IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 4, 2019, accepted May 31, 2019, date of publication June 28, 2019, date of current version September 17, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2925566

An Ensemble Learning Approach for Fault Diagnosis in Self-Organizing Heterogeneous Networks

YUEPING WANG[1], KUN ZHU[1], MENGYUN SUN[2], AND YUEYU DENG[1]

[1]College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China
[2]Huawei Company, Nanjing 210005, China

Corresponding author: Kun Zhu (zhukun@nuaa.edu.cn)

This work was supported in part by the Natural Science Foundation of China under Grant 61701230, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20170805, and in part by the Fundamental Research Funds for the Central Universities under Grant NE2018107.

ABSTRACT Self-organizing networks (SON) aim to offer high quality services while reducing both capital and operational expenditures by enabling three main functions: self-configuration, self-optimization, and self-healing. Though there exits only few studies on self-healing, it plays an important role in intelligent network maintaining. In this paper, we firstly propose an ensemble learning based fault diagnosis system for self-organizing heterogeneous networks. Specifically, the base learner is strengthened in each iteration and the final diagnosis result is obtained from the combination of all base classifications for performance improvement. Moreover, traditional classification algorithms are designed with the premise of balanced data set. However, when they are applied to fault diagnosis in cellular network which has imbalanced data, the classification accuracy for minority classes is not satisfying. To address this issue, synthetic minority over-sampling technique (SMOTE) is applied to handle the data-imbalance. Furthermore, considering the fact that misclassification is unavoidable, and most existing schemes aim to achieve a low detection error rate, while ignoring the fact that different type of misclassification errors can cause different economic losses to the operators. We consider the cost sensitivity and use rescaling method to help the classifier differentiate the importance of different samples, so that a minimal total loss can be achieved. Also, for handling the sparse data and dense deployment issues in small cells of a heterogeneous network, we provide a distributed diagnose system for lowering the communications cost. Extensive simulations are performed and the results show the effectiveness of the proposed system.

INDEX TERMS Heterogeneous wireless network, ensemble learning, fault diagnosis, imbalanced data, cost-sensitivity.

I. INTRODUCTION

With the development of cellular networks towards 5G/B5G, the challenging technical requirements (e.g., data rate and capacity) drives the network evolving to more complex structures featured by heterogeneity and dense deployment. In this case, traditional manually based schemes for network deployment, configuration, optimization, and maintenance will be in-efficient and incur huge operational and maintaining expenditures. To counter these problems, the concept of self-organizing network (SON) was proposed which aims to realize three main functions: self-configuration, self-optimization, and self-healing. Specifically, SON introduces adaptive and automatic mechanisms into cellular networks, which makes mobile network have more flexible planning and deployment, efficient optimization and maintenance, less manual intervention, lower capital expenditure and operational expenditure [1]–[3].

As an important component of SON, self-healing aims to provide fault detection, diagnosis, compensation, and even recovery in an automatic manner when the cellular service failure or degradation appear [4], [5]. In this work, we focus on the problem of fault diagnosis, which is responsible for determining the root cause of the fault from known

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

network status information. Diagnosis plays an important role in self-healing since it can not only filter the fault detection results, but also offer references for subsequent processes, e.g., fault recovery.

Note that fault diagnosis requires sufficient information to determine the possible causes of the problem [6]. The information can be collected from multiple sources and different network layers, e.g., system alarms, configuration parameters, key performance indicators (KPIs), and contextual information, etc. All of these indicators could reveal the cause in various aspects and degrees.

Though some existing work have been done for self-configuration and self-optimization [7]–[9], only few work have been done for self-healing and fault diagnosis. Specifically, in [10], the Bayesian network (BN) is firstly used in an automated diagnosis model for cellular system. While in [11], the authors propose a diagnosis system based on unsupervised Bayesian network. In [12], a Lagrangian-relaxation based self-repairing mechanism is proposed for Wi-Fi networks. Fuzzy logic controllers (FLC) are used to analyse root cause in [13] and [14]. In [15] an automatic diagnosis system based on different unsupervised techniques with Self-organization map (SOM) is proposed. In [16]–[18], the problems of cell outage are investigated in self-organizing femtocell networks. In [19], a machine learning aided context-aware outage detection scheme was proposed based on support vector data description. In [20], a mmWave UAV mesh network is considered and a beam management and self-healing scheme were proposed. In [21], deep learning is adopted for detecting the sleeping cells in next generation cellular networks. In [6], the authors state that diagnosis phase can be designed as a classifier and combination of classifiers could enhance the diagnosis performance by combining their outputs. In general, to get a good combination, the classifiers should be as accurate and diverse as possible. However, the method applied in [6] is a simple combination without considering the diversity among different classifiers. Furthermore, existing studies lack of discussions on the cost-sensitivity in fault diagnosis.

In this paper, we firstly propose to use ensemble learning for fault diagnosis exploiting the diversity of classifiers. Specifically, we first introduce a boosting algorithm to design our diagnosis system, and diverse base classifiers are generated in sequential manner where the accuracy of a base classifier has influence on the generation of subsequent classifiers.

Also, data imbalance poses a challenging issue for fault diagnosis in cellular networks. Specifically, a cellular network is functioning well during most of the running time, and only service failure or degradation appear with a relative low probability. Accordingly, the amount of normal status data overwhelms that of abnormal data, which then generates imbalanced training data. When handling imbalanced data, traditional methods usually lead to a bias towards classification to majority class. To address this problem, a data level resampling technique termed synthetic minority over-sampling technique (SMOTE) is adopted in this work to learn from practical imbalanced cellular network data sets. Note that more advanced over-sampling techniques can also be applied. In this work, SMOTE is used due to its simplicity and effectiveness for our problem.

Furthermore, we extend our previous work in [22] by considering that misclassifications are unavoidable in real implementations. However, most existing schemes aim to achieve a low detection error rate, while ignoring the fact that different types of misclassification errors can cause different economic losses to the operators. For example, it might be troublesome if a healthy cell is misclassified as a faulty one, but it would be much more serious if a problematic cell is misclassified as healthy, which will lead to a long-term performance degradation. So simply pursuing a minimum error rate cannot minimize the total loss. In this work, we further consider the cost-sensitivity. Even if two classifiers have the same error rate, the losses will differ due to the different types of misclassification errors. Therefore, it is required to propose a reasonable framework to distinguish the cost of misclassifying different samples. To this end, we use rescaling method to help the classifier differentiate the importance of different samples.

Also, note that the fault diagnosis for small cells in a heterogeneous cellular network faces the problems of sparse data and dense deployment. For handling these issues, we make further extensions and propose a distributed diagnose system for lowering the communications cost.

The main novelty and contributions of this work can be summarized as follows:

- We propose an ensemble learning approach for fault diagnosis in SON, and we consider the issue of data imbalance for which SMOTE is used to improve the diagnosis accuracy.
- To the best of our knowledge, we are the first to formulate the fault diagnosis as a cost sensitive learning problem to quantify the loss caused by different type of misclassifications. The proposed method can minimize the total loss of the classifier, which is much more practical than only maximizing the detection accuracy.
- For handling the sparse data and dense deployment issues in small cells of a heterogeneous network, we propose a distributed diagnose method for lowering the communications cost.

