




Research Article

A Novel Reformed Reduced Kernel Extreme Learning Machine with RELIEF-F for Classification

Zongying Liu ¹, Jiangling Hao ¹, Dongrui Yang,² Ghalib Ahmed Tahir,² and Mingyang Pan ¹

¹Dalian Maritime University, Faculty of Navigation, No. 1 Linghai Road, Dalian 116085, China

²University of Malaya, Kuala Lumpur 50603, Malaysia

Correspondence should be addressed to Jiangling Hao; haojl@dlmu.edu.cn

Received 25 August 2021; Revised 15 February 2022; Accepted 28 February 2022; Published 24 March 2022

Academic Editor: Simone Ranaldi

Copyright © 2022 Zongying Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the exponential growth of the Internet population, scientists and researchers face the large-scale data for processing. However, the traditional algorithms, due to their complex computation, are not suitable for the large-scale data, although they play a vital role in dealing with large-scale data for classification and regression. One of these variants, which is called Reduced Kernel Extreme Learning Machine (Reduced-KELM), is widely used in the classification task and attracts attention from researchers due to its superior performance. However, it still has limitations, such as instability of prediction because of the random selection and the redundant training samples and features because of large-scaled input data. This study proposes a novel model called Reformed Reduced Kernel Extreme Learning Machine with RELIEF-F (R-RKELM) for human activity recognition. RELIEF-F is applied to discard the attributes of samples with the negative values in the weights. A new sample selection approach, which is used to further reduce training samples and to replace the random selection part of Reduced-KELM, solves the unstable classification problem in the conventional Reduced-KELM and computation complexity problem. According to experimental results and statistical analysis, our proposed model obtains the best classification performances for human activity data sets than those of the baseline model, with an accuracy of 92.87 % for HAPT, 92.81 % for HARUS, and 86.92 % for Smartphone, respectively.

1. Introduction

In recent decades, the rapid advancement in technology increased computation capacity that welcomed the second spring of Artificial Intelligence (AI). As the backbone of AI, Machine Learning (ML) touches our daily life, and even we do not notice. For example, some functions are applied in the wearable devices, including sporting detection, fall detection, and activity detection. These applications based on the classification algorithms are implemented successfully in our real-world life. Many classical neural networks, such as Artificial Neural Network, Support Vector Machine, and Back-propagation algorithm, performed well for the classification tasks [1–3]. However, the main limitation of these algorithms is the heavy computation, especially for the large-scale data. In the support vector machine, the kernel method, which connects the input layer of the model with the hidden layer, increases the computational complexity. At the same

time, the main reason of backpropagation and artificial neural network with heavy computation is to compute suitable input weights and output weights for the neural network.

To solve the problem of complex computation, in 2004, Huang et al. proposed a single-layer feed-forward network called Extreme Learning Machine [4]. Due to the random selection of input weights between the input and hidden layer, it was faster thousands of times and achieved better performance in classification than that of the traditional algorithms [5]. After that, Extreme Learning Machine with Kernel (KELM) was proposed [6], which used Gaussian function to connect input layer and hidden layer and then found a least-squares solution. It obtained better performance in classification than that of conventional extreme learning machine [7]. However, the computation of the kernel method is heavy, especially for the large-scale data. In 2016, Deng et al. proposed a fast kernel algorithm called

Reduced Kernel Extreme Learning Machine (Reduced-KELM) [8]. It randomly selects a certain percentage of training samples. Although this strategy reduced the computation complexity and solved the limitation of KELM, the random selection method became an unstable element that leads to the unstable forecasting performance.

To overcome the limitations mentioned above, there are two main aims in this study. The first one is to filter out redundant features based on feature selection methods, because the large-scaled data usually appears in the training process. In aspect of feature selection, RELIEF-F is the one of the efficient algorithms that is used to select features in the different models. Paper [9] applied RELIEF-F to select training features for the classifier on the facial expression recognition. Yahdin et al. employed RELIEF-F for the feature selection in the prediction of the relevance education background [10]. In 2021, Cui et al. applied machine learning methods with feature selection method, RELIEF-F, to classify the wood materials [11]. These classification algorithms with RELIEF-F have better performance than those without RELIEF-F. Besides, paper [12] concluded that RELIEF-F had much better performance in feature selection than other feature selection methods. Therefore, RELIEF-F plays a significant role in the feature selection and assisting in enhancing performance of classification.

The other one is to overcome the limitation of random element in the model Reduced-KELM and enhance the performance in the classification. The aim of randomly selected sample in Reduced-KELM is to select samples that represent the all features from the training data. However, random selection approach cannot select all samples with the different features and probably miss important features. This situation leads to the decrease forecasting performance and unstable prediction performance. To solve random influence in the process of selecting training features, clustering method is applied to select suitable samples in the training phase or reduce the complex computation of training part. For example, Wu et al. combined K-means clustering method with KELM. It successfully reduced the complexity of computation in the training process [13]. Huang et al. proposed a clustering method with extreme learning machine for classification, which increased the ability of classification [14]. Moreover, samples selection method also impacts on the model performance. Liu et al. applied samples selection method based on correlation analysis, and Fisher is proposed, which could remove the redundant features that had close correlations with each other, to extreme learning machine [15]. It showed the role of samples selection method in the speech emotion recognition model, which increased the speed of discriminating emotional states of different speakers from speech. These proved that the good samples selection method played a vital role in increase efficient and speed of training model.

Inspired by these summary and conclusions, this study applies RELIEF-F to select reliable features. It discards those insignificant features from the data set, which reduces the computation complexity in the training process. Moreover, a novel sample selection method called Reformed Sample Selection Method (RSSM) is proposed. It takes the

advantages of K-means and Correlation Detection Selection (CDS) method and takes new strategy to seek more important samples from the training data by modification of K-means and CDS. This study applies RSSM to successfully replace the random selection part in the conventional Reduced-KELM. The proposed model is called Reformed Reduced Kernel Extreme Learning Machine. It not only solves the limitation of random selection in Reduced-KELM, but also improves the performance in classification. Therefore, the main contributions of this study are summarized as follows:

- (i) RELIEF-F algorithm is applied to select relevant features for the training phase. It directly reduces the computation complexity and has less training time than the baseline model Reduced-KELM.
- (ii) We proposed a novel reformed reduced kernel extreme learning machine. It uses an efficient sample selection method to replace the random part of Reduced-KELM and obtain better performance in classification than that of the compared models.
- (iii) The proposed model performed better than the baseline models on both benchmark data and real-world data. Especially for human activity recognition, our proposed model has superior ability in human activity recognition task.

This paper is organized as follows. Section 2 reviews Extreme Learning machine, Kernel Extreme Learning Machine, and the works with RELIEF-F. Section 3 represents Reduced Kernel Extreme Learning Machine, RELIEF-F for the features reduction, three types of Sample Selection Method (K-means, Correlation Detection Selection, and Reformed Sample Selection Method), and our proposed model. Section 4 reports on data description, experimental design, the experimental results, and discussion on these results. Section 5 represents the comparison by the statistical method. Finally, the conclusion is represented in Section 6.

2. Related Works

Since the rapid development of machine learning algorithms, the artificial intelligence technologies were applied in various domains and achieved good performance, such as face recognition [16, 17], time series prediction [18, 19], and classification [20, 21]. These algorithms involve some traditional and classical neural networks. Taking Back-propagation Neural Network (BPNN) [22] and Support Vector Machine (SVM) [23] as examples, they showed the superability in classification and regression [24–27]. With the appearance of the ‘big data era,’ huge-scale data is collected. However, due to the characteristics of the traditional and classical neural networks, these algorithms cannot afford the heavy computation with large-scale data. The computation cost is a barrier to the implementation of these algorithms in the real world.

In the recent decade, random projection algorithms attracted lots of attention of researchers. Due to the random selection of weights, these types of algorithms solved the

heavy computation problem. Extreme Learning Machine (ELM), which was proposed by Huang et al. [4], was one of the random projection algorithms. Paper [28] indicated that ELM was thousands of times faster in the training time and achieved better performance than the traditional neural networks, such as BPNN and SVM in classification and regression. In recent years, this algorithm and its variant algorithms were widely used in many domains, such as the stock market prediction [29], image classification [30], flight control [31], and speech emotion recognition [15].

