Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 3183469, 11 pages
https://doi.org/10.1155/2022/3183469

Research Article

A Novel Approach of Feature Space Reconstruction with Three-Way Decisions for Long-Tailed Text Classification

Xin Li, [1] **Lianting Hu**, [2,4] **Peixin Lu**, [1] **Tianhui Huang**, [1] **Wei Yang**, [5] **Quan Lu**, [1] **Huiying Liang**, [2,4] **and Long Lu** [1,3]

[1] *School of Information Management, Wuhan University, Wuhan, China*
[2] *Medical Big Data Center, Guangdong Provincial People's Hospital, Guangdong Academy of Medical Sciences, Guangdong, China*
[3] *Institute of Pediatrics, Guangzhou Women and Children's Medical Center, Guangzhou Medical University, Guangzhou, China*
[4] *Guangdong Cardiovascular Institute, Guangzhou, Guangdong, China*
[5] *Wuhan National Laboratory for Optoelectronics, Huazhong University of Science and Technology, Wuhan, China*

Correspondence should be addressed to Xin Li; xinli2020@whu.edu.cn and Long Lu; lulong@whu.edu.cn

Received 1 December 2021; Accepted 21 March 2022; Published 15 April 2022

Academic Editor: Ahmed Mostafa Khalil

Text classification is widely studied by researchers in the natural language processing field. However, real-world text data often follow a long-tailed distribution as the frequency of each class is typically different. The performance of current mainstream learning algorithms in text classification suffers when the training data are highly imbalanced. The problem can get worse when the categories with fewer data are severely undersampled to the extent that the variation within each category is not fully captured by the given data. At present, there are a few studies on long-tailed text classification which put forward effective solutions. Encouraged by the progress of handling long-tailed data in the field of image, we try to integrate effective ideas into the field of long-tailed text classification and prove the effectiveness. In this paper, we come up with a novel approach of feature space reconstruction with the help of three-way decisions (3WDs) for long-tailed text classification. In detail, we verify the rationality of using a 3WD model for feature selection in long-tailed text data classification, propose a new feature space reconstruction method for long-tailed text data for the first time, and demonstrate how to effectively generate new samples for tail classes in reconstructed feature space. By adding new samples, we enrich the representing information of tail classes, to improve the classification results of long-tailed text classification. After some comparative experiments, we have verified that our model is an effective strategy to improve the performance of long-tailed text classification.

1. Introduction

Due to the rapid development of Internet technology and information construction, it becomes easier to obtain valuable text data to study and analyze problems in certain fields. As a consequence, text classification is widely studied by researchers in the natural language processing field [1–3], and many applications based on text classification technology have been developed, such as news filtering and organization [4], e-mail classification and spam filtering [5], web page mining [6], and medical documents' classification [3]. Text classification is a task that assigns textual documents to predefined classes based on the knowledge extracted from their content [7]. A wide variety of techniques has been designed for text classification, which is mainly based on machine learning models, such as Naive Bayes (NB) [8], Decision Tree (DT) [9], Support Vector Machines (SVMs) [10], and deep learning models, such as convolutional neural networks (CNNs) [11, 12], recurrent neural network (RNN) [13], bidirectional long short-term memory (BI-LSTM) [14, 15], and transformer models [16]. Calculating text representation, training classification models, and predicting class labels for class-unknown documents are the main steps of text classification [17]. The main steps based on these models in text classification are shown in Figure 1.

As we all know, sufficient data are the premise of superior performances of these artificial intelligence learning methods; especially, deep learning models are developed to train massive data initially [18, 19]. But real-world text data often follow a long-tailed distribution as the frequency of each class is typically different, such as news topic classification, clinical name entities recognition, and disease diagnosis for electronic medical records [20–22]. It means that a dataset can have a large number of under-represented classes (tail classes) and a few classes with more than sufficient data (head classes). The performance of supervised learning algorithms in the process of text classification suffers when the training data are highly imbalanced [23, 24]. The problem can get worse when the categories with a fewer data are severely undersampled to the extent that the variation within each category is not fully captured by the given data.

At present, few studies have studied the problem of long-tailed text classification and put forward effective solutions. Some methods have been proposed to handle the problem of imbalanced datasets in text classification [25–29]. Finding and fine tuning the network model of classifiers suitable for specific data sets is one of these methods, such as [25] describing the proposed approach for text classification in an unbalanced data environment based on an implementing individual LSTM neural network. Data augmentation is also a frequently used way in scenarios with few samples or unbalanced categories [26–29], such as back translation [26], which has become an effective way of data augmentation; for example, researchers translate some English text into Chinese and then translate them back into English, they can get various new training data, and the size of the dataset is doubled. Oversampling [27] and undersampling [28] are both common methods to deal with unbalanced datasets. But those methods which repeatedly take samples only consider the processing at the data level, so the generated data are still redundant and invalid in the feature space, and the improvement of classification effect is not obvious. The method of finding and fine tuning appropriate classifiers in specific data sets does not have universal applicability. At present, research studies on real-world long-tailed data mainly focus on the field of image classification [30–34].

Those techniques for learning long-tailed distributions generally fall into three groups: resampling [30], reweighting and cost-sensitive learning [31, 32], and feature manipulation [33, 34]. Especially, methods based on feature space augmentation of long-tailed image data have effectively improved the classification effect recently [33, 34].

Encouraged by the progress of handling long-tailed data in the field of image [30–34], considering texts carry a lot of semantic information, analysis is more complex than image data, we try to integrate effective ideas into the field of long-tailed text classification, and prove the effectiveness. In this paper, we come up with a novel approach of feature space reconstruction with 3WD for long-tailed text classification in Figure 2. In detail, with the help of 3WD model, the main innovations of this paper can be summarized into three aspects: (1) We verify the effectiveness of using a 3WD model for feature selection in long-tailed text data classification. (2) We propose a new feature space reconstruction method for long-tailed text classification and demonstrate how to effectively generate new samples for tail classes in reconstructed feature space. By adding new samples, we enrich the representing information of tail classes, to improve the classification results of long-tailed text classification. (3) After some comparative experiments, we have verified that our model is an effective strategy to improve the performance of long-tailed text classification, which can effectively improve the accuracy values of tail classes. The rest of paper is organized as follows: Section 2 introduces the novel approach of feature space reconstruction with 3WD for long-tailed text classification related works. In Section 3, we design some experiments and provide the corresponding results in detail. Finally, we draw some conclusions and some possible future works in Section 4.