The rest of the paper is organized as follows. In Section II, the problem of fault diagnosis is introduced. In Section III, the preprocess of imbalanced data is introduced. In Section IV, Ada-boost based diagnosis approach is proposed. And in Section V, we further consider the cost-sensitivity. A distributed scheme for small cells is proposed in Section VI. Simulation results and analysis are described in Section VII. And the conclusions are drawn in Section VIII. Some common notations used int this paper are described in Table I.

II. PROBLEM STATEMENT

In this work, we consider the fault diagnosis in self-organizing cellular networks. As shown in Fig. 1, as a consequent step

TABLE 1. Notations.

Symbols	Definition
X	The training data set
N_+	The number of positive samples
N_-	The number of negative samples
C_+	The cost of misclassifying a positive sample
C_-	The cost of misclassifying a negative sample
C_{total}	The overall loss caused by prediction errors
C_{01}	The cost of dividing a negative sample as positive
C_{10}	The cost of dividing a positive sample as negative
P_+	Probability of positive sample occurrence
P_-	Probability of negative sample occurrence

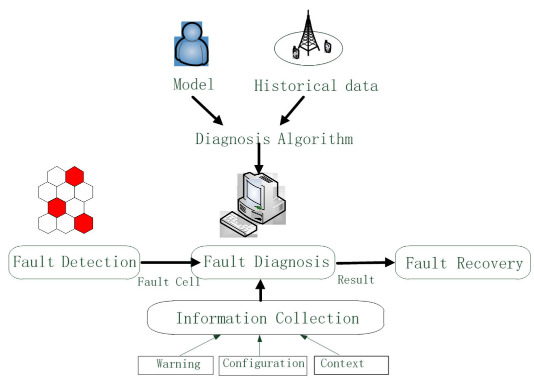


FIGURE 1. Fault diagnosis system.

of fault detection, fault diagnosis is performed based on the information collected. On the one hand, fault diagnosis is responsible for determining the root causes of the service failure. For example, the failure can be caused by hardware or software errors, configuration errors, inappropriate parameter setting, and interference, etc,. On the other hand, fault diagnosis verifies the fault detection results and exclude the detection errors, which enhances the robustness of the self-healing system.

Traditionally, modeling based fault diagnosis is adopted, which could work well for simple networks. However, it requires accurate modeling and rules establishment which may not be practical with the increasing complexity of the network. In this work, we consider data-driven fault diagnosis which focus on data processing and classifier design. Data-driven fault diagnosis includes design phase and exploitation phase. In the design phase, firstly the required information need to be collected for self-healing system. For a data-driven system, larger amount of data from multi-source would be beneficial for improving the performance. In a cellular network, following categories of information can be collected and exploited: a) configuration information: the real configuration parameters of base stations, e.g., transmit power, frequency, antenna angle; b) network layer information: the load and throughput information, handover success rate; c) physical layer information: RSRP, RSRQ, SINR; d) context information: the environment information of deployed base stations, e.g., terrain and wether conditions. After collecting the information, we can obtain the raw data which is then needs pre-procession. One of the most important step in pre-procession is data labeling, which categories a data sample into a specific class. For example, the commonly used labels are weak coverage, coverage hole, inter-system interference, and normal state. Then the pre-processed data are input into selected classification algorithms for training appropriate classifiers. And in exploitation phase, the trained model will be used to classify unknown data for fault diagnosis.

III. PREPROCESS OF IMBALANCED DATA BASED ON SMOTE

A. CLASSIFICATION IN IMBALANCED DATA

In practical cellular system, the amount of normal status data overwhelms that of abnormal data, which then generates imbalanced training data. There are majority classes and minority classes in an imbalanced data set. Most number of instances belong to majority classes. Learning from imbalanced data poses great challenges for classification algorithms, as standard classifiers will be biased towards the majority class, leading to a higher predictive accuracy over the majority class while poorer predictive accuracy over the minority class(es) [25]. Furthermore, [26] points out that classifiers trained by imbalanced data sets tend to incorrectly classify cases from minority classes as the majority class.

In literature, two main categories of methods exist for handling data imbalance, i.e., data level methods and algorithm level methods. Specifically, the main idea of data level approaches is to resample the imbalanced data set to make it balanced. There are two types of resmapling approaches, i.e., over-sampling and under-sampling. Under-sampling approaches re-balance the data through abandoning certain amount of majority class data. In contrast, over-sampling approaches usually increase the number of minority class data by reproducing to make the data set balanced. And for algorithm level approaches, the main idea is to adjust existing algorithms (e.g., adjusting the classification boundary and cost function) to improve the classification accuracy for minority instances. In this work, a data level approach termed Synthetic minority over-sampling technique (SMOTE) [24] is applied to handle the imbalanced cellular network data.

B. SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE

A most straightforward over-sampling method is random over-sampling, which increases the amount of minority class data through simple random duplication. However, it has the drawback of overfitting. To address this problem, SMOTE as an improved scheme based on random over-sampling algorithm was proposed, which generates new minority class samples by means of synthesizing two neighboring samples. And K-nearest neighbor algorithm is used to select the neighbouring samples. To a large extent, SMOTE can avoid the over-fitting problem. As shown in Fig. 2, a new minority sample can be generated by following steps. Firstly,

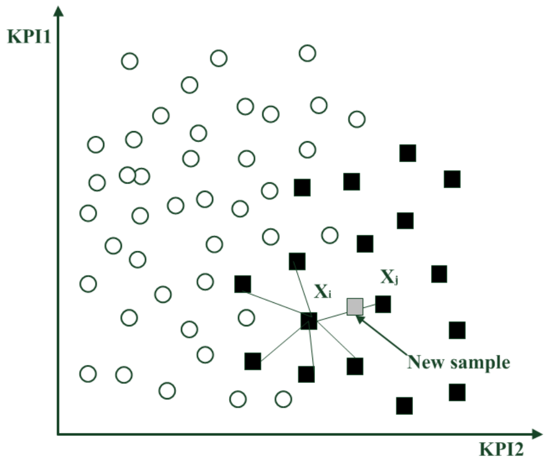


FIGURE 2. **Obtain a new sample.**

for a minority class sample X_i in M, the algorithm find the K-nearest neighbors to X_i. Then a sample X_j is randomly selected from the above neighbors, and the difference $diff = X_i - X_j$ is obtained. Finally, a new sample can be generated by the following formula:

$$X_n = X_i + rand(0, 1) * diff. \tag{1}$$

IV. FAULT DIAGNOSIS BASED ON ENSEMBLE LEARNING

Fault diagnosis is essential a classification problem. To improve the efficiency and accuracy of a classifier for fault diagnosis in cellular networks, we propose to apply ensemble learning. Specifically, AdaBoost as a famous boosting type ensemble algorithm is applied in cellular network fault diagnosis for the first time. In this section we will first introduce ensemble learning and then describe the proposed AdaBoost based fault diagnosis in detail.

A. INTRODUCTION TO ENSEMBLE LEARNING

Ensemble learning is a machine learning method that uses a series of learners to learn and integrates various results using a certain fusion rule to obtain better performance than a single learner. As shown in Fig. 3, ensemble learning trains a set of classifiers (i.e. base classifiers) and combines their classification results to achieve a better one when classifying instances in a classification problem. The generalization ability of the classifier improved by using multiple decision to jointly determine the label of each instance.