Because ELM is a modified Single Layer Feed-forward Network (SLFN), before discussing ELM, SLFN should be introduced. The structure of SLFN can be shown in Figure 1, which includes three layers, regarding the input, hidden, and output layers.

We assume that there are N arbitrary samples (X, T) , where the input samples represent $X = [x_1, x_2, \dots, x_L, \dots, x_N] \in \mathfrak{R}^{N \times L}$, and its corresponding target values are $T = [t_1, t_2, \dots, t_L, \dots, t_N] \in \mathfrak{R}^{N \times D}$. L stands for the number of training samples, and D is the number of output nodes. The hidden neurons can be shown as hidden matrix (H) that is calculated by the activation function ($g(\cdot)$). The input weights (a) connect between the input layer and hidden layer. Output weights (β) connect the hidden layer with the output layer. Then, the output (\hat{T}) of a feed-forward neural network with S hidden neurons can be expressed as follows:

$$\beta G(x) = \sum_{i=1}^S \beta_i g(a_i x + b_i) = \hat{T}, \quad (1)$$

where S is the number of hidden neurons, β represents the output weights with dimension of $(S \times D)$, a is the input weights with dimension of $(S \times L)$, and b is a bias matrix with dimension of $(L \times S)$. If there is no error between the activation function $g(x)$ with S hidden neurons in the single-layer feed-forward network and actual target values, the mathematical equation can be shown as follows:

$$\sum_{i=1}^L \|T_i - \hat{T}_i\| = 0. \quad (2)$$

It can be extended as

$$\sum_{i=1}^S \sum_{j=1}^L \beta_j g(a_j x_i + b_j) = T_j. \quad (3)$$

Traditionally, the main aim of training SLFN is the minimization of the cost function for finding the corresponding weights and bias. In this processing, the BP learning algorithm is used from the output to the input. The cost function is shown in

$$E = \sum_{i=1}^L \left(\sum_{j=1}^S \beta_j g(a_j x_i + b_j) - T_i \right)^2. \quad (4)$$

Unlike SLFN, ELM applies the gradient-based algorithms and proposes an efficient learning algorithm for feed-forward neural networks in order to solve the drawbacks of BP learning algorithm. Based on the theory of ELM, unlike the traditional activation function that requires adjusting the

input weights and biases, the input weights and biases of hidden layer can be selected randomly. Then, the training process of ELM is to find a least-squares solution $\hat{\beta}$ of (3), which is shown in the following equation:

$$\|H\hat{\beta} - T\| = \min_{\hat{\beta}} \|H\hat{\beta} - T\|, \quad (5)$$

where H is the hidden matrix based on the activation function. It is a non-squared matrix that can be calculated by (6). The input weights (a) and hidden biases b were selected randomly.

$$H = \begin{bmatrix} h(x_1) \\ \vdots \\ h(x_L) \end{bmatrix} = \begin{bmatrix} g(a_1 x_1 + b_1) & \cdots & g(a_S x_1 + b_S) \\ \vdots & \vdots & \vdots \\ g(a_1 x_L + b_1) & \cdots & g(a_S x_L + b_S) \end{bmatrix}. \quad (6)$$

Finally, Huang [4] proposed that the smallest norm least-squares solution is

$$\beta = H^\dagger T, \quad (7)$$

where H^\dagger represents the Moore-Penrose generalized inverse of matrix H . Its mathematical equation can be shown as $H^\dagger = (H^T H)^{-1} H^T$, where the superscript (τ) of H stands for the transpose operator. Therefore, the training process can be shown in Algorithm 1.

At the same time, with the advent of the era of big data, the large-scaled data is widely used for training model. However, it also brings the huge computation and decreases the training efficiency. Although the training speed of ELM is faster than that of the conventional algorithms, it also faces this situation. Furthermore, the dimension of training samples impacts the complexity of computation. An efficient filter-method called RELIEF, which was proposed by Kiral [32], showed attributes based on how well their values distinguish among samples that are near each other. After that, Kononenko et al. updated the RELIEF algorithm [33] and proposed the RELIEF-F algorithm. It used the Manhattan (L_1) norm to compute the distance between the near-hit and near-miss instances. It reported that RELIEF-F algorithm is an efficient method that takes absolute differences rather than the square of those differences. Besides, to reduce complex computation and increase training efficiency, researchers pay more attention to deal with input features before going through training phase in ELM. For example, Tian et al. applied RELIEF-F as feature selection method in ELM for the gait recognition [34]. In Paper [35], RELIEF-F algorithms is used to propose a feature selection technique for the purpose to eliminate redundancy. It reported that this structure of model with feature selection technique showed significant improvements than other existing forecasting models in terms of forecast accuracy and convergence rate. Many studies [36–39] concluded that RELIEF-F, as a feature selection technique, is an efficient and common approach for eliminating redundant features.

However, due to the random selection of input weights in ELM, the forecasting results are not the same under the same parameters setting of ELM, which causes the unstable forecasting performance, while the number of hidden neurons is also required to define by user. To solve unstable

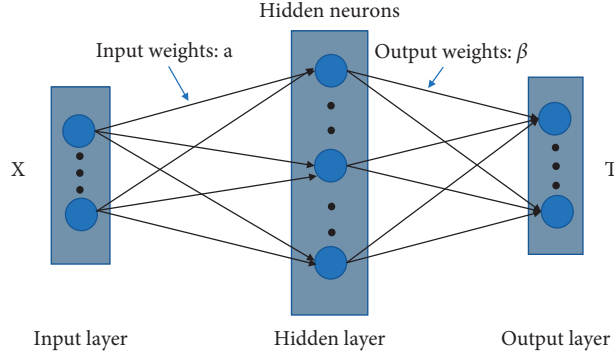


FIGURE 1: The structure of SLFN.

Require: Input data matrix X , the corresponding target values T with D output nodes, the number of hidden neurons S , and activation function, $g(\cdot)$.

Ensure: the output weights β .

- (1) Random select input weights (a) and biases (b);
- (2) Calculate hidden matrix H by (6);
- (3) Calculate output weights β by (7).

ALGORITHM 1: The training process of ELM.

forecasting performance problem, Kernel Extreme Learning Machine (KELM) was proposed by Huang in 2011 [5]. It applied the kernel method to connect input layer and hidden layer, which avoided the unstable forecasting performance from ELM caused by the random selection of the input weights.

In KELM, the hidden matrix (K) is calculated by Gaussian function $k(\cdot)$, which is represented as

$$K = k(X_L, X_L), \quad (8)$$

where the training samples represent $X_L = \{x_1, x_2, \dots, x_L\}$. The output weights (β) of KELM can be computed by

$$\beta = \left(K + \frac{I}{C} \right)^{-1} T_L, \quad (9)$$

where I is an identity matrix, C represents the regularization parameter that generally is defined as 1, and $T_L = \{T_1$ is the corresponding training target values ($T_L = \{T_1, T_2, \dots, T_L\}$). The forecasting values (\hat{T}) can be calculated by

$$\hat{T} = K^T \beta. \quad (10)$$

The papers indicating kernel functions played a vital role in KELM compared with conventional ELM in regression and classification [6, 7, 40]. However, the kernel method with large-scale data generates the huge dimensional kernel matrix, which directly leads to the heavy time consumption in the training process of KELM.

To overcome the limitation of KELM, Deng et al. proposed an efficient and fast model called Reduced Kernel Extreme Learning Machine (Reduced-KELM) [8]. It applied random method to select part of training sample to calculate the hidden kernel matrix, which to some extent reduces the

computation. However, due to random selection for training samples, its forecasting performance is not stable. Based on above revision, Table 1 briefly summarizes the advantages and drawbacks of ELM, KELM, and Reduced-KELM.