2. Methods

In this section, we propose a novel approach of feature space reconstruction with 3WD for long-tailed text classification. As mentioned above, calculating text representations, training classification models, and predicting class labels for class-unknown documents are the main steps of text classification. Firstly, with the help of 3WD, two evaluation functions are used to generate the optimal set of features, which can effectively reduce the number of feature words, make the extracted feature words more representative of categories, and text representations can be calculated accordingly. Then, because texts containing sufficient information in tail-class data only account for a small part, some information will be lost in the process of text representations, and it is easy to find that the representation vectors of tail classes are very sparse. Thus, we augment the data in tail classes by reconstructing the feature space with the features learned from the head classes with ample samples. In particular, we decompose the features of each class into class-generic features and class-specific features. Novel samples of tail classes are then generated randomly by fusing the class-specific features from the tail classes with the class-generic features from head classes. Finally, the new samples are added to existing data to train the classification models and predict class labels to verify the effect of our method.

2.1. Feature Selection by 3WD.

Considering the decision risk and uncertainty, the 3WD method was initially proposed by Yao [35] based on rough sets and Bayesian decision procedure. The 3WD method which has attracted extensive attention in various fields provides a new tool for text classification [36–38]. The core idea of the 3WD method is to divide the universe into three disjoint regions, including an acceptance decision region, a deferment decision region, and a rejection decision region. If the decision maker has enough information, he (or she) can quickly make a decision, i.e., acceptance and rejection. Otherwise, the decision maker can choose to postpone the decision.

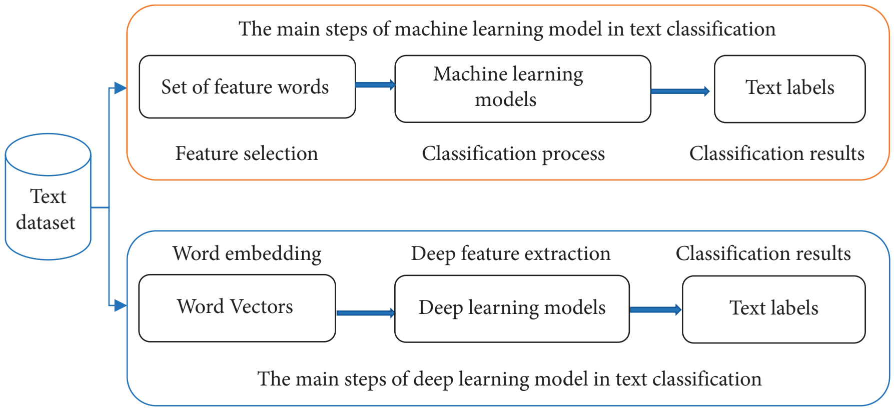
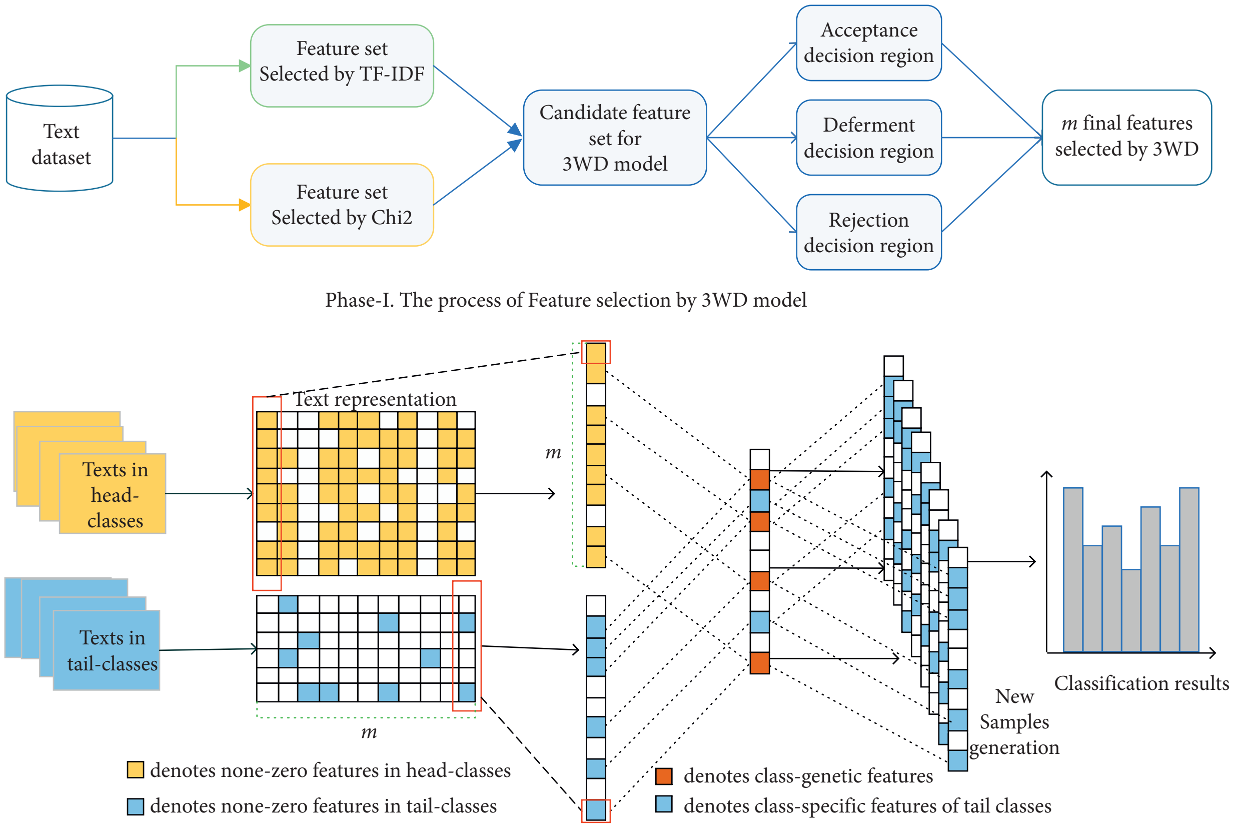


FIGURE 1: The main steps based on machine learning and deep learning models in text classification.


Text dataset
Feature set Selected by TF-IDF
Feature set Selected by Chi2
Candidate feature set for 3WD model
Acceptance decision region
Deferment decision region
Rejection decision region
m final features selected by 3WD
Texts in head-classes
Texts in tail-classes
Text representation
m
m
New Samples generation
Classification results
denotes none-zero features in head-classes
denotes none-zero features in tail-classes
denotes class-genetic features
denotes class-specific features of tail classes

Phase-I. The process of Feature selection by 3WD model

Phase-II. The process of feature space reconstruction and new samples generation in tail-classes

FIGURE 2: The proposed model architecture in our paper.