Ensemble learning methods can be divided into two types according to whether the base classifiers belongs to the same type. Heterogeneous ensemble learning uses a variety of base classifiers for integration. The two main representatives of heterogeneous ensemble learning are Stack Generalization and Meta Learning. The base classifiers are all the same type in homomorphic ensemble learning, but the parameters between these base classifiers are different. The base classifiers mainly include Naive Bayesian, Decision Tree, Neural Network, K-Nearest-Neighbor in homomorphic ensemble learning. Ensemble learning needs to meet the following two prerequisites in order to achieve a better performance: First, the error rate of each base classifier should be less than 0.5, otherwise the ensemble result will increase the error rate. Second, there should be differences between these base classifiers. If the classification results of each base classifier are similar, there would be no difference between the ensemble results and the decision made by a base classifier. Therefore, the key problem to ensemble learning is how to obtain a set of differential base classifiers and how to ensemble the results of these classifiers. There are three main ways to construct a differential classifier: (1) Differential classifiers can be trained by a series of training subsets which divided from original training set. The popular methods include bagging and boosting. (2) Differential classifiers can be obtained by selecting different feature subsets randomly from original feature set. The main feature selections include random subspace and small residual. (3) Differential classifiers can be achieved by setting different parameters in classifier model. For example, various decision trees can be trained by choosing different maximum depth. After the training of differential base classifiers is completed, each base classifier can provide only one label or a label subsets when used for the classification of each test instance. The final classification result is obtained by vote of each base classifier, mainly including simple voting, weighted voting and bayesian voting. The fault diagnosis system designed in this section constructs a differential classifier by processing the data set, and assigns a weight to each sample data in the data set. When training the base classifier, by changing this weight, each training produces a different basis and the final result is obtained by simple voting.

B. DECISION STUMP BASED ADABOOST

Decision stump, also known as one level decision tree, consists of an internal node and several terminal nodes, which are directly connected to the former. Therefore, decision stump based classifier determines the final classification result according to only one property. That is, only single feature takes effect. If there are multiple properties, we need to find the lowest error rate property as the basis for decision. Decision stump based classifier is often used as the base classifier in ensemble learning.

AdaBoost is an iterative ensemble learning algorithm, in which base classifier is generated linearly. By increasing the weight of misclassified data, it changes the distribution of the samples, guiding the classifier to focus on samples that are difficult to classify. Firstly, it assigns equal weights to all training samples $D_t = (w_1, w_2, \ldots, w_N), w_i = 1/N$, where t is t-th iteration and N is the total number of samples. From the training data set M and weight vectors D_t, the algorithm trains base classifier $h_t : X \to Y$ according to base learning algorithm. Then, it measures the error rate of h_t and acquires updated weight vectors D_{t+1} by increasing the weight of incorrectly classified samples. Next, the algorithm acquires another base classifier from the training data set and

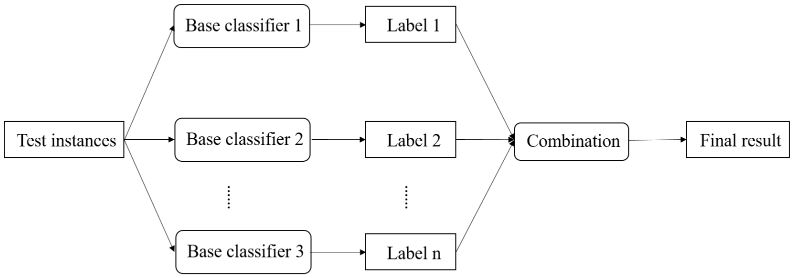


FIGURE 3. Model of ensemble learning.

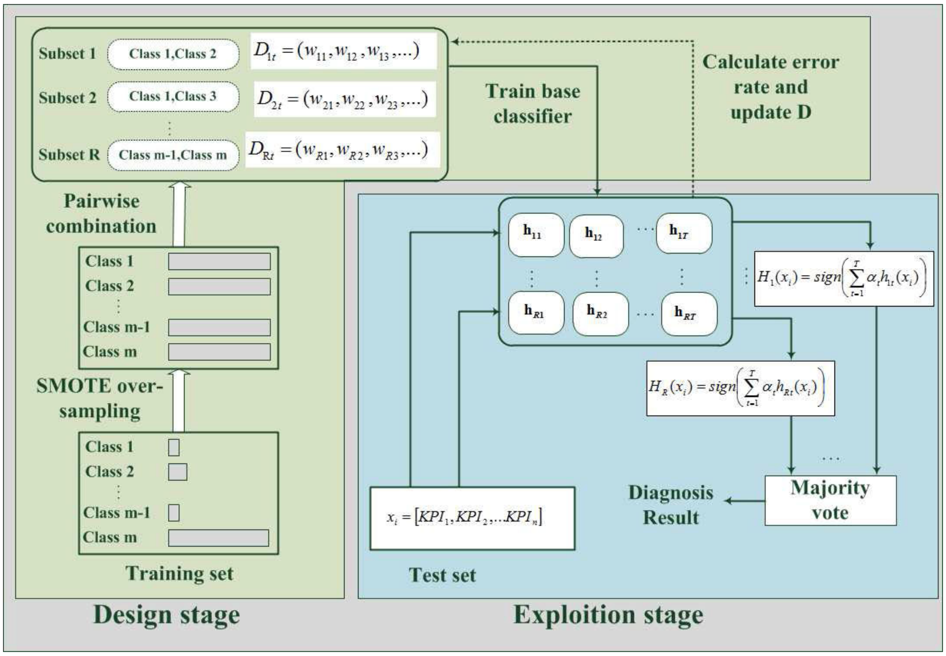


FIGURE 4. Proposed ensemble learning based fault diagnosis system.

newly obtained D_{t+1}, which is repeated for T times. The final classifier is derived by combining T base classifiers.The AdaBoost algorithm is often used to solve classification problems without suffering from overfitting.

C. PROPOSED ADABOOST BASED FAULT DIAGNOSIS SYSTEM

Note that the service outage or degradation could be caused by various factors. Some common causes include interference from neighbouring cells, inappropriate setting of transmit power, inappropriate antenna angle, handover failure etc,. We denote class i as a specific class of causes. It is worth noting that decision stump was originally designed for binary classification, while in our case the extension to multi-classification is required.

A main idea is to decompose the multi-classification problem into binary classification problems. And two main approaches exsit for such decomposition, i.e., One-Against-One (OAO) and One-Against-ALL (OAA). OAO constructs all possible pairwise, in which one class as positive class and another as negative class, while OAA selects one class to be positive, leaving all rest classes to be negative. Recent study show that OAO outperforms OAA [27], accordingly, OAO is applied in our system.

As shown in Fig. 4, the fault-diagnosis system consists of a design stage and an exploition stage. In design stage, the training set is firstly pre-processed by using SMOTE, and then is regrouped into subset of two classes, to which a weight vector is added. The base classifiers are trained with theses subsets. In exploition stage, base classifiers are used to classify each input case, and final result is obtained through combination and majority vote. In the following section, we will describe the two stages in detail.

