Journal of Network Intelligence
Taiwan Ubiquitous Information
©2023 ISSN 2414-8105
Volume **8**, Number **4**, November **2023**

Severe Imbalanced Big Dataset Classification Based on the Combination of Enhanced WOA algorithm and Quasi-Recurrent Neural Network

Yufeng Zhao*

School of Electronics and IoT Engineering
Chongqing Industry Polytechnic College
Chongqing,401120, China
zhaoyf@cqipc.edu.cn

He Jie

Lee Kong Chian Faculty of Engineering and Science
Tunku Abdul Rahman University
Kuala Lumpur, 43000, Malaysia
hejie@cqipc.edu.cn

*Corresponding author: Yufeng Zhao
Received May 29, 2023, revised July 2, 2023, accepted September 16, 2023.

Abstract. *A classification method based on enhanced whale optimization algorithm (EWOA) and deep learning model is proposed to solve severe class imbalance issue in large-scale dataset classification. The method consists of three stages: feature selection, preprocessing, and classification. To improve classification accuracy, EWOA is designed to search for optimal feature subset in the imbalanced data, removing redundant and uncorrelated features. The dataset is processed with modified Synthetic Minority Oversampling Technique (SMOTE) in which the data is projected onto the kernel space using Support Vector Machine (SVM) to recognize support vectors of boundary sample points. Then, on account of kernel distances, neighbors of support vectors of samples from the minority class are calculated and new samples are adaptively synthesized by selecting either interpolation or extrapolation according to distribution of the neighboring sample classes. Finally, preprocessed dataset is classified using a quasi-recurrent neural network (QRNN). Experiment results validate that the proposed QRNN model combined with improved resampling algorithm is beneficial to alleviate the impact of data imbalance, and the proposed method precedes comparison methods in classifying severely imbalanced data sets, proving the superiority and efficacy of the proposed method.*
Keywords: Unbalanced Big Data Classification; Enhanced Whale Optimization Algorithm; Deep Learning; Quasi-Recurrent Neural Network; Synthetic Minority Oversampling Technique; Support Vector Machine.

1. **Introduction.** Unbalanced data refers to a dataset where the quantity of samples in some classes are much fewer than quantity of samples in other classes, classes with fewer samples are often regarded as positive classes, while classes with more samples are regarded as the negative classes [1]. The issue of unbalanced data resides in many fields, such as fraud detection, medical diagnosis, industrial fault detection, and so on [2,3].

Traditional classification algorithms generally assume that the class distribution is roughly balanced. Therefore, when the class distribution in a dataset is imbalanced, the decisional boundaries of classifiers may be shifted, resulting in misclassification, where

positive class samples are misclassified as negative class samples [4]. In practical applications, misclassification of positive class samples often leads to more serious consequences, such as misclassifying industrial failures as normal or diagnosing illnesses as normal in medical diagnosis. Therefore, it is vital to improve the classification performance of imbalanced data, especially to improve the recognition accuracy of positive class samples [5, 6]. Galar et al. [7] pointed out that traditional undersampling methods, such as Random Undersampling (RAMU), randomly delete majority class samples in the dataset to realize data balancing effect. However, importance of majority samples may not be the same, and this deletion method is prone to drop unique information in majority samples, exacerbate intra-class imbalance, and reduce the generalization of the model, thus causing negative impacts on the performance of the classification model. In contrast, traditional oversampling methods, such as Random Oversampling (RAMO) [8], randomly select minority class samples in the dataset and duplicate them to achieve a balance in different categories of samples. However, the generated data may not be representative and may excessively strengthen some minority class samples, leading to the model overfitting problem [9]. Therefore, researchers have conducted in-depth research on the problems caused by traditional resampling techniques and proposed undersampling and synthetic oversampling methods. Błaszczyński et al. [10] proposed distance-based undersampling methods, which preserve representative majority class samples by calculating distance of samples from majority and minority classes. However, distance calculation often takes more time. Based on the synthetic oversampling theory, Synthetic Minority Oversampling Technology (SMOTE) [11] focuses on minority class samples to generate new data between minority class sample points to achieve sample expansion. On this basis, the density-based SMOTE (DBSMOTE) method [12] used reachable distance and kernel distance as standards to determine the quantity of generated samples for minority classes with different attributes. In addition to adding constraints to SMOTE when generating minority samples, SMOTE-IPF [13] method performed denoising processing on the data generated by SMOTE through iterative partition filters. The SMOTE for multi-class classification (SMOM) algorithm [14] addresseed the over-generalization problem by assigning different weights to negative class samples. The geometric SMOTE (G-SMOTE) algorithm [15] generated synthetic samples in the geometric are in vicinity of each selected minority sample to enhance data generation mechanism.

Improving existing algorithms to achieve better performance in imbalanced classification problems is a common solution. Representative methods in this category mainly include feature selection (FS) and cost-sensitive (CS) methods [16]. The basic criterion of FS techniques is to pick out a more representative feature subset, enabling model to effectively distinguish between different types of samples and improve classification performance [17]. Viegas et al. [18] brought out concept of genetic programming to wrapper FS algorithms to extract more effective features. Moayedikia [19] used symmetric uncertainty to measure the relevance from selected features to actual classes, and selected the most relevant subset to realize better effect for high-dimensional unbalanced data. The fundamental design of CS learning technology is to import cost-aware coefficients in the model's training process, and to consider minority class samples more by selecting, transforming, or bringing in cost-sensitivity to the features. Teisseyre et al. [20] proposed an adaptive weight cost-sensitive classification method for multi-label classification tasks. Zhang et al. [21] proposed a cost-aware dictionary-based method, which separately took into account misclassification price during different stages to obtain minimum loss.

