Equilibrium optimization algorithm	Laser cutting parameters for polymethylmethacrylate sheets
2021	Zayed et al. [127]	RVFL-CHOA	Sign, tribas, sigmoid, hardlim, radbas	Chimp optimization algorithm (CHOA)	Solar power forecasting
2021	Ganaie et al. [77]	Co-trained RVFL (coRVFL)	-	Closed form	Classification problem
2021	Dash et al. [128]	Empirical wavelet transform based robust minimum variance RVFL (EWT-RRVFLN)	Local sigmoid function, global morlet wavelet	Closed form (LS)	Short term solar power forecasting
2021	Gao et al. [126]	Walk forward empirical wavelet transformation based RVFL (WFEWT-RVFL)	-	-	Time series forecasting
2021	Dai et al. [95]	Incremental learning paradigm with privileged information for random vector functional-link networks: IRVFL+	Sigmoid, sine, triangular, radial	Incremental learning	Classification and regression problems
2020	Ganaie and Tanveer [112]	LSTSVM classifier with enhanced features from pre-trained RVFL	Radbas	Closed form	Classification problem
2020	Ganaie et al. [74]	Total-Var-RVFL, Class-Var-RVFL	Relu	Closed form	Classification problem
2020	Hazarika and Gupta [129]	Wavelet-coupled RVFL (WCRVFL) network	Relu, sigmoid	Closed form (LS)	COVID-19 cases forecasting
2020	Dudek [64]	Standard RVFL	Sigmoid	Closed form (LS)	Regression problem
2020	Chakravorti and Satyanarayana [41]	Exponentially extended RVFL network (ERVFLN), Kernel ERVFLN (KERVFLN)	Tanh(.)	Closed form (LS)	Nonlinear system identification
2020	Cao et al. [62]	-	Sigmoid, radial basis function (RBF), sine	-	Classification problem
2020	Lu et al. [70]	MobileNet-RVFL-CBA, MobileNet-ELM-CBA and MobileNet-SNN-CBA	-	Closed form (LS)	Brain MRI image classification problem

2020	Rasheed et al. [63]	Standard RVFL	Sine, hardlim, sigmoid, tribas, radbas, sign	Closed form (LS)	Respiratory motion prediction
2020	Zhang and Yang [89]	RVFL+, kernel RVFL+	Sigmoid, sine, hardlim, triangular basis function (TBF), radial basis function (RBF)	Closed form (LS)	Classification and regression problems
2020	Ye et al. [93]	$L_2 - L_1$ -RVFL	Sigmoid	Iterative method	Classification Problem
2020	Abd Elaziz et al. [130]	MPA-RVFL	Sigmoid, sine, hardlim, tribas, radbas	Marine predators algorithm (MPA)	Tensile behavior prediction
2020	Essa et al. [131]	RVFL-AEO	Sign, tribas, sigmoid, hardlim, radbas	Artificial ecosystem-based optimization (AEO) algorithm	Forecasting power consumption and water productivity of seawater
2020	Sharshir et al. [132]	Firefly algorithm based RVFL (FA-RVFL)	Sigmoid, radbas, hardlim, sine, sign, tribas	Firefly algorithm	Thermal performance and modeling prediction of developed pyramid solar
2020	Pan et al. [68]	Jaya-RVFL	-	Jaya algorithm	Transient stability assessment of power systems
2020	Parija et al. [91]	Empirical mode decomposition (EMD) and water cycle algorithm (WCA) based weighted multi-kernel RVFL network (WMKRVFLN) (WCA-EMD-WMKRVFLN)	-	Closed form (LS)	Epileptic EEG signal classification
2019	Hussein et al. [133]	Moth search algorithm based RVFL (MSA-RVFL)	-	Closed form (LS)	Water quality analysis
2019	Zhang et al. [42]	Sparse auto encoder with l_1 norm based RVFL (SP-RVFL)	Sigmoid	Closed form (LS)	Classification problem
2019	Nayak et al. [90]	kernel RVFL (KRVFL)	-	Closed form (LS)	Brain abnormalities detection
2019	Zhou et al. [134]	Improved orthogonal incremental RVFL (I-OI-RVFL)	Sigmoid	Iterative method	Data modeling
2019	Ruz and Henríquez [101]	Naive Bayes classifier based RVFL (RVFL-NB)	Sigmoid	closed form	Classification problem of mixed data
2019	Cao et al. [65]	Standard RVFL	Sigmoid	Closed form (LS)	Classification problem
2019	Bisoi et al. [39]	Variational mode decomposition based RVFL (VMD-RVFL)	Tanh	Closed form (LS)	Crude oil price forecasting
2019	Borah and Gupta [110]	Unconstrained convex minimization based implicit Lagrangian twin RVFL for binary classification (UL-TRVFLC)	Multiquadratic	Newton-Armijo stepsize method	Classification problem
2019	Sahani and Dash [122]	Class-specific weighted RVFL (CSWRVFL)	Tanh	Closed form (LS)	Power quality disturbances
2019	Fan et al. [67]	Random subspace fisher linear discriminant (FLD) based RVFL	-	Closed form (LS)	Image steganalysis
2018	Wang et al. [104]	Standard RVFL	-	Closed form (LS)	Optical fiber pre-warning system
2018	Pratama et al. [103]	Parsimonious RVFL (pRVFL)	-	FWGRLS method	Data stream
2018	Henriquez and Ruz [102]	Neural networks with random weights (NNRW)	Sigmoid	Closed form (LS)	Regression and classification problem
2018	Vuković et al. [44]	Orthogonal polynomial expanded RVFL (OPE-RVFL)	Tansig, logsig, tribas	Closed form (ridge regression/ Moore-Penrose pseudoinverse)	Regression problem
2018	Dash et al. [135]	Standard RVFL, regularized online sequential network (ROS-RVFL)	Radbas	Closed form (LS)	Indian summer monsoon rainfall prediction
2018	Nhabangue and Pillai [136]	Empirical mode decomposition based improved RVFL (EMD-IRVFL)	-	Closed form (LS)	Wind speed forecasting
2017	Cui et al. [79]	Feature selection based RVFL (RVFL-FS)	-	Alternative direction method of multiplier (ADMM) algorithm	Fingerprinting based indoor positioning system
2017	Scardapane et al. [92]	Bayesian inference RVFL (B-RVFL)	-	Bayesian inference algorithm	Classification problem
2017	Cao et al. [43]	Standard RVFL	Sigmoid	Closed form (LS)	Regression problem
2017	Dai et al. [81]	Robust regularized RVFLN (RR-RVFLN), online robust regularized RVFL (ORR-RVFLN)	-	Closed form (LS)/ Iterative method	Industrial application

2017	Zhou et al. [84]	Cauchy distribution weighted M-estimation-based robust RVFL (Cauchy-M-RVFLN)	Sigmoid	Iterative method	Blast furnace iron-making process
2017	Xu et al. [137]	Kernel RVFL (K-RVFL)	-	Closed form (LS)	Thermal process
2017	Dai et al. [78]	Manifold learning based RVFL	-	Closed form (LS)	Analysis of alzheimer's disease
2016	Zhang et al. [138]	Multivariable incremental RVFL (M-I-RVFLN)	-	Iterative method	Molten iron quality prediction
2016	Zhang and Suganthan [37]	Standard RVFL	Sigmoid, sine, hardlim, tribas, radbas, sign	Closed form (ridge regression/ Moore-Penrose pseudoinverse)	Classification problem
2015	Zhou et al. [139]	Online sequential RVFL (OS-RVFL)	-	Iterative method	Molten iron quality prediction
2015	Ren et al. [140]	Standard RVFL	Logistic sigmoid	Closed form (LS)	Wind power ramp detection

5. Semi-supervised methods based on RVFL model

There are datasets in which small number of samples are labeled in many applications. RVFL variants have been successfully employed in diverse domains, i.e., classification and regression, etc. But there is very little research for solving semi-supervised learning problems with the RVFL model. Table 3 summarizes the RVFL models developed for the semi-supervised learning. Peng et al. [141] proposed a joint optimized semi-supervised RVFL model, i.e., JOSRVFL, in which a novel approach is used to optimize the objective function of the JOSRVFL model. There are many techniques in machine learning to improve the generalization performance of a model, and fuzzy theory is one of them. In [121], a novel fuzziness-based RVFL model has been proposed for a semi-supervised learning problem. Inspired by transductive SVM [142], Scardapane et al. [143] proposed a transductive RVFL (TR-RVFL) model that defines box-constrained quadratic (BCQ) problem solvable in polynomial time. The TR-RVFL model performs better than many state-of-art algorithms based on the manifold regularization (MR) theory. In [144], two algorithms, i.e., horizontally distributed semi-supervised learning (HDSSL) and vertically DSSL, were proposed. Both algorithms are based on the RVFL model and alternating direction method of multipliers (ADMM) strategy. The HDSSL and VDSSL algorithms solve DSSL problems with horizontally and vertically partitioned data, respectively. Therefore, the RVFL model performs well in semi-supervised problems.

Table 3: The summary of semi-supervised RVFL models

Year	Literature	Model description	Activation function	Hyper-parameter optimization or Solution	Application
2020	Peng et al. [141]	JOSRVFL and JOSELM	-	Iterative method	Classification problem
2020	Xie et al. [144]	Horizontally distributed semi-supervised learning (HDSSL), vertically DSSL (VDSSL)	-	Iterative method	Classification problem
2017	Cao et al. [121]	F-RVFL	Sigmoid	Closed form	Classification problem
2016	Scardapane et al. [143]	TR-RVFL	Sigmoid	Closed form	Classification problem

6. Clustering methods based on RVFL model

Clustering is an unsupervised learning problem where samples are categorized into clusters (groups) based on their similarities. In the literature, there are various clustering techniques, i.e., point-based clustering methods [145] and plane based clustering [146, 147], etc. Zhang et al. [148] proposed an unsupervised discriminative RVFL (UDRVFL) model for the clustering problem. To capture the local information within data, the local manifold learning concept has been used while global biased knowledge of the data has also been considered so that data can be clustered in an optimal manner.

7. Ensemble frameworks based on RVFL model

Ensemble learning utilizes multiple learning algorithms, which are named base learners. The performance of a single RVFL model is often unstable because of the random nature of its hidden features. To improve the model's

stability and performance, it can be beneficial to average the outputs from multiple RVFL models that each have different hidden features. This approach is commonly known as ensemble learning. Ensembles of randomized models, such as random forests, are effective in reducing the variance resulting from the random feature space. Thus, researchers working with RVFL models have explored the development of ensemble RVFLs to improve the model’s stability and performance. In general, there are two steps in ensemble learning. First, a pool of base learners is constructed parallel or sequential. Second, the base learners are combined for decisions according to some rules. Therefore, the ensemble RVFL method either trains multiple RVFLs or utilizes a meta-RVFL to connect the outputs of the base learners. Table 4 summarizes the representative literature about the ensemble RVFL models.

7.1. Ensemble RVFL-based on bagging

Bagging is short for bootstrap aggregating, which trains the base learner using a subset from the training data. The subsets are drawn randomly with replacement [149]. For instance, the base learners of a rotated forest are replaced by RVFLs for classification problems [150]. Bagging generates different subsets with different features to train the RVFL with different structures [151].

7.2. Ensemble RVFL-based on boosting

Boosting constructs the ensemble RVFL incrementally by paying more attention to the samples which are not correctly learned by the base RVFL. In [152], Maximum Relevance Minimum Redundancy is utilized to select features, and the ensemble RVFL is constructed using the Adaboost scheme.

7.3. Ensemble RVFL-based on stacking

Stacking refers to training a meta-learner that works on the outputs from all base learners. In [153], a meta-RVFL is trained to combine the results from all the base RVFL networks with different activation functions. In [154], an individual RVFL network is trained to forecast each sub-series generated by the decomposition, and an incremental RVFL is introduced to aggregate all forecasts. In [155], the short-term load is decomposed by a two-stage decomposition, and an incremental RVFL aggregates the forecasts from each sub-series RVFL.

7.4. Ensemble RVFL-based on decomposition

Another branch of ensemble RVFL is the decomposition-based ensemble framework shown in Figure 4. The time series is decomposed into sub-series carrying different frequencies and each sub-series is modeled by an individual RVFL network. Finally, the aggregation of all forecasts is the output. There are many mature signal decomposition algorithms, such as the empirical mode decomposition (EMD) [156], bi-variate empirical mode decomposition (BEMD) [157], ensemble empirical mode decomposition (EEMD) [158], complete ensemble empirical mode decomposition (CEEMD) [159], hybrid decomposition [160, 161] and other algorithms [162, 163]. For the decomposition block, there

are many hybrid ensemble RVFL with signal decomposition algorithms, such as EMD [164, 154, 50], EEMD [165, 166, 167, 168], CEEMD [169, 170, 171] and two-stage decomposition [155]. In [155], the discrete wavelet transform (DWT) is utilized to decompose the modes generated by EMD into sub-series and an incremental RVFL is trained to aggregate the forecasts from all sub-series. Technical indicators are also utilized to augment the decomposed features for stock forecasting and stock trend classification [160, 161].

7.5. Ensemble weights

Determination of the ensemble weights is crucial for the final performance. If large weights are assigned to a bad base RVFL, it is a disaster for the overall performance. Most of the ensemble RVFL employs the equal-weight scheme [164, 169, 50, 165, 166, 167, 168, 170]. Besides the equal-weight scheme, different algorithms are proposed and applied to learn such unequal ensemble weights. For instance, evolutionary algorithms are adopted to learn, and ensemble weights, as well as the base learners' parameters [151, 172, 173]. A negative correlation learning strategy is utilized to learn the ensemble weights in [174, 175].

7.6. Diverse model pool

Besides the above ensemble RVFL, whose base learners are all RVFL networks, some researchers utilize multiple machine learning models, including RVFL, to construct the ensemble pool, which increases the models' diversity. For instance, ELM, RVFL, and Schmidt neural networks (SNNs) are trained on the same features generated from a DL model in [174]. Finally, the majority voting mechanism combines the outputs from these three neural networks. In [176], the successive projections algorithm is utilized to build ensemble ELM, RVFL, and feedforward networks with random weights. In [165], ELM, BPNN, and RVFL are employed to forecast each sub-series after decomposition. In [177], the model pool consists of RVFL and ELM, which are trained in an offline fashion first, and only a subset of them is randomly selected for online updating. In [178], the fast Fourier transform and Relief algorithms extract features for ensemble ELM and RVFL.