Let the dataset of long-tailed texts denotes as $X = \{X_{\text{Head}}, X_{\text{Tail}}\} = \{x_1, x_2, \ldots, x_i, \ldots, x_n\}$, where X_{Head} is the set of the classes with ample samples (head classes), X_{Tail} is the set of the samples of under-represented classes (tail classes), and the total amount of all texts in X is n. $C = \{C_{\text{Head}}, C_{\text{Tail}}\} = \{C_1, C_2, \ldots, C_k, \ldots, C_K\}$, C_k denotes the kth class of X, and there are K classes in X. Firstly, we need to preprocess the long-tailed samples and do word segmentation for Chinese texts. $V = \{v_1, v_2, \ldots, v_j, \ldots, v_N\}$ be the vocabulary mined from X.

The 3WD model used in this paper uses the TF-IDF algorithm [39] and Chi2 Statistics [40] as double-decision functions to select features in the long-tailed dataset, which can consider the frequency and distribution of vocabularies, as well as label-related information. M_μ is defined as the feature set selected by decision function μ, and M_ω is defined as the feature set selected by decision function ω. For $\forall v_j \in V$:

$$\mu(v_j) = \mathrm{TF}(v_j, x_i) \times \mathrm{IDF} = \frac{n_{j,i}}{\sum_s n_{s,i}} \times lg \frac{|n| + 1}{|\{i\text{: } v_j \in x_i\}| + 1}. \quad (1)$$

Here, $n_{i,j}$ denotes the number of times vocabulary v_j appeared in document x_i, $\sum_k n_{k,i}$ is the total times of all vocabularies that appeared in document x_i, and the total amount of all texts in X is $|n|$. $|\{i\text{: } v_j \in x_i\}|$ is the total number of documents with v_j. Adding 1 to prevent v_j is not in X, resulting in a calculation error caused by zero denominators.

$$\begin{aligned}\omega(v_j) &= x^2(v_j, C_k) \\ &= \frac{n(A_{jk}D_{jk} - C_{jk}B_{jk})^2}{(A_{jk} + C_{jk})(B_{jk} + D_{jk})(A_{jk} + B_{jk})(D_{jk} + C_{jk})},\end{aligned} \quad (2)$$

where A_{jk} is the number of documents that contain v_j and belong to C_k, B_{jk} is the number of documents that contain v_j but do not belong to C_k, C_{jk} is the number of documents that do not contain v_j but belong to C_k, and D_{jk} is the number of documents that do not contain v_j and not belong to C_k.

The feature sets M_μ and M_ω of the longed-tailed text data are generated by two evaluation functions, we map the feature words existing in both feature sets to the positive field $\mathrm{POS}(\mu, \omega)$ of the 3WD model. The feature words that only exist in one feature set M_μ or M_ω are added to the boundary domain $\mathrm{BND}(\mu, \omega)$ of the 3WD model, further processing is carried out with these features according to the classification results to decide to retain them or not, and the feature words with no appearance in feature sets are added to the negative domain $\mathrm{NEG}(\mu, \omega)$. 3WD rules can be constructed through three fields: the positive field $\mathrm{POS}(\mu, \omega)$ corresponds to acceptance, the negative field $\mathrm{NEG}(\mu, \omega)$ corresponds to rejection, and the boundary field $\mathrm{BND}(\mu, \omega)$ corresponds to non commitment. Then, $\forall v_j \in V$:

$$\begin{cases} v_j \in \mathrm{POS}(\mu, \omega), \textit{ if } v_j \in M_\mu \text{ and } v_j \in M_\omega, \\ v_j \in \mathrm{BND}(\mu, \omega), \text{ if } (v_j \in M_\mu \text{ but } v_j \notin M_\omega) \text{ or } (v_j \notin M_\mu \text{ but } v_j \in M_\omega), \\ v_j \in \mathrm{NEG}(\mu, \omega), \text{if } v_j \notin M_\mu \text{ and } v_j \notin M_\omega. \end{cases} \quad (3)$$

and $\mathrm{NEG}(\mu, \omega) = (\mathrm{POS}(\mu, \omega) \bigcup \mathrm{BND}(\mu, \omega))^C$.

The final feature set $T = \{t_1, t_2, \ldots, t_m\}$ is generated accordingly, which is a collection of m feature words used for classification. For the effectiveness of this method for feature selection in the process of long-tailed text data classification, we evaluate in Section 3.2 of the paper.

2.2. Feature Space Reconstruction and Novel Sample Generation for Tail Classes.

In Section 2.1, we obtain the final feature set $T = \{t_1, t_2, \ldots, t_m\}$ with m optimal feature words; thus, document x_i can be represented as a vector of features $F_i = \{f_1, f_2, \ldots, f_m\}$, and the collection of long-tailed text data X can be represented as equation:

$$M = \begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1m} \\ f_{21} & f_{22} & & f_{2m} \\ \vdots & & \ddots & \vdots \\ f_{n1} & f_{n2} & \cdots & f_{\mathrm{nm}} \end{bmatrix}, \quad (4)$$

where element f_{ij} represents the binary weight of t_j from F_i. If t_j appears in x_i, $f_{ij} = 1$, and otherwise $f_{ij} = 0$.

Thus, for a given class C_k, the matrix of C_k can be expressed as equation:

$$M_{C_k} = \begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1m} \\ f_{21} & f_{22} & & f_{2m} \\ \vdots & & \ddots & \vdots \\ f_{S1} & f_{S2} & \cdots & f_{Sm} \end{bmatrix}. \quad (5)$$

Then, we can get the class feature vector of C_k which is denoted as $F_{C_k} = \{f_1^k, f_2^k, \ldots, f_m^k\}$, where $f_i^k = \mathrm{sgn}(\sum_S f_{si})$, $\mathrm{sgn}(x) = 1$ for $x > 0$, $\mathrm{sgn}(x) = 0$ for $x = 0$. F_{C_k} represents the binary weight of feature words in all documents of class C_k, namely, if t_j appears in x_i, $f_{ij} = 1$, otherwise $f_{ij} = 0$.

By comparing with head classes, we find that the class feature vectors of tail classes are extremely sparse, which contain small numbers of nonzero eigenvalues. This is because texts containing sufficient information in tail-class data only account for a small part, and some information is lost in the process of text representations, which leads to poor classification results. The data in the head class are sufficient. It seems natural to use the knowledge learned from the head class to help recover the missing information in the tail class.