Deep learning can automatically extract more discriminative deep-level feature representations without the need for manual feature design. Over recent decades, many deep learning (DL) models have been designed, including Convolutional Neural Networks

(CNNs), Deep Belief Neural Networks (DBNs), et al. These models have been implemented in various classification or pattern recognition tasks, achieving better effect than traditional techniques [22, 23]. Ando et al. [24] presented a DL-based oversampling method, which first uses CNN to learn deep space mapping, then maps each original sample to a multi-dimensional deep space through CNN. Then, artificial samples are synthesized within linear subspace of the nearest-neighbors of minorities in embedded space, achieving better sampling performance. Khan et al. [25] designed a feature learning scheme on basis of cost-aware learning, which synchronously optimizes learning procedures of features and classifier during CNN modeling, making the learned features more robust and discriminative. Douzas et al. [26] utilized conditional Generative Adversarial Networks (CGAN) to synthesize minority class samples, effectively improving quality of synthetic samples. Shen et al. [27] proposed a deep undersampling technique that utilizes DL model to select a smaller, sensitive, representative subset of samples from multiple classes, reducing the influences of data unbalancing issue.

Based on investigation of existing big data classification methods, we propose an scheme to perform classification of severely-imbalanced dataset based on Quasi-Recurrent Neural Network (QRNN) and enhanced Whale Optimization Algorithm (EWOA). The algorithm consists of three stages: FS, preprocessing, and classification. In FS stage, the EWOA algorithm is used to eliminate irrelevant and redundant features to improve classification accuracy. In the preprocessing stage, the SMOTE-SVM method is used to find the support vectors of the training set using SVM and adaptively interpolate them both internally and externally in the kernel space to address the class imbalance problem and achieve balance between different classes in the dataset. During classification, the QRNN-based DL method is used to classify the preprocessed dataset.

The remaining sections of this paper are organized as follows. Chapter II presents the proposed method in detail, which includes an IWOA algorithm for feature selection, the SMOTE-SVM algorithm for data preprocessing, and an imbalanced big data classification model based on QRNN. Chapter III gives the experiment results and analysis. Finally, Chapter IV provides a summary of the entire paper.

2. **Unbalanced dataset classification based on EWOA and QRNN.**

A classification scheme for severely-unbalanced big dataset is proposed based on EWOA and QRNN . The algorithm consists of three stages: FS, preprocessing, and classification. Figure 1 illustrates the proposed method's flowchart. In the FS stage, an improved WOA algorithm is used to optimize the original features by redundant and impertinent features to boost classification accuracy. In preprocessing stage, the SMOTE algorithm is incorporated with support vector machine (SVM) to identify boundary samples of training set, i.e., support vectors, and adaptively interpolate them both internally and externally in the kernel space to solve class unbalancing issue. During classification, QRNN model is used to classify the preprocessed dataset.

2.1. **Feature Selection.**

Impertinent or redundant features can significantly affect classification effect. The major objective of FS stage is to obtain optimal feature subset to improve classification accuracy. At this stage, each subset of features is considered as a whale's location in EWOA algorithm. Random subsets of features will be generated, with each subset containing a varying number of features that is less than or equal to the total number of features in original dataset. The optimal solutions will be identified as whales that exhibit both high classification accuracy and a minimal number of features.

The WOA algorithm is inspired by the hunting behavior of whale populations. Compared to other traditional algorithms, WOA has received attention and recognition from

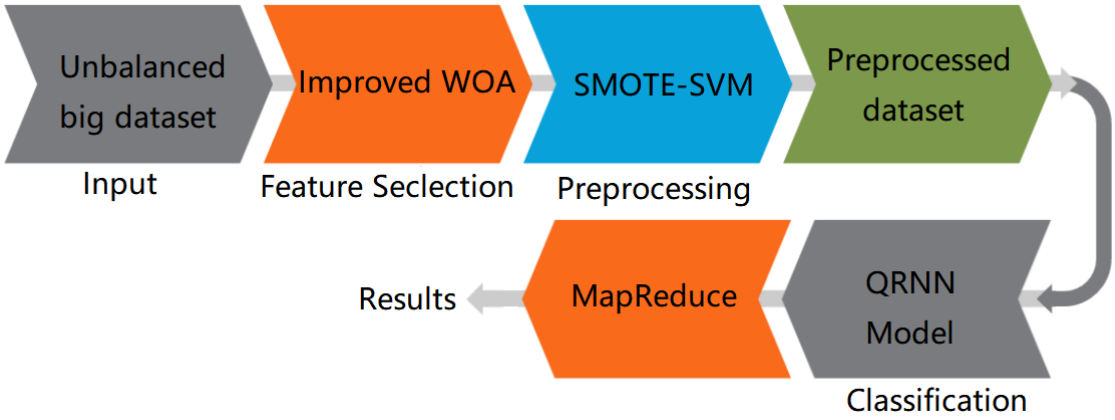

Figure 1. Flowchart of the Proposed Method

scholars in various fields due to its high efficiency, simplicity, and fewer parameters [28]. However, as a relatively new swarm intelligence algorithm, WOA has demonstrated its effectiveness in solving optimization problems, it still faces common challenges such as being prone to getting trapped in local optima, low optimization accuracy, and slow convergence speed. In this regard, this paper proposes an improved algorithm called EWOA that incorporates quadratic interpolation. Specifically, in EWOA, adaptive weights are designed to optimize the convergence speed of the population and balance the global search and local development capabilities. Furthermore, by combining quadratic interpolation and greedy selection strategies, new optimal solutions are generated using the extreme points of quadratic functions, which boosts computational accuracy and avoids the population getting trapped in local optima.

In the proposed EWOA, an inertia weight parameter is designed to adjust global searching and local developing capabilities, which is applied in prey encircling and attacking stages. To account for the role of the best whale during initial hunting process in original WOA, a weight w is added to the target prey position.