This study is inspired by the idea of RELIEF-F. Firstly, it applies RELIEF-F to discard useless features of training data. Secondly, to solve the limitation of Reduced-KELM, we propose a novel sample selection method to replace random selection method of Reduced-KELM. Finally, we propose a model named Reformed Reduced Kernel Extreme Learning Machine with RELIEF-F. The following section describes details of the proposed techniques.

3. Methodology

This section explains a novel framework for reducing training computation and improving performance during classification. Firstly, RELIEF-F algorithm is proposed for feature selection, which discards the irrelevant features and reduces the training time of the classifier. Secondly, two sample selection methods, including K-mean and correlation detection selection, are introduced. Then, a novel sample selection method named Reformed Sample Selection Method is proposed, which is combined with K-means and Correlation Detection Selection method. Finally, this novel sample selection method successfully replaced the random part of Reduced-KELM, which generates a model called Reformed Reduced Kernel Extreme Learning Machine with RELIEF-F.

3.1. Reduced Kernel Extreme Learning Machine. Before describing our proposed methods, the baseline model Reduced-KELM needs to be introduced. The conventional

TABLE 1: The advantages and drawbacks of ELM and variant ELM models.

Model	Advantages	Drawbacks
ELM	Fast training speed; same or better performance than the conventional models	Unstable forecasting performance; parameter dependency (2 parameters)
KELM	Stable performance	Heavy computation; parameter dependency (1 parameter)
Reduced-KELM	Less computation than KELM	Unstable forecasting performance; parameter dependency (1 parameter)

KELM applies all training samples to generate the hidden matrix by Gaussian activation function. The main idea of model Reduced-KELM is to reduce complexity in the computation of kernel matrix by randomly selecting a certain percentage of training samples from all training samples to compute the hidden kernel matrix. It is less time-consuming as it uses only 10 percent of nodes. Paper [8] concluded that Reduced-KELM, randomly selecting ten percentage of nodes, rapidly decreased the training time and achieved almost the same performance as KELM. In the following experiments, we apply ten percent as the random selection percentage in Reduced-KELM.

It is assumed that $\tilde{X} = \{x_i\}_{i=1}^m$ is certain percentage of training samples that are randomly selected, where \hat{m} is the total number of selected samples. Then, the hidden matrix of Reduced-KELM is computed by using the following equation:

$$\hat{K} = k(\tilde{X}, X_L) = \begin{bmatrix} k(x_1, x_1) & \cdots & k(x_{\hat{m}}, x_1) \\ \vdots & \ddots & \vdots \\ k(x_1, x_L) & \cdots & k(x_{\hat{m}}, x_L) \end{bmatrix}. \quad (11)$$

The dimension of the hidden matrix (\hat{K}) in Reduced-KELM is reduced from $(L \times L)$ to $(L \times \hat{m})$, which directly decreases the computation of the training process. It computes output weights by

$$\beta = \left(\frac{I}{C} + \hat{K}^T \hat{K} \right)^{-1} \hat{K}^T T_L. \quad (12)$$

The training process of Reduced-KELM is summarized in Algorithm 2.

Reduced-KELM has less training time than the conventional KELM due to the random selection of support vectors for computing kernel matrices. However, the classification result of Reduced-KELM is unstable. To overcome this limitation and to further reduce the training time, this study proposes a RELIEF-F algorithm for selecting features of observations. It represents a novel sample selection method to replace the random selection process of support vectors for enhancing classification performance. The following subsections will introduce the details.

3.2. RELIEF-F Algorithm for Features Reduction. In this study, inspired by the characteristic of the RELIEF-F algorithm and successful application on regression and classification models, it is applied to select features from data sets. The following is the process of feature selection by the RELIEF-F algorithm. Firstly, a feature is selected randomly as R . Then, its P has searched nearest neighbors from the

same class that are named as the nearest hits (B). At the same time, it also searches P nearest neighbors from other different classes as M . It updates the quality estimation W_D for all features based on the selected features R and M by (13). The updated formula is similar to that of RELIEF. Our proposed algorithm weighs the contribution from each class of the misses with the prior probability of that class (P). The contributions of hits and misses in each step will be in the range between zero and one. The values of W determine the ranking of the importance of features. It discards all features with values that are less than zero. The rest of the features continue to process in the training part of the model.

$$W_D = \frac{W_D - \text{diff}\{A, R, B\}}{m} + \frac{\sum_{C \neq \text{class}(R)} [P(C) \text{diff}\{A, R, M\}]}{m}, \quad (13)$$

where the initial weight W_D is set as zero, diff is a function for calculating the absolute difference, and $P(C)$ stands for the probability of this attribute appears in class C . This algorithm seeks M for each different class and averages their contribution for updating estimates W_D , which estimates the ability of features for the target values. In order to reduce useless features from the data set, this study applies the RELIEF-F algorithm to calculate the weight values of each attribute in the data set and then discards all features with negative weight values for reducing the dimension of feature vectors. Besides that, RELIEF-F originally needs to search P nearest neighbors from the same class. The number P requires to be defined by the user. In this study, P is defined as 10 based on the reference paper [33]. In our proposed model, RELIEF-F is used before starting training model. It is an efficient algorithm for reducing dimension of features and complex computation in the further process.

RELIEF-F algorithm using filter-method approach calculates a score (weight) for each feature to identify which features are most relevant to the set of instances. A weight is linked to each attribute, where the most relevant attribute has the highest weight. If a feature value difference is observed in a neighboring instance pair with the same class, the weight decreases. Alternatively, if a feature value difference is observed in a neighboring instance pair with different class values, the weight increases. Compared to positive weight features, negative weight features will have more chance in the same or closed class [41]. Moreover, Kira and Rendell demonstrated that, statistically, the relevance level of a relevant feature was expected to be larger than zero and that of an irrelevant one was expected to be zero or negative [32]. Therefore, generally, the threshold of RELIEF-F (τ) should be defined such as $\tau > 0$.

Require: Training input data matrix X_L , the corresponding target values T_L with D output nodes, and kernel function, $k(\cdot)$.
Ensure: the output weights β .

- (1) Random select \hat{m} samples from all training observation as support vectors \hat{X} ;
- (2) Calculate reduced hidden matrix \hat{K} by (11);
- (3) Calculate output weights by (12).

ALGORITHM 2: The training process of Reduced-KELM.

3.3. Sample Selection Methods. To overcome the random selection limitation of Reduced-KELM and to enhance classification performance, this study proposes a novel sample selection method to replace the random selection part of model Reduced-KELM. This new method applies modified K-means and correlation detection selection methods to select the efficient samples. Before introducing the proposed approach, we described the two new sample selection methods.

3.3.1. K-Means. K-means [42] is a classical clustering method, which solves the optimal clustering center and optimal classification by learning. It has a high learning efficiency and can process large-scale data [43]. In this paper, the K-means algorithm clusters the data for achieving stable prediction and higher accuracy than the conventional Reduced-KELM.

K-means is an unsupervised learning clustering algorithm and one of the most popular clustering algorithms at present. It applies the Euclidean distance metric as the standard similarity analysis method and divides the whole into a certain number of classes with high similarity. It assists in decreasing the number of samples and applying the cluster centroid position to stand for the original samples. The main goal of the K-means algorithm is to minimize the sum of the squared errors on all Z clusters. Its mathematical equation is as follows:

$$J(z) = \sum_{x \in Z} \|x_i - \mu_z\|^2 \quad (14)$$

where μ_z represents the average value of all data that belonged to cluster Z ($Z = \{1, 2, \dots, Z\}$).

It is assumed that the data set contains N sample data, and the number of clusters is set as Z . Firstly, Z observations are selected from the whole data and set as the cluster center of the initial partition. Secondly, according to the similarity measurement method, it computes the distances between the undivided sample data and each cluster center point. After that, it divides the observations that are closer to the cluster center into the corresponding cluster. Then, it calculates the sum of square error between the center position and the corresponding observations for all classes. With moving the cluster center, observations belonged to each class are redivided until there is no change in the sum of squared errors of class. Finally, K-means will return the center position of each cluster. Algorithm 3 summarizes the process of the K-means sample selection method. The returned values can be used to replace the random part of Reduced-KELM for achieving stable forecasting performance.