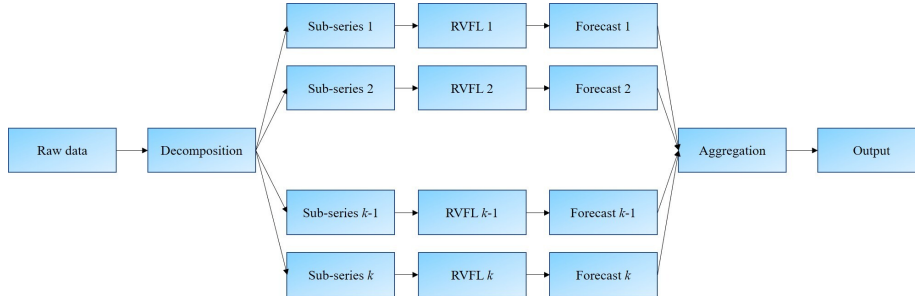


Figure 4: The architecture of decomposition-based ensemble RVFL.

7.7. Other diversity strategies

Besides the diversity strategies of typical ensemble learning, novel diversity strategies specifically designed for RVFL networks are investigated [151, 179]. In [179], different RVFLs' enhancement features are initialized according to different distributions. Five novel diversity strategies, such as data quantity diversity, sampling interval diversity, parameter diversity, ensemble number diversity, and ensemble method diversity, are proposed and investigated in [151]. Zhang and Suganthan [180] propose a novel and efficient strategy to increase the diversity of ensemble RVFL. A single RVFL is trained, and its hyper-parameters are optimized by cross-validation. Then the other RVFLs' hyper-parameters are generated by adding a noisy deviation to the optimal value.

7.8. Others

RVFL also helps split the dataset into subsets, and each subset is learned by the oblique decision tree [52]. In [51], a cascaded ensemble RVFL where the shallow layers' RVFL generate predicted values for the successive layers. Malik et al. [181] proposed a novel ensemble model (en-efRVFL) which has extended features based RVFL model as base model and the output of each base model is integrated by averaging method. The en-efRVFL model has three kinds of features, i.e., original features, supervised randomized features (newly generated) and unsupervised randomized features and therefore generates more accurate and diverse base model in the ensemble. In [182], conv-eRVFL model combines the CNN model with ensemble RVFL model and implement it for the diagnosis of Alzheimer's Disease. An ensemble of RVFL models is fed with the features that an eight-layer trained CNN derives from multiple layers. The s-membership fuzzy function is incorporated into the RVFL network as an activation function to help deal with outliers. In order to reach a judgement, the outputs of all the bespoke RVFL classifiers are averaged and supplied to the RVFL classifier.

Table 4: Summary of ensemble RVFL models

Year	Literature	Model pool	Diversity strategy	Ensemble strategy	Application
2022	Malik et al. [181]	Extended feature RVFL (efRVFL)	-	Averaging	Classification Problem
2021	Yu et al. [151]	RVFL	Data quantity diversity Sampling interval diversity Parameter diversity Ensemble number diversity Ensemble method diversity	Averaging Adaboost RVFL	Forecasting
2021	Lu et al. [183]	RVFL, ELM and Schmidt NN	-	Majority voting mechanism	Cerebral microbleed diagnosis
2021	Hu et al. [184]	RVFL	Solutions' angle	Evolutionary algorithm	Forecasting
2021	Malik et al. [150]	RVFL	Bagging	Rotated forest	Classification
2020	Lu et al. [179]	RVFL	Different distributions	Majority voting mechanism	Classification
2020	Tahir et al. [153]	RVFL	Stacking	RVFL	Multichannel fall detection
2019	Chen et al. [185]	RVFL	Bootstrap	Game theory	Classification
2019	Musikawan et al. [186]	RVFL	Metaheuristic algorithm	Linear regression	Regression
2019	Xia et al. [178]	RVFL and ELM	Diversity strategy	Ensemble strategy	IGBT open-circuit fault diagnosis
2018	Shi et al. [51]	RVFL	Diversity strategy	Ensemble strategy	Parkinson's disease diagnosis
2018	Mesquita et al. [176]	RVFL, ELM and randomized feedforward NN	Successive Projections Algorithm	Successive Projections Algorithm	Regression
2018	Lu et al. [172]	RVFL	De-correlation	Negative correlation learning	Forecasting
2018	Katwal and Suganthan [52]	Oblique decision tree	The RVFL splits the data into subsets.	-	Classification
	Li et al. [167] Qu et al. [164] Sun et al. [168]	RVFL	Signal decomposition	Summation	Forecasting
	Tang et al. [166, 165] Zhang et al. [50]	RVFL	Signal decomposition	RVFL	Forecasting
2017	Qin et al. [169]	Kernel ridge regression	De-correlation	Negative correlation learning	Prediction interval
2017	Miskowiak and Wang [175]	RVFL	De-correlation	Negative correlation learning	Prediction interval

8. Deep architectures based on RVFL model

The success of deep learning (DL) is based on the hierarchical representations of the raw data [187]. DL stacks multiple hidden layers and optimizes the weights using any variants of the back-propagation algorithm. With the help of the deep architecture, the DL can extract multi-scale features automatically. Inspired by the idea of DL, deep RVFL with multiple enhancement layers has been proposed [60].

Table 5 summarizes the representative literature about different deep RVFL networks. The main distinction among them is the utilization of direct links. Some literature only utilizes the direct link to connect input layer to output layer to assist in learning the linear patterns [188, 189]. Some literature utilizes direct links to connect all hidden layers and output layers [190, 3, 191, 60, 192]. Therefore, the raw features are utilized to provide clean information to each level’s representation. Furthermore, Katuwal and Suganthan [193] proposed dense connections of all hidden layers. Another characteristic of the deep RVFLs is the number of output layers. Multiple output layers benefit from the ensemble learning and improving the performance [190, 3, 191, 60, 192].

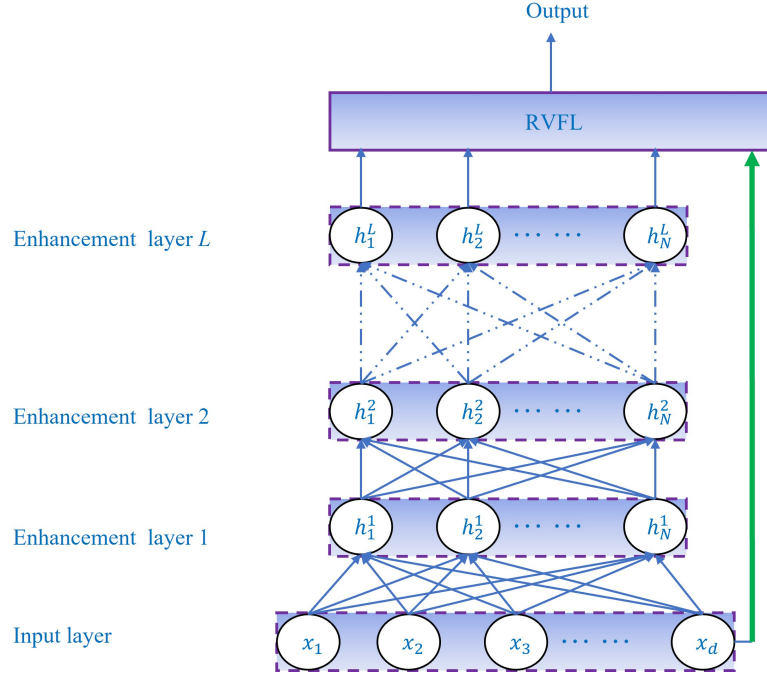
The main research problem of deep RVFL is the architecture design. This section reviews the state-of-the-art deep RVFL’ architectures in detail. The state-of-the-art deep RVFL architectures are shown in Figure 5. These architectures can be classified into three categories, the stacked deep RVFL, the hybrid deep RVFL, and the ensemble deep RVFL (edRVFL).

Table 5: Summary of deep RVFL models

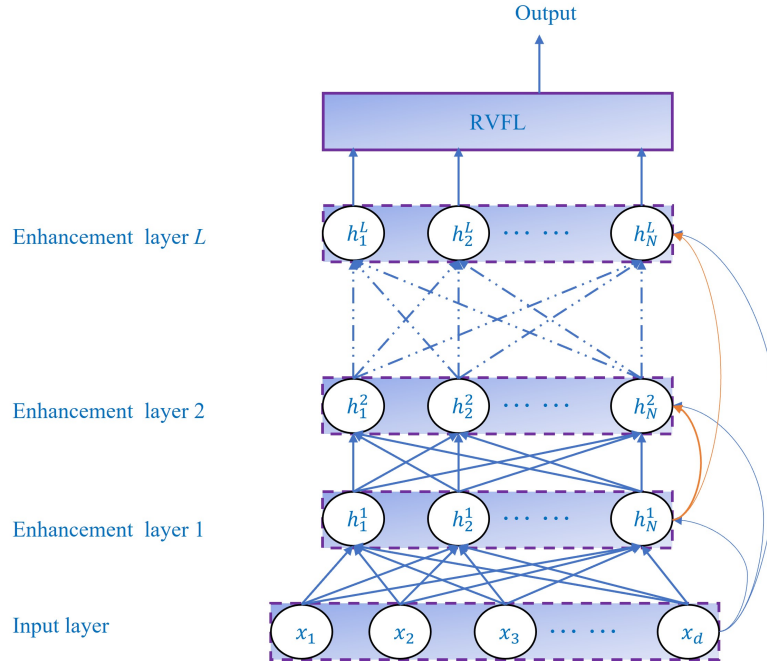
Year	Literature	Direct link	Output layer	Activation function	Application
2022	Hu et al. [194]	Random skip connections	Multiple output layers	Four activation functions	Regression
2022	Du et al. [195]	From input to each hidden and output layer	Dynamic ensemble	<i>sigmoid</i>	Forecasting
2022	Yu et al. [196]	From input to each output layer	Multiple output layers	Na	Landslide displacement prediction
2022	Hu et al. [197]	Automatic search	Multiple output layers	Automatic search	Classification
2022	Shi et al. [198]	From input to each hidden and output layer	Multiple output layers	<i>Sigmoid</i>	Semi-supervised classification
2022	Shi et al. [191]	From input to each hidden and output layer	Multiple output layers	<i>Sigmoid</i>	Forecasting
2022	Gao et al. [9]	From input to each hidden and output layer	Multiple output layers	<i>Sigmoid</i>	Forecasting
2022	Ganai and Tanveer [199]	From input to each hidden and output layer	Multiple output layers	Different activations	Diagnosis of Alzheimer disease
2022	Malik and Tanveer [200]	From input to each hidden and output layer	Multiple output layers	Different activations	Diagnosis of Alzheimer disease
2021	Sharma et al. [201]	-	-	s-fuzzy activation function	Diagnosis of Alzheimer disease
2021	Shi et al. [60]	From input to each hidden and output layer	Multiple output layers	<i>Sigmoid</i>	Classification
2021	Cheng et al. [192]	From input to each hidden and output layer	Multiple output layers	Five activation functions	Time series classification
2021	Dai et al. [202]	Without direct link	Last hidden layer’s features	Sign	SAR target recognition
2019	Katuwal and Suganthan [193]	Densely connected to hidden layers	Last hidden layer’s features	<i>Sigmoid</i>	Classification
2019	Katuwal and Suganthan [193]	From input to output layer	Last hidden layer’s features	<i>Sigmoid</i>	Classification
2018	Henriquez and Ruiz [189]	From input to output layer	Last hidden layer’s features	<i>Sigmoid</i>	Sentiment classification
2017	Zhang and Suganthan [188]	From input to output layer	Last hidden layer’s features	<i>Relu</i>	Visual tracking

8.1. Stacked deep RVFL

The stacked deep RVFL utilizes multiple enhancement layers to achieve multi-scale feature extraction. The consistent characteristic of the deep RVFL architectures is the multiple stacked enhancement layers. The main difference among them lies in how the direct links are connected. The sdRVFL is the most straightforward deep RVFL architecture, which stacks multiple hidden layers, and the direct link only connects the input layer and output layer [189, 193, 188]. The convolutional deep RVFL also establishes the direct connection in this fashion, but the enhancement layers are convolution layers with random weights [188, 202]. However, the direct links are not equipped with hidden layers, which



(a) sdRVFL [193]



(b) sdRVFL(dense) [193]

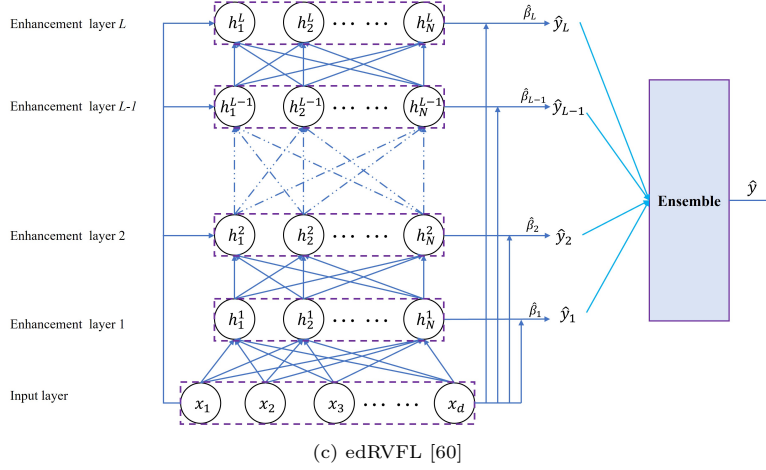


Figure 5: Different types of architectures of deep RVFL model.

weakens the unsupervised feature extraction. The direct links are densely connected to the hidden layers to guide the random features' generation, and the sdRVFL(dense) is proposed [193]. The above architectures only utilize the last hidden layer's features for decision, which may lose valuable information from the intermediate layers. A dRVFL is proposed to take advantage of the rich information from all hidden layers in [60]. A reservoir layer is utilized to extract features for the following deep RVFL [203].

8.2. Ensemble deep RVFL

The stacked deep RVFL does not fully use the features from intermediate layers. However, the dRVFL's utilization of all features requires an inversion of a super large matrix [60]. Therefore, the edRVFL is proposed to achieve a trade-off between computational efficiency and features utilization [60, 190]. The structure of edRVFL is shown in Figure 5. In edRVFL, the direct link connects each enhancement layer with the input layer to guide the random features' generation. An individual output layer with a direct link to the input layer is built for each enhancement layer. Such design splits the inversion of a large matrix into multiple mini-matrix inversion and takes advantage of all features [60]. After training all the output layers, an ensemble block generates the final output. The majority voting mechanism is adopted for classification tasks [192, 60] and mean/median operation is applied for forecasting tasks [190]. Recently, a comparative study shows that edRVFL outperforms ensemble deep ELM on human joint angles prediction [204]. In addition to using all hidden layers for decision making, Yu et al. [196] proposes to utilize a genetic algorithm for selection. Since the hidden neurons are randomized without optimization, there may be inferior neurons which hampers the generation of high-quality features in deeper layers. Therefore, Shi et al. [191] prune the inferior neurons

before generating the next layer’s neurons. In addition, a weighting scheme, which assigns different weights to the training samples in each hidden layer, is proposed to improve the performance. The wrongly classified samples are assigned larger weights in the next layer to increase the diversity and accuracy simultaneously. The norm features from an edRVFL are concatenated with the privileged information from another edRVFL with different activation [199]. This concatenation is fed into another deep RVFL for classifications. Recently, a strategy of random skip connections is proposed to enhance the representation ability of the edRVFL in [194]. Instead of using all output layers, an edRVFL with a selective ensemble method is designed and succeeds in landslide displacement prediction [196]. Following the principle of determining important output layers, an edRVFL with dynamic ensemble based on online performance is proposed for time series forecasting [195]. An approach for the automatic design of ensemble deep randomized neural networks is proposed in [197].