Assuming that N whales distributing in a d-dimensional space, location of the i-th whale in the at iteration T is denoted as $X_i^T = (X_{i,1}, X_{i,2}, X_{i,3}, ..., X_{i,d})$, where $i = 1, 2, 3, ..., N$; $T = 1, 2, 3, ..., T_{MAX}$. T_{MAX} is the highest number of repetitions, and X_b^T represents position of the best whale individual (position of target prey) when searching up to the T-th generation of the population. Then the updates of whale positions are calculated as:

$$\begin{cases} X_i^{T+1} = w \cdot X_b^T - A \times D_1 & p < 0.5 \\ X_i^{T+1} = w \cdot X_b^T + D_2 \cdot e^{ZM} \cdot \cos(w\pi M) & p \geq 0.5 \end{cases} \tag{1}$$

Where A is a coefficient variable, D_1 represents the encircling step size during the prey encircling stage, D_2 represents the distance between the current prey and the whale, Z depends on the spiral shape, M is a arbitrary numerical value within [-1, 1], and p is a uniform probability factor between [0, 1]. If $p \geq 0.5$, the algorithm comes in spiral updating phase; If $p < 0.5$, it comes in prey encircling phase. The magnitude of the whale's positional changes is controlled by adjusting the weight w:

$$w = w_1 + (w_1 - w_2)\frac{2}{\pi} \cdot \arccos(\frac{T}{T_{MAX}}) \tag{2}$$

Where w_1 is the initial weight value, w_2 is the final weight value, T is the current iteration, and T_{MAX} is the largest allowable count of iterations. It has been verified that the

algorithm's optimization performance is best when $w_1 = 0.8$ and $w_2 = 0.3$. Therefore, the designed adaptive nonlinear inertia weight utilizes the monotonically decreasing characteristic of the inverse cosine function on $(-\infty, \infty)$ to make the weight of the EWOA algorithm nonlinearly decrease with the increase of iteration number. When the weight coefficient is relatively large at the beginning of the algorithm, the larger weight will enable the algorithm to quickly reach the vicinity of the optimal position, and the population will have good global exploration ability. Corresponding to the growth of iterations, the weighting coefficient gradually decreases, and the smaller weight causes the whale to move towards the vicinity of optimal location, and influence of the optimal whale location gradually increases. This makes each iteration closer to hypothetical best possible result, thereby improving the convergence speed of the population and having good local development ability, avoiding falling into local optima.

2.1.1. *Quadratic Interpolation Strategy.* In EWOA algorithm, the quadratic interpolation [30] is applied to further refine local search competence and convergence. After using quadratic interpolation, the population is updated to a new population. In the process of generating a new population at each iteration, the whales are re-ranked in descending order according to their fitness in the population, and three individuals x_j , x_{j+1} , and x_{j+2} , $j = 1, 2, 3, ..., N-2$, are selected from the population in turn for quadratic interpolation to obtain a new individual $\vec{x}_j$. The fitness greedy strategy is then applied to evaluate fitness of x_j and $\vec{x}_j$:

$$x_j = \begin{cases} \vec{x}_j, f(\vec{x}_j) < f(x_j) \\ x_j, f(\vec{x}_j) \geq f(x_j) \end{cases} \tag{3}$$

As it can be seen that by applying quadratic interpolation to three individuals that come close to the finest resolution, the searching ability and speed of the algorithm have been enhanced. When the three individuals remain distant from the most favorable result, quadratic interpolation increases the variety within community, broadens searching range for solutions, and enhances ability for breaking free from confines of near-best solution. Therefore, applying quadratic interpolation as a local search operator in the WOA can optimize convergence and precision of the population comprehensively, and enhance its local search ability.

2.2. **Preprocessing Stage.** The core notion underlying SMOTE is to address attribute level rather than case level by fabricating instances of the under-represented class, thereby manufacturing artificial examples of minority class. Synthetic instances are fabricated by taking k nearest-neighbors of each minority instance and fabricating new examples in proportion to the linear distances between those neighbors. However, the reliability of the generated samples using linear interpolation method may not be sufficient when applied to solve nonlinear classification problems. Therefore, the SMOTE-SVM algorithm is adopted for new sample generation. Figure 2 depicts the algorithm procedure. Based on the distribution of the nearest-neighbor samples of support vectors of marginalized group sample set under the kernel distance, interpolation or extrapolation methods are designed to generate new samples. Assuming there is a training sample set S, where set of positive samples $S_{\min} \in S$ and set of negative samples $S_{\max} \in S$. The algorithm takes the following steps:

Utilize an SVM classifier to obtain the support vectors SV_+ of $S_{\min}$.

Searching for k nearest-neighbors of SV_+ in S based on kernel distance. The kernel distance is calculated as:

$$d^{\varphi}(x_i, x_j)^2 = ||\varphi(x_i) - \varphi(x_j)||^2 = K(x_i, x_i) - 2K(x_i, x_j) + K(x_j, x_j) \tag{4}$$

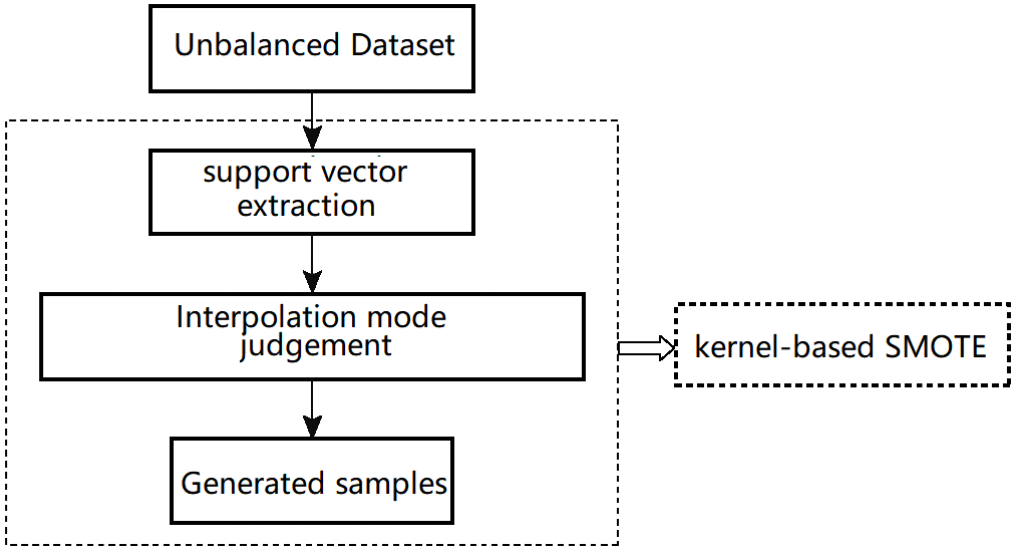

Figure 2. Flowchart of the SMOTE-SVM algorithm

Where x_i and x_j are any two points in SV_+. The kernel function $K(x_i, x_i) = \varphi(x_i)^{\mathrm{T}}\varphi(x_j)$ is the dot product of x_j and x_i when expressed in terms of their feature vectors.