3.3.2. Correlation Detection Selection Method. There are many samples for the different classes in the classification. Generally, these samples are not all useful for the training model. As compared to the K-means clustering for sample selection, this study proposes an efficient technique named as Correlation Detection Selection method (CDS). It mainly finds the correlation among samples and discards the samples with high correlation values. Discarding the samples with similar information not only plays a positive role in training the model, but also replaces unstable random parts in Reduced-KELM. It increases the classification performance.

The main idea of CDS is to select memory without the high correlation values from all training data observations. Firstly, we initialize the threshold of CDS as δ , and the initial memory (mem) is defined as the first observation in the training data ($\text{mem} = X_1$). Secondly, this method calculates the average value of correlation between the coming sample (X_i) and filtered memory (mem). The coming sample will add into the filtered memory when the average correlation value is smaller than the threshold of CDS (δ). In contrast, it will exclude the coming data from the filtered memory. Algorithm 4 shows the pseudocode of CDS.

3.3.3. Reformed Sample Selection Method. In this section, a new sample selection method named Reformed Sample Selection Method (RSSM), which applies the advantage of K-means and CDS to seek the more suitable samples for calculating kernel matrix, is proposed.

In RSSM, randomly set Cent as the initial centroids from the input matrix X_L and find out the samples that are nearest to each centroid based on Euclidean distance. Based on these samples, we can recalculate the position of centroids of Cent. Then, J can be computed. Cent can be computed until the value of J is not changed based on (14). In the next step, we initialize memory as $\text{mem} = \text{Cent}_1$. Start from the second Cent, and calculate the average value of correlation (AC) between coming sample from Cent and mem. Based on the condition, mem can be updated. Finally, the matrix of mem can be returned. Algorithm 5 shows the detail about the pseudocode of Reformed Sample Selection Method.

3.4. Proposed Model: Reformed Reduced Kernel Extreme Learning Machine with RELIEF-F. For fair comparison and to further decrease the computation, the data is processed by RELIEF-F algorithm firstly. It is a first step to deal with input features. The output of RELIEF-F can be set as a new input data for the further steps. Based on the sample selection

Require: Training input data matrix X_L , and the number of clusters Z .
Ensure: the centroid positions $Cent$.

- (1) Initialize Z centroids randomly;
- (2) Associate each samples with the nearest centroid by Euclidean distance;
- (3) Recalculate the position of centroids $Cent$;
- (4) Repeat step 2 and 3 until result of (14) is no change; **return** $Cent$

ALGORITHM 3: The process of K-means.

Require: Training input data matrix X_L , and the threshold of CDS δ .
Ensure: the filtered memory mem .

Initial Part:

- (1) Sort training samples by class;
- (2) Set the threshold of CDS as δ ;
- (3) Set the initial filter memory $mem = X_1$; **The Selecting Part:**
- (4) **for** $i \in \{2, \dots, L\}$ **do**
- (5) Calculate the average value of correlation (AC) between X_i and mem ;
- (6) **if** $AC < \delta$ **then**
- (7) $mem = [mem; x_i]$;
- (8) **else**
- (9) $mem = mem$.
- (10) **end if**
- (11) **end for**
- (12) **return** mem

ALGORITHM 4: The process of Correlation Detection Selection method.

Require: Training input data matrix X_L , and the threshold of CDS δ .
Ensure: the filtered memory mem .

- (1) random set $Cent$ as the initial centroids;
- (2) Associate each samples with the nearest centroid by Euclidean distance;
- (3) Recalculate the position of centroids $Cent$;
- (4) Repeat step 2 and 3 and calculate J based on (14);
- (5) Get $Cent$ with labels until value of J is no change;
- (6) Sort $Cent$ by class;
- (7) Set the threshold of CDS as δ ;
- (8) Set the initial filter memory $mem = Cent_1$;
- (9) **for** $i \in \{2, \dots, L\}$ **do**
- (10) Calculate the average value of correlation (AC) between $Cent_i$ and mem ;
- (11) **if** $AC < \delta$ **then**
- (12) $mem = [mem; Cent_i]$;
- (13) **else**
- (14) $mem = mem$.
- (15) **end if**
- (16) **end for**
- (17) **return** mem

ALGORITHM 5: The process of Reformed Sample Selection Method.

approaches, this study proposes a Reformed Reduced Kernel Extreme Learning Machine with RELIEF-F (R-RKELM), which employs the output of RSSM to replace the random selection part of Reduced-KELM. To prove the classification ability of the proposed model R-RKELM, it is compared with two other models, including Reduced Kernel Extreme Learning Machine with K-means and RELIEF-F (K-

RKELM), and Reduced Kernel Extreme Learning Machine with Correlation Detection Selection and RELIEF-F (C-RKELM).

Firstly, based on RELIEF-F algorithm, the original features are processed. Then, the first model K-RKELM has applied the centroid positions of each cluster by K-means and all training samples to calculate the reduced kernel

matrix. Model C-RKELM employs selected memory to replace the randomly selected data samples for developing conventional model Reduced-KELM.

The centroid positions of each cluster by K-means and the selected memory by CDS cooperated with the training samples to calculate the reduced kernel matrix, which replaces the kernel matrix calculated by random data samples in the conventional Reduced-KELM.

The proposed model applies the output of RSSM to replace the randomly selected samples in the conventional Reduced-KELM. RSSM is an unsupervised method. It mainly concentrates the training samples by K-means and obtains the centroid positions. And then, it applies CDS to discard the elements of the centroid positions with the high correlation value. The final remaining output replaces the random samples of conventional Reduced-KELM.

The pseudocode of R-RKELM is shown in Algorithm 6.

4. Experimental Works

To enhance the ability of classification and overcome the limitation of Reduced-KELM, this section designs two experiments. They employed the eight data sets, including benchmarks and real-world human activity data, to evaluate the classification ability for the RELIEF-F algorithm and Reduced-KELM with the different sample selection methods, respectively. This section mainly introduces data description, experimental design, and parameter setting. Lastly, based on the experimental design, the experimental results and discussion will be introduced.

4.1. Data Description. In the experimental section, the five benchmarks data sets and three human activity data sets are used for evaluating the classification ability.

A set of commonly used benchmarks includes German, Image, Ringnorm, Twonorm, and Waveform, available at UCI Machine Learning Repository [44]. These data sets contain binary class classification tasks.

Furthermore, with the data explosion and popularity of portable devices, researchers and developers pay more attention to human activity recognition, such as fall detection and sports detection in portable devices. Then, this study employs three real-world data sets to evaluate, including the Human Activities and Postural Transitions Recognition using Smartphone Data (HAPT) [45], Human Activity Recognition Using Smartphones Data Set (HARUS) [46], and Smartphone Data set for Human Activity Recognition in Ambient Assisted Living Data Set (Smartphone).

Besides, we separated the percentage of training and testing data in all benchmarks as 70% and 30%, respectively. The training and testing data of all real-world data sets are divided by their data source. We used the same division in our experiments. These data sets involve multiclass classification tasks. Table 2 shows the details of each data set.

4.2. Experiment Design and Parameter Setting. To evaluate the ability of our proposed methods and compared models fairly, this study designs two experiments. All experiments

are simulated on Matlab2014a in the laptop with Windows 10, 16 GB RAM environment.