In [34], authors use deep convolution neural networks to augment the feature space of long-tailed image data. Through image feature sampling, the class features of these images are divided into class-generic and class-specific features. It is proved that the class general features from the head class are regarded as transferable knowledge for the feature space expansion of the tail class, and the effectiveness of this idea is proved in the image filed.

In this paper, we explore the effectiveness of feature reconstruction in long-tailed text data classification. For a given head class and a tail class, $F^{\text{head}} = \{f_1^h, f_2^h, \ldots, f_m^h\}$ and $F^{\text{tail}} = \{f_1^t, f_2^t, \ldots, f_m^t\}$ are class feature vectors of them correspondingly, and m is the total number of feature words. By analyzing their class feature sets, we map the feature words existing in both class feature sets to the class-generic feature set M^G, and the feature words only existing in tail-class feature set to tail class-generic feature set M^{TS}, thus for $\forall t_i \in T$:

$$\begin{cases} t_i \in M^G, \text{ if } f_i^h * f_i^t = 1, \\ t_i \in M^{TS}, \text{if } f_i^h * f_i^t = 0 \text{ and } \operatorname{sgn}\left(f_i^t - f_i^h\right) = 1. \end{cases} \tag{6}$$

Using the class-generic feature set M^G and t tail class-generic feature set M^{TS}, we reconstruct the feature space of tail-class documents. The class-specific features from the class are then combined with the class-generic features from the Na classes linearly. A random combination ratio is generated to guide the fusion by randomly drawing class-generic and class-specific feature vectors to form an augmented sample for the tail class.

2.3. Long-Tailed Text Classification. In Section 2.2, novel samples of tail classes are generated by fusing the class-specific features from the tail classes with the class-generic features from classes with ample data. Finally, the new samples are added to existing data to train the classification models and predict class labels to verify the effect of our method. In our paper, we use XGBoost as our classifier which is a method of gradient boosting decision tree.

We evaluate the classification performance of the proposed method in our paper and compare it with other existing machine learning and deep learning methods. For each class C_k, we use accuracy precision, recall, and F1-measure as experiment metrics defined in Table 1.

$$\begin{aligned} \text{Accuracy}\left(C_k\right) &= \frac{\text{TP}\left(C_k\right) + \text{TN}\left(C_k\right)}{\text{TP}\left(C_k\right) + \text{TN}\left(C_k\right) + \text{FP}\left(C_k\right) + \text{FN}\left(C_k\right)}, \\ \text{Precision}\left(C_k\right) &= \frac{\text{TP}\left(C_k\right)}{\text{TP}\left(C_k\right) + \text{FP}\left(C_k\right)}, \\ \text{Recall}\left(C_k\right) &= \frac{\text{TP}\left(C_k\right)}{\text{TP}\left(C_k\right) + \text{FN}\left(C_k\right)}, \\ \text{F}_1\left(C_k\right) &= \frac{2 * \text{Precision}\left(C_k\right) * \text{Recall}\left(C_k\right)}{\text{Precision}\left(C_k\right) + \text{Recall}\left(C_k\right)}. \end{aligned} \tag{7}$$

In multiclass classification problems, the overall performance can be measured by averaging the evaluation methods. Microaverage and macroaverage are used widely for this purpose. In this study, in cases of evaluating long-tailed distribution, it is better to use macroaverage scores than microaverage scores since the data size of categories is not considered in the microaverage calculation. Taking the definition of F1-value as an example, the definitions of macro-F1 and micro-F1 are shown in equations as follows:

$$\begin{aligned} \text{Macro} - \text{F1} &= \frac{1}{K} \sum_{i=1}^{K} F_1\left(C_k\right), \\ \text{Micro} - \text{F1} &= \frac{2 * \sum_{i=1}^{K} \text{TP}\left(C_k\right)}{2 * \sum_{i=1}^{K} \text{TP}\left(C_k\right) + \sum_{i=1}^{K} \text{FP}\left(C_k\right) + \sum_{i=1}^{K} \text{FN}\left(C_k\right)}. \end{aligned} \tag{8}$$

Table 1: Meanings of TP, TN, FP, and FN.

Sample x_i in the corpus X	Result in C_k	Not a result in C_k
Belongs to C_k	$\text{TP}\left(C_k\right)$	$\text{FP}\left(C_k\right)$
Does not belong to C_k	$\text{FN}\left(C_k\right)$	$\text{TN}\left(C_k\right)$

3. Experiments

3.1. Dataset. In this paper, we use Fudan University corpus as a long-tailed text classification corpus. The Fudan University TC corpus is from the Chinese NLP group in the Department of Computer Information and Technology, Fudan University of China. We randomly selected some category data to simulate the distribution of real-world long-tailed datasets. The numbers and distribution of our dataset can be seen in Table 2 and Figure 3. There are 4 head classes and 12 tail classes. The ratio of training set to test set is 7 : 3.

In this study, we carried out 3 experiments on the dataset. All experiments were implemented on a 64 bit MAC computer with 8 GB internal storage. The experimental code was written in Python language using Scikit-learn (sk-learn) and TensorFlow. Sk-learn and TensorFlow are commonly used third-party modules in machine learning and deep learning which encapsulate many commonly used machine learning and deep learning algorithms, such as SVM, XGBoost, RNN, CNN, and others. In preprocessing, all documents were segmented into words by the opensource tool Jieba, and stop words were removed in this process.

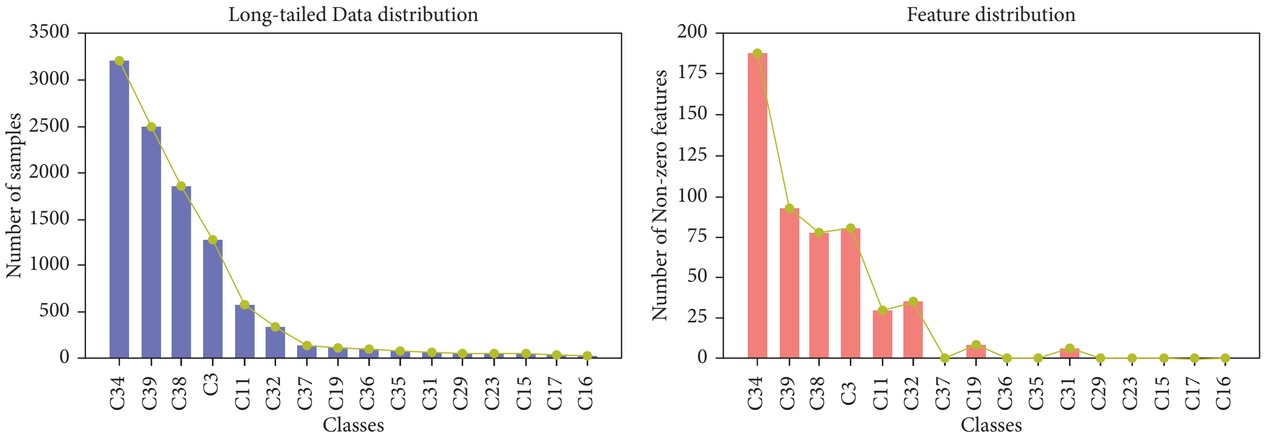
By the final feature set obtained by the 3WD model, we can calculate the text representations of our long-tailed text dataset described in Section 2.1. And the number of nonzero features of each class is obtained, as shown in Figure 3(b). It can be seen that the tail classes contain less information, and some information is lost in the process of text representations; it is easy to find that the representation vectors of tail class are very sparse.