In addition to the supervised learning based on edRVFL, Hu and Suganthan [205] propose a clustering algorithm based on edRVFL’s features. The unsupervised learning is achieved by the manifold regularization. Then, the k means is developed based on the edRVFL features. The consensus clustering method is related to the ensemble block of the edRVFL. Recently, a novel edRVFL for semi-supervised tasks is proposed by Shi et al. [198]. The proposed jointly optimized semi-supervised edRVFL (JOSedRVFL) minimizes the loss function consisting of three components, the error term, L_2 norm regularization and manifold regularization. The L_2 norm regularization aims at reducing the model’s complexity, and the manifold regularization ensures the conditional probabilities of similar samples are close. In [200], geometrical informations under the graph embedded framework are employed while calculating the output parameters of each hidden layer (base model) and therefore, has better generalization performance than edRVFL model.

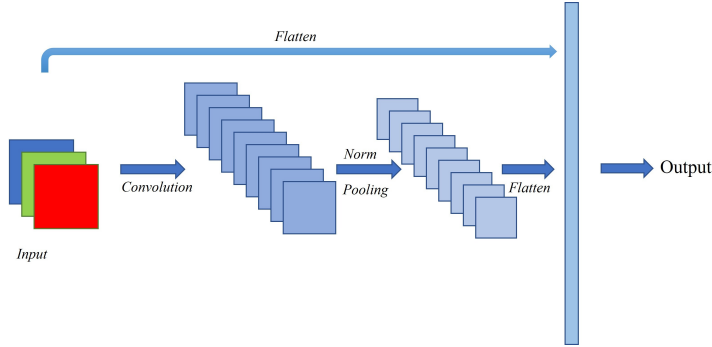


Figure 6: The architecture of deep convolutional RVFL.

8.3. Hybrid deep RVFL

Unlike the above deep RVFLs, the hybrid ones utilize other advanced feature extractions techniques, like DL, to generate the input of the decision block.

The decision block can be any RVFL’s variants, including shallow and deep architectures. For instance, features extracted from a pre-trained ResNet-50 are fed into RVFL whose activation is s-fuzzy membership function [201]. In [192], the ResNet extracts features from time series data, and these features are fed into multiple edRVFL. The convolutional sparse coding deep network extracts features and feeds them into a stacked deep RVFL [206]. In [183], the base model in the ensemble learning framework is a hybrid deep RVFL whose DL performs feature extraction, and RVFL makes decisions.

9. Hyper-parameters optimization and experimental setup

The RVFL’s performance heavily depends on hyper-parameters optimization. This section first separately summarizes the hyper-parameters optimization of single-layer and deep RVFLs. Finally, experimental setup, including data partitioning, evaluation metrics, and statistical tests, are presented.

9.1. Hyper-parameters optimization for single-layer RVFL

For the canonical RVFL, these hyper-parameters include input scaling, number of hidden nodes, activation functions, regularization strength, and distribution of random weights. Most literature utilizes a comprehensive grid search to tune these parameters. Grid search is straightforward to implement and succeeds on many tasks, such as classification [123, 37], forecasting [126, 190]. A comprehensive evaluation of RVFL for classification is conducted by Zhang and Suganthan [37]. A grid search based on 4-fold cross-validation is utilized to select the number of hidden nodes and regularization strength for the RVFLs with different configurations. This study achieves some significant findings. First, the results demonstrate the superiority of the direct links. Second, the output layer’s bias must be tuned based on the specific task. Third, the *hadlim* and *sign* activation functions usually degenerate the accuracy. Finally, tuning the scaling of randomization weights (input scaling) also increases the performance. Gao et al. [126] divides the time series into training, validation, and test set in chronological order. Then, a grid search is conducted to select the hyper-parameters, number of hidden nodes, and regularization strength, according to the forecasting errors on the validation set. Choosing the optimal activation function and number of hidden neurons is also an challenging job so some researchers adopt incremental learning techniques [95] and kernel trick [41] to avoid these issues. Table 1 shows the list of activation functions used in the literature.

However, exhaustive grid search has several drawbacks. Some literature implements evolutionary algorithms for hyper-parameters optimization, such as levy flight based PSO [128] and chimp optimization [127]. The evolutionary algorithms encode the hyper-parameters into an individual, obtaining the optimal configuration after many generations. Each generation selects the individuals whose performance is outstanding. Therefore, generation after generation, the suitable configuration survives.

Based on the above descriptions, there are two main branches of hyper-parameters tuning of single-layer RVFL, the grid search, and evolutionary optimizations. The performance of grid searches heavily relies on the researchers' experience because they have to define the hyper-parameters for selection manually. A detailed discussion and analysis of single-layer RVFL's hyper-parameters are given in [37]. For evolutionary optimization of the RVFL, the researchers can define a large search region to allow the evolutionary algorithm to explore the best configuration. However, the evolutionary algorithms are time-consuming because each generation trains multiple RVFLs.

The following suggestions about tuning single-layer RVFLs are provided for researchers and practitioners. First, large hidden dimensions are preferred for large datasets with huge input dimensions. Second, tuning the input scaling parameter may boost the performance on specific tasks, although the most common practice is to set the scaling factor to one. Third, it is advisable to increase the hidden nodes when tuning the input scaling worsens the performance. Fourth, the regularization strength of ridge regression plays a critical role in improving performance.

9.2. Hyper-parameters optimization for deep RVFL

As the RVFL becomes deep, the hyper-parameters that wait to be optimized grow exponentially. Whether each layer needs a different set of hyper-parameters is still an open problem. Katuwal and Suganthan [193] utilizes a grid search to optimize the number of hidden nodes and regularization, but the number of layers is fixed as three.

Some hyper-parameters tuning strategy is proposed for the deep RVFLs. For instance, a two-stage tuning strategy is proposed to obtain the best configurations of edRVFL for classification [60]. The optimal number of hidden nodes and regularization parameters are selected for a two-layer network in the first stage. Optimizing the hyper-parameters for a two-layer network saves computational time and also considers the effects of deep representations. Then, the second stage fine-tunes the hyper-parameters within the neighborhood of the obtained number of hidden nodes and regularization parameter from the first stage optimization. Similar to the two-stage tuning, a three-stage method is proposed by Cheng et al. [192] for time series classification. In [192], the authors tune the number of hidden nodes and regularization parameters for the first and second hidden layer in the first and second tuning stages. Then, the third stage imposes a random deviation on the optimal hyper-parameter obtained from the previous steps. The random variations can enhance the generalization ability and reduce the computational burden of the hyper-parameters tuning. Gao et al. [190] proposes a layer-wise grid search algorithm to determine the deep RVFL's hyper-parameters layer by layer. The cross-validation is conducted to obtain the best hyper-parameters. Once the hyper-parameters tuning is finished, the hyper-parameters and hidden states for this layer are fixed. The cross-validation is applied to the next layer. This process is repeatedly until the hyper-parameters for the last layer are determined. The layer-wise tuning offers

each layer an opportunity to own different configurations. The layer-wise tuning method is also utilized in [3].

When the neural networks become deep, the hyper-parameters tuning becomes more challenging, because there are more hyper-parameters and the computation burden increases. For the deep RVFLs, the tuning of hyper-parameters must take into account both computational efficiency and performance. Hence, to achieve these two goals, stage-based [60, 192] and layer-wise tuning algorithms are proposed [3, 190]. In summary, the stage-based tuning divides the tuning process into stages to save computation time and ensure efficient hyper-parameter exploration. The first stage of stage-based tuning searches hyper-parameters within a coarse region for saving computation time and exploring a large space. The following stages are the fine tuning to further improve the accuracy. The layer-wise tuning considers each hidden layer and the corresponding output layer as an independent model. Hence, each layer’s tuning only works on a single-layer RVFL whose computation is much faster than that of deep architectures. Furthermore, such tuning ensures that each output layer performs exceptionally well. Finally, the layer-wise tuning can offer different layers with different hyper-parameters to increase the diversity.

The preceding discussion offers some insights into tuning hyper-parameters for the deep RVFLs. First, tuning for shallow RVFLs or each layer can save computational time. Second, searching within a coarse region is an efficient way to explore distinctive hyper-parameters. Third, fine-tuning based on the selected hyper-parameters from wide ranges can further improve the performance. Fourth, assigning different hyper-parameters to each layer contributes to the diversification of the edRVFL.

9.3. Experimental setup

This section presents and summarizes the experimental procedures, including data partitioning, normalization, evaluation metrics, and statistical tests. For classification problems, the researchers usually adopt the k -fold cross-validation for hyper-parameters tuning and evaluate the models on the remaining test set [37]. If the partition schemes of the dataset are available, the researchers must follow the same partitioning for fair comparisons, such as annealing and audiology-std dataset [37]. With a given experimental setup, there is a need to benchmark the models for fair comparison. There have been multiple attempts in the literature to benchmark the performance of the models on a given experimental setup Fernández-Delgado et al. [207]. Recently, self normalizing networks [208] released the publicly available data partitions for reproducibility and the benchmarking of the models. Following self normalizing networks [self normalizing ref], several studies like [60] followed this setup for fair evaluation of the models. Recently, Del Ser et al. [209] presented a thorough survey based on randomization based machine learning models with renewable energy prediction problems and compared them. There is still a gap for the benchmarking of the models like lack of evaluation of the models across different hyperparameters, their range and so on. Thus, benchmarking of the models to ensure the progress

of the literature, reproducibility of the results and fair comparison is needed in randomization based models.

The test set is always located at the end for time series datasets, and the remaining observations are utilized for training and hyper-parameters tuning. There are two approaches to split the training and validation set. The first approach is the same as the cross-validation for regression and forecasting. Some researchers formulate the observations into input patterns and response values. Then, a k -fold cross-validation is conducted to tune the hyper-parameters [164]. The second approach splits the training and validation set in chronological order [126].

All models have limitations, and therefore evaluation and comparison of machine learning models depend on the specific dataset. A fruitful set of evaluation metrics is utilized to evaluate the RVFL's performance. For classification, the classification accuracy is always the first choice [60]. Table 6 summarizes the forecasting errors utilized in the literature, where, x_j and $\hat{x}_j$ represent the raw observation and its forecast, L and T represent the size of training and test set, respectively. MAE and RMSE can be utilized when the time series are of the same scale, although RMSE is more sensitive to outliers. MAPE is a popular percentage error with high interpretability. Finally, MASE is a scaled metric and can be utilized for comparisons on different time series. In addition to the forecasting errors, the direction statistics are utilized for comparison in some literature [166, 165]. The definition of direction statistics ($Dstat$) is

$$Dstat = \frac{1}{L} \sum_1^L a_i \times 100\%, \quad (14)$$

where $a_i = 1$ if $(\hat{x}_j - x_{j-1})(x_j - x_{j-1}) > 0$, or otherwise $a_i = 0$.

Table 6: Forecasting errors in the literature about RVFL.

Metric	Formula
Mean absolute error (MAE)	$\frac{1}{L} \sum_1^L \hat{x}_j - x_j $
Mean absolute scaled error (MASE)	$\frac{1}{L} \sum_1^L \frac{ \hat{x}_j - x_j }{\frac{1}{T-1} \sum_{t=2}^T x_t - x_{t-1} }$
Mean squared error (MSE)	$\frac{1}{L} \sum_1^L (\hat{x}_j - x_j)^2$
Root mean squared error (RMSE)	$\sqrt{\frac{1}{L} \sum_1^L (\hat{x}_j - x_j)^2}$
Mean absolute percentage error (MAPE)	$\frac{1}{L} \sum_1^L \left \frac{\hat{x}_j - x_j}{x_j} \right $

In addition to the above evaluation metrics, the literature also utilizes statistical tests to compare the different models' performance on various datasets. In general, these tests can be classified into two groups, the group-wise and pair-wise tests. Group-wise tests can determine the overall ranking of the models on all the datasets and group them based on the statistical distance. The literature about RVFL utilizes the Nemenyi test to compare the models in a group-wise fashion [126, 164]. The pair-wise tests assist in comparing the models in a pair-wise fashion, which is straightforward to show the better model.

For instance, the Wilcoxon test is utilized to ascertain how many algorithms edRVFL significantly outperforms [60]. There are several others statistical tests such as Friedman test, sign-test and so on, to evaluate the performance of machine learning models. We refer the reader to [210, 207, 211] for more detailed information about the application of these tests to machine learning models.

10. Time series forecasting and other applications

Time series forecasting refers to establishing the model using historical observations, and this model is utilized to make extrapolations for future steps. Accurate and reliable forecasts help the stakeholders and decision-makers plan, organize, maintain and develop the system in advance in a data-driven fashion. Table 7 summarizes the representative literature about forecasting by RVFL and its variants. RVFL and the improved versions have demonstrated their outstanding performance on various forecasting tasks from different domains, such as electricity load [126, 164], solar power [212], wind power [169], financial time series [154] and other data [129]. Table 7 shows that signal decomposition algorithms are popular for feature extraction on forecasting tasks. The signal decomposition can splits the time series into multiple sub-series with different frequencies. Then the RVFL works on these sub-series for forecasting. This section presents the details of all the literature about RVFL-based forecasting.