The first experiment compares the classification performance of model Reduced-KELM with RELIEF-F algorithm with that of the conventional Reduced-KELM. It indicates the role the RELIEF-F algorithm plays in the features dimension reduction in Reduced-KELM. The performances of all benchmarks and human activity data in model Reduced-KELM are compared with those of Reduced-KELM with the RELIEF-F algorithm. The main aim of the RELIEF-F algorithm is to rank the features based on their importance in the classes and keep reliable attributes for the following training phase. Based on this algorithm, the feature selection process not only improves the classification performance, but also decreases the training time rather than conventional Reduced-KELM. To compare models fairly, the design of parameter setting needs to make sure that every model has the best performance under specific parameter setting. In the first experiment, the number of P nearest neighbors needs to be defined, which is critical to the performance of RELIEF-F algorithm. Based on the conclusion of paper [33], P is defined as ten in the first experiment. At the same time, we set the percentage of random selection as ten for all models in the first experiment, including the conventional Reduced-KELM and Reduced-KELM with RELIEF-F. Because the reference paper [8] concluded that Reduced-KELM randomly selected ten percentage of nodes that assisted on rapidly decreasing the training time, the performance of Reduced-KELM obtained was almost at the same level as that of KELM. Besides, due to the implementation of the kernel method, the kernel parameter impacts the performance in classification. For fair comparison among the models, the value of the kernel parameter is defined as one for all models in the first experiment.

On the other hand, the second experiment mainly observes the role the three different sample selection methods played in classification by model Reduced-KELM. These three methods successfully replace the random part of Reduced-KELM, respectively. This experiment shows the superior ability of selecting samples in the different sample selection methods and the ability of reducing the complexity computation of training model. It compares the performance of the proposed model R-RKELM with the conventional Reduced-KELM, K-RKELM, and C-RKELM. To reflect the connection between the first experiment and the second experiment, the second experiment applies the data sets that are processed by RELIEF-F algorithm. The parameters of RELIEF-F algorithm and kernel method in the second experiment are the same as the first one. To exhibit the performances of model under the different measurements, except for the common measurement accuracy and the corresponding Standard Deviation (SD) and Time, Sensitivity, Specificity, and Precision are employed to evaluate the performance in all experiments as well. At the same time, to observe the generalization ability, the fifty times will be run, and then calculate their average values of measurements when the model has a random selection method. A high standard deviation indicates that the accuracy values among fifty times are spread out over a wider range, and vice versa.

Require: Training input data matrix X_L ; the corresponding target values T_L with D output nodes; kernel function, $k(\cdot)$; the number of clusters, Z .

Ensure: the output weights β .

RELIEF-F Algorithm:

(1) Based on RELIEF-F, the input features are processed and the output matrix (Rel) can be obtained; **Reformed Sample Selection**

Method:

(2) Set Rel as input data;

(3) Return mem based on Algorithm 5;

(4) $\hat{X} = \text{mem}$; **Training Model:**

(5) Calculate reduced hidden matrix \hat{K} by (11);

(6) Calculate output weights β by (12).

ALGORITHM 6: The training process of R-RKELM.

TABLE 2: Details of Data sets.

Data	Training samples	Testing samples	Feature	Class
German	700	300	20	2
Image	1460	626	18	2
Ringnorm	5180	2220	20	2
Twonorm	5180	2220	20	2
Waveform	3500	1500	21	2
HAPT	7767	3162	561	6
HARUS	7352	2947	561	6
Smartphone	4252	1492	561	6

4.3. Experimental Results and Discussion. The first experiment demonstrates the differences between the conventional Reduced-KELM and Reduced-KELM with RELIEF-F algorithm (Relief-F). Relief-F algorithm is applied in the benchmarks and real-world data sets. The ranking of predictor weights is shown in Figure 2, which represents the level of importance of features.

Based on these bar charts in Figure 2, the values of the vertical axis represent the level of importance for features. Because of the conclusion from paper [32], compared to the positive weight features, the negative weight features have more chance in the same or closed class. These features probably are redundancy. And paper [41] showed that the features with positive weights have much better performance than that with negative weights. Therefore, this study discards the features with values that are below zero.

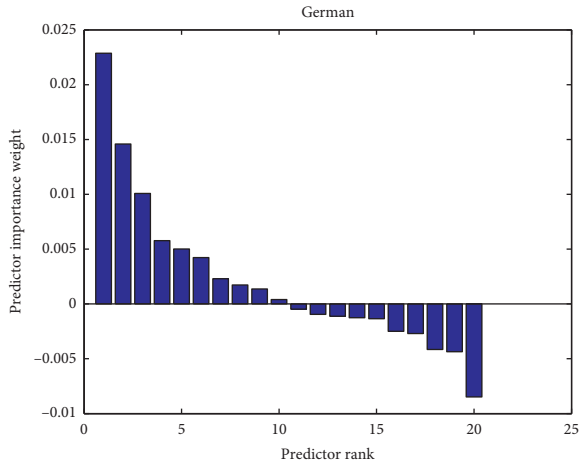
The dimension of data sets can be reduced by RELIEF-F algorithm. The final dimension of each data set is shown in Table 3. The column named ‘Difference’ represents the number of features that the RELIEF-F algorithm has reduced. For example, the German data filters half of the features from original data, and Ringnorm has only reduced one attribute by RELIEF-F.

Table 4 indicates the performances between Reduced-KELM and Reduced-KELM with RELIEF-F in the eight data sets. It represents the performance of accuracy, Difference (Accuracy of Reduced-KELM - Accuracy of Relief-F), Standard Deviation (SD), and training time (Time). Besides, the other three measurements are also shown in Table 4, including Sensitivity, Specificity, and Precision.

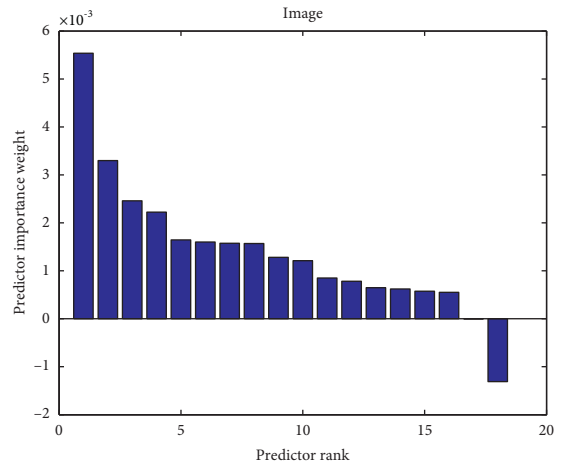
In terms of accuracy, only one data set (Twonorm) shows the best performance in model Reduced-KELM. The rest of the data sets obtain the super classification ability in the

Relief-F model rather than the conventional Reduced-KELM. On average, the growth rate of accuracy by relief-F in these data sets reaches 1.33 %. At the same time, the positive value of difference indicates that the super classification ability in model relief-F is better than the conventional model Reduced-KELM, and vice versa. The maximum difference of accuracy appears in German data. On the contrary, the image obtains the minimum difference. Although three data sets (including Twonorm, Waveform, and HAPT) have the same performance in SD for the conventional Reduced-KELM, Relief-F obtains the minimum value in standard deviation for the rest of data sets. In aspect of training efficiency, the main achievement of Relief-F is saving the training time. Especially for the data with high dimensions, such as HAPT, HARUS, and smartphones, the training time (minutes) is reduced by relief-F with 0.0371, 0.1102, and 0.031, respectively. In other measurements, the Relief-F algorithm has reduced Sensitivity for the majority of the data sets. This situation indicates that the Relief-F algorithm has a more stable prediction ability than the conventional Reduced-KELM. At the same time, the same performances appear in Specificity and Precision. Expect for Twonorm, HAPT, and HARUS, the Relief-F algorithm shows much better classification performance than the Reduced-KELM model. Therefore, the Relief-F method not only improves the accuracy of classification in benchmarks and real-world data sets, but also has saved the training time.

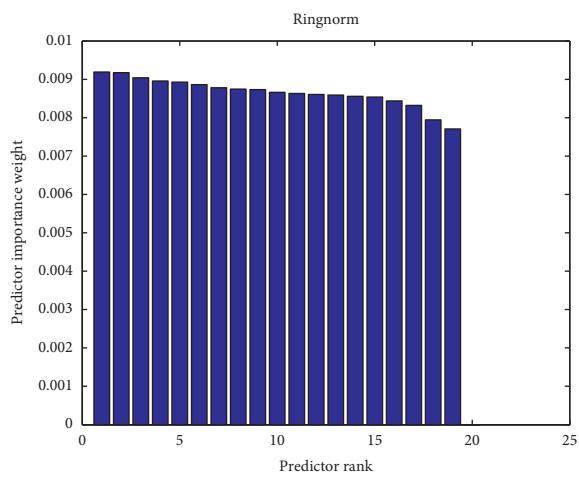
The second experiment compares the performance in classification of the proposed model R-RKELM with the model K-RKELM and C-RKELM. Table 5 collects the information about accuracy, SD, time, sensitivity, specificity, and precision for the model K-RKELM, C-RKELM, and R-RKELM.