1) DESIGN STAGE

The training data set is organized as follow:

$$X = \left[KPI_1, KPI_2, KPI_3, \ldots, Class\right], \tag{2}$$

where KPI_i and *Class* represent key performance indicators and fault causes, respectively. After data re-balancing through SMOTE, new training data set reform R subset, which includes only two classes. The number of subset is $R = m*(m-1)/2$, where m is the number of class in original train set. Then base classifier are trained from these subsets. First, a weight vector $D_{it} = (w_1, w_2, \ldots, w_N)$ is assigned to each subset, where i is the i-th subset and t refer to the t-th iteration. Moreover, w_i is initialized to $1/N$, where N is the number of samples in subset and is different for each subset. Second, base classifier h_{it} is trained according to subset i and corresponding weight vector. Lastly, the error of h_{it} is measured as ϵ_{it} and weight vector is updated as follow:

$$D_{i,t+1} = \begin{cases} \dfrac{D_{i,t}e^{\alpha_t}}{Z_t}, & h_{it}(x_i) = \text{class label}, \\ \dfrac{D_{i,t}e^{-\alpha_t}}{Z_t}, & h_{it}(x_i) \neq \text{class label}, \end{cases} \tag{3}$$

where Z_t is a normalization factor and α is calculated as follow:

$$\alpha_t = \frac{1}{2}\ln\left(\frac{1-\epsilon_{it}}{\epsilon_{it}}\right), \tag{4}$$

where ϵ_{it} is the error rate of base classifier generated in last iteration. After iterating the last two steps for T times, we obtain a base classifier matrix as follows:

$$H = \begin{bmatrix} h_{11} & h_{12} & \ldots & h_{1T} \\ h_{21} & h_{22} & \ldots & h_{2T} \\ \ldots & & & \\ h_{R1} & h_{R2} & \ldots & h_{RT} \end{bmatrix}. \tag{5}$$

Here, we complete the train of all base classifiers.

2) EXPLOITION STAGE

At exploition stage, all parameters and alarms of undiagnosed cell are organized into a *KPI* vector. Then each vector is sent to all base classifiers in matrix, and each row of the matrix aims at the same binary classification.

$$H(x_i) = \begin{bmatrix} h_{11}(x_i) & h_{12}(x_i) & \ldots & h_{1T}(x_i) \\ h_{21}(x_i) & h_{22}(x_i) & \ldots & h_{2T}(x_i) \\ \ldots & & & \\ h_{R1}(x_i) & h_{R2}(x_i) & \ldots & h_{RT}(x_i) \end{bmatrix}. \tag{6}$$

The result of a specific row in a matrix can only be one of the two classes that were originally combined. Result of each row is combined as follow:

$$H_r(x_i) = sign\left(\sum_{t=1}^{T} \alpha_t h_{rt}(x_i)\right). \tag{7}$$

For each input vector, each row of the matrix generates a diagnosis result, and the final diagnosis result is obtained according to the majority-voting rule.

V. COST-SENSITIVE FAULT DIAGNOSIS

In this section, based on previous analysis, we further consider the fact that the losses or damages which are the consequences of misclassifying different samples are different. Specifically, we propose a cost-sensitive fault diagnosis scheme based on cost-sensitive learning, and use rescaling method to help the classifier differentiate the importance of different samples.

Specifically, the state of each cell is denoted by sample $\mathbf{x}$, consisting of N features ($\mathbf{x} \in R^N$) and y for its label. The purpose of a traditional classification issue is to find a hypothesis $\phi(\mathbf{x})$ to minimize the classification error rate: $Err = E_{\mathbf{x},y}(\mathbb{I}(\phi(\mathbf{x}) \neq y))$. However, for a fault detection problem, this method is no longer appropriate, because the data set is often imbalanced, which means that the number of healthy cells will overwhelm the number problematic cells. In this case, majority class will dominate the result. For reducing the impact of majority class, two different metrics have been proposed.

A. TRITIONAL FAULT DIAGNOSIS METRICS

The confusion matrix is an effective metrics for measuring the classification results, as shown in TABLE 1, where positive samples are divide as TP and FP, denoting the correctly classified positive samples and misclassified ones, respectively, TN and FN refers to the negative samples that are correctly or incorrectly identified by the classifier. In the above definition, the positive samples represents the one that needs more attention. Here, it stands for problematic cells.

Although this method distinguishes the accuracy of the classifier on different samples, it does not quantify the importance of each sample. Meanwhile, this method is only used as a performance measure rather than a modification in learning process. Though the objective of fault diagnosis is to achieve a high detection accuracy, misclassifications are un-avoidable in real systems. Note that, different faults incur different costs and misclassifications could introduce different costs to the detection system, which makes the classification problem cost sensitive. Hence, developing a cost sensitive method to minimize the total loss appears as a pressing need.

B. PROPOSED COST-SENSITIVE METRICS

In our scheme, a cost matrix is used to reveal the losses caused by misclassifying different samples. The cost of negative samples is equivalent to the expense of one more additional fault diagnosis. The cost of a positive samples can be represented by a mean value of the losses that a problematic cell will produce over a period of time. Since the diagonal line represents that the prediction value is the same with the actual value, and a right classification will not cause any loss. So the entries on diagonal line are generally 0. Thus, a cost matrix can be constructed in Table 2.

Note that the optimal decision will not change if each entry is multiplied by the same positive constant. Therefore, we can use this rule to simplify the cost matrix. A common method

TABLE 2. Confusion matrix.

	Predict Positive	Predict Negative
Actual Positive	TP (True Posotive)	FN (False Negative)
Actual Negative	FP (False Positive)	TN (True Negative)

TABLE 3. Cost matrix.

	Predict Positive	Predict Negative
Actual Positive	0	$C(0,1) = c_{01}$
Actual Negative	$C(1,0) = c_{10}$	0

is to set the cost of negative sample to 1, that is $c_{01} = 1$. Accordingly, the cost of positive class will become $c'_{10} = c_{10}/c_{01}$. Note that the cost of misclassifying a faulty cell is always lager than a healthy cell, so c' is usually greater than 1.

It is clear that fault diagnosis problem is a cost sensitive problem in which the expected cost: $Cost = E_{\mathbf{x},y}\left(C_{y\phi(\mathbf{x})}\right)$ needs to be minimized. We observe that $E_{\mathbf{x},y}\left(C_{y\phi(\mathbf{x})}\right) = E_{\mathbf{x}}(E_{y|\mathbf{x}}C_{y\phi(\mathbf{x})}|\mathbf{x})$, then the problem of minimizing $E_{\mathbf{x},y}\left(C_{y\phi(\mathbf{x})}\right)$ can be converted to the problem of minimizing $(E_{y|\mathbf{x}}C_{y\phi(x)}$ on each $\mathbf{x}$. Based on the expected loss of predicting $\mathbf{x}$ by ϕ ($\mathbf{x}$)can be calculated as: $loss(\mathbf{x}, \phi(\mathbf{x})) = E_{y|\mathbf{x}}(C_{y\phi(\mathbf{x})})$ [9]. In this work, we have

$$loss(x, \phi(\mathbf{x})) = \begin{cases} P(1|\mathbf{x})C_{10} & if\, \phi(\mathbf{x}) = 0, \\ P(0|\mathbf{x})C_{01} & if\, \phi(\mathbf{x}) = 1, \end{cases} \quad (8)$$

For minimizing the total cost, we can find the optimal prediction of x by solving

$$\phi^*(\mathbf{x}) = \underset{\phi(\mathbf{x})\in\{1,0\}}{argmin} \quad loss(\mathbf{x}, \phi(\mathbf{x})). \quad (9)$$

The proposed cost-sensitive fault diagnosis system is to solve the above minimization problem. In the next subsection, a detailed description of this method is given.