According to group attribute of k nearest-neighbors of x_i in SV_+ , the approach used for fabricating novel instance, either interpolation or extrapolation, is determined. Let m be the count of instances from the over-represented class within the nearest k instances to x_i. If $m = k$, x_i is considered as a noise sample and re-labeled. If $m > k/2$, interpolation is performed on x_i . If $m >< k/2$, extrapolation is performed on x_i. This strategy is essentially based on oversampling using support vectors as the classification boundary. It generates minority class samples based on a decision mechanism, eliminates noise points, and helps the positive class expand into areas of low density of negative class samples, thereby improving the accuracy of classification.

Search for N nearest-neighbors of x_i in S_{min} , where N is the multiple of the number of SV_+ , and perform interpolation or extrapolation to generate new samples. Assuming x_j is one of the neighbors, the newly generated point x^{ij} in the feature space is represented as:

$$x^{ij} = \varphi(x_i) + \delta^{ij}(\varphi(x_j) - \varphi(x_i)) \tag{5}$$

Where δ^{ij} is a random number. When the new sample is generated by interpolation, the newly generated point is located on the line segment between x_i and x_j, and δ^{ij} takes a value between 0 and 1. Otherwise, the newly generated point is on the extension of the line segment between x_i and x_j , and δ^{ij} takes a value between -1 and 0.

SMOTE-SVM improves the decision boundary by generating new samples in different ways based on distribution of different instances. Within the shared space between positive and negative samples, the sample distribution in this area can cause the decision boundary to tilt toward the positive class. This algorithm selects interpolation to increase the number of positive samples in this area, which moves the decision boundary toward the negative class. In the area where positive and negative instances are far apart, the algorithm performs extrapolation to expand the positive boundary outward and obtain more distribution space for positive samples. The entire process is performed in the kernel space. When negative samples dominate among the nearest-neighbors of the boundary sample, the boundary sample is in the overlapping area between positive and negative classes in the kernel space, so the interpolation is performed on the boundary sample.

If positive samples dominate among the nearest-neighbors of the boundary sample, the boundary sample is in the area where the two classes are far apart, so extrapolation is performed on the boundary sample.

2.3. **QRNN-Based Classification Model.** In the classification stage, QRNN serves to categorize the preprocessed dataset. The commonly used long short-term memory (LSTM) network relies on the output of previous time step when computing each time step, which makes it difficult to perform efficient computing when dealing with large-scale data, further reducing the modeling capacity of the data. QRNN, conversely, uses a neural sequence modeling method with alternating convolutional layers to fully utilize the sequential details of the input series at output stage, enabling parallel processing of data across time steps. In addition, QRNN simplifies the LSTM structure by only computing the forget gate and output gate, lessening the calculations required by the system. Convolutional layers and pooling layers are main components of QRNN, and Figure 3 depicts the network architecture. The convolutional layers are used to extract feature information of input data and convolve it with the gate function, while pooling layers extract feature information of convolutional layer output to reduce feature dimensionality. Assuming

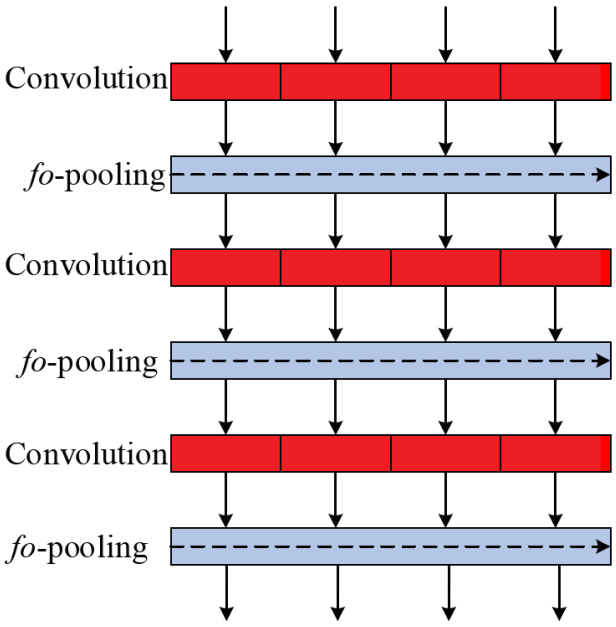

Figure 3. Illustration of the QRNN network.

sequence $X = (x_1, x_2, ..., x_T)$ of length T is the input sequence of QRNN convolutional layer, feature information of input sequence is first convolved with filters of quantity w and width r in the time dimension in a convolutional manner, so that current and past time information are input to the QRNN unit. The QRNN unit structure is given in Figure 4. Let the input sequence at time t be $X = (x_{(t-r+1)}, ..., x_t)$, then the calculation process for the output sequence z_t , forget gate f_t , and output gate o_t is as follows:

$$z_t = \tanh(W_z^1 x_{(t-r+1)} + W_z^2 x_{(t-r+2)} + W_z^r x_t) \tag{6}$$

$$f_t = \text{Sigmoid}(W_f^1 x_{(t-r+1)} + W_f^2 x_{(t-r+2)} + W_f^r x_t) \tag{7}$$

$$o_t = \text{Sigmoid}(W_o^1 x_{(t-r+1)} + W_o^2 x_{(t-r+2)} + W_o^r x_t) \tag{8}$$

Where W_z, W_f , and W_o represent weight matrices, and sigmoid and tanh are activation functions. When the filter width increases, the model can compute more features.