3.2. The Effectiveness of Feature Selection Using 3WD. In this paper, we use the 3WD model to generate the final feature set $T = \{t_1, t_2, \ldots, t_m\}$ of long-tailed text data, which is a collection of m feature words used for classification. In this section, we use word frequency, CHI2 [40], and TF-IDF [39] as comparative methods for feature selection, to test the effectiveness of the 3WD model in our dataset. For all methods mentioned above, we select the top 2,000 feature words and use SVM as a classification model. The corresponding results are shown in Table 2.

From the results of Table 3, we can see that the 3WD algorithm has improved the long-tailed text classification performance in the scores of accuracy, precision, recall, and F1-value than other methods. It is about 10 ~ 20% improvement than other methods on average. From the average point of view, 3WD model's overall performance is better than other methods.

Table 2: The numbers of the long-tailed text dataset used in our paper.

No.	Class	Class label	Samples
1	Head	C34-Economy	3201
2	Head	C39-Sports	2507
3	Head	C38-Politics	1854
4	Head	C3-Art	1282
5	Tail	C11-Space	582
6	Tail	C32-Agriculture	348
7	Tail	C37-Military	150
8	Tail	C19-Computer	120
9	Tail	C36-Medical	104
10	Tail	C35-Law	103
11	Tail	C31-Environment	89
12	Tail	C29-Transport	67
13	Tail	C23-Mine	67
14	Tail	C15-Energy	65
15	Tail	C17-Communication	55
16	Tail	C16-Electronics	32



(a)

(b)

Figure 3: (a) The distribution of our long-tailed text dataset; (b) the distribution of features.

Table 3: The performance results of 3WD models and baseline methods for feature selection.

Method	Accuracy	Macroprecision	Macrorecall	Macro-F1 score
Word frequency	0.56	0.487	0.518	0.51
Chi2	0.70	0.577	0.591	0.58
TF-IDF	0.79	0.638	0.627	0.631
3WD (our paper)	0.87	0.697	0.637	0.656

3.3. The Effectiveness of Feature Reconstruction of Tail Class. In Section 2.2, we augment the tail-class data by reconstructing the feature space with the features learned from the head classes with ample samples. In particular, we decompose the features of each class into class-generic features and class-specific features. Novel samples of tail classes are then generated randomly by fusing the class-specific features from the tail classes with the class-generic features from head classes. In this section, the new samples are added to existing data to train the classification models and predict class labels to verify the effect of our method. According to the number of samples generated, it is divided into 6 levels, namely, 0%, 10%, 20%, 30%, 40%, 50%, and 60%, which represent the degrees of sample generation.

In the first stage of our experiment, we use the 3WD algorithm as a feature selection method to select 2,000 effective feature words; the introduction and distribution of our long-tailed text dataset can be seen in Section 3.1. From the results of Figure 4, we can see that when we add new samples of tail-class data by reconstructing the feature space by learning information from head-classes based on the original samples, the classification accuracy is significantly

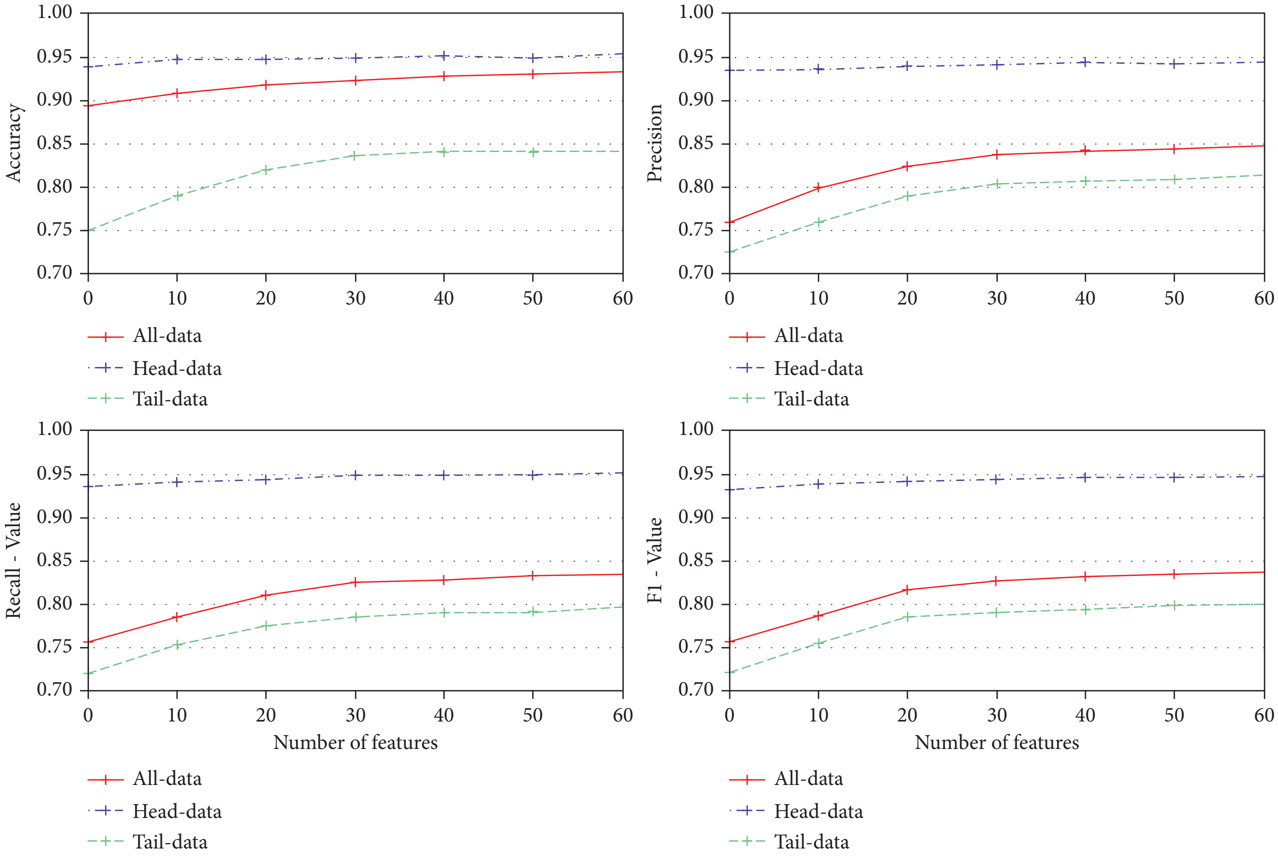


FIGURE 4: The performance results of our proposed model with different generation degrees.