Table 7: Summary of RVFLs for forecasting

Year	Literature	Feature extraction	Learning category	Hyper-parameter optimization	Field
2022	Gao et al. [3]	-	edRVFL	Layer-wise grid search	Inpatient discharges
2021	Zayed et al. [127]	-	Kernelized RVFL	Chimp Optimization Algorithm	Solar power
2021	Dash et al. [128]	Signal decomposition	Expanded RVFL	Particle swarm optimization	Solar power
2021	Majumder et al. [213]	-	Online sequential kernel RVFL	-	Solar power
2021	Gao et al. [126]	Signal decomposition	Single model	Grid search	Electricity load
2021	Hu et al. [184]	-	Ensemble RVFL	Evolutionary optimization	Electricity load
2021	Manibardo et al. [214]	Deep architecture	RVFL, deep RVFL and edRVFL	Bayesian optimization	Road traffic
2020	Cheng and Wang [215]	Signal decomposition	Decomposition-based ensemble learning	-	Wind speed
2018/2020	Qiu et al. [154], Tang et al. [166], Zhang et al. [50]	Signal decomposition	Decomposition-based ensemble learning	Grid search	Crude oil price
2020	Wu et al. [171]	Signal decomposition	Decomposition-based ensemble learning	Sine cosine algorithm	Crude oil price
2020	Wu et al. [170]	Signal decomposition	Decomposition-based ensemble learning	whale optimization algorithm	Financial time series
2020	Zhang et al. [216]	Stacked auto-encoder	Incremental RVFL	Grid search	FCCU end-point quality
2020	Hazarika and Gupta [129]	Signal decomposition	Single RVFL	Grid search	COVID-19 cases
2019	Bisoi et al. [39]	Signal decomposition	Single RVFL	Grid search	Crude oil price
2019	Kushwaha and Pindoriya [217]	Signal decomposition	Decomposition-based ensemble learning	Grid search	Solar power
2019	Majumder et al. [218]	-	Kernelized RVFL	Water cycle algorithm	Solar power
2018	Mondliki et al. [219]	-	RVFL with different regularizations	Grid search	Financial time series
2018	Lian et al. [220]	-	Ensemble RVFL	Grid search	Landslide displacement
2018	Li et al. [167]	Signal decomposition	Decomposition-based ensemble learning	Grid search	Travel time
2018	Lu et al. [174]	-	Ensemble RVFL	Grid search	Production rate
2017/2018	Qiu et al. [169], Nhabangue and Pillai [136]	Signal decomposition	Decomposition-based ensemble learning	Grid search	Wind power
2016/2018	Qiu et al. [164, 155]	Signal decomposition	Decomposition-based ensemble learning	Grid search	Electricity load
2016	Zhang et al. [138]	-	Incremental RVFL	Grid search	Molten iron quality
2015	Zhou et al. [139]	PCA	Online sequential RVFL	Grid search	Molten iron quality

10.1. Electricity load

Electricity load forecasting is crucial for the development, maintenance, and planning of power systems. Among the abundant forecasting methods, RVFL demonstrates its success by many researchers. For instance, a quantile scaling method is proposed to re-distribute the randomly weighted inputs of RVFL to avoid the saturation effects and suppress the outliers in [38]. Signal decomposition techniques are utilized to remedy the unsupervised features of RVFL in [126, 155, 164]. For instance, EMD decomposes the load data into several modes, and then each mode is predicted by an RVFL. Finally, the summation

is conducted to combine the predictions for each mode [164]. In [155], a two-stage decomposition method is proposed to decompose the load data, then each load is predicted by an individual RVFL, and finally, all forecasts are aggregated by another RVFL with explanatory variables. Different from the above ensemble methods, a single RVFL is built on all the components generated by EWT in [126]. The same decomposition scheme is combined with edRVFL for short-term load forecasting in [190] and the results demonstrate edRVFL's superiority over a single RVFL. Moreover, a multi-modal evolutionary algorithm is utilized to optimize the enhancement weights, bias, and combination weights of the ensemble RVFL for short-term load forecasting [184].

10.2. Solar power

With renewable energy development, solar power forecasting is an emerging area. In [212], the authors compare RVFL with SLFN and RWSLFN, and the results demonstrate the superiority of the direct link. Signal decomposition methods also work for solar time series. For instance, in [217], maximum overlap DWT decomposes the power data into sub-series, and an individual RVFL predicts each series. Finally, aggregation of all forecasts is the forecast for solar power. Moreover, some researchers utilize meta-heuristics algorithms to optimize RVFL's parameters automatically. For instance, in [218], the multi-kernel RVFL whose kernel parameters are optimized by water cycle algorithm is proposed to forecast short-term solar power. In [127], Chimp Optimization Algorithm (CHOA) is utilized to determine RVFL's hyper-parameters for predicting output power and the monthly power production of a solar dish/Stirling power plant. Some researchers integrate signal decomposition, evolutionary optimization, and RVFL together to boost forecasting accuracy. For example, in [128], the EWT is utilized to decompose the time series, and the residue is discarded. The remaining sub-series are expanded using trigonometric activation in the direct link, and the enhancement states are a linear combination of two activation functions. Finally, the RVFL is trained with a novel robust objective function by minimizing the variance of training data. Moreover, the added activation functions' hyper-parameters are also optimized by PSO. The new cost function also shows its improvement on RVFL in terms of forecasting accuracy. In [213], an Online Sequential Kernel-based Robust RVFL is trained based on Hampel's cost function to forecast solar and wind power.

10.3. Wind power

A comparison of RVFL and other machine learning models on wind speed forecasting is conducted in [221]. Some literature about wind power forecasting combines signal decomposition techniques with RVFL [136, 169]. For instance, CEEMD is applied to decompose the raw data into modes, and a kernel ridge regression predicts each mode. Finally, instead of using a simple summation, the RVFL is trained to combine the forecasts of all methods for wind power ramp prediction in [169]. In [136], Chebyshev expansion is utilized as functional nodes to reduce the number of activation nodes. Then it is combined with EMD

for wind speed forecasting. In addition, Hampel’s cost function is utilized for training an online sequential kernel-based robust RVFL to forecast solar, and wind power in [213]. In Cheng and Wang [215], a multi-objective salp swarm optimizer is adopted to determine the weights which are used to combine the forecasts from four networks, including RVFL, for wind speed forecasting.

10.4. Financial time series

The financial time series is different from the above data with strong cycles. The financial time series is much volatile, and it is difficult to extract features. Among all the RVFL-related financial time series forecasting literature, most focus on crude oil prices. Similar to the other kinds of time series forecasting literature, many researchers combine RVFL and signal decomposition algorithms for crude oil price forecasting [50, 154, 166]. EEMD decomposes crude oil price, and then different RVFLs are trained for each mode, including the residue. Finally, the summation of all RVFL’s outputs is the forecast in [166]. In [154], CEEMD decomposes the raw data into modes, and an individual RVFL is established on each mode. Finally, forecasts of all modes are aggregated using an incremental RVFL. The same decomposition-based structure is utilized. The difference is that improved CEEMD with adaptive noise acts as the decomposition in [170, 171] and sine cosine algorithm optimizes all the parameters in [171]. In [50], bivariate EMD is utilized to decompose the original time series into sub-series, and an individual RVFL predicts each series. Finally, aggregate the forecasts via summation. The modes generated from VMD are fed into RVFL, and the experimental results demonstrate the superiority of VMD over EMD in [39]. Besides the decomposition-based RVFL, a novel ensemble RVFL with five diversity strategies is proposed for crude oil price forecasting in [151].

Besides the literature about crude oil price forecasting, RVFL also succeeds on other financial time series. For example, in [219], different regularization parameters are imposed to the output weights of the direct link and enhancement nodes to forecast discount rates. In [170], the improved CEEMD with adaptive noise decomposes the data into sub-series, and each sub-series is predicted by an individual RVFL whose parameters are optimized by a whale optimization algorithm. Finally, the output is the summation of all forecasts. In [222], multilingual search engine data is utilized to derive the input for RVFL to forecast crude oil price.

10.5. Other Applications

Besides the above popular areas with a large need for forecasting, RVFL and its improved versions have also succeeded in various areas, such as temperature [152], landslide displacement [220], COVID-19 cases [129], travel time [167], molten iron quality [138, 139], energy consumption [168], signal-to-noise ratio [223], algae missing values [133], temperature in subway station [224], inpatient discharges [3] and so on [214].

Among these literature, many utilize different heuristic algorithms to optimize the hyper-parameters [99, 130, 131, 132], weights [225] or select the input

features [133]. For instance, firefly algorithm is utilized to select RVFL’s hyper-parameters, number of enhancement nodes, bias, direct link, distribution and activation function, for thermal performance prediction [129]. In [226], an incremental method to adjust RVFL’s structure is proposed for time series prediction, where the network increases its enhancement nodes when the performance degrades. For instance, in [133], a moth search algorithm is utilized to select input features for RVFL to predict missing values of algae. In [130], Marine Predators Algorithm is utilized to optimize RVFL’s hyper-parameters for tensile behavior prediction. In [131], an artificial ecosystem-based optimization algorithm is utilized to optimize RVFL’s hyper-parameters for forecasting power consumption and water productivity of seawater. In [99], mayfly-based optimization is utilized to optimize RVFL’s hyper-parameters to forecast the performance of Photovoltaic/Thermal Collector. In [225], the RVFL trained by PSOGA is utilized to generate prediction intervals for landslide displacement. The RVFL is first pre-trained using reconstructed intervals, and then the PSOGA trains the RVFL with transferred weights based on original data.

Besides the RVFL-based on meta-heuristics, some literature also focuses on ensemble RVFL [167, 168, 174, 220]. For instance, GA is utilized to assign ensemble weights for each RVFL trained with bootstrap samples, and the RVFLs whose weights are higher than the threshold are selected to construct prediction intervals for landslide displacement [220]. In [167], EEMD is utilized to decompose the travel time into modes, and a different RVFL predicts each mode. Finally, each mode’s forecasts are combined with linear addition. In [152], ensemble RVFL is trained based on AdaBoost after selection features via MRMR. Each RVFL is trained using iteratively reweighted least squares for temperature forecasting. In [168], EEMD decomposes time series into sub-series, and the features with high correlation with target variables are used for the corresponding RVFL to forecast building energy consumption. In [174], Negative Correlation Learning is utilized for training ensemble RVFL networks for production rate forecasting. Manibardo et al. [214] applies RVFL, deep RVFL, and edRVFL to the road traffic dataset, and the hyper-parameters are determined by Bayesian optimization. Manibardo et al. [214] claims that the direct link is the fundamental reason for RVFL and its variants’ superiority over ELM-based models.

Incremental (online) RVFL also succeeds on various time series [135, 138, 139, 216, 227]. Incremental RVFL updates its structure or weights when new observations are available. For example, in [138], the incremental RVFL adds a new node and updates its weight incrementally until the performance degrades for the prediction of molten iron quality. In [139], the online sequential RVFL is trained using the principal components and its estimation of the previous steps to predict molten iron quality, too. In [135], RVFL and online sequential RVFL are compared on rainfall prediction, and the results demonstrate OS-RVFL’s superiority for rainfall forecasting. In [227], an online RVFL-based on sliding-window is trained to temperature forecasting. In [216], a stacked auto-encoder is trained in offline fashion first, and then an incremental RVFL is established based on the SAE’s output when a concept drift is detected.

In addition, different novel RVFLs are proposed for other time series. In [129], the level one sub-series generated from discrete wavelet transformation are fed into RVFL for COVID-19 cases forecasting. In [228], Schmidt orthogonalization is utilized to orthogonalize the output vectors, and the hidden nodes are pruned according to the output weights to predict product quality.

Although RVFL and its variants succeed in various time series forecasting tasks, the research on spatial-temporal time series is not mature. There is only one RVFL-related paper touching this problem to the authors' best knowledge. In [137], a kernel RVFL is established to predict the temporal dynamics decomposed via Karhunen–Loève method from the spatial-temporal process.

11. Comparison with other state of the art machine learning techniques

An insightful discussion about the comparison between the RVFLs and other state of the art machine learning techniques significantly contributes to the literature. This section mainly discusses about the pros and cons of the RVFLs and other machine learning models.

The early work with randomization techniques can be traced with perceptron and standard feed-forward neural network (SLFN). In perceptron, the weights between sensor units and response units can be generated randomly whereas the rest weights from the associator units and the response units are calculated via reinforcement learning [229, 230]. SLFN [30] also uses randomization technique but there is no direct links in this network. Jacobian neural Network (JNN) [231] is a polynomial time randomized algorithm which give optimal network with probability one. Moreover, the paper [35] has some theoretical justification for RVFL and other neural networks. Back-propagation based trained ANNs are sensitive to learning rate setting, slow convergence and trapped into local minima [16, 17, 18]. On the other hand, RVFL resolves these issues by generating the weights randomly from input layer to hidden layer and the rest weights (hidden layer to output layer) are calculated via closed form solution. The direct links in RVFL plays an important role in both classification and regression problem [37, 44]. These direct connections separates RVFL from other randomized networks such as radial basis function (RBF) [27] and extreme learning machine (ELM) [232] and so on. RVFL and its deep variants have shown superior performance than ELM, Hierarchical ELM (H-ELM) and multi-layer kernel ELM (ML-KELM) [193]. Support vector machine (SVM) has strong mathematical foundation and has shown state of the arts results [233, 234, 235]. However, RVFL with privileged information (RVFL+) and its kernel extension (KRVFL+) have shown superior performance than SVM and its variants such as gSMO-SVM+ and fast SVM+ [89]. From optimization perspective, RVFL+ has simpler constraints than SVM+, that results in closed form solution. Furthermore, the DRVFL shows superiority over SVM and random forest on Twitter sentiment datasets. The DRVFL with fuzzy activation outperforms EML and kernel-ELM on the ADNI dataset.

One popular state of the art deep learning method is the Resnet. The Resnet constructs a quite deep architecture with the help of residual links. It utilizes back-propagation algorithm to optimize the weights and bias, which takes much more time than training the RVFL networks. In addition, the literature has demonstrated the superiority of deep RVFL over the Resnet on tabular data classification [60, 191]. Besides tabular datasets, time series forecasting is also a valuable problem. For forecasting, the long short-term memory (LSTM) and temporal convolutional network (TCN) are two common state-of-the-art methods. Comparing with edRVFLs, the training is much slower. However, many literature show that the advanced RVFLs outperform the LSTM and TCN [190]. Furthermore, the deep stacked RVFL method outperforms stacked denoising auto-encoder on two benchmark MR brain datasets (MD-1 and MD-2). Some literature has demonstrated that RVFL-based models outperform the BPNN [39].

However, RVFL does not include CNN-type feature extraction layers for image or sequence data. The convolution filters aim at mining local patterns from different spaces. Then, multiple stacks of these filters assist in learning a global representation. Finally, the gradient descent algorithms help to learn these features in an end-to-end fashion. Although RVFL does not own the CNN’s feature extraction layers, the features learned by CNNs can be utilized as input to the RVFL variants. In other words, the RVFL variants can be the decision module for the features from all kinds of gradient-based deep networks [192].

12. Conclusions and future directions

Randomized neural networks (RNNs) have shown their strength among machine learning models. A special kind of RNN, RVFL model, has been emerged as a very successful model. The history of RVFL model can be traced in late 20th century. This review paper summarizes the developments of RVFLs from theoretical foundations to various applications. As per our knowledge, this is the first review paper focused for RVFL model. The RVFL is a feed-forward neural network with random features and direct links. The randomized features introduce non-linear representations of the input features, and the direct links reserves the linear pattern. The hidden layer’s weights are randomly initialized and frozen during training, and only the output layer is trained with a closed-form approach. The randomized features render RVFLs at a fast computational speed.