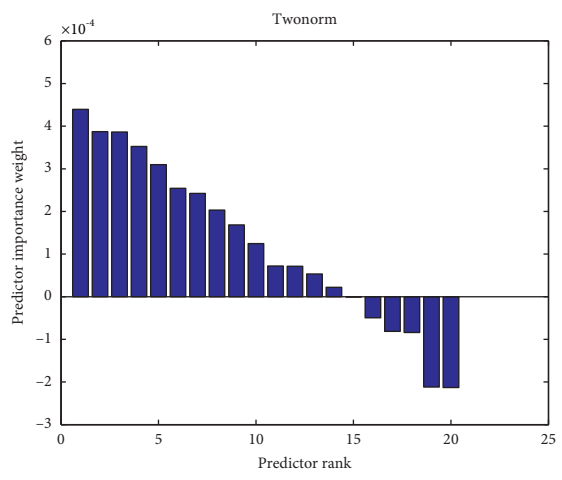
(a)



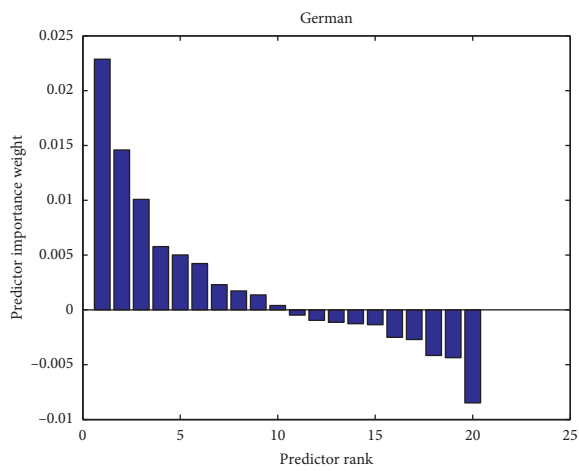
(b)



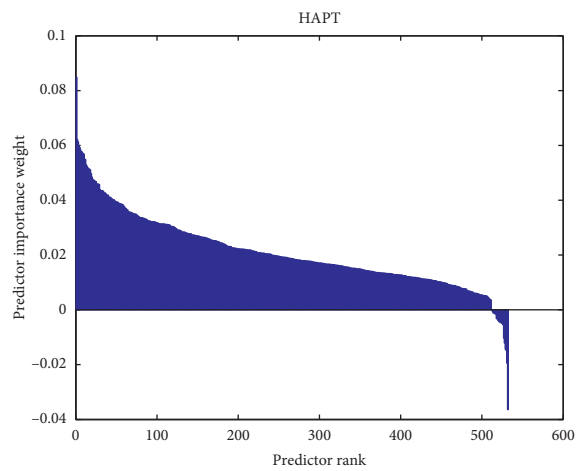
(c)



(d)



(e)



(f)

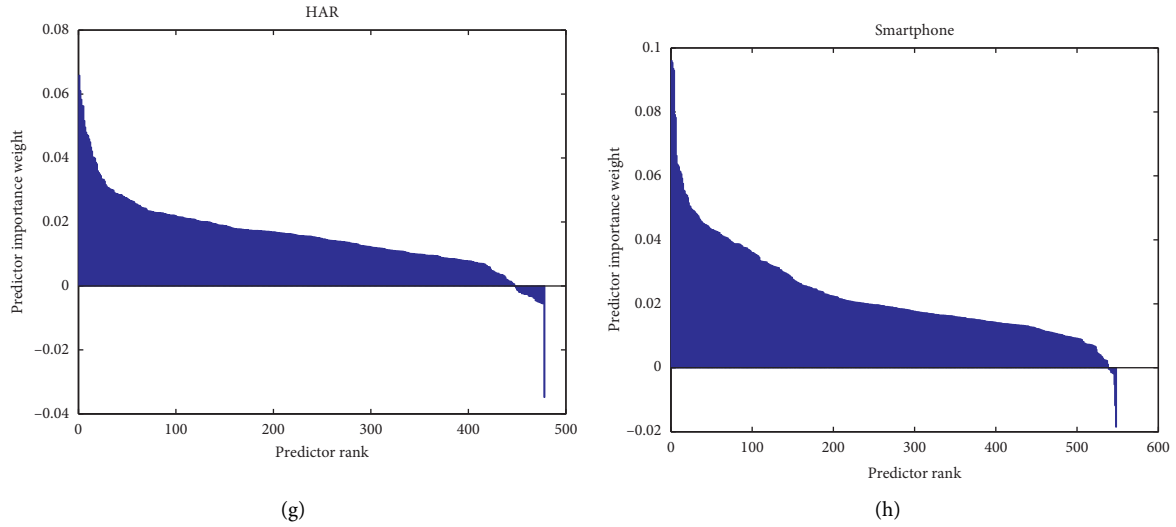


FIGURE 2: The distribution of predictor importance weights in features.

TABLE 3: The reduction of features dimension by RELIEF-F.

Data	Feature	Dimension after RELIEF-F	Difference
German	20	10	10
Image	18	16	2
Ringnorm	20	19	1
Twonorm	20	14	6
Waveform	21	10	11
HAPT	561	532	29
HARUS	561	478	83
Smartphone	561	548	13

TABLE 4: The performances of Reduced-KELM and Relief-F.

Data	Model	Accuracy (%)	Difference	SD	Time	Sensitivity	Specificity	Precision
German	Reduced-KELM	71.53	8.74%	0.06	0.0013	0.5790	0.4962	0.5175
	Relief-F	80.27						
Image	Reduced-KELM	85.70	0.16%	0.03	0.0086	0.8623	0.8510	0.8555
	Relief-F	85.86						
Ringnorm	Reduced-KELM	60.06	0.85%	0.02	0.1128	0.5961	0.6079	0.7786
	Relief-F	60.91						
Twonorm	Reduced-KELM	94.10	-1.65%	0.01	0.1022	0.9401	0.9377	0.9115
	Relief-F	92.45						
Waveform	Reduced-KELM	84.29	0.93%	0.01	0.0474	0.8324	0.8090	0.8148
	Relief-F	85.22						
HAPT	Reduced-KELM	88.52	0.95%	0.08	0.4155	0.9580	0.9434	0.8731
	Relief-F	89.47						
HARUS	Reduced-KELM	84.02	5.66%	0.07	0.3826	0.9470	0.9419	0.8643
	Relief-F	89.68						
Smartphone	Reduced-KELM	85.52	1.03%	0.07	0.1261	0.7749	0.7535	0.7280
	Relief-F	86.55						

In terms of accuracy, the proposed model, R-RKELM, successfully assists on enhancing classification performance in model Reduced-KELM. For the whole data sets, model R-RKELM obtains the best performance in accuracy than the other two models. Moreover, due to the random characteristic in the K-means algorithm, there is a minor difference in performance appearing in forecasting results when K-RKELM and

R-RKELM are run repeatedly. The standard deviation represents the degree of forecasting difference in all predictions. Except for HARUS and Smartphone data, model R-RKELM obtains the lowest value in Standard Deviation. In the aspect of training time, benchmarks data sets take longer training time in model C-RKELM than model R-RKELM. In HAPT and HARUS data, model R-RKELM takes the similar time in the training process

TABLE 5: The performance between relief-F and RKELM with two sample selection methods.