improved from 0 to 30% generation degree; the experiment shows that the classification results are not significantly improved after 30–40% generation degree. Especially, the accurate score of tail-class data has improved from 0.752 to 0.84 when the sample generation degree increased to 30%, from 30% degree to 60% degree, and the accuracy is not improved. The precision score of tail-class data has improved from 0.725 to 0.813, the recall score of tail-class data has improved from 0.72 to 0.797, the F1-value score of tail-class data has improved from 0.722 to 0.801 when the sample generation degree increased to 30%. It can also be seen from Figure 4 that the classification results of tail-class data have improved without reducing the accuracy of all data and head-class data.

3.4. Comparing with Mainstream Methods. In our experiments, we use XGBoost as our classifier and compare it with four baseline training methods: TF-IDF with Linear SVM, CNN, RNN, and Bi-LSTM. XGBoost is a method of gradient boosting decision tree. It strives to maximize speed and efficiency, so it is called X (extreme) gboosted. Parameter settings for XGBoost in our experiments: max_depth:6, eta: 0.5, objective: multi: SoftMax, and num_class:16.

Convolutional neural network (CNN) is a variant of multilayer perceptron (MLP), which essentially is a multilayer perceptron. The key to its success lies in its way of local connection and sharing weights. On the one hand, it reduces the number of weights, which makes the network easy to optimize and reduces the risk of overfitting. Parameter settings for CNN in our experiments: embedding_dim = 100, seq_length = 600, num_classes = 16, num_filters = 256, kernel_size = 5, hidden_dim = 128, dropout_keep_prob = 0.5, learning_rate = 1e-3, and batch_size = 64.

RNN (recurrent neural network) is a kind of a neural network used to process sequence data. The biggest difference of RNN is that it also establishes weight connections between neurons and between layer to capture the information at previous time points. Parameter settings for RNN in our experiments: embedding_dim = 100, seq_length = 600, num_classes = 16, num_layers = 3, hidden_dim = 128, rnn = "gru," dropout_keep_prob = 0.8, learning_rate = 1e-3, and batch_size = 128.

Bi-LSTM is the abbreviation of bidirectional long short term memory, which is a combination of forwarding LSTM and backwarding LSTM. Both are often used to model context information in natural language processing tasks. LSTM is a kind of RNN that is very suitable for modeling time-series data, such as text data. Parameter settings refer to RNN.

In this section, the results of accuracy, macroprecision, macrorecall, and macro-F1 score in our experiment are shown in Table 4 and Figure 5. It is easy to see that the proposed method in our paper outperformed baseline training methods in all cases, especially in tail classes. The macro-F1 score obtained by our method in tail classes is 80.1%, which is 5~10% higher than that obtained by deep

Table 4: The performance results of our proposed model and baseline methods.

Method	Datatype	Accuracy	Macroprecision	Macrorecall	Macro-F1 score
TF-IDF	All data	0.79	0.65	0.635	0.64
	Head class	0.872	0.854	0.857	0.857
	Tail class	0.635	0.597	0.581	0.585
CNN	All data	0.915	0.765	0.732	0.747
	Head class	0.943	0.937	0.928	0.933
	Tail class	0.747	0.731	0.692	0.709
RNN	All data	0.905	0.783	0.767	0.775
	Head class	0.932	0.927	0.916	0.918
	Tail class	0.782	0.759	0.728	0.742
BI-LSTM	All data	0.927	0.82	0.796	0.81
	Head class	0.954	**0.947**	0.94	0.943
	Tail class	0.821	0.778	0.747	0.752
Our method	All data	**0.933**	**0.842**	**0.825**	**0.832**
	Head class	0.953	0.945	**0.951**	**0.948**
	Tail class	**0.841**	**0.813**	**0.797**	**0.801**

Bold values means the best values in accuracy, macroprecision, macrorecall, and macro-F1 score.