C. FAULT DIAGNOSIS BASED ON COST-SENSITIVE LEARNING

The cost-sensitive method for fault diagnosis proposed in this work is mainly composed of two phases: a training phase and a validation phase (see Fig. 5). And the data set is divided into two sets: a training set and a validation set. In addition, the positive and negative samples in each dataset follow the same distribution. Since the number of healthy cells in the fault diagnosis phase overwhelm the faulty cells in real applications, the two data sets are unbalanced. The first step of training phase is pre-processing. In this step, we standardize the mean of all data to 0, and the variance to 1.

The next step is rescaling the training set $\mathbf{X}$, So that each sample $\mathbf{x}_i \in \mathbf{X}$ can get a appropriate weight w_i. This method proved to be effective in terms of data imbalance and unequal misclassification costs. It is also one of the most convenient way to implement cost-sensitive learning, and can be easily applied to various existing algorithms (e.g., decision tree,

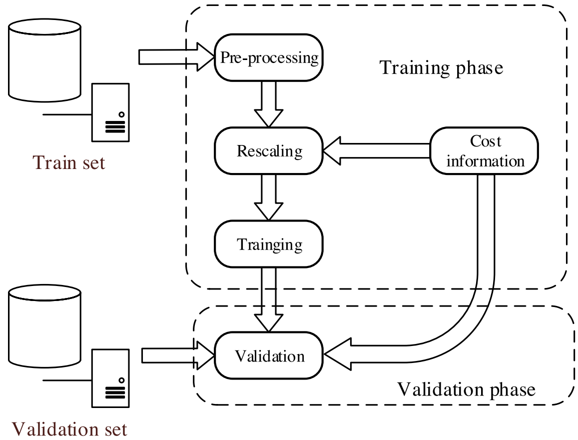


FIGURE 5. Cost-sensitive fault diagnosis.

support vector machine, and neural networks) without any modification on the original algorithm.

An practical investigation on the influence of data imbalance on cost-sensitive learning reveals that when data imbalance and unequal misclassification cost occurs simultaneously, a balance rescale ratio is required first [29], as shown in Eq. 10

$$ri_{+,-} = \frac{N_-}{N_+}, \quad (10)$$

where N_+ and N_- represents the number of positive and negative samples, respectively.

Though in [30], Ciraco argued that altering the sample distribution does not mean changing the sample cost, it has been proved equivalent when we use Rescaling [29]. Therefore, we can use the rescaling method to make the weight of the sample corresponding to their cost ratio. In this work, we assume that the balance stage is not suitable for all situations. This method is effective only when the occurrence probability is the same. If probability of each class is unequal, it is obviously that the high probability class should have a higher priority when making judgement. Therefore, after the balance the class, a weight based on the class probability is needed. The modified balance ratio is show in Eq.11 as

$$ri'_{+,-} = \frac{N_-}{N_+} \cdot \frac{P_+}{P_-}, \quad (11)$$

where P_+ and P_- represent the probability of positive and negative sample occurrence in practical applications, respectively. Note that these two balance ratio are equivalent when the occurrence probability is the same.

In the traditional rescaling problem, each class will have the same distribution and occurrence probability, and the optimal rescale ratio is correspond to their cost ratio, as shown in Eq. 12

$$rc_{+,-} = \frac{C_+}{C_-}, \quad (12)$$

After the above balance rescaling, we assume that the samples have reached the optimal state and can be directly applied the optimal rescale ratio. A natural way is to combine the balance rescale ratio and optimal rescale ratio into a single ratio r, as shown in Eq. 13

$$r_{+,-} = \frac{C_+}{C_-} \cdot \frac{N_-}{N_+} \cdot \frac{P_+}{P_-}, \tag{13}$$

where C_+ and C_- represent the cost of misclassifying a positive and negative sample, respectively.

At this point, the processed training set will be used to train the classifier. In this work, a soft margin support vector machine (SVM) is used to find a hyperplane $h(x)$ that linearly separates negative and positive samples. Since the cost information has been added to the training set through rescaling, the SVM can be directly learning from the rescaled samples without any modification. The hyperplane can be derived through an optimization problem as follows:

$$\begin{aligned} &\min_{w,b,\xi} \quad \frac{1}{2}||w||^2 + C\sum_{i=1}^{N}\xi_i. \\ &s.t. \quad y(w \cdot \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i. \end{aligned} \tag{14}$$

where ξ_i represents the distance from the sample x_i to the hyperplane, and the hyperplane is defined by weight vector w together with threshold b. C is the penalty coefficient for misclassification.

Finally, in the validation phase, the actual performance of the classifier will be further verified. According to the results, some parameters (e.g., kernel of the SVM, penalty coefficient) in this method will be further optimized.

In this work, the cost information needs to be provided by the user. Although in some special cases, the user may not know the damage caused by a misclassification, only a rough range is available. Such problems can also be solved by converting the cost interval to the mean or the maximum and minimum values. Therefore, in this work, we only consider the cases where the accurate cost information is already given.

VI. DISTRIBUTED DIAGNOSIS SYSTEM FOR SMALL CELLS

A. ISSUES FOR FAULT DIAGNOSIS IN SMALL CELLS

As a supplement and extension of macrocells, small cells can be deployed in blind spot areas to increase network coverage. On the other hand, they can be deployed in hotspots to increase system capacity. In a single area, small cells can also be deployed dynamically in a plug-and-play manner as traffic increases. Though the deployment in micro-cells increases the complexity of the network, the difficulty of operation and management, and the economic cost, the resource utilization and service quality can be significantly enhanced. Therefore, the use of small cells is an inevitable requirement for network development. Compared with macrocells, small cells have the characteristics of small coverage, low transmission power, and convenience and flexibility. It is these characteristics that determine the fault diagnosis under small cells is far different from macrocells. Specifically, there are following two issues.

1) SPARSE USER DATA IN EACH CELL

In general, the coverage of macro cells can reach several kilometers or even tens of kilometers, while the coverage of small cells may be only tens of meters. The number of small cell service users is far less than that of macro cells. This would lead to the following two results:

(1) It is difficult to obtain statistical performance data. The number of users in a single cell decreases sharply, and user data acquired by cellular networks is scarce. The KPIs based on statistical results, such as the average throughput often used in macrocells and handover success rate, are difficult to obtain. This is mainly due to the fact that it is difficult for these KPIs to reach a certain number of statistics in a small cell network. Even if the time span is increased to meet the requirements, it cannot reflect the real-time status of the network and then cannot be used as the basis for fault diagnosis. In addition, in a normally working network, the number of faulty cells is much smaller than that of normal cells. That is, the above-mentioned cellular failure data is much smaller than normal data, which further increases the difficulty in obtaining useful KPI data in the small cell networks.

(2) The algorithms that require higher data volume are not applicable. Regardless of macrocells or small cells, the probability of occurrence of a fault condition is much lower than that of the normal condition. Accordingly, the faulty cellular data is much lower than normal cellular data. In a small cell, this phenomenon is more obvious. In the existing macro-cellular diagnostic system, most of the diagnostics are based on machine learning classification algorithms. The training of the classification model requires a certain amount of training data sets. For some of the algorithms, the performance is highly related to the size of the training set. The small number of users in a small cell and the low probability of failure of a single cell make these algorithms that require a certain number of training sets no longer applicable. Based on the above two reasons, fault diagnosis of small cells needs to find a suitable classification algorithm. The algorithm must maintain high classification accuracy when the training set is limited. At the same time, it can be based on real-time data rather than statistical data reflecting overall performance to classify datasets with fewer attribute attributes. In this paper, we propose to use SVM algorithm to classify small sample data and implement fault diagnosis. The SVM algorithm determines the classifying hyperplane according to the support vector. If the data is small, it still selects the higher performance. Specific description and related The solution is given in the subsequent sections of this section.