With the renaissance of deep learning, researchers have extend the shallow RVFL to deep architectures to enhance its representation ability. In the deep architectures of RVFL, the hidden neurons are also randomly initialized and frozen during the training step. Only the output layers are trained, which reduces the computational burden of back-propagation. The RVFLs with deep architectures have demonstrated their superiority over shallow ones on classification, regression, and forecasting, etc. Therefore, the authors claim that the hierarchical enhancement features offer a large modeling capacity and increase

performance. Different ensemble learning algorithms, such as bagging, boosting, and stacking, are shown to significantly boost the single RVFL’s performance. In addition, the ensemble RVFLs based on signal decomposition demonstrate tremendous success on various forecasting tasks. The signal decomposition algorithms first dis-aggregate the complex sequential data into multiple components, which assists in the RVFL’s representation ability. Then RVFL-based models are built on each component, and the ensemble of all forecasts is the output. The RVFL-based models have achieved significant success in various domains because of their fast computational speed, high accuracy, and powerful representation ability and these models also have achieved state-of-the-art performance in the time series forecasting domain on wind speed, solar energy, electricity load, etc. we hope that this paper offers treasure information about the RVFL model to the researchers. we presented a thorough survey on the developments of the RVFL model in many aspects, i.e., shallow RVFL, ensemble algorithms based on the RVFL model, deep RVFL variants, etc. Also, we presented applications of the RVFL model that shows its applicability in the real world. The literature has demonstrated the superiority of the RVFL-based deep models over tabular datasets [191]. While reviewing the papers in the literature, we found some potential research directions that the researchers in future should explore.

- Weights initialization techniques (WITs) have significant impact over the performance of RVFL models. A few research [37, 43, 65, 66] suggest that this topic needs to be discussed further with mathematical justifications. Moreover, several others strategies [236] can be explored with RVFL model such as interval based initialization [237], variance scaling based initialization [238], data-driven initialization [239], hybrid initialization [240, 241], cluster based initialization [242] and data statistics based initialization [243].
- Outliers or noisy samples influence the modeling capability of standard RVFL and hence, leads to poor performance. Kernalized RVFL models are robust but can’t be employed for large scale dataset. Therefore, for large scale, different techniques such as random Fourier features [244] can be used to handle the same [245]. Moreover, RVFL with fuzzy neural networks [246], ensemble learning [45] or other advanced techniques can be employed to develop robust RVFL variants.
- Ensemble learning and deep learning are two individual growing fields [59]. Researchers have recently combined them to develop a more accurate and efficient model that can perform well on real-world data. The RVFL model has fast training speed and good generalization performance and has been employed successfully in various engineering domains. Therefore, this can be a hot topic for researchers to explore RVFL in these research directions.
- In the literature [52, 247], RVFL and decision tree have been employed together to develop a model with better performance. Recently, deep forest [248, 249] with better interpretability and less tunable parameters

as compare to deep neural networks (DNNs) is a growing research field. Studying the RVFL model and decision tree with deep forest architecture can be a new research field.

- To increase the generalization performance of machine learning models, learning with global/local data consistency (topological properties of data) has shown its importance among the machine learning community. The RVFL model transforms the original features into randomized features in unsupervised manner. Hence, the randomization process might ruin the original feature space’s topological properties and leads to an inefficient model. Therefore, works [124, 72] indicates that incorporating the idea of manifold learning into the RVFL model can develop more accurate models.
- The standard RVFL handles balanced data effectively. The imbalance learning problem seriously deteriorates the performance of the RVFL model. In general, techniques addressing imbalance data are divided into two categories, i.e., data-level approach [120] and algorithm approaches [250] can be used with RVFL model to classify imbalance data, effectively. Therefore, it is an opportunity for researchers to develop other techniques to explore this research direction.
- Developments in the RVFL model have been focused on supervised problems, i.e., classification and regression problems. In the real-world, unlabeled data consist of only a few labeled samples but many unlabeled samples. There are very less work with RVFL to handle semi-supervised problems. Therefore, it is desirable to develop variants of the RVFL model that can be employed for semi-supervised problems effectively.
- The model pool of ensemble RVFL mainly consists of learning algorithms. However, statistical methods can be included and improve the performance. For instance, statistical forecasting methods, such as ARIMA and exponential smoothing can be included in the model pool for forecasting tasks. Therefore, the models’ diversity is increased significantly.
- Most deep RVFLs networks are designed based on the conventional feed-forward architecture. However, the deep learning community has proposed various advanced architectures, such as the Transformer [251] and graph convolution neural network, etc. Combining the advanced architectures and the idea of deep RVFL may maintain high performance and reduce training time simultaneously.
- The existing literature utilizes static aggregation for the ensemble block of edRVFL. However, such a static ensemble does not consider the evolving characteristic or the concept drift problem. Recently, a dynamic ensemble algorithm that computes the ensemble weights based on the latest accuracy is proposed for forecasting by Liang et al. [252]. The output layers of edRVFL can be considered as different models. Hence, the dynamic ensemble can be applied to combine all output layers’ forecasts. Therefore,

a dynamic ensemble that assigns evolving weights can be combined with edRVFL to boost the performance further.

- The tuning process of edRVFL imposes a significant effect on the performance. A layer-wise tuning algorithm is proposed for time series forecasting in [190]. Such a tuning procedure benefits the diverse and optimal architecture of the edRVFL. However, Gao et al. [190] only implements a layer-wise tuning algorithm with Bayesian optimization. In the future, more advanced optimization algorithms can be combined with layer-wise tunings, such as evolutionary algorithms [253]. The marriage of layer-wise tuning and advanced optimization algorithms will develop the RVFLs into auto-ML in the future.
- Although the random features offer non-linearity and fast computation, the random nature carries redundant information. Therefore, an intelligent selection of the random features owns the strong potential to increase the performance [192, 191]. The inferior features of random layers may deteriorate the performance. The existing works only consider linear feature selection, and pruning techniques [192, 191]. However, there are more advanced feature selection algorithms [254]. The effects of applying different feature selection algorithms on the random features must be studied.
- Although RVFL and its variants show superior forecasting, there are still directions worth exploring. For instance, the augmentation of RVFL’s random features is not mature yet, although signal decomposition shows its effectiveness. If the decomposition is done correctly, the elements generated are always high-dimensional. In RVFL and deep RVFL, the direct links are connected to the linear output layers. Therefore, effective treatment of such high-dimensional features is critical. Potential solutions can be dimensionality reduction, feature selection and double regularizations. Dimensionality reduction algorithms can be utilized to transform the huge input feature matrix into a low-dimensional space. Then, a linear layer is trained in the low-dimensional space. Feature selection only selects a few best features for the linear layer to learn. As for the double regularizations, different regularizations are imposed on the direct link and random features. If the direct link is of high dimension, its regularization would prefer sparsity.
- RVFL’s forecasting ability on spatial-temporal time series is not investigated yet. The spatial-temporal time series is a temporal sequence of graph signals. However, the conventional version of RVFL is not suitable for graph data. Therefore, it is a promising direction to develop RVFL for graph data.
- Recently, multi-label learning has been emerging as an exciting research domain. Therefore, researchers may develop randomized neural networks to handle multi-label data. RVFL model doesn’t have enough work to

manage the multi-label datasets. The efficient and effective variants of the RVFL model should be developed for multi-label tasks.

- Unlike supervised learning, unsupervised learning problems doesn't have target variable information. Standard RVFL model needs target information while calculating the final parameters. Unlabeled data are clustered (or grouped) by considering their topological properties or other properties of the data. Therefore, the needful works are required to handle unlabeled data via RVFL model.
- The community lacks a thorough investigation that compares the performance of randomised neural networks on datasets that are openly accessible, utilising standardised metrics, evaluation procedures, and several datasets. A benchmark for comparing the various randomised architectures is thus needed in the field. This will encourage future efforts to improve a randomised model from various angles, including precision, trustworthiness, and training/inference efficiency.
- Most of the RVFL-based architectures are based on the offline training wherein all the data is available for the training at once. However, in online scenarios the sequential streaming data needs to be processed. RVFL models can be adapted to handle such scenarios. Moreover, one can also focus on the development of deep RVFL architectures for the online learning process. As for the edRVFL, the output layers can be trained in an online fashion and the ensemble can be online, too.

Acknowledgement

This work is supported by the National Supercomputing Mission under DST and Miety, Govt. of India under Grant No. DST/NSM/R&D_HPC_Appl/2021/03.29, as well as the Department of Science and Technology under Interdisciplinary Cyber Physical Systems (ICPS) Scheme grant no. DST/ICPS/CPS-Individual/2018/276 and Mathematical Research Impact-Centric Support (MATRICS) scheme grant no. MTR/2021/000787. Mr. Ashwani kumar Malik acknowledges the financial support (File no - 09/1022 (0075)/2019-EMR-I) given as scholarship by Council of Scientific and Industrial Research (CSIR), New Delhi, India. We are grateful to IIT Indore for the facilities and support being provided.

References

- [1] C. Zhang, Y. Lu, Study on artificial intelligence: The state of the art and future prospects, *Journal of Industrial Information Integration* 23 (2021) 100224.
- [2] S. Russell, Artificial intelligence: The future is superintelligent, *Nature* 548 (2017) 520–521.

- [3] R. Gao, W. X. Cheng, P. Suganthan, K. F. Yuen, Inpatient discharges forecasting for Singapore hospitals by machine learning, *IEEE Journal of Biomedical and Health Informatics* (2022).
- [4] M. Li, Z. Zhu, Spatial-temporal fusion graph neural networks for traffic flow forecasting, in: *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 2021, pp. 4189–4196.
- [5] P. Giudici, E. Raffinetti, Shapley-lorenz explainable artificial intelligence, *Expert Systems with Applications* 167 (2021) 114104.
- [6] R. Gao, R. Li, M. Hu, P. N. Suganthan, K. F. Yuen, Dynamic ensemble deep echo state network for significant wave height forecasting, *Applied Energy* 329 (2023) 120261.
- [7] P. J. Lisboa, A review of evidence of health benefit from artificial neural networks in medical intervention, *Neural Networks* 15 (2002) 11–39.
- [8] D. M. Himmelblau, Applications of artificial neural networks in chemical engineering, *Korean Journal of Chemical Engineering* 17 (2000) 373–392.
- [9] S.-Y. King, J.-N. Hwang, Neural network architectures for robotic applications, *IEEE Transactions on Robotics and Automation* 5 (1989) 641–657.
- [10] K. J. Hunt, D. Sbarbaro, R. Żbikowski, P. J. Gawthrop, Neural networks for control systems—a survey, *Automatica* 28 (1992) 1083–1112.
- [11] S. Chen, S. A. Billings, P. Grant, Non-linear system identification using neural networks, *International Journal of Control* 51 (1990) 1191–1214.
- [12] B. C. Csáji, Approximation with artificial neural networks, *Faculty of Sciences, Eötvös Loránd University, Hungary* 24 (2001) 7.
- [13] I. A. Basheer, M. Hajmeer, Artificial neural networks: fundamentals, computing, design, and application, *Journal of Microbiological Methods* 43 (2000) 3–31.
- [14] M. Leshno, V. Y. Lin, A. Pinkus, S. Schocken, Multilayer feedforward networks with a nonpolynomial activation function can approximate any function, *Neural Networks* 6 (1993) 861–867.
- [15] Y. Chauvin, D. E. Rumelhart, *Backpropagation: theory, architectures, and applications*, Psychology Press, 2013.
- [16] R. A. Jacobs, Increased rates of convergence through learning rate adaptation, *Neural Networks* 1 (1988) 295–307.
- [17] M. Gori, A. Tesi, On the problem of local minima in backpropagation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14 (1992) 76–86.

- [18] G. D. Magoulas, M. N. Vrahatis, G. S. Androulakis, Improving the convergence of the backpropagation algorithm using learning rate adaptation methods, *Neural Computation* 11 (1999) 1769–1796.
- [19] L. Zhang, P. N. Suganthan, A survey of randomized algorithms for training neural networks, *Information Sciences* 364 (2016) 146–155.
- [20] P. N. Suganthan, R. Katuwal, On the origins of randomization-based feedforward neural networks, *Applied Soft Computing* 105 (2021) 107239.
- [21] J. Park, I. W. Sandberg, Universal approximation using radial-basis-function networks, *Neural Computation* 3 (1991) 246–257.
- [22] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, *Neural Networks* 2 (1989) 359–366.
- [23] F. Scarselli, A. C. Tsoi, Universal approximation using feedforward neural networks: A survey of some existing methods, and some new results, *Neural Networks* 11 (1998) 15–37.
- [24] J. Zhang, G. G. Walter, Y. Miao, W. N. W. Lee, Wavelet neural networks for function learning, *IEEE Transactions on Signal Processing* 43 (1995) 1485–1497.
- [25] Y.-H. Pao, G.-H. Park, D. J. Sobajic, Learning and generalization characteristics of the random vector functional-link net, *Neurocomputing* 6 (1994) 163–180.
- [26] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, *SIAM Journal on Imaging Sciences* 2 (2009) 183–202.
- [27] D. S. Broomhead, D. Lowe, Radial basis functions, multi-variable functional interpolation and adaptive networks, Technical Report, Royal Signals and Radar Establishment Malvern (United Kingdom), 1988.
- [28] W. Cao, X. Wang, Z. Ming, J. Gao, A review on neural networks with random weights, *Neurocomputing* 275 (2018) 278–287.
- [29] C. Huang, M. Li, F. Cao, H. Fujita, Z. Li, X. Wu, Are graph convolutional networks with random weights feasible?, *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022).
- [30] W. F. Schmidt, M. A. Kraaijeveld, R. P. Duin, Feed forward neural networks with random weights, in: *International conference on pattern recognition*, IEEE Computer Society Press, 1992, pp. 1–1.
- [31] Y.-H. Pao, Y. Takefuji, Functional-link net computing: theory, system architecture, and functionalities, *Computer* 25 (1992) 76–79.

- [32] Y.-H. Pao, S. M. Phillips, D. J. Sobajic, Neural-net computing and the intelligent control of systems, *International Journal of Control* 56 (1992) 263–289.
- [33] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, *IEEE transactions on systems, man, and cybernetics* (1985) 116–132.
- [34] C. G. Looney, Radial basis functional link nets and fuzzy reasoning, *Neurocomputing* 48 (2002) 489–509.
- [35] B. Igel'nik, Y.-H. Pao, Stochastic choice of basis functions in adaptive function approximation and the functional-link net, *IEEE Transactions on Neural Networks* 6 (1995) 1320–1329.
- [36] D. Needell, A. A. Nelson, R. Saab, P. Salanevich, Random vector functional link networks for function approximation on manifolds, *arXiv preprint arXiv:2007.15776* (2020).
- [37] L. Zhang, P. N. Suganthan, A comprehensive evaluation of random vector functional link networks, *Information Sciences* 367 (2016) 1094–1105.
- [38] Y. Ren, P. N. Suganthan, N. Srikanth, G. Amaratunga, Random vector functional link network for short-term electricity load demand forecasting, *Information Sciences* 367 (2016) 1078–1093.
- [39] R. Bisoi, P. Dash, S. Mishra, Modes decomposition method in fusion with robust random vector functional link network for crude oil price forecasting, *Applied Soft Computing* 80 (2019) 475–493.
- [40] J. Del Ser, D. Casillas-Perez, L. Cornejo-Bueno, L. Prieto-Godino, J. Sanz-Justo, C. Casanova-Mateo, S. Salcedo-Sanz, Randomization-based machine learning in renewable energy prediction problems: critical literature review, new results and perspectives, *arXiv preprint arXiv:2103.14624* (2021).
- [41] T. Chakravorti, P. Satyanarayana, Non linear system identification using kernel based exponentially extended random vector functional link network, *Applied Soft Computing* 89 (2020) 106117.
- [42] Y. Zhang, J. Wu, Z. Cai, B. Du, S. Y. Philip, An unsupervised parameter learning model for RVFL neural network, *Neural Networks* 112 (2019) 85–97.
- [43] W. Cao, J. Gao, Z. Ming, S. Cai, H. Zheng, Impact of probability distribution selection on RVFL performance, in: *International Conference on Smart Computing and Communication*, Springer, 2017, pp. 114–124.