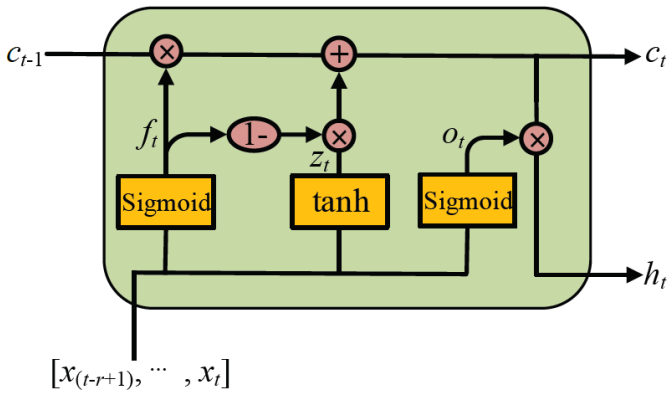

Figure 4. Unit structure of the QRNN

The pooling layer updates the memory cell state c_t at time t using the fo-pooling pooling method [31]:

$$c_t = f_t \odot c_{t-1}(1 - f_t) \odot z_t \tag{9}$$

where $\odot$ represents element-wise multiplication, ct-1 represents the previous memory cell state. Finally, the output h_t of the QRNN unit at time t is determined from output gate o_t and memory cell state c_t :

$$h_t = o_t \odot c_t \tag{10}$$

3. **Experiment and Analysis.** The experiment was conducted on a PC with an Intel Core i5-10400F 2.90 GHz processor and an NVIDIA GeForce RTX 2070 SUPER environment, implemented using the PyTorch framework. The experiment was effectively evaluated through 5-fold cross-validation. Therefore, the dataset was randomly divided into 5 equal parts, where 4 parts were selected as the training set and 1 part was selected as the test set for each experiment. This process was repeated 5 times, and the average value of each experiment was used as the final evaluation result.

3.1. **Datasets.** To thoroughly validate the effectiveness of the proposed method, we selected six severely imbalanced datasets from different fields [32]. These datasets include the Pageblocks and Kddcup datasets in the text classification domain, the Yeast and Abalone19 datasets in the bioinformatics domain, and the Thyroid and Contraceptive datasets in the disease prevention and diagnosis domain. Among them, Kddcup and Abalone19 are binary classification datasets, while Pageblocks, Yeast, Thyroid, and Contraceptive are multi-class datasets. Table 1 provides details of the datasets, where Imbalance Ratios (IRs) represent the quotient of the amount of examples relative to the fewest instances among all classes.

Table 1. Statistics of the experiment datasets.

Dataset	Samples #	Class #	Feature dimension	IRs
Pageblocks	5472	5	10	175.46: 11.75: 4.11: 3.14: 1
Kddcup	2233	2	41	73.4: 1
Yeast	1484	5	8	92.6: 85.8: 48.8: 32.6:1
Abalone19	4177	2	8	129.5:1
Thyroid	720	3	21	39.18: 2.18: 1
Contraceptive	1473	3	9	1.89: 1.53: 1

3.2. **Evaluation Metrics.** To assess the predictive ability for unbalanced datasets, two commonly used evaluation metrics are G-mean and F-value, both of which are drawing from confusion matrix shown in Table 2. In confusion matrix, *TP* represents number of true positive samples classified properly, *TN* represents number of true negative samples classified properly, and *FP* and *FN* represent number of false positive samples classified incorrectly and number of false negative samples classified erroneously, respectively.

Table 2. Confusion Matrix

Classification	Positive	Negative
Classified Positive	TP	FP
Classified Negative	FN	TN

G-mean represents geometric mean of the classification accuracy for both the positive and negative categories, and requires that both values are high for G-mean to be high. A higher G-mean indicates a more robust model. Its definition is as follows:

$$G - mean = \sqrt{\frac{TP}{TP + FN} \times \frac{TN}{TN + FP}} \tag{11}$$

F-value means weighted average of precision and recall, where precision and recall represent the sample's ability to correctly predict the positive class and the share of actual positive instances that are accurately forecasted, respectively. $P = TP/(TP + FP)$, $R = TP/(TP + FN)$. F-value is given as:

$$F - value = \frac{(1 + \beta^2) \times P \times R}{\beta^2 \times P + R} \tag{12}$$

In the experiment, β is set to 1, indicating that both P and R matter equally. A higher F-value indicates that fewer positive samples are misclassified, and the cost of misclassification is smaller.

3.3. **Results and Analysis.** Firstly, the performance differences of using SMOTE-SVM versus original SMOTE, DBSMOTE [12], SMOTE-IPF [13], and G-SMOTE [15] were analyzed in the data pre-processing stage when using the proposed scheme. Table 3 lists G-means on different datasets. Baseline indicates directly using the imbalanced dataset for classification. From the results, it can be observed that all the resampling methods improve the performance to some extent compared to the baseline method directly using the original imbalanced data for classification. This indicates that data balancing is beneficial for alleviating the negative impact of class imbalance and improving classification performance. When using the SMOTE algorithm for sampling, since no specific distinction was made among the positive class samples, the odds of misclassifying the samples at the boundary was relatively high. SMOTE-IPF and G-SMOTE enhanced the boundary to a certain extent by removing noise points, which works beneficially for classification performance. SMOTE-SVM achieved the best performance by not only enhancing the decision boundary but also taking into account the possibility of poor points when projecting samples into the feature space. In SMOTE-SVM, the oversampling ratio pertains to the multiple of the count of positive support vectors that need to be interpolated, and it is an important parameter that affects the final classification results. In this study, we conducted experiments with different oversampling ratios for the SMOTE-SVM method, where $N = 1$ means generating new samples equal to quantity of support vectors in the current positive class samples. During the experiment, N was increased from 1 until the

Table 3. G-means results with different preprocessing techniques. (%)

Methods	Pageblocks	Kddcup	Yeast	Abalone19	Thyroid	Contraceptive
baseline	65.29	72.38	51.33	57.05	73.61	49.98
SMOTE	89.88	94.22	68.41	72.39	87.11	58.20
DBSMOTE	90.07	94.19	68.70	73.14	88.09	60.08
SMOTE-IPF	93.08	95.44	69.88	75.26	90.01	61.89
G-SMOTE	92.77	96.49	72.85	77.30	91.94	62.17
SMOTE-SVM	94.40	99.13	74.76	78.25	94.37	65.82

dataset was roughly balanced. The outcomes demonstrated that the best classification performance was not achieved when the dataset was perfectly balanced.