Data	Model	Accuracy (%)	SD	Time	Sensitivity	Specificity	Precision
German	K-RKELM	83.53	0.01	0.0011	0.5531	0.5881	0.6045
	C-RKELM	81.67	0.02	0.0419	0.9157	0.1538	0.8787
	R-RKELM	85.44	0.01	0.0056	0.6753	0.3940	0.7184
Image	K-RKELM	87.43	0.01	0.0063	0.8694	0.8736	0.8816
	C-RKELM	87.70	0.00	0.2610	0.9634	0.7819	0.8294
	R-RKELM	87.75	0.00	0.2536	0.9040	0.8424	0.8700
Ringnorm	K-RKELM	55.32	0.00	0.1265	0.5556	0.5541	0.7634
	C-RKELM	67.66	0.00	10.2167	1.0000	0.3555	0.6064
	R-RKELM	69.37	0.00	6.1088	0.7986	0.5910	0.7454
Twonorm	K-RKELM	94.52	0.00	0.1482	0.9382	0.9393	0.9195
	C-RKELM	95.23	0.00	9.1533	0.9385	0.9566	0.8735
	R-RKELM	95.35	0.00	5.7655	0.9479	0.9539	0.9096
Waveform	K-RKELM	85.51	0.01	0.0305	0.8454	0.8258	0.8298
	C-RKELM	85.47	0.00	3.7682	0.8757	0.8117	0.9045
	R-RKELM	85.84	0.00	1.6029	0.8594	0.8344	0.8597
HAPT	K-RKELM	89.50	0.09	0.3997	0.9709	0.8914	0.7367
	C-RKELM	90.58	0.06	18.8531	0.8932	0.8309	0.7481
	R-RKELM	92.87	0.06	24.1748	0.9760	0.9197	0.7869
HARUS	K-RKELM	89.79	0.08	0.3253	0.9618	0.8945	0.7552
	C-RKELM	88.83	0.00	20.1623	0.8468	0.8898	0.6087
	R-RKELM	92.81	0.05	20.1696	0.9649	0.9221	0.8029
Smartphone	K-RKELM	86.68	0.06	0.2197	0.8055	0.7888	0.7092
	C-RKELM	86.68	0.00	5.6465	0.7945	0.8057	0.7178
	R-RKELM	86.92	0.06	1.3800	0.8126	0.8090	0.7268

TABLE 6: The Wilcoxon signed-rank test between R-RKELM and Relief-F.

Data	Model		Difference	Signed	Rank
	R-RKELM (%)	Relief-F (%)			
German	85.44	80.27	5.17	+	7
Image	87.75	85.86	1.89	+	3
Ringnorm	69.37	60.91	8.46	+	8
Twonorm	95.35	92.45	2.90	+	4
Waveform	85.84	85.22	0.64	+	2
HAPT	92.87	89.47	3.40	+	6
HARUS	92.81	89.68	3.13	+	5
Smartphone	86.92	86.55	0.37	+	1

than model C-RKELM. Model R-RKELM in Smartphone data takes less than five times as much training time as model C-RKELM. In Sensitivity, model C-RKELM has the best performance than other models. Model R-RKELM shows the best performance in Specificity for the majority of data sets. Compared with performance in Precision, only real-world data sets achieve the best performances in model R-RKELM. However, there is a small gap between R-RKELM and other models. Therefore, the three sample selection methods play a positive role in Reduced-KELM for enhancing classification performance. And the proposed model R-RKELM has the best achievement in terms of classification performance.

5. Statistical Analysis

According to the comparison results in Tables 4 and 5, the best performances are achieved by model Relief-F and R-RKELM, respectively. To measure the level of classifying ability between R-RKELM and Relief-F, this study has

applied Wilcoxon-signed Rank Test to test whether R-RKELM has superior in classification ability to Relief-F.

Table 6 reports the accuracy of R-RKELM and Relief-F in each data set. The difference between these two models in terms of accuracy for all the eight data sets is computed. The ranking number based on these absolute difference values is shown. Then, values of $R+$ and $R-$ are computed. $R+$ represents the sum of ranks for the positives, and $R-$ stands for the sum of ranks for the negatives in Table 6. $R+$ is 36 and $R-$ is 0. Based on the table of critical values, at the confidence level of $p=0.05$, the difference between the algorithms is significant if the value of $R-$ is less than 3. Based on the result, we conclude that model R-RKELM has the superior classification ability with model Relief-F statistically.

6. Conclusion

This study introduces a novel classifier called Reformed Reduced Kernel Extreme Learning Machine with RELIEF-F

(R-RKELM) for human action recognition. The proposed framework has two stages. In the first stage, it employs RELIEF-F to discard the irrelevant features with the negative values in the weight. The second stage focuses on the training samples selection for the reduction of computation complexity. Moreover, the proposed approach NSSM in R-RKELM takes advantage of K-means and CDS to replace the randomly reduced part of conventional Reduced-KELM, which reduces the unstable element for classification. Based on the experimental evaluation on eight data sets and statistical analysis, R-RKELM has much better performance in terms of classification and training time than conventional Reduced-KELM than other baselines. The accuracy of the proposed model reached around 90 %. In the future, we will focus on the parameter dependency in our proposed model. The kernel parameter impacts the performance of classification.

Data Availability

All data sets used in paper are from UCI Machine Learning Repository

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

All authors have participated in (a) conception and design, or analysis and interpretation of the data; (b) drafting the article or revising it critically for important intellectual content; and (c) approval of the final version.

Acknowledgments

This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant nos. 3132019400 and 3132021129.

References

- [1] J. Qi Lim and N. Alias, "Comparison of ann and svm for classification of eye movements in eog signals," *J. Phys. Conf. Ser.*, vol. 971, 2018.
- [2] S. Stéfano Frizzo, B. Nathielle Waldrigues, A. Nied et al., "Analysis of training techniques of ann for classification of insulators in electrical power systems," *IET Generation, Transmission & Distribution*, vol. 14, no. 8, pp. 1591–1597, 2020.
- [3] R. Yu, X. An, B. Jin, J. Shi, O. A. Move, and Y. Liu, "Particle classification optimization-based BP network for telecommunication customer churn prediction," *Neural Computing & Applications*, vol. 29, no. 3, pp. 707–720, 2018.
- [4] G.-B. Huang, Q.-Yu Zhu, and C.-K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks, 2," in *Proceedings of the 2004 IEEE international joint conference on neural networks (IEEE Cat. No. 04CH37541)*, pp. 985–990, IEEE, Budapest, Hungary, 25 July 2004.
- [5] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 2, pp. 513–529, 2011.
- [6] A. Iosifidis, A. Tefas, and I. Pitas, "On the kernel extreme learning machine classifier," *Pattern Recognition Letters*, vol. 54, pp. 11–17, 2015.
- [7] Z. Liu, K. Chu, N. Masuyama, and K. Pasupa, "Multiple steps time series prediction by a novel recurrent kernel extreme learning machine approach," in *Proceedings of the 2017 9th International Conference on Information Technology and Electrical Engineering (ICITEE)*, pp. 1–4, IEEE, Phuket, Thailand, 12 October 2017.
- [8] W.-Y. Deng, Y.-S. Ong, and Q.-H. Zheng, "A fast reduced kernel extreme learning machine," *Neural Networks*, vol. 76, pp. 29–38, 2016.
- [9] M. B. Abdulrazaq, M. R. Mahmood, R. M. Subhi, M. H. Abdulwahab, R. R. Zebari, and B. Amira, "An analytical appraisal for supervised classifiers' performance on facial expression recognition based on relief-f feature selection," *Journal of Physics: Conference Series*, vol. 1804, Article ID 012055, 2021.
- [10] S. Yahdin, A. Desiani, N. Gofar, and K. Agustin, "Application of the relief-f algorithm for feature selection in the prediction of the relevance education background with the graduate employment of the universitas sriwijaya," *Computer Engineering and Applications Journal*, vol. 10, no. 2, pp. 71–80, 2021.
- [11] X. Cui, Q. Wang, K. Wei, G. Teng, and X. Xu, "Laser-induced breakdown spectroscopy for the classification of wood materials using machine learning methods combined with feature selection," *Plasma Science and Technology*, vol. 23, no. 5, Article ID 055505, 2021.
- [12] M. R. Mahmood, "Two feature selection methods comparison chi-square and relief-f for facial expression recognition," in *Journal of Physics: Conference Series* vol. 1804, IOP Publishing, Article ID 012056, 2021.
- [13] L. Wu, Y. Peng, J. Fan, Y. Wang, and G. Huang, "A novel kernel extreme learning machine model coupled with k-means clustering and firefly algorithm for estimating monthly reference evapotranspiration in parallel computation," *Agricultural Water Management*, vol. 245, p. 106624, 2021.
- [14] J. Huang, L. Zhu, and Z. Gu, "A clustering method based on extreme learning machine," *Neurocomputing*, vol. 277, pp. 108–119, 2018.
- [15] Z.-T. Liu, M. Wu, W.-H. Cao, J.-W. Mao, J.-P. Xu, and G.-Z. Tan, "Speech emotion recognition based on feature selection and extreme learning machine decision tree," *Neurocomputing*, vol. 273, pp. 271–280, 2018.
- [16] W. Jin, J.-Qi Zhang, and X. Zhang, "Face recognition method based on support vector machine and particle swarm optimization," *Expert Systems with Applications*, vol. 38, no. 4, pp. 4390–4393, 2011.
- [17] M. Coşkun, A. Uçar, Ö. Yildirim, and Y. Demir, "Face recognition based on convolutional neural network," in *Proceedings of the 2017 International Conference on Modern Electrical and Energy Systems (MEES)*, pp. 376–379, IEEE, Kremenchuk, Ukraine, 15 November 2017.
- [18] H. Yan and H. Ouyang, "Financial time series prediction based on deep learning," *Wireless Personal Communications*, vol. 102, no. 2, pp. 683–700, 2018.
- [19] J. Chen, G.-Q. Zeng, W. Zhou, W. Du, and K.-D. Lu, "Wind speed forecasting using nonlinear-learning ensemble of deep learning time series prediction and extremal optimization,"