2) DENSE DEPLOYMENT OF SMALL CELLS

Another significant feature of small cell is the high deployment density. In a macrocell coverage area, hundreds or thousands of small cells may be deployed. In crowded areas, multiple small cells may even overlap. For example, under the coverage of one macro cell, multiple microcells are clustered together. The subscribers connected to a small cell base station are both interfered by the neighboring small cells and the macrocells in the area at the same time. The interference could be strong due to the small distance between dense small cells. This makes the interference problem particularly prominent in hierarchical heterogeneous networks. And interference should be more considered for fault diagnosis in small cells.

Also, the diagnosis architecture needs to be changed due to the dense deployment. In single-tier networks where only macrocells exist, fault diagnosis is mostly based on the data stored in the OM center. These data are processed and integrated to form KPIs that reflect network performance. Although the centralized fault diagnosis does not need to consider the problem of sparse user data, it can also make the best judgment according to the overall situation of the network, but this centralized fault diagnosis based on the OM center is no longer applicable in small cell networks. On the one hand, the number of small cells in a hierarchical heterogeneous network is very large. The centralized self-healing function requires that all small cells regularly report information to the OM centers. This will bring huge communication costs and the OM system will also be overloaded due to the management of large number of nodes, and accordingly the monitoring and management functions cannot be realized. On the other hand, since the self-healing function requires high real-time performance, the communication between the OM system and the small cell base station will cause delays. Excessive communication overheads will overload the network and cause higher delays, which will seriously affect the real-time fault diagnosis performance.

Obviously, distributed fault diagnosis without excessive communication overhead is more suitable for fault diagnosis in small cell networks. One way is to design fault diagnosis system based on a single small cell. This method deploys fault diagnosis functions on each small base station. The other way is to deploy faulty diagnosis functions on macrocell base stations. And each MBS is responsible for fault diagnosis of all small cells within the coverage of the macrocell. The former method reduces the flexibility of the small base station, complicates the function of the small base station. And when the base station itself fails, the system could lose the fault diagnosis function. The latter method aggregates the data of the small cells into macrocell base station instead of OM centers, which reduces the communication overheads and the burden of the operation and management center. In addition, the aggregation of data of multiple small cells alleviates the sparse user data problem to some extent. This approach is the fault diagnosis framework adopted in this paper.

B. DESIGN OF DISTRIBUTED FAULT DIAGNOSIS SYSTEM

1) DATA SOURCE AND FAULT TYPES

In centralized fault diagnosis system, the information used for centralized fault diagnosis includes configuration parameters, alarm information, network count information, performance statistics, etc. These information are all from the OM center of the cellular network which are further processed and integrated into KPIs for fault diagnosis. While in distributed fault diagnosis systems, these offline data in OM center are difficult to obtain since distributed fault diagnosis does not bring all the information together. Instead, in this paper, we use user equipment measurement (UE measurement) as the main source of data for fault diagnosis. Note that UE measurements and KPIs are widely used in coverage optimization, load balancing, fault detection, and energy conservation in self-organizing networks. In addition, since the coverage of a small cell is small, a user can quickly move from the center area of a small cell to the edge area. In this case, the UE measurement value changes greatly which could result in classification errors, and thus affecting the final fault diagnosis results. The literature [31] has proven that it is more obvious to distinguish the fault cases by adding user location information. Therefore, the user location information as an important parameter should be included for fault diagnosis in small cells.

The process of obtaining fault diagnosis information from the UE is described as follows. In the normal state, UE measurement values are continuously collected to the small cell, which are then transmitted to the macrocell through a logical link. In the case of a fault state in the small cell, the radio link between the user terminal and the small cell is disconnected, the user terminal switches from the current faulty cell to the neighboring cell, and the UE sends the faulty cellular information that has been recorded while not yet reported to the neighboring cell to the macro cell. In this way, the macro cell can timely obtains relevant faulty cellular information and achieves real-time diagnosis. With the UE information collected in this way, the delay mainly comes from the handover from the failed cell to the neighbor cell, and it can reflect the network state in real time.

In small cell networks, the deployment of small cells is dense. This dense deployment makes the coverage of small cells overlap and interferes with each other. In this work, the design of this system considers three typical fault conditions: 1) High interference. The coexistence of macro cells and small cells causes interference to small cell users from both small cells and macrocells at the same time. With dense deployment of microcells, overlapping coverage areas create more interference, resulting in ultra-high interference which will affect the communications. 2) Low signal strength. In indoor environment, the presence of obstacles such as walls causes the signal to attenuate faster, and the signal strength received by the user is weak, which affects user communications. 3) Lost connections. A small cell stops transmitting signals if there is no power supply or the backhaul link is down, and users in the area can only switch to the distant neighbor cells or distant macro cells to maintain communications. In order to diagnose the above-mentioned failures, we need to collect the relevant information of the cell, which mainly include RSRP, SINR and distance.

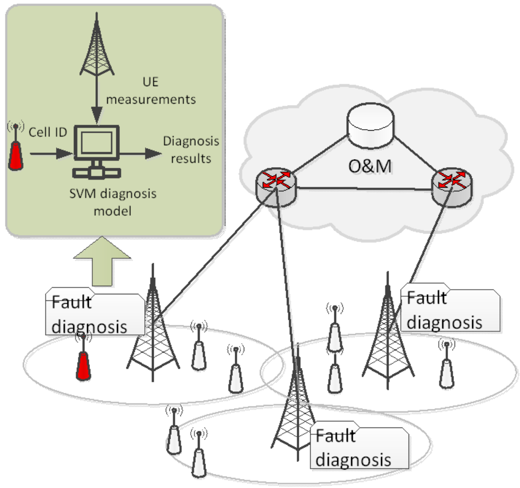
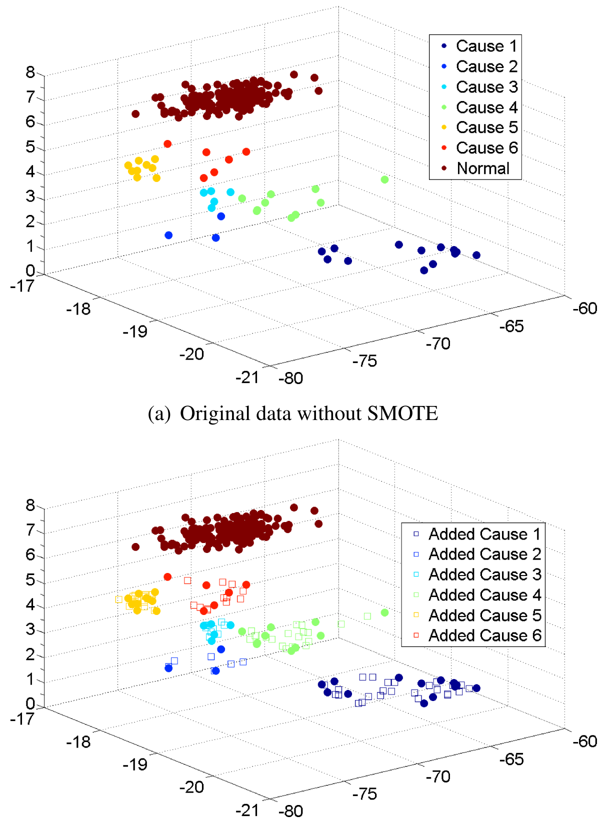


FIGURE 6. Distributed diagnosis model for small cells.