Taking the Yeast dataset as an example, when the oversampling ratios were set to 1, 5, 10, 15, 20, 25, 30, and 35, the F-value results obtained for each classification are given in Figure 5. The best classification performance was achieved when $N = 23$. It should be noted that the dataset was perfectly balanced when N takes value of 26. The oversampling ratio directly affects the number of newly generated samples, where too many new samples can cause data redundancy and too few new samples may not improve the classification accuracy. Both of these scenarios are not conducive to improving the classification performance. Therefore, the specific oversampling ratio for each dataset needs to be further determined through experiments based on the actual situation. The oversampling ratio for other datasets in this study was determined using the same method.

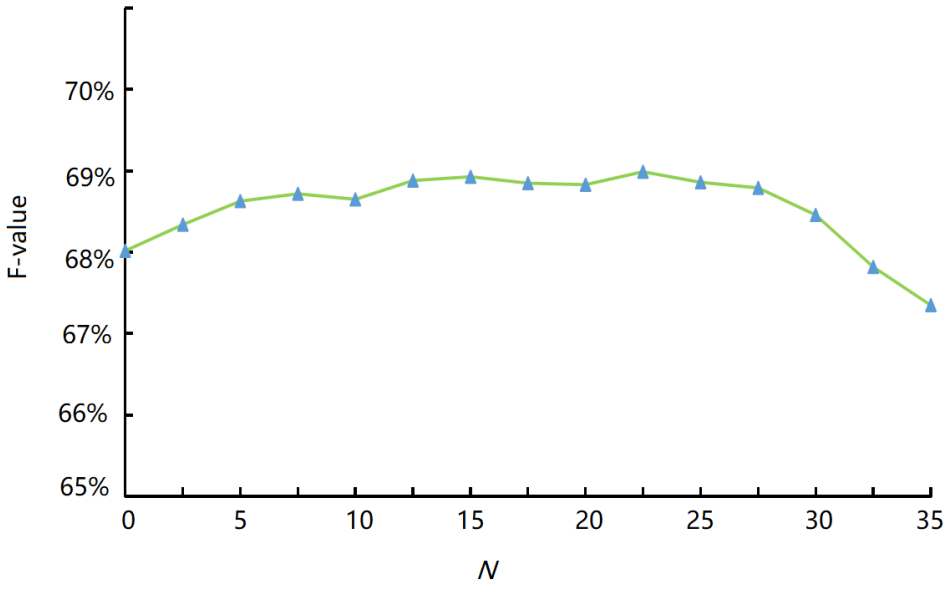

Figure 5. The impact of N on the F-measure of the proposed method on Yeast dataset.

Table 4 and Table 5 present the G-means and F-value results of proposed and other advanced methods on six experimental datasets, where best performance is highlighted in bold. Among them, the method proposed in [20] is a traditional FS method, and its performance is not ideal in all indicators. DL methods significantly improve the classification performance on severely imbalanced large datasets. Ando et al. [24] used CNN to map the original samples to a multi-dimensional space, which can generate synthetic samples in the linear subspace of the closest instances of the underrepresented class, improving the sampling performance. However, the computational complexity is high and it is not suitable for datasets with numerous attributes. Khan et al. [25] conducted attribute

inference utilizing cost-aware training, optimizing both feature and classifier learning, making the learned features more robust and discriminative. However, the method is too sensitive to the choice of hyperparameters. Douzas et al. [26] used CGAN to create minority class samples, effectively improving the quality of synthetic samples, but it is not suitable for extremely imbalanced datasets. Shen et al. [27] used a DL model to select a smaller, sensitive, and typical subset of samples from multiple classes, reducing the impact of class imbalance. However, the method performs poorly on datasets with numerous categories. In comparison, the proposed method uses EWOA for optimal FS, preprocesses data with SMOTE-SVM, and finally uses QRNN for classification. While improving the efficiency of processing large datasets, the proposed method can effectively solve the classification problem of datasets with multiple classes, large feature dimensions, and severe class imbalance, outperforming other traditional and DL methods.

Table 4. G-means result comparison with different methods (%).

Methods	Pageblocks	Kddcup	Yeast	Abalone19	Thyroid	Contraceptive
[20]	76.54	87.44	55.33	58.22	71.43	51.08
[24]	89.88	95.50	68.05	69.30	88.70	58.77
[25]	90.49	97.52	68.72	71.45	91.85	60.94
[26]	92.85	96.73	70.43	70.32	92.99	61.85
[27]	94.02	98.94	72.09	76.64	94.41	63.77
Proposed	**94.40**	**99.13**	**74.76**	**78.25**	**94.37**	**65.82**

Table 5. *F-value* results comparison with different methods.

Methods	Pageblocks	Kddcup	Yeast	Abalone19	Thyroid	Contraceptive
[20]	59.88	81.54	50.88	57.05	60.33	42.17
[24]	64.57	92.37	58.45	66.24	79.41	46.08
[25]	66.35	96.88	61.08	68.93	83.09	46.25
[26]	68.12	96.49	63.72	70.17	87.54	48.73
[27]	69.47	97.02	65.14	70.38	89.47	50.03
Proposed	**71.85**	**97.62**	**68.92**	**72.63**	**90.99**	**54.78**

To evaluate the impact of imbalanced ratios on the performance of different algorithms, Figure 6 analyzes the variations in *F-value* results of different methods when adjusting the IR values on the experimental dataset. From the figure, it can be observed that when the IR value is 1, i.e., the large dataset does not exhibit any imbalanced phenomenon, the performance differences among different methods are relatively small. However, when the IR value increases to 40, the *F-value* of the methods in [20], [24], [25], and [26] begin to decrease significantly, indicating that these methods cannot handle severely imbalanced large datasets well. The proposed method and the method in [27] both ensure stable performance at different IR values. When the IR value exceeds 110, the proposed method's performance is significantly better than that of [27], demonstrating the effectiveness of the proposed method in tackling the classification task of severely imbalanced large-scale datasets.