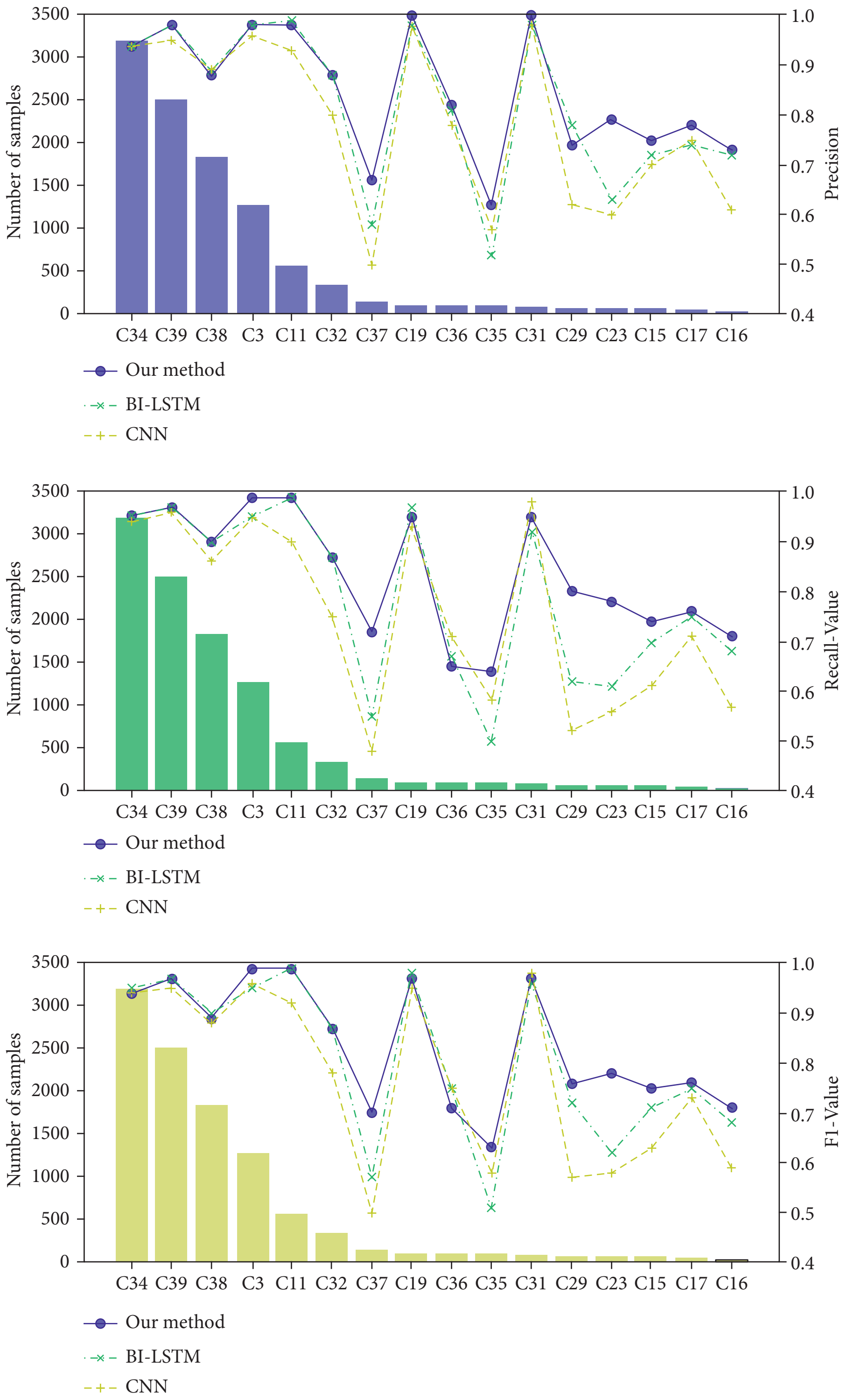


Figure 5: The performance results of our proposed model and deep learning methods.

learning models and 20% higher than that obtained by TF-IDF. Meanwhile, other results reflect similar improvements to our method.

4. Discussion

Long tail text classification is a problem that needs to be solved in the real world. The performance of current mainstream learning algorithms in text classification suffers when the training data are highly imbalanced. In this paper, a novel approach of feature space reconstruction with the help of 3WD for long-tailed text classification is proposed to greatly reduce the impact from the long-tailed distribution of datasets. In detail, we use Fudan University corpus as a long-tailed text classification corpus with 4 head classes and 12 tail classes, and verify the effectiveness of using the 3WD model for feature selection in long-tailed text data. Then, we propose a new feature space reconstruction method for long-tailed text data and demonstrate how to effectively generate new samples for tail classes in reconstructed feature space. By adding new samples, we enrich the representing information of tail classes, to improve the classification results of long-tailed text classification. After performing some comparative experiments, we have verified that our model is an effective strategy to improve the performance of long-tailed text classification with the comparison with mainly used deep learning methods.

As mentioned above, real-world text data often follow a long-tailed distribution as the frequency of each class is typically different. For example, the classification of diseases in the auxiliary diagnosis model based on electronic medical records usually presents a long tail distribution; the distribution of different types of entities extracted from biomedical texts often shows long-tailed state. Rather than using back translation for data augmentation, this paper verifies the effectiveness of feature space reconstruction for long-tailed data classification. In the next research, we will apply the method in this paper to the construction of an auxiliary diagnosis model to verify the effectiveness of our method and whether it can improve the accuracy of disease diagnosis.

Data Availability

This study uses the Fudan University TC corpus which is from the Chinese NLP group in Department of Computer Information and Technology, Fudan University of China. All data are public and available online.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Xin Li and Lianting Hu designed the study, implemented the algorithm, preprocessed the data, and contributed equally to this paper. Long Lu, Quan Lu, and Huiying Liang gave critical suggestions. Xin Li and Tianhui Huang wrote the paper. Peixin Lu, Wei Yang, and Xin Li proofread the manuscript. Long Lu supervised the project and secured the grant funding.

Acknowledgments

This research was funded by National Natural Science Foundation of China (nos. 61936013 and 71921002), National Key R&D Program of China (2019YFC0120003), and Natural Science Foundation of Hubei Province of China (2019CFA025). The numerical calculations in this paper have been done on the supercomputing system in the Supercomputing Center of Wuhan University.

References

[1] Z. Jiang, B. Gao, Y. He, Y. Han, P. Doyle, and Q. Zhu, "Text classification using novel term weighting scheme-based improved TF-IDF for internet media reports," *Mathematical Problems in Engineering*, vol. 2021, no. 6, Article ID 6619088, 1–30 pages, 2021.

[2] N. R. Aljohani, A. Fayoumi, and S. U. Hassan, "An in-text citation classification predictive model for a scholarly search system," *Scientometrics*, p. 126, 2021.

[3] H. Y. Liang, Y. Brian, Tsui, and H. Ni, "Evaluation and accurate diagnoses of pediatric diseases using artificial intelligence," *Nature Medicine*, vol. 25, no. 3, 2019.

[4] J. S. Li, L. C. Chen, J. V. Monaco, P. Singh, and C. C. Tappert, "A comparison of classifiers and features for authorship authentication of social networking messages," *Concurrency and Computation: Practice and Experience*, vol. 29, no. 14, Article ID e3918, 2017.

[5] J. R. Méndez, T. R. Cotos-Yañez, and D. Ruano-Ordás, "A new semantic-based feature selection method for spam filtering," *Applied Soft Computing*, vol. 76, pp. 89–104, 2019.

[6] H. Lu, D. Zhan, L. Zhou, and D. He, "An improved focused crawler: using web page classification and link priority evaluation," *Mathematical Problems in Engineering*, vol. 2016, Article ID 6406901, 10 pages, 2016.

[7] T. Sabbah, A. Selamat, M. H. Selamat et al., "Modified frequency-based term weighting schemes for text classification," *Applied Soft Computing*, vol. 58, pp. 193–206, 2017.

[8] A. Mccallum and K. Nigam, "A comparison of event models for Naive Bayes text classification," in *Proceedings of the AAAI-98 Workshop on Learning for Text Categorization*, pp. 41–48, Madison WI, USA, 1998.

[9] B. Charbuty and A. Abdulazeez, "Classification based on decision tree algorithm for machine learning," *Journal of Applied Science and Technology Trends*, vol. 2, no. 1, pp. 20–28, 2021.