(a) Original data without SMOTE

(b) New data added by SMOTE

FIGURE 7. Synthetic Minority Over-sampling.

2) DISTRIBUTED SYSTEM DESIGN

Different from centralized fault diagnosis, the distributed fault diagnosis function mainly runs on each macrocell base station. When the fault detection function discovers that a cell is faulty, the fault diagnosis system collects related information according to the ID of the cell and sends the information into the already trained SVM fault diagnosis model. Then the model classifies the data, and determines the cause of the failure according to the classification result. As shown in Figure 6, a small cell aggregates the information into the macro cell and then the macro cell aggregates the relevant information to the OM center through the backhaul link. The small cell fault diagnosis runs on the macro cell base station and only reports the fault diagnosis result to the OM center to provide references for subsequent self-healing steps. Although this architecture increases the difficulty and complexity of macrocell base station deployment to a certain extent, it reduces the communication overheads from the perspective of the overall system and has high feasibility.

VII. PERFORMANCE EVALUATIONS

In this part, we evaluate the performance of the proposed schemes. Specifically, we first evaluate the effectiveness of the proposed ensemble learning fault diagnosis, and the superiority of is demonstrated by comparing with other classification algorithms. Then the cost-sensitive and distributed fault diagnosis schemes are evaluated.

A. SYSTEM SETTINGS

Note that the data set is obtained from [24], which is the collection of real data from a cellular network operated by a Spanish mobile operator. For the data set, the main features used in training process are retainability, handover success rate (HOSR), received signal receiving power (RSRP), reveived signal receiving quality (RSRQ), signal to interference ratio (SINR), average throughput, and distance. In addition to these features, each training sample is accompanied by a label that indicates the fault cause of each cell. There are 7 kinds of labels used, which are Antenna Downtilt, Coverage Hole, Inter-System Interference, Overshoot Coverage, Weak Coverage and normal.

To indicate the number of added samples by SMOTE, we define the over-sampling rate as

$$R_o^i = \frac{N_{add}^i}{N_{original}^i}, \tag{15}$$

where N_{add}^i refers to the number of added samples of particular class i and $N_{original}^i$ represents original sample number of the same class. Therefore, the total number of samples in new train set is expressed as

$$\begin{aligned} N_{total} &= \sum_{i=1}^{m} \left(N_{add}^i + N_{original}^i\right) \quad (16)\\ &= \sum_{i=1}^{m} \left(R_o^i + 1\right) * N_{original}^i. \quad (17) \end{aligned}$$

Taking randomly selected 200 samples and $R_o^i = 200\%(i = 1, 2, \ldots 6)$ as an example, the three-dimensional scatter plot of original train set is displayed in Fig. 7a and the situation of new train set added by SMOTE is shown in Fig. 7b.

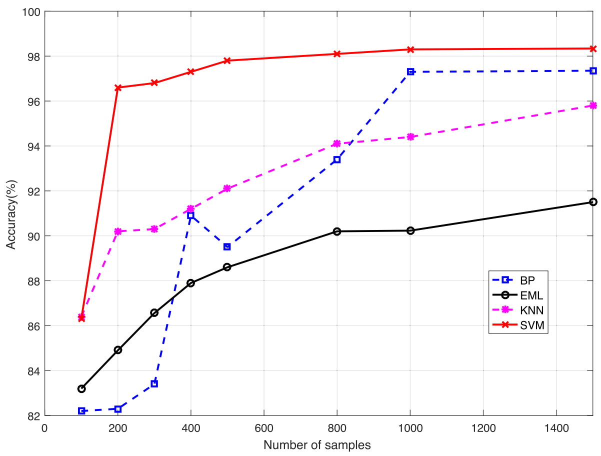


FIGURE 8. Diagnosis accuracy with different number of train samples.

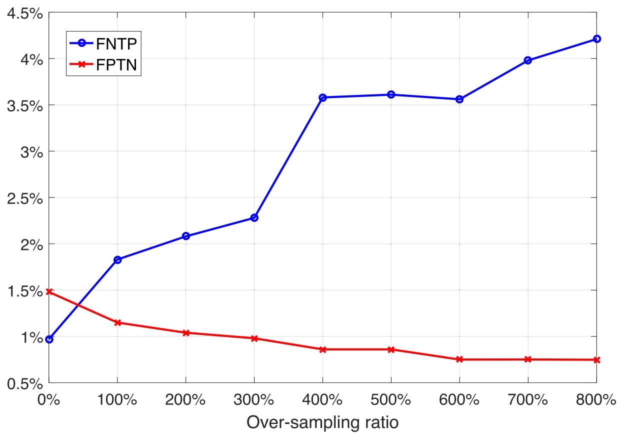


FIGURE 9. FNTP and FPTN at different over-sampling ratios.

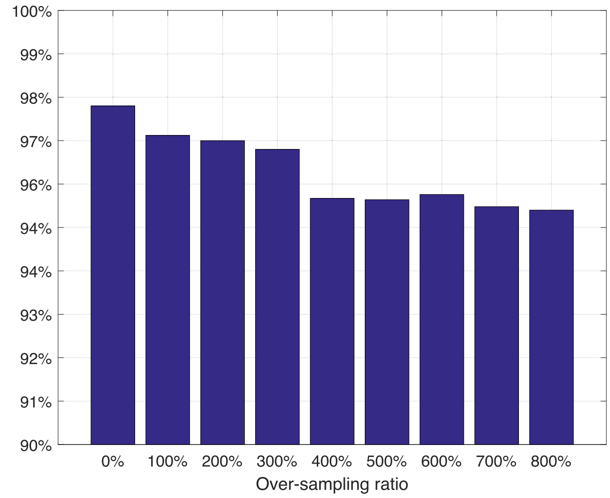


FIGURE 10. Accuracy at different over-sampling ratios.

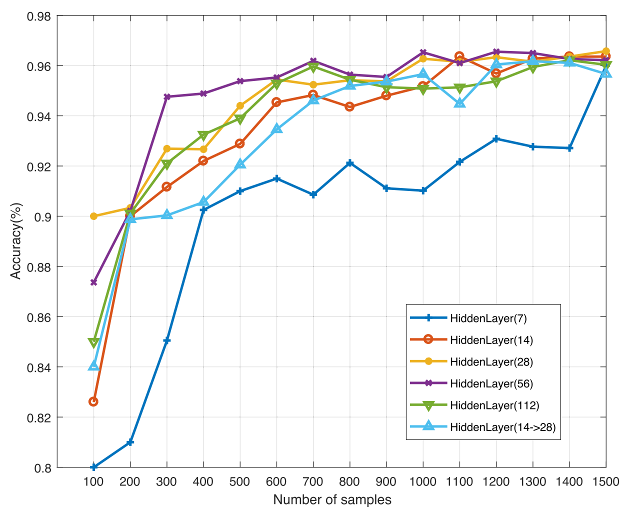


FIGURE 11. Diagnosis accuracy with deep learning approach under different parameter settings.