To analyze the impact of different components in the proposed method on the classification performance of large datasets, Table 6 shows the *G-mean* and *F-value* results obtained using different components on the Yeast dataset. The first row represents the performance when only using the LSTM network without any optimization algorithms or preprocessing methods. The last row represents the classification result when using the complete proposed method. The results show that the improved WOA algorithm can significantly improve the performance of feature selection and neural network parameter

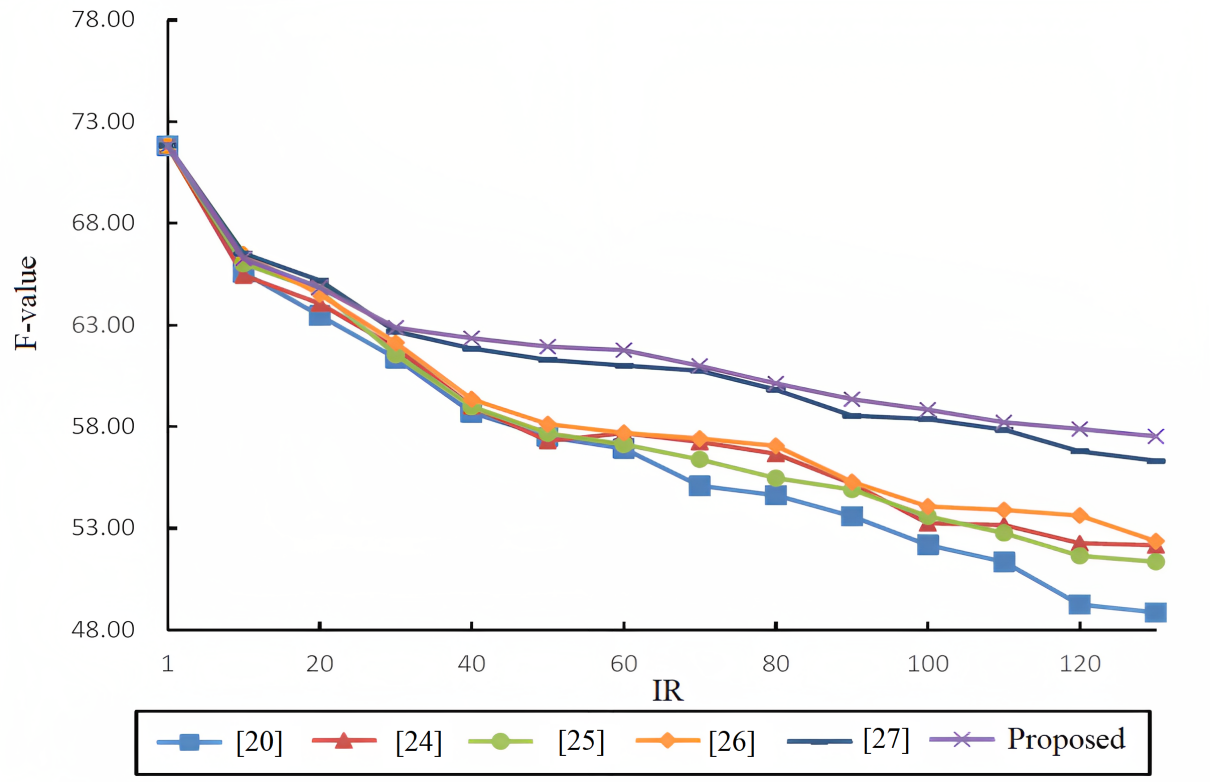

Figure 6. Performance comparison under different IR values.

optimization. Compared with LSTM, the QRNN network can improve the processing capability for large-scale data while reducing computational costs.

Table 6. Ablation study results on Yeast dataset (%)

WOA	Improved WOA	SMOTE	SMOTE-SVM	LSTM	QRNN	*G-MEANS*	*F-value*
×	×	×	×	✓	×	55.87	50.03
×	×	×	×	×	✓	57.94	52.45
✓	×	✓	×	×	✓	65.92	58.37
×	✓	✓	×	×	✓	68.41	62.75
×	✓	×	✓	×	✓	74.76	68.92

4. **Conclusion.** A novel scheme for unbalanced big data classification combining EWOA and DL model is proposed to address the issues of low classification accuracy and prone to fall to local optima in most existing imbalanced data classification algorithms. The EWOA algorithm is utilized to optimize features of imbalanced data, eliminating redundant features and finding the optimal feature subset. A kernel-based nonlinear interpolation method is proposed to effectively solve the problem of inconsistency between sample generation and classification space. Two interpolation methods are used to adjust the classification boundary, ensuring diversity and reliability of sample generation. Finally, the QRNN network is adopted to perform classification on the preprocessed dataset. Findings from testing indicate that the suggested approach can handle extremely unbalanced large datasets and achieve high accuracy in classification. It should be noted that the proposed method may have a long processing time. Therefore, how to enhance the practicality and speed of the algorithm is a problem that needs to be further addressed.

Acknowledgement. This work is supported by project of The Ministry of Education's College Student Department's Supply and Demand Matching Employment and Education Project(20230103588) Project of the Informationization Education Guidance Committee of the Ministry of Education(KT22620) Science and Technology Research Project of Chongqing Education Commission(KJQN202203213)

REFERENCES

[1] A. Viloria, O. Lezama, N. Mercado-Caruzo, "Unbalanced data processing using oversampling: Machine Learning," *Procedia Computer Science*, vol. 175, pp. 108-113, 2020.

[2] H. Wang, Z. Xu, H. Fujita, and S. Liu, "Towards felicitous decision making: An overview on challenges and trends of Big Data," *Information Sciences*, vol. 367, pp. 747-765, 2016.

[3] Y. Ma, Y. Peng, T.-Y. Wu, "Transfer learning model for false positive reduction in lymph node detection via sparse coding and deep learning," *Journal of Intelligent & Fuzzy Systems*, vol. 43, no. 2, pp. 2121-2133, 2022.

[4] H.-X. Guo, Y.-J. Li, S. Jennifer, M.-Y. Gu, Y.-Y. Huang, and B. Gong, "Learning from class-imbalanced data: Review of methods and applications," *Expert systems with applications*, vol. 73, pp. 220-239, 2017.