- [44] N. Vuković, M. Petrović, Z. Miljković, A comprehensive experimental evaluation of orthogonal polynomial expanded random vector functional link neural networks for regression, *Applied Soft Computing* 70 (2018) 1083–1096.
- [45] Y. Ren, L. Zhang, P. N. Suganthan, Ensemble classification and regression-recent developments, applications and future directions, *IEEE Computational Intelligence Magazine* 11 (2016) 41–53.
- [46] X. Dong, Z. Yu, W. Cao, Y. Shi, Q. Ma, A survey on ensemble learning, *Frontiers of Computer Science* 14 (2020) 241–258.
- [47] L. Breiman, Bagging predictors, *Machine Learning* 24 (1996) 123–140.
- [48] Y. Freund, R. E. Schapire, Experiments with a new boosting algorithm, in: *ICML*, volume 96, Citeseer, 1996, pp. 148–156.
- [49] D. H. Wolpert, Stacked generalization, *Neural Networks* 5 (1992) 241–259.
- [50] C. Zhang, F. Jiang, S. Wang, W. Shang, A novel hybrid approach with a decomposition method and the RVFL model for crude oil price prediction, in: *2020 IEEE International Conference on Big Data (Big Data)*, IEEE, 2020, pp. 4446–4449.
- [51] J. Shi, Z. Xue, Y. Dai, B. Peng, Y. Dong, Q. Zhang, Y. Zhang, Cascaded multi-column RVFL+ classifier for single-modal neuroimaging-based diagnosis of Parkinson’s disease, *IEEE Transactions on Biomedical Engineering* 66 (2018) 2362–2371.
- [52] R. Katuwal, P. N. Suganthan, Enhancing multi-class classification of random forest using random vector functional neural network and oblique decision surfaces, in: *2018 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2018, pp. 1–8.
- [53] C. Gallicchio, S. Scardapane, Deep randomized neural networks, *Recent Trends in Learning From Data* (2020) 43–68.
- [54] A. Voulodimos, N. Doulamis, A. Doulamis, E. Protopapadakis, Deep learning for computer vision: A brief review, *Computational Intelligence and Neuroscience* 2018 (2018).
- [55] Y. Li, C. Huang, L. Ding, Z. Li, Y. Pan, X. Gao, Deep learning in bioinformatics: Introduction, application, and perspective in the big data era, *Methods* 166 (2019) 4–21.
- [56] S. M. Marvasti-Zadeh, L. Cheng, H. Ghanei-Yakhdan, S. Kasaei, Deep learning for visual tracking: A comprehensive survey, *IEEE Transactions on Intelligent Transportation Systems* (2021).

- [57] A. B. Nassif, I. Shahin, I. Attili, M. Azzeh, K. Shaalan, Speech recognition using deep neural networks: A systematic review, *IEEE Access* 7 (2019) 19143–19165.
- [58] Y. Cao, T. A. Geddes, J. Y. H. Yang, P. Yang, Ensemble deep learning in bioinformatics, *Nature Machine Intelligence* 2 (2020) 500–508.
- [59] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, P. N. Suganthan, Ensemble deep learning: A review, *Engineering Applications of Artificial Intelligence* (2022).
- [60] Q. Shi, R. Katuwal, P. N. Suganthan, M. Tanveer, Random vector functional link neural network based ensemble deep learning, *Pattern Recognition* 117 (2021) 107978.
- [61] C. R. Rao, S. K. Mitra, Further contributions to the theory of generalized inverse of matrices and its applications, *Sankhyā: The Indian Journal of Statistics, Series A* (1971) 289–300.
- [62] W. Cao, L. Hu, J. Gao, X. Wang, Z. Ming, A study on the relationship between the rank of input data and the performance of random weight neural network, *Neural Computing and Applications* 32 (2020) 12685–12696.
- [63] A. Rasheed, A. Adebisi, K. C. Veluvolu, Respiratory motion prediction with random vector functional link (RVFL) based neural networks, in: *Journal of Physics: Conference Series*, volume 1626, IOP Publishing, 2020, p. 012022.
- [64] G. Dudek, Are direct links necessary in random vector functional link networks for regression?, in: *International Conference on Artificial Intelligence and Soft Computing*, Springer, 2020, pp. 60–70.
- [65] W. Cao, M. J. Patwary, P. Yang, X. Wang, Z. Ming, An initial study on the relationship between meta features of dataset and the initialization of NNRW, in: *2019 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2019, pp. 1–8.
- [66] M. Tanveer, M. A. Ganaie, P. N. Suganthan, Ensemble of classification models with weighted functional link network, *Applied Soft Computing* 107 (2021) 107322.
- [67] L. Fan, W. Sun, G. Feng, Image steganalysis via random subspace fisher linear discriminant vector functional link network and feature mapping, *Mobile Networks and Applications* 24 (2019) 1269–1278.
- [68] J. Pan, J. Fan, A. Dong, Y. Li, Random vector functional link network optimized by jaya algorithm for transient stability assessment of power systems, *Mathematical Problems in Engineering* 2020 (2020).

- [69] R. Rao, Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems, *International Journal of Industrial Engineering Computations* 7 (2016) 19–34.
- [70] S.-Y. Lu, S.-H. Wang, Y.-D. Zhang, A classification method for brain MRI via mobilenet and feedforward network with random weights, *Pattern Recognition Letters* 140 (2020) 252–260.
- [71] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, MobileNetV2: inverted residuals and linear bottlenecks, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4510–4520.
- [72] X. Li, Y. Yang, N. Hu, Z. Cheng, J. Cheng, Discriminative manifold random vector functional link neural network for rolling bearing fault diagnosis, *Knowledge-Based Systems* 211 (2021) 106507.
- [73] X. Guo, W. Zhou, Q. Lu, A. Du, Y. Cai, Y. Ding, Assessing dry weight of hemodialysis patients via sparse Laplacian regularized RVFL neural network with $L_{2,1}$ -norm, *BioMed Research International* 2021 (2021).
- [74] M. A. Ganaie, M. Tanveer, P. N. Suganthan, Minimum variance embedded random vector functional link network, in: *International Conference on Neural Information Processing*, Springer, 2020, pp. 412–419.
- [75] S. Parija, R. Bisoi, P. Dash, M. Sahani, Deep long short term memory based minimum variance kernel random vector functional link network for epileptic EEG signal classification, *Engineering Applications of Artificial Intelligence* 105 (2021) 104426.
- [76] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Computation* 9 (1997) 1735–1780.
- [77] M. A. Ganaie, M. Tanveer, P. N. Suganthan, Co-trained random vector functional link network, in: *2021 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2021, pp. 1–8.
- [78] P. Dai, F. Gwady-Sridhar, M. Bauer, M. Borrie, X. Teng, Healthy cognitive aging: a hybrid random vector functional-link model for the analysis of Alzheimer’s disease, in: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2017, pp. 4567–4573.
- [79] W. Cui, L. Zhang, B. Li, J. Guo, W. Meng, H. Wang, L. Xie, Received signal strength based indoor positioning using a random vector functional link network, *IEEE Transactions on Industrial Informatics* 14 (2017) 1846–1855.
- [80] D. Samal, P. K. Dash, R. Bisoi, Modified added activation function based exponential robust random vector functional link network with expanded version for nonlinear system identification, *Applied Intelligence* (2021) 1–27.

- [81] W. Dai, Q. Chen, F. Chu, X. Ma, T. Chai, Robust regularized random vector functional link network and its industrial application, *IEEE Access* 5 (2017) 16162–16172.
- [82] J. Chen, S. Yang, D. Zhang, Y. Nanekaran, A turning point prediction method of stock price based on RVFL-GMDH and chaotic time series analysis, *Knowledge and Information Systems* (2021) 1–26.
- [83] J.-A. Mueller, A. Ivachnenko, F. Lemke, GMDH algorithms for complex systems modelling, *Mathematical and Computer Modelling of Dynamical Systems* 4 (1998) 275–316.
- [84] P. Zhou, Y. Lv, H. Wang, T. Chai, Data-driven robust RVFLNs modeling of a blast furnace iron-making process using cauchy distribution weighted m-estimation, *IEEE Transactions on Industrial Electronics* 64 (2017) 7141–7151.
- [85] A. K. Malik, M. A. Ganaie, M. Tanveer, P. N. Suganthan, for the Alzheimer’s Disease Neuroimaging Initiative, Alzheimer’s disease diagnosis via intuitionistic fuzzy random vector functional link network, *IEEE Transactions on Computational Social Systems* (2022). doi:10.1109/TCSS.2022.3146974.
- [86] A. K. Malik, M. A. Ganaie, M. Tanveer, Graph embedded intuitionistic fuzzy weighted random vector functional link network, in: *IEEE Symposium Series on Computational Intelligence*, IEEE, 2022.
- [87] N. Ahmad, M. A. Ganaie, A. K. Malik, K. T. Lai, M. Tanveer, Minimum variance embedded intuitionistic fuzzy weighted random vector functional link network, in: *International Conference on Neural Information Processing*, IEEE, 2022.
- [88] B. B. Hazarika, D. Gupta, 1-norm random vector functional link networks for classification problems, *Complex & Intelligent Systems* (2022) 1–17.
- [89] P.-B. Zhang, Z.-X. Yang, A new learning paradigm for random vector functional-link network: RVFL+, *Neural Networks* 122 (2020) 94–105.
- [90] D. R. Nayak, R. Dash, B. Majhi, U. R. Acharya, Application of fast curvelet tsallis entropy and kernel random vector functional link network for automated detection of multiclass brain abnormalities, *Computerized Medical Imaging and Graphics* 77 (2019) 101656.
- [91] S. Parija, P. K. Dash, R. Bisoi, Multi-kernel-based random vector functional link network with decomposed features for epileptic EEG signal classification, *IET Signal Processing* 14 (2020) 162–174.
- [92] S. Scardapane, D. Wang, A. Uncini, Bayesian random vector functional-link networks for robust data modeling, *IEEE Transactions on Cybernetics* 48 (2017) 2049–2059.

- [93] H. Ye, F. Cao, D. Wang, A hybrid regularization approach for random vector functional-link networks, *Expert Systems with Applications* 140 (2020) 112912.
- [94] A. Alalimi, L. Pan, M. A. Al-Qaness, A. A. Ewees, X. Wang, M. Abd Elaziz, Optimized random vector functional link network to predict oil production from tahe oil field in china, *Oil & Gas Science and Technology–Revue d’IFP Energies nouvelles* 76 (2021) 3.
- [95] W. Dai, Y. Ao, L. Zhou, P. Zhou, X. Wang, Incremental learning paradigm with privileged information for random vector functional-link networks: IRVFL+, *Neural Computing and Applications* (2022) 1–13.
- [96] W. Jiao, S. Song, H. Han, W. Wang, Q. Zhang, Artificially intelligent differential diagnosis of enlarged lymph nodes with random vector functional link network plus, *Medical Engineering & Physics* (2022) 103939.
- [97] V. H. A. Ribeiro, R. Santana, G. Reynoso-Meza, Random vector functional link forests and extreme learning forests applied to uav automatic target recognition, *Engineering Applications of Artificial Intelligence* 117 (2023) 105538.
- [98] Y. Zhang, R. Guo, Y. Peng, W. Kong, F. Nie, B.-L. Lu, An auto-weighting incremental random vector functional link network for eeg-based driving fatigue detection, *IEEE Transactions on Instrumentation and Measurement* 71 (2022) 1–14.
- [99] M. Abd Elaziz, S. Senthilraja, M. E. Zayed, A. H. Elsheikh, R. R. Mostafa, S. Lu, A new random vector functional link integrated with mayfly optimization algorithm for performance prediction of solar photovoltaic thermal collector combined with electrolytic hydrogen production system, *Applied Thermal Engineering* 193 (2021) 117055.
- [100] A. H. Elsheikh, T. A. Shehabeldeen, J. Zhou, E. Showaib, M. Abd Elaziz, Prediction of laser cutting parameters for polymethylmethacrylate sheets using random vector functional link network integrated with equilibrium optimizer, *Journal of Intelligent Manufacturing* 32 (2021) 1377–1388.
- [101] G. A. Ruz, P. A. Henríquez, Random vector functional link with naive bayes for classification problems of mixed data, in: *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, IEEE, 2019, pp. 1749–1752.
- [102] P. A. Henriquez, G. A. Ruz, A non-iterative method for pruning hidden neurons in neural networks with random weights, *Applied Soft Computing* 70 (2018) 1109–1121.
- [103] M. Pratama, P. P. Angelov, E. Lughofer, M. J. Er, Parsimonious random vector functional link network for data streams, *Information Sciences* 430 (2018) 519–537.

- [104] Y. Wang, D. Gong, L. Pang, D. Yang, RVFL-based optical fiber intrusion signal recognition with multi-level wavelet decomposition as feature, *Photonic Sensors* 8 (2018) 234–241.
- [105] E. M. El-Said, M. Abd Elaziz, A. H. Elsheikh, Machine learning algorithms for improving the prediction of air injection effect on the thermohydraulic performance of shell and tube heat exchanger, *Applied Thermal Engineering* 185 (2021) 116471.
- [106] J. A. Suykens, J. Vandewalle, Least squares support vector machine classifiers, *Neural Processing Letters* 9 (1999) 293–300.
- [107] M.-L. Zhang, Z.-H. Zhou, A k-nearest neighbor based algorithm for multi-label classification, in: 2005 IEEE International Conference on Granular Computing, volume 2, IEEE, 2005, pp. 718–721.
- [108] D. Candel, R. Nanculef, C. Concha, H. Allende, A sequential minimal optimization algorithm for the all-distances support vector machine, in: *Iberoamerican Congress on Pattern Recognition*, Springer, 2010, pp. 484–491.
- [109] A. J. Smola, B. Schölkopf, A tutorial on support vector regression, *Statistics and Computing* 14 (2004) 199–222.
- [110] P. Borah, D. Gupta, Unconstrained convex minimization based implicit lagrangian twin random vector functional-link networks for binary classification (ULTRVFLC), *Applied Soft Computing* 81 (2019) 105534.
- [111] M. A. Kumar, M. Gopal, Least squares twin support vector machines for pattern classification, *Expert Systems with Applications* 36 (2009) 7535–7543.
- [112] M. A. Ganaie, M. Tanveer, LSTSVM classifier with enhanced features from pre-trained functional link network, *Applied Soft Computing* 93 (2020) 106305.
- [113] L. Tang, C. Zhang, L. Li, S. Wang, A multi-scale method for forecasting oil price with multi-factor search engine data, *Applied Energy* 257 (2020) 114033.
- [114] S. S. S. Mary, A. Sasithradevi, S. M. M. Roomi, J. J. Immanuvel, A random vector functional link network based content based image retrieval, in: 2019 Fifth International Conference on Science Technology Engineering and Mathematics (ICONSTEM), volume 1, IEEE, 2019, pp. 486–492.
- [115] Z. Zhou, B. Wu, Adaptive sliding mode control of manipulators based on fuzzy random vector function links for friction compensation, *Optik* 227 (2021) 166055.