- Energy Conversion and Management*, vol. 165, pp. 681–695, 2018.
- [20] M. Amrane, S. Oukid, I. Gagaoua, and T. Ensari, “Breast cancer classification using machine learning,” in *Proceedings of the 2018 Electric Electronics, Computer Science, Biomedical Engineerings’ Meeting (EBBT)*, pp. 1–4, IEEE, Istanbul, Turkey, 18 April 2018.
- [21] C.-W. Chang and N. T. Dinh, “Classification of machine learning frameworks for data-driven thermal fluid models,” *International Journal of Thermal Sciences*, vol. 135, pp. 559–579, 2019.
- [22] R. Hecht-Nielsen, “Theory of the backpropagation neural Network**Based on “nonindent” by robert hecht-nielsen, 1992,” in *Proceedings of the International Joint Conference on Neural Networks*, pp. 65–93, Elsevier, Washington, D.C., 18 June 1989.
- [23] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [24] Y.-R. Zeng, Y. Zeng, B. Choi, and L. Wang, “Multifactor-influenced energy consumption forecasting using enhanced back-propagation neural network,” *Energy*, vol. 127, pp. 381–396, 2017.
- [25] Z. Ye and M. K. Kim, “Predicting electricity consumption in a building using an optimized back-propagation and Levenberg-Marquardt back-propagation neural network: case study of a shopping mall in China,” *Sustainable Cities and Society*, vol. 42, pp. 176–183, 2018.
- [26] B. Richhariya and M. Tanveer, “Eeg signal classification using universum support vector machine,” *Expert Systems with Applications*, vol. 106, pp. 169–182, 2018.
- [27] A. Zendejboudi, B. Ma, and R Saidur, “Application of support vector machine models for forecasting solar and wind energy resources: a review,” *Journal of Cleaner Production*, vol. 199, pp. 272–285, 2018.
- [28] G.-B. Huang, Q.-Yu Zhu, and C.-K. Siew, “Extreme learning machine: theory and applications,” *Neurocomputing*, vol. 70, no. 1-3, pp. 489–501, 2006.
- [29] X. Li, H. Xie, R. Wang et al., “Empirical analysis: stock market prediction via extreme learning machine,” *Neural Computing & Applications*, vol. 27, no. 1, pp. 67–78, 2016.
- [30] J. Cao, K. Zhang, M. Luo, C. Yin, and X. Lai, “Extreme learning machine and adaptive sparse representation for image classification,” *Neural Networks*, vol. 81, pp. 91–102, 2016.
- [31] B. Xu, Y. Pan, D. Wang, and F. Sun, “Discrete-time hypersonic flight control based on extreme learning machine,” *Neurocomputing*, vol. 128, pp. 232–241, 2014.
- [32] K. Kira and L. A. Rendell, “A practical approach to feature selection,” *Machine Learning Proceedings 1992*, vol. 1992, pp. 249–256, 1992.
- [33] I. Kononenko, “Estimating attributes: analysis and extensions of relief,” in *European Conference on Machine Learning*, pp. 171–182, Springer, Berlin, Heidelberg, 1994.
- [34] Y. Tian, W. Chen, L. Li, X. Wang, and Z. Liu, “Gait recognition via coalitional game-based feature selection and extreme learning machine,” *NeuroQuantology*, vol. 16, no. 2, 2018.
- [35] G. Hafeez, I. Khan, M. Usman, K. Aurangzeb, and A. Ullah, “Fast and accurate hybrid electric load forecasting with novel feature engineering and optimization framework in smart grid,” in *Proceedings of the 2020 6th Conference on Data Science and Machine Learning Applications (CDMA)*, pp. 31–36, IEEE, Riyadh, Saudi Arabia, 4 March 2020.
- [36] O. Cigdem, F. Yeganli, and H. Demirel, “Performance analysis of different feature selection methods on Parkinson’s disease diagnosis,” *Journal of Neuroscience Methods*, vol. 309.
- [37] A. M Helmi, M. A. A. Al-Qaness, A. Dahou, R. Damaševičius, T. Krilavičius, and M. A. Elaziz, “A novel hybrid gradient-based optimizer and grey wolf optimizer feature selection method for human activity recognition using smartphone sensors,” *Entropy*, vol. 23, no. 8, p. 1065, 2021.
- [38] J. Xie and M. Zhu, “Investigation of acoustic and visual features for acoustic scene classification,” *Expert Systems with Applications*, vol. 126, pp. 20–29, 2019.
- [39] R. Zhang and Y. Xu, “Data-driven dynamic security assessment and control of power systems: an online sequential learning method,” *Journal of Energy Engineering*, vol. 145, no. 5, Article ID 04019019, 2019.
- [40] S. Shamshirband, K. Mohammadi, H.-L. Chen, G. Narayana Samy, D. Petković, and C. Ma, “Daily global solar radiation prediction from air temperatures using kernel extreme learning machine: a case study for Iran,” *Journal of Atmospheric and Solar-Terrestrial Physics*, vol. 134, pp. 109–117, 2015.
- [41] I. Kononenko, E. Šimec, and M. Robnik-Šikonja, “Overcoming the myopia of inductive learning algorithms with relief,” *Applied Intelligence*, vol. 7, no. 1, pp. 39–55, 1997.
- [42] E. W. Forgy, “Cluster analysis of multivariate data: efficiency versus interpretability of classifications,” *Biometrics*, vol. 21, pp. 768–769, 1965.
- [43] H. Jia, S. Ding, and Z. Shi, “Approximate weighted kernel k-means for large-scale spectral clustering,” *Journal of Software*, vol. 26, no. 11, pp. 2836–2846, 2015.
- [44] D. Dua and C. Graff, “UCI machine learning repository,” 2017, <https://archive.ics.uci.edu/ml/index.php>.
- [45] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, “Transition-aware human activity recognition using smartphones,” *Neurocomputing*, vol. 171, pp. 754–767, 2016.
- [46] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, “A public domain dataset for human activity recognition using smartphones,” *Esann*, vol. 3, no. 3, 2013.