[5] T.-Y. Wu, A. Shao, J.-S. Pan, "CTOA: Toward a Chaotic-Based Tumbleweed Optimization Algorithm," *Mathematics*, vol. 11, no. 10, 2339, 2023.

[6] T.-Y. Wu, H.-N. Li, S.-C. Chu, "CPPE: An Improved Phasmatodea Population Evolution Algorithm with Chaotic Maps," *Mathematics*, vol. 11, no. 9, 1977, 2023.

[7] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 463-484, 2011.

[8] A. Moreo, A. Esuli, F. Sebastiani, "Distributional random oversampling for imbalanced text classification," in *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*. IEEE, 2016, pp. 805-808.

[9] M. S. Santos, J. P. Soares, P. H. Abreu, H. Araujo, and J. Santos, "Cross-validation for imbalanced datasets: avoiding overoptimistic and overfitting approaches," *IEEE Computational Intelligence Magazine*, vol. 13, no. 4, pp. 59-76, 2018.

[10] J. Błaszczyński, J. Stefanowski, "Neighbourhood sampling in bagging for imbalanced data," *Neurocomputing*, vol. 150, pp. 529-542, 2015.

[11] A. Fernández, S. Garcia, F. Herrera, N.-V. Chawla, "SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary," *Journal of Artificial Intelligence Research*, vol. 61, pp. 863-905, 2018.

[12] C. Bunkhumpornpat, K. Sinapiromsaran, C. Lursinsap, "DBSMOTE: density-based synthetic minority over-sampling technique," *Applied Intelligence*, vol. 36, pp. 664-684, 2012.

[13] J. A. Sáez, J. Luengo, J. Stefanowski, and F. Herrera, "SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering," *Information Sciences*, vol. 291, pp. 184-203, 2015.

[14] T. Zhu, Y. Lin, Y. Liu, "Synthetic minority oversampling technique for multiclass imbalance problems," *Pattern Recognition*, vol. 72, pp. 327-340, 2017.

[15] G. Douzas, F. Bacao, "Geometric SMOTE a geometrically enhanced drop-in replacement for SMOTE," *Information Sciences*, vol. 501, pp. 118-135, 2019.

[16] F. Feng, K.-C. Li, J. Shen, Q.-G. Zhou, and X.-H. Yang, "Using cost-sensitive learning and feature selection algorithms to improve the performance of imbalanced classification," *IEEE Access*, vol. 8, pp. 69979-69996, 2020.

[17] S. Maldonado, J. López, "Dealing with high-dimensional class-imbalanced datasets: Embedded feature selection for SVM classification," *Applied Soft Computing*, vol. 67, pp. 94-105, 2018.

[18] F. Viegas, L. Rocha, M. Gonçalves, F. Mourão, G. Sá, T. Salles, G. Andrade, and I. Sandin, "A genetic programming approach for feature selection in highly dimensional skewed data," *Neurocomputing*, vol. 273, pp. 554-569, 2018.

[19] A. Moayedikia, K. L. Ong, Y. L. Boo, W. G. Yeoh, and R. Jensen, "Feature selection for high dimensional imbalanced class data using harmony search," *Engineering Applications of Artificial Intelligence*, vol. 57, pp. 38-49, 2017.

[20] P. Teisseyre, D. Zufferey, M. Słomka, "Cost-sensitive classifier chains: Selecting low-cost features in multi-label classification," *Pattern Recognition*, vol. 86, pp. 290-319, 2019.
[21] G. Zhang, F. Porikli, H. Sun, Q. Sun, G. Xia, and Y. Zheng, "Cost-sensitive joint feature and dictionary learning for face recognition," *Neurocomputing*, vol. 391, pp. 177-188, 2020.
[22] A. Shrestha, A. Mahmood, "Review of deep learning algorithms and architectures," *IEEE access*, vol. 7, pp. 53040-53065, 2019.
[23] F.-Q. Zhang, T.-Y. Wu, Y. Wang, R. Xiong, G. Ding, P. Mei, and L.-Y. Liu, "Application of quantum genetic optimization of LVQ neural network in smart city traffic network prediction," *IEEE Access*, Vol. 8, pp. 104555-104564, 2020.
[24] S. Ando, C.-Y. Huang. "Deep over-sampling framework for classifying imbalanced data," in *Machine Learning and Knowledge Discovery in Databases: European Conference (ECML PKDD 2017)*. IEEE, 2017, pp. 770-785.
[25] S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel and R. Togneri, "Cost-sensitive learning of deep feature representations from imbalanced data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3573-3587, 2017.
[26] G. Douzas, F. Bacao. "Effective data generation for imbalanced learning using conditional generative adversarial networks," *Expert Systems with Applications*, vol. 91, pp. 464-471, 2018.
[27] W. Shen, Y. Li, Y. Han, L. Chen, D. Wu, Y. Zhou, and B. Xu, "Boundary sampling to boost mutation testing for deep learning models," *Information and Software Technology*, vol. 130, 106413, 2021.
[28] J. Nasiri, F. M. Khiyabani, "A whale optimization algorithm (WOA) approach for clustering," *Cogent Mathematics & Statistics*, vol. 5, no. 1, 1483565, 2018.
[29] I. N. Trivedi, P. Jangir, A. Kumar, N. Jangir, and R. Totlani, "A novel hybrid PSO–WOA algorithm for global numerical functions optimization," in *Advances in Computer and Computational Sciences: Proceedings of ICCCCS 2016*. IEEE, 2016, pp. 53-60.
[30] X. Chen, C. Mei, B. Xu, K. Yu, and X. Huang, "Quadratic interpolation based teaching-learning-based optimization for chemical dynamic system optimization," *Knowledge-Based Systems*, vol. 145, pp. 250-263, 2018.
[31] M. Wang, X. Wu, Z. Wu, S. Kang, D. Tuo, G. Li, D. Su, D. Yu, H. Meng, "Quasi-fully convolutional neural network with variational inference for speech synthesis" in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 7060-7064.
[32] M. Galar, A. Fernández, E. Barrenechea, and F. Herrera, "EUSBoost: Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling," *Pattern Recognition*, vol. 46, no. 12, pp. 3460-3471, 2013.