B. PERFORMANCE OF THE PROPOSED ENSEMBLE LEARNING FRAMEWORK WITH SMOTE

We evaluate the performance of the proposed scheme under different number of training samples. Also, we compare our shceme with a Support Vector Machine (SVM) method, a K-Nearest-Neighbor (KNN) method, and a feed-forward Back propagation Neural Network (BP) method. The results are shown in Fig. 8, where EML refers to the proposed Adaboost based Ensemble learning algorithm. Firstly, we observe that the accuracy of all four algorithms can be improved with the increase of training samples. And the proposed EML algorithm outperforms the other three algorithms in terms of higher accuracy on varying number of samples. Moreover, the results also show that the proposed algorithm could train more accurate classifier with a small number of training samples.

We also evaluate the performance of the proposed scheme considering different over-sampling ratios. The results are shown in Fig. 9. We can observe that $FNTP$ decreases with the increase of over-sampling ratio, while $FPTN$ increases. Also, when the over-sampling ratio is greater than 300%, we can observe a significant increase on $FPTN$, while $FNTP$ only decreases slightly. Considering the tradeoff $FNTP$ and $FPTN$, the optimal over-sampling ratios are different for different numbers of samples.

Finally, we investigate the impact of over-sampling ratio on the diagnosis accuracy. The results are shown in shown in Fig. 10. It is worth pointing that within certain regions, the different choices of over-sampling ratios do not significantly impact the diagnosis accuracy. It indicates that SMOTE improves the classification accuracy of the minority class, and will not significantly reduce the overall accuracy of the system.

For comparison purpose, we also apply deep learning in our work. The result is shown in Fig. 11. We can see that the result heavily relies on the parameter selections. We set different number of neurons in hidden layers. The performance with the choices of 14, 28, 56 are better than the choice of 7, 56, etc. Also, for some choices the results are not as good as the proposed method. The reason is that deep learning is usually suitable for more complicated tasks with huge amount of training data. While in this work, the benefits of deep learning are not realized.

C. EVALUATION OF COST-SENSITIVE LEARNING

To verify the effectiveness of cost-sensitive learning in helping traditional classifiers reducing the total cost, we test the influence of different rescaling ratio on SVM. In this test, the only variable is the weight of the each sample. The training set and validation set used in each experiment are the same. Since the cost matrix is a constant matrix related to the actual application scenario, we set the cost information c'_{01} to 4, 5, 6, 7 for comparison experiments.

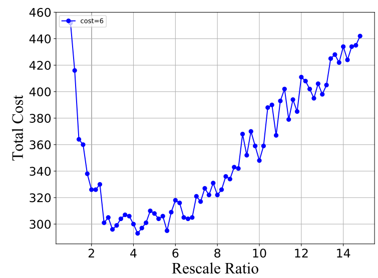
The results of the simulation are shown in Fig. 12, where we can visually see the impact of rescaling on SVM. When the rescaling ratio increases, the total cost shows a downward trend and then begin to rise after reach the optimal solution. Note that the location of the optimal solution is also related to the cost information. When the cost of misclassifying a positive sample increases, the optimal solution will shift to the right. However, it could be found that this optimal solution becomes less obvious when the cost of positive samples becomes expensive. This is because when the positive samples are too expensive, the classifier will try to classify more ambiguous samples as positive to reduce the total cost. When most of the fuzzy samples are classified as positive, the overall cost tends to be saturated. Therefore, cost-sensitive learning is more effective when the cost of a positive sample is not too high.

We also compare our proposed method with cost-blind methods, decision tree (DT) and k-Nearest Neighbors (KNN). The cost information is used as a variable to test the performance of each method at different costs. In order to make the results more comparable, the total cost has been normalized, as shown in Eq.8. where $n_{predict}$ indicates the number of the samples that need to be predicted, $C(x)$ represents the loss caused by misclassifying sample x

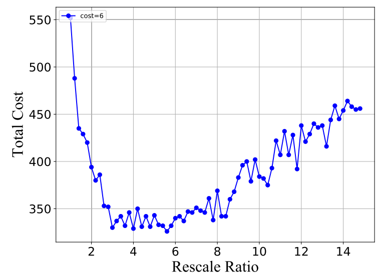
$$C_{total} = \frac{\sum C(\mathbf{x}) \cdot sign(y - f(\mathbf{x}))}{n_{predict}}. \tag{18}$$

This experiment shows the superiority of cost-sensitive learning, as shown in Fig. 13. Obviously, we can see that the total cost will raise if cost ratio increases. Among these cost-blind algorithms, SVM has the best performance, and this is why we choose it to add cost sensitivity. Note that the raise of cost blind methods is almost linear. This is because it does not make any change to adapt the different cost ratios. When the cost ratio is not high, there is no significant difference between these methods. However, when the cost ratio becomes higher, proposed cost-sensitive methods can significantly mitigate the increase in total cost.

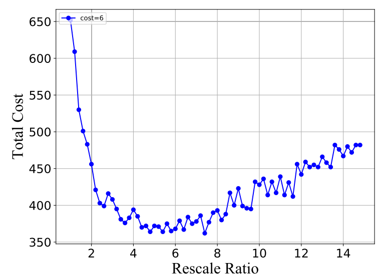
Finally, it can be found that the proposed cost-sensitive method is outperform than other cost-blind method, and more suitable for practical applications.



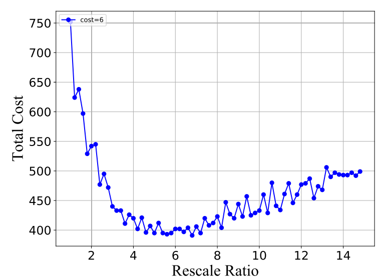
(a) c'_{01}=4



(b) c'_{01}=5



(c) c'_{01}=6



(d) c'_{01}=7

FIGURE 12. Performance tendency.

D. EVALUATION OF DISTRIBUTED SCHEME

For distributed diagnosis system. In order to obtain better diagnostic results, we compare SVM models based on trainings of different kernel functions, and select the most accurate model as the final fault diagnosis model. For finding the most suitable kernel function for diagnosis, we use different numbers of training samples to perform training based on three different kernel functions. As shown in Fig. 14, we compare the fault diagnosis accuracy of the linear kernel function, the polynomial kernel function, and the Gaussian kernel function under different sample sizes. Obviously,

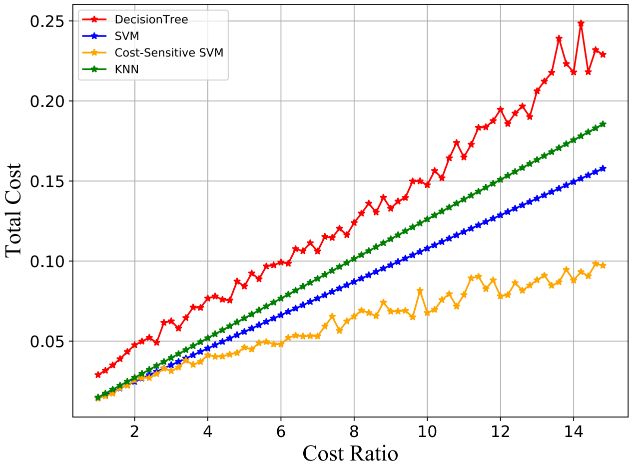


FIGURE 13. Fault diagnosis methods in different cost ratios.