[10] A. M. Mesleh and G. Kanaan, "Support vector machine text classification system: using Ant Colony Optimization based feature subset selection," in *Proceedings of the 2008. ICCES 2008. International Conference on Computer Engineering & Systems*, Cairo, Egypt, November 2008.

[11] Y. Zhang, "Design of the piano score recommendation image analysis system based on the big data and convolutional neural network," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 4953288, 9 pages, 2021.

[12] Z. Tan, J. Chen, Q. Kang, M. C. Zhou, and K. Sedraoui, "Dynamic embedding projection-gated convolutional neural networks for text classification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 99, pp. 1–10, 2021.

[13] H.-Y. Lu, N. Kang, Y. Li, Q.-Y. Zhan, J.-Y. Xie, and C.-J. Wang, "Utilizing recurrent neural network for topic discovery in short text scenarios," *Intelligent Data Analysis*, vol. 23, no. 2, pp. 259–277, 2019.

[14] X. Lin, M. Zhang, and X. Wang, "Supervised learning algorithm for multilayer spiking neural networks with long-term memory spike response model," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 8592824, 16 pages, 2021.

[15] M. V. V. P. Kantipudi, S. Kumar, and A. Kumar Jha, "Scene text recognition based on bidirectional LSTM and deep neural network," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 2676780, 11 pages, 2021.

[16] B. Myagmar, J. Li, and S. Kimura, "Cross-domain sentiment classification with bidirectional contextualized transformer language models," *IEEE Access*, vol. 7, no. 99, p. 1, 2019.

[17] C. C. Aggarwal and C. Zhai, "A survey of text classification algorithms," in *Mining Text Data*, pp. 163–222, Springer, Boston, MA, USA, 2012.

[18] K. Kowsari, D. E. Brown, M. Heidarysafa, K. J. Meimandi, and L. E. Barnes, "HDLTex: hierarchical deep learning for text classification," in *Proceedings of the IEEE International Conference on Machine Learning and Applications*, Cancum, Mexico, 2017.

[19] A. Mikolajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," in *Proceedings of the International Interdisciplinary Phd Workshop*, pp. 117–122, IEEE, Poland, May 2018.

[20] M. Wang, "Construction of financial management early warning model based on improved ant colony neural network," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 6748920, 12 pages, 2021.

[21] L. Ma and T. Yang, "Construction and evaluation of intelligent medical diagnosis model based on integrated deep neural network," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 7171816, 10 pages, 2021.

[22] P. Ghasiya and K. Okamura, "Investigating COVID-19 news across four nations: a topic modeling and sentiment analysis approach," *IEEE Access*, vol. 9, pp. 36645–36656, 2021.

[23] Z. Ling and H. Dai, "Reducing performance bias for unbalanced text mining," in *Proceedings of the IEEE International Conference on Data Mining-Workshops*, IEEE, Hong Kong, China, December 2006.

[24] C. Lanera, P. Berchialla, A. Sharma, C. Minto, D. Gregori, and I. Baldi, "Screening PubMed abstracts: is class imbalance always a challenge to machine learning?" *Systematic Reviews*, vol. 8, 2019.

[25] S. Parshutin, A. Kirshners, Y. Kornijenko, V. Zabiniako, M. Gasparovica-Asite, and A. Rozkalns, "Classification with LSTM networks in user behaviour analytics with unbalanced environment," *Automatic Control and Computer Sciences*, vol. 55, no. 1, pp. 85–91, 2021.

[26] G.-X. Luo, R. Dong, Y.-H. Chen, and W.-B. Zhang, "A joint back-translation and transfer learning method for low-resource neural machine translation," *Mathematical Problems in Engineering*, vol. 2020, Article ID 6140153, 11 pages, 2020.

[27] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, no. 1, pp. 321–357, 2002.

[28] X. Xu-Ying Liu, J. Zhi-Hua Zhou, and Z. Zhou, "Exploratory undersampling for class-imbalance learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 2, pp. 539–550, 2009.

[29] T. Tang, X. Tang, and T. Yuan, "Fine-tuning BERT for multilabel sentiment analysis in unbalanced code-switching text," *IEEE Access*, vol. 8, pp. 193248–193256, 2020.

[30] C. Huang, Y. Li, C. C. Loy, and X. Tang, "Learning deep representation for imbalanced classification[C]. Computer vision & pattern recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5375–5384, Las Vegas, NV, USA, June 2016.

[31] Y.-X. Wang, D. Ramanan, and M. Hebert, "Learning to model the tail," in *Advances in Neural Information Processing Systems*, pp. 7029–7039, 2017.

[32] Z. Liu, Z. Miao, X. Zhan, J. Wang, B. Gong, and S. X. Yu, "Large-scale long-tailed recognition in an open world," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2537–2546, Long Beach, CA, USA, 2019.

[33] J. Liu, Y. Sun, C. Han, Z. Dou, and W. Li, "Deep representation learning on long-tailed data: a learnable embedding augmentation perspective," 2020, https://arxiv.org/abs/2002.10826.

[34] P. Chu, X. Bian, S. Liu, and H. Ling, "Feature space augmentation for long-tailed data," in *Proceedings of the European Conference on Computer Vision*, Glasgow, UK, August 2020.

[35] Y. Yao, "Three-way decisions with probabilistic rough sets," *Information Sciences*, vol. 180, no. 3, pp. 341–353, 2010.

[36] D. Liang and B. Yi, "Two-stage three-way enhanced technique for ensemble learning in inclusive policy text classification," *Information Sciences*, vol. 547, pp. 271–288, 2020.

[37] Y. Cheng and K. Rui, "Text classification of minimal risk with three-way decisions," *Journal of Information and Optimization Sciences*, vol. 39, no. 4, pp. 973–987, 2018.

[38] Y. Jin and H. Feng, "Chinese text classification algorithm based on Three-way Decisions," *Journal of Nanjing University(Natural ence)*, vol. 54, no. 4, pp. 794–803, 2018.

[39] G. Forman, "BNS feature scaling: an improved representation over tf-idf for svm text classification," in *Proceedings of the 17th ACM conference on Information and knowledge management*, ACM, Napa Valley, CA, USA, October 2008.

[40] S. M. Pourhashemi, A. Osareh, and B. Shadgar, "Presenting a hybrid feature selection method using Chi2 and DMNB wrapper for E-mail spam filtering," *International Journal of Computer Science & Network Solutions*, vol. 9, pp. 216–227, 2014.