- [116] E. T. Elkabbash, R. R. Mostafa, S. I. Barakat, Android malware classification based on random vector functional link and artificial jellyfish search optimizer, *PloS One* 16 (2021) e0260232.
- [117] Z. Zhou, Z. Ma, Y. Wang, Z. Zhu, Fabric wrinkle rating model based on resnet18 and optimized random vector functional-link network, *Textile Research Journal* 93 (2023) 172–193.
- [118] J. Huang, C.-M. Vong, W. Qian, Q. Huang, Y. Zhou, Online label distribution learning using random vector functional-link network, *IEEE Transactions on Emerging Topics in Computational Intelligence* (2022).
- [119] W. Cao, P. Yang, Z. Ming, S. Cai, J. Zhang, An improved fuzziness based random vector functional link network for liver disease detection, in: 2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing,(HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS), IEEE, 2020, pp. 42–48.
- [120] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, *Journal of Artificial Intelligence Research* 16 (2002) 321–357.
- [121] W. Cao, J. Gao, Z. Ming, S. Cai, Z. Shan, Fuzziness based random vector functional-link network for semi-supervised learning, in: 2017 International Conference on Computational Science and Computational Intelligence (CSCI), IEEE, 2017, pp. 782–786.
- [122] M. Sahani, P. K. Dash, FPGA-based online power quality disturbances monitoring using reduced-sample HHT and class-specific weighted RVFLN, *IEEE Transactions on Industrial Informatics* 15 (2019) 4614–4623.
- [123] V. Chauhan, A. Tiwari, Randomized neural networks for multilabel classification, *Applied Soft Computing* (2021) 108184.
- [124] M. Ganaie, M. Tanveer, A. Malik, P. N. Suganthan, Minimum variance embedded random vector functional link network with privileged information, in: 2022 International Joint Conference on Neural Networks (IJCNN), IEEE, 2022.
- [125] C. Zhang, S.-K. Oh, W. Pedrycz, Z. Fu, S. Lu, Reinforced fuzzy clustering-based rule model constructed with the aid of exponentially weighted L2 regularization strategy and augmented random vector functional link network, *Fuzzy Sets and Systems* (2021). doi:10.1016/j.fss.2021.09.022.
- [126] R. Gao, L. Du, K. F. Yuen, P. N. Suganthan, Walk-forward empirical wavelet random vector functional link for time series forecasting, *Applied Soft Computing* 108 (2021) 107450.

- [127] M. E. Zayed, J. Zhao, W. Li, A. H. Elsheikh, M. Abd Elaziz, D. Yousri, S. Zhong, Z. Mingxi, Predicting the performance of solar dish stirling power plant using a hybrid random vector functional link/chimp optimization model, *Solar Energy* 222 (2021) 1–17.
- [128] D. R. Dash, P. Dash, R. Bisoi, Short term solar power forecasting using hybrid minimum variance expanded RVFLN and sine-cosine levy flight PSO algorithm, *Renewable Energy* 174 (2021) 513–537.
- [129] B. B. Hazarika, D. Gupta, Modelling and forecasting of Covid-19 spread using wavelet-coupled random vector functional link networks, *Applied Soft Computing* 96 (2020) 106626.
- [130] M. Abd Elaziz, T. A. Shehabeldeen, A. H. Elsheikh, J. Zhou, A. A. Ewees, M. A. Al-qaness, Utilization of random vector functional link integrated with marine predators algorithm for tensile behavior prediction of dissimilar friction stir welded aluminum alloy joints, *Journal of Materials Research and Technology* 9 (2020) 11370–11381.
- [131] F. Essa, M. Abd Elaziz, A. H. Elsheikh, Prediction of power consumption and water productivity of seawater greenhouse system using random vector functional link network integrated with artificial ecosystem-based optimization, *Process Safety and Environmental Protection* 144 (2020) 322–329.
- [132] S. W. Sharshir, M. Abd Elaziz, M. Elkadeem, Enhancing thermal performance and modeling prediction of developed pyramid solar still utilizing a modified random vector functional link, *Solar Energy* 198 (2020) 399–409.
- [133] A. M. Hussein, M. Abd Elaziz, M. S. A. Wahed, M. Sillanpää, A new approach to predict the missing values of algae during water quality monitoring programs based on a hybrid moth search algorithm and the random vector functional link network, *Journal of Hydrology* 575 (2019) 852–863.
- [134] P. Zhou, Y. Jiang, C. Wen, T. Chai, Data modeling for quality prediction using improved orthogonal incremental random vector functional-link networks, *Neurocomputing* 365 (2019) 1–9.
- [135] Y. Dash, S. K. Mishra, S. Sahany, B. K. Panigrahi, Indian summer monsoon rainfall prediction: a comparison of iterative and non-iterative approaches, *Applied Soft Computing* 70 (2018) 1122–1134.
- [136] M. F. Nhabangue, G. Pillai, Wind speed forecasting using improved random vector functional link network, in: 2018 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, 2018, pp. 1744–1750.
- [137] K.-K. Xu, H.-X. Li, H.-D. Yang, Kernel-based random vector functional-link network for fast learning of spatiotemporal dynamic processes, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 49 (2017) 1016–1026.

- [138] L. Zhang, P. Zhou, M. Yuan, T.-y. Chai, Multivariable dynamic modeling for molten iron quality using incremental random vector functional-link networks, *Journal of Iron and Steel Research International* 23 (2016) 1151–1159.
- [139] P. Zhou, M. Yuan, H. Wang, Z. Wang, T.-Y. Chai, Multivariable dynamic modeling for molten iron quality using online sequential random vector functional-link networks with self-feedback connections, *Information Sciences* 325 (2015) 237–255.
- [140] Y. Ren, X. Qiu, P. N. Suganthan, G. Amaratunga, Detecting wind power ramp with random vector functional link (RVFL) network, in: 2015 IEEE Symposium Series on Computational Intelligence, IEEE, 2015, pp. 687–694.
- [141] Y. Peng, Q. Li, W. Kong, F. Qin, J. Zhang, A. Cichocki, A joint optimization framework to semi-supervised RVFL and ELM networks for efficient data classification, *Applied Soft Computing* 97 (2020) 106756.
- [142] J. Wang, X. Shen, W. Pan, On transductive support vector machines, *Contemporary Mathematics* 443 (2007) 7–20.
- [143] S. Scardapane, D. Comminiello, M. Scarpiniti, A. Uncini, A semi-supervised random vector functional-link network based on the transductive framework, *Information Sciences* 364 (2016) 156–166.
- [144] J. Xie, S. Liu, H. Dai, Y. Rong, Distributed semi-supervised learning algorithms for random vector functional-link networks with distributed data splitting across samples and features, *Knowledge-Based Systems* 195 (2020) 105577.
- [145] A. K. Jain, R. C. Dubes, *Algorithms for clustering data*, Prentice-Hall, Inc., 1988.
- [146] M. Tanveer, T. Gupta, M. Shah, A. D. N. Initiative, Pinball loss twin support vector clustering, *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 17 (2021) 1–23.
- [147] B. Richhariya, M. Tanveer, A. D. N. Initiative, Least squares projection twin support vector clustering (LSPTSVC), *Information Sciences* 533 (2020) 1–23.
- [148] Y. Zhang, Q. Zhu, Y. Peng, W. Kong, An unsupervised discriminative random vector functional link network for efficient data clustering, in: 2021 4th International Conference on Pattern Recognition and Artificial Intelligence (PRAI), IEEE, 2021, pp. 347–352.
- [149] Z.-H. Zhou, Ensemble learning, in: *Machine Learning*, Springer, 2021, pp. 181–210.

- [150] A. K. Malik, M. A. Ganaie, M. Tanveer, P. N. Suganthan, A novel ensemble method of RVFL for classification problem, in: 2021 International Joint Conference on Neural Networks (IJCNN), IEEE, 2021, pp. 1–8.
- [151] L. Yu, Y. Wu, L. Tang, H. Yin, K. K. Lai, Investigation of diversity strategies in RVFL network ensemble learning for crude oil price forecasting, *Soft Computing* 25 (2021) 3609–3622.
- [152] L. Zhang, X. Zhang, H. Chen, H. Tang, A robust temperature prediction model of shuttle kiln based on ensemble random vector functional link network, *Applied Thermal Engineering* 150 (2019) 99–110.
- [153] A. Tahir, G. Morison, D. A. Skelton, R. M. Gibson, A novel functional link network stacking ensemble with fractal features for multichannel fall detection, *Cognitive Computation* 12 (2020) 1024–1042.
- [154] X. Qiu, P. N. Suganthan, A. G. Amaratunga, Ensemble incremental random vector functional link network for short-term crude oil price forecasting, in: 2018 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, 2018, pp. 1758–1763.
- [155] X. Qiu, P. N. Suganthan, G. A. Amaratunga, Ensemble incremental learning random vector functional link network for short-term electric load forecasting, *Knowledge-Based Systems* 145 (2018) 182–196.
- [156] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, H. H. Liu, The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis, *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 454 (1998) 903–995.
- [157] G. Rilling, P. Flandrin, P. Gonçalves, J. M. Lilly, Bivariate empirical mode decomposition, *IEEE Signal Processing Letters* 14 (2007) 936–939.
- [158] Z. Wu, N. E. Huang, Ensemble empirical mode decomposition: a noise-assisted data analysis method, *Advances in Adaptive Data Analysis* 1 (2009) 1–41.
- [159] M. E. Torres, M. A. Colominas, G. Schlotthauer, P. Flandrin, A complete ensemble empirical mode decomposition with adaptive noise, in: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2011, pp. 4144–4147.
- [160] X. Qiu, P. N. Suganthan, G. A. Amaratunga, Fusion of multiple indicators with ensemble incremental learning techniques for stock price forecasting, *Journal of Banking and Financial Technology* 3 (2019) 33–42.
- [161] W. X. Cheng, P. N. Suganthan, X. Qiu, R. Katuwal, Classification of stock market trends with confidence-based selective predictions, in: *Swarm, Evolutionary, and Memetic Computing and Fuzzy and Neural Computing*, Springer, 2019, pp. 93–104.

- [162] J. Gilles, Empirical wavelet transform, *IEEE Transactions on Signal Processing* 61 (2013) 3999–4010.
- [163] K. Dragomiretskiy, D. Zosso, Variational mode decomposition, *IEEE Transactions on Signal Processing* 62 (2013) 531–544.
- [164] X. Qiu, P. N. Suganthan, G. A. Amaratunga, Electricity load demand time series forecasting with empirical mode decomposition based random vector functional link network, in: *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, IEEE, 2016, pp. 001394–001399.
- [165] L. Tang, Y. Wu, L. Yu, A randomized-algorithm-based decomposition-ensemble learning methodology for energy price forecasting, *Energy* 157 (2018) 526–538.
- [166] L. Tang, Y. Wu, L. Yu, A non-iterative decomposition-ensemble learning paradigm using RVFL network for crude oil price forecasting, *Applied Soft Computing* 70 (2018) 1097–1108.
- [167] L. Li, X. Qu, J. Zhang, H. Li, B. Ran, Travel time prediction for highway network based on the ensemble empirical mode decomposition and random vector functional link network, *Applied Soft Computing* 73 (2018) 921–932.
- [168] H. Sun, W. Zhai, Y. Wang, L. Yin, F. Zhou, Privileged information-driven random network based non-iterative integration model for building energy consumption prediction, *Applied Soft Computing* 108 (2021) 107438.
- [169] X. Qiu, Y. Ren, P. N. Suganthan, G. A. Amaratunga, Short-term wind power ramp forecasting with empirical mode decomposition based ensemble learning techniques, in: *2017 IEEE Symposium Series on computational intelligence (SSCI)*, IEEE, 2017, pp. 1–8.
- [170] J. Wu, T. Zhou, T. Li, A hybrid approach integrating multiple ICEEM-DANs, WOA, and RVFL networks for economic and financial time series forecasting, *Complexity* 2020 (2020).
- [171] J. Wu, F. Miu, T. Li, Daily crude oil price forecasting based on improved CEEMDAN, SCA, and RVFL: a case study in WTI oil market, *Energies* 13 (2020) 1852.
- [172] J. Lu, J. Ding, C. Liu, Y. Jin, Prediction of physical properties of crude oil based on ensemble random weights neural network, *IFAC-PapersOnLine* 51 (2018) 655–660.
- [173] C. Lian, L. Zhu, Z. Zeng, Y. Su, W. Yao, H. Tang, Constructing prediction intervals for landslide displacement using bootstrapping random vector functional link networks selective ensemble with neural networks switched, *Neurocomputing* 291 (2018) 1–10.

- [174] J. Lu, C. Li, J. Ding, Ensemble random weights neural network based online prediction model of the production rate for mineral beneficiation process, *IFAC-PapersOnLine* 51 (2018) 1–6.
- [175] B. Miskony, D. Wang, A randomized algorithm for prediction interval using RVFL networks ensemble, in: *International Conference on Neural Information Processing*, Springer, 2017, pp. 51–60.
- [176] D. P. Mesquita, J. P. P. Gomes, L. R. Rodrigues, S. A. Oliveira, R. K. Galvao, Building selective ensembles of randomization based neural networks with the successive projections algorithm, *Applied Soft Computing* 70 (2018) 1135–1145.
- [177] C. Ren, Y. Xu, Y. Zhang, R. Zhang, A hybrid randomized learning system for temporal-adaptive voltage stability assessment of power systems, *IEEE Transactions on Industrial Informatics* 16 (2019) 3672–3684.
- [178] Y. Xia, Y. Xu, B. Gou, A data-driven method for IGBT open-circuit fault diagnosis based on hybrid ensemble learning and sliding-window classification, *IEEE Transactions on Industrial Informatics* 16 (2019) 5223–5233.
- [179] Y. Liu, W. Cao, Z. Ming, Q. Wang, J. Zhang, Z. Xu, Ensemble neural networks with random weights for classification problems, in: *2020 3rd International Conference on Algorithms, Computing and Artificial Intelligence*, 2020, pp. 1–5.
- [180] L. Zhang, P. N. Suganthan, Benchmarking ensemble classifiers with novel co-trained kernel ridge regression and random vector functional link ensembles [research frontier], *IEEE Computational Intelligence Magazine* 12 (2017) 61–72.
- [181] A. K. Malik, M. A. Ganaie, M. Tanveer, P. N. Suganthan, Extended features based random vector functional link network for classification problem, *IEEE Transactions on Computational Social Systems* (2022) 1–10. doi:10.1109/TCSS.2022.3187461.
- [182] R. Sharma, T. Goel, M. Tanveer, P. N. Suganthan, I. Razzak, R. Murugan, Conv-eRVFL: Convolutional neural network based ensemble RVFL classifier for alzheimer’s disease diagnosis, *IEEE Journal of Biomedical and Health Informatics* (2022).
- [183] S.-Y. Lu, D. R. Nayak, S.-H. Wang, Y.-D. Zhang, A cerebral microbleed diagnosis method via featurenet and ensembled randomized neural networks, *Applied Soft Computing* (2021) 107567.
- [184] Y. Hu, B. Qu, J. Wang, J. Liang, Y. Wang, K. Yu, Y. Li, K. Qiao, Short-term load forecasting using multimodal evolutionary algorithm and random vector functional link network based ensemble learning, *Applied Energy* 285 (2021) 116415.

- [185] S. Chen, J. Gao, R. Zhao, H. Fu, Selective ensemble modeling method based on random vector functional link network and game theory, in: 2019 6th International Conference on Systems and Informatics (ICSAI), IEEE, 2019, pp. 584–588.
- [186] P. Musikawan, K. Sunat, Y. Kongsorot, P. Horata, S. Chiewchanwattana, Parallelized metaheuristic-ensemble of heterogeneous feedforward neural networks for regression problems, *IEEE Access* 7 (2019) 26909–26932.
- [187] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, L. Farhan, Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions, *Journal of big Data* 8 (2021) 1–74.
- [188] L. Zhang, P. N. Suganthan, Visual tracking with convolutional random vector functional link network, *IEEE Transactions on Cybernetics* 47 (2017) 3243–3253.
- [189] P. A. Henríquez, G. A. Ruz, Twitter sentiment classification based on deep random vector functional link, in: 2018 International Joint Conference on Neural Networks (IJCNN), IEEE, 2018, pp. 1–6.
- [190] R. Gao, L. Du, P. N. Suganthan, Q. Zhou, K. F. Yuen, Random vector functional link neural network based ensemble deep learning for short-term load forecasting, *Expert Systems with Applications* 206 (2022) 117784.
- [191] Q. Shi, M. Hu, P. N. Suganthan, R. Katuwal, Weighting and pruning based ensemble deep random vector functional link network for tabular data classification, *Pattern Recognition* (2022) 108879.
- [192] W. X. Cheng, P. N. Suganthan, R. Katuwal, Time series classification using diversified ensemble deep random vector functional link and resnet features, *Applied Soft Computing* (2021) 107826. doi:<https://doi.org/10.1016/j.asoc.2021.107826>.
- [193] R. Katuwal, P. N. Suganthan, Stacked autoencoder based deep random vector functional link neural network for classification, *Applied Soft Computing* 85 (2019) 105854.
- [194] M. Hu, J. H. Chion, P. N. Suganthan, R. K. Katuwal, Ensemble deep random vector functional link neural network for regression, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2022) 1–12. doi:[10.1109/TSMC.2022.3213628](https://doi.org/10.1109/TSMC.2022.3213628).
- [195] L. Du, R. Gao, P. N. Suganthan, D. Z. Wang, Time series forecasting using online performance-based ensemble deep random vector functional link neural network, in: 2022 International Joint Conference on Neural Networks (IJCNN), IEEE, 2022, pp. 1–7.

- [196] X. Yu, C. Lian, Y. Su, B. Xu, X. Wang, W. Yao, H. Tang, Selective ensemble deep bidirectional rvfln for landslide displacement prediction, *Natural Hazards* 112 (2022) 725–745.
- [197] M. Hu, R. Gao, P. N. Suganthan, M. Tanveer, Automated layer-wise solution for ensemble deep randomized feed-forward neural network, *Neurocomputing* 514 (2022) 137–147.
- [198] Q. Shi, P. N. Suganthan, J. Del Ser, Jointly optimized ensemble deep random vector functional link network for semi-supervised classification, *Engineering Applications of Artificial Intelligence* 115 (2022) 105214.
- [199] M. A. Ganaie, M. Tanveer, Ensemble deep random vector functional link network using privileged information for Alzheimer’s disease diagnosis, *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (2022).
- [200] A. K. Malik, M. Tanveer, Graph embedded ensemble deep randomized network for diagnosis of alzheimer’s disease, *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (2022).
- [201] R. Sharma, T. Goel, M. Tanveer, S. Dwivedi, R. Murugan, FAF-DRVFL: Fuzzy activation function based deep random vector functional links network for early diagnosis of alzheimer disease, *Applied Soft Computing* 106 (2021) 107371.
- [202] Q. Dai, G. Zhang, Z. Fang, B. Xue, SAR target recognition with modified convolutional random vector functional link network, *IEEE Geoscience and Remote Sensing Letters* (2021).
- [203] M. Hu, R. Gao, P. Suganthan, Deep reservoir computing based random vector functional link for non-sequential classification, in: *2022 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2022, pp. 1–8.
- [204] S. Yang, R. Gao, L. Li, W. T. Ang, Deep randomized feed-forward networks based prediction of human joint angles using wearable inertial measurement unit: Performance comparison, in: *2022 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2022, pp. 01–08.
- [205] M. Hu, P. N. Suganthan, Representation learning using deep random vector functional link networks for clustering, *Pattern Recognition* 129 (2022) 108744.
- [206] K. Maeda, S. Takahashi, T. Ogawa, M. Haseyama, Convolutional sparse coding-based deep random vector functional link network for distress classification of road structures, *Computer-Aided Civil and Infrastructure Engineering* 34 (2019) 654–676.

- [207] M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim, Do we need hundreds of classifiers to solve real world classification problems?, *The Journal of Machine Learning Research* 15 (2014) 3133–3181.
- [208] G. Klambauer, T. Unterthiner, A. Mayr, S. Hochreiter, Self-normalizing neural networks, *Advances in neural information processing systems* 30 (2017).
- [209] J. Del Ser, D. Casillas-Perez, L. Cornejo-Bueno, L. Prieto-Godino, J. Sanz-Justo, C. Casanova-Mateo, S. Salcedo-Sanz, Randomization-based machine learning in renewable energy prediction problems: critical literature review, new results and perspectives, *Applied Soft Computing* (2022) 108526.
- [210] J. Demšar, Statistical comparisons of classifiers over multiple data sets, *The Journal of Machine Learning Research* 7 (2006) 1–30.
- [211] J. Carrasco, S. García, M. Rueda, S. Das, F. Herrera, Recent trends in the use of statistical tests for comparing swarm and evolutionary computing algorithms: Practical guidelines and a critical review, *Swarm and Evolutionary Computation* 54 (2020) 100665.
- [212] A. Aggarwal, M. Tripathi, Short-term solar power forecasting using random vector functional link (RVFL) network, in: *Ambient Communications and Computer Systems*, Springer, 2018, pp. 29–39.
- [213] I. Majumder, P. Dash, S. Dhar, Real-time energy management for PV–battery–wind based microgrid using on-line sequential kernel based robust random vector functional link network, *Applied Soft Computing* 101 (2021) 107059.
- [214] E. L. Manibardo, I. Laña, J. Del Ser, Random vector functional link networks for road traffic forecasting: Performance comparison and stability analysis, in: *2021 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2021, pp. 1–8.
- [215] Z. Cheng, J. Wang, A new combined model based on multi-objective salp swarm optimization for wind speed forecasting, *Applied Soft Computing* 92 (2020) 106294.
- [216] X. Zhang, Y. Zou, S. Li, Enhancing incremental deep learning for FCCU end-point quality prediction, *Information Sciences* 530 (2020) 95–107.
- [217] V. Kushwaha, N. M. Pindoriya, A SARIMA-RVFL hybrid model assisted by wavelet decomposition for very short-term solar PV power generation forecast, *Renewable Energy* 140 (2019) 124–139.
- [218] I. Majumder, P. K. Dash, R. Bisoi, Short-term solar power prediction using multi-kernel-based random vector functional link with water cycle algorithm-based parameter optimization, *Neural Computing and Applications* 32 (2020) 8011–8029.

- [219] T. Moudiki, F. Planchet, A. Cousin, Multiple time series forecasting using quasi-randomized functional link neural networks, *Risks* 6 (2018) 22.
- [220] C. Lian, L. Zhu, Z. Zeng, Y. Su, W. Yao, H. Tang, Constructing prediction intervals for landslide displacement using bootstrapping random vector functional link networks selective ensemble with neural networks switched, *Neurocomputing* 291 (2018) 1–10. doi:<https://doi.org/10.1016/j.neucom.2018.02.046>.
- [221] D. Gupta, N. Natarajan, M. Berlin, Short-term wind speed prediction using hybrid machine learning techniques, *Environmental Science and Pollution Research* (2021) 1–19.
- [222] J. Li, L. Tang, S. Wang, Forecasting crude oil price with multilingual search engine data, *Physica A: Statistical Mechanics and its Applications* 551 (2020) 124178.
- [223] X. Xue, W. Sun, J. Wang, Q. Li, G. Luo, K. Yu, RVFL-LQP: RVFL-based link quality prediction of wireless sensor networks in smart grid, *IEEE Access* 8 (2020) 7829–7841.
- [224] Y. Zhang, B. Li, Y. Wang, Q. Tian, The forecast of the temperature in subway station based on RVFL neural network, in: 2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), IEEE, 2018, pp. 1–5.
- [225] C. Lian, Z. Zeng, X. Wang, W. Yao, Y. Su, H. Tang, Landslide displacement interval prediction using lower upper bound estimation method with pre-trained random vector functional link network initialization, *Neural Networks* 130 (2020) 286–296.
- [226] C. P. Chen, S. R. LeClair, Y.-H. Pao, An incremental adaptive implementation of functional-link processing for function approximation, time-series prediction, and system identification, *Neurocomputing* 18 (1998) 11–31.
- [227] H. Qu, S. Fu, L. Pang, C. Ding, H. Zhang, Rapid temperature prediction method for electronic equipment cabin, *Applied Thermal Engineering* 138 (2018) 83–93.
- [228] P. Zhou, Y. Jiang, C. Wen, T. Chai, Data modeling for quality prediction using improved orthogonal incremental random vector functional-link networks, *Neurocomputing* 365 (2019) 1–9.
- [229] F. Rosenblatt, The perceptron: a probabilistic model for information storage and organization in the brain., *Psychological Review* 65 (1958) 386.
- [230] F. Rosenblatt, Principles of neurodynamics. perceptrons and the theory of brain mechanisms, Technical Report, Cornell Aeronautical Lab Inc Buffalo NY, 1961.

- [231] A. Elisseeff, H. Paugam-Moisy, Jnn, a randomized algorithm for training multilayer networks in polynomial time, *Neurocomputing* 29 (1999) 3–24.
- [232] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: theory and applications, *Neurocomputing* 70 (2006) 489–501.
- [233] L. Wang, Support vector machines: theory and applications, volume 177, Springer Science & Business Media, 2005.
- [234] C. Cortes, V. Vapnik, Support-vector networks, *Machine Learning* 20 (1995) 273–297.
- [235] M. Tanveer, C. Gautam, P. N. Suganthan, Comprehensive evaluation of twin svm based classifiers on UCI datasets, *Applied Soft Computing* 83 (2019) 105617.
- [236] M. V. Narkhede, P. P. Bartakke, M. S. Sutaone, A review on weight initialization strategies for neural networks, *Artificial Intelligence Review* 55 (2022) 291–322.
- [237] S. S. Sodhi, P. Chandra, S. Tanwar, A new weight initialization method for sigmoidal feedforward artificial neural networks, in: 2014 International Joint Conference on Neural Networks (IJCNN), IEEE, 2014, pp. 291–298.
- [238] H. Yang, X. Ding, R. Chan, H. Hu, Y. Peng, T. Zeng, A new initialization method based on normed statistical spaces in deep networks, *Inverse Problems & Imaging* 15 (2021) 147.
- [239] S. Li, Z. Zhao, T. Liu, R. Hu, X. Du, Initializing convolutional filters with semantic features for text classification, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 1884–1889.
- [240] D. Mishkin, J. Matas, All you need is a good init, *arXiv preprint arXiv:1511.06422* (2015).
- [241] D. Aguirre, O. Fuentes, Improving weight initialization of relu and output layers, in: International Conference on Artificial Neural Networks, Springer, 2019, pp. 170–184.
- [242] P. Steiner, A. Jalalvand, P. Birkholz, Cluster-based input weight initialization for echo state networks, *IEEE Transactions on Neural Networks and Learning Systems* (2022).
- [243] S. Koturwar, S. Merchant, Weight initialization of deep neural networks (dnns) using data statistics, *arXiv preprint arXiv:1710.10570* (2017).
- [244] A. Rahimi, B. Recht, Random features for large-scale kernel machines, *Advances in Neural Information Processing Systems* 20 (2007).

- [245] S. Mehrkanoon, J. A. Suykens, Deep hybrid neural-kernel networks using random fourier features, *Neurocomputing* 298 (2018) 46–54.
- [246] P. V. de Campos Souza, Fuzzy neural networks and neuro-fuzzy networks: A review the main techniques and applications used in the literature, *Applied soft computing* 92 (2020) 106275.
- [247] R. Katuwal, P. N. Suganthan, L. Zhang, An ensemble of decision trees with random vector functional link networks for multi-class classification, *Applied Soft Computing* 70 (2018) 1146–1153.
- [248] Z.-H. Zhou, J. Feng, Deep forest: Towards an alternative to deep neural networks., in: *IJCAI*, 2017, pp. 3553–3559.
- [249] Z.-H. Zhou, J. Feng, Deep forest, *National Science Review* 6 (2019) 74–86.
- [250] V. López, A. Fernández, J. G. Moreno-Torres, F. Herrera, Analysis of pre-processing vs. cost-sensitive learning for imbalanced classification. open problems on intrinsic data characteristics, *Expert Systems with Applications* 39 (2012) 6585–6608.
- [251] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.
- [252] D. Liang, G. Ruobin, P. N. Suganthan, D. Z. Wang, Bayesian optimization based dynamic ensemble for time series forecasting, *Information Sciences* (2022). doi:<https://doi.org/10.1016/j.ins.2022.01.010>.
- [253] S. Das, P. N. Suganthan, Differential evolution: A survey of the state-of-the-art, *IEEE Transactions on Evolutionary Computation* 15 (2010) 4–31.
- [254] G. Chandrashekar, F. Sahin, A survey on feature selection methods, *Computers & Electrical Engineering* 40 (2014) 16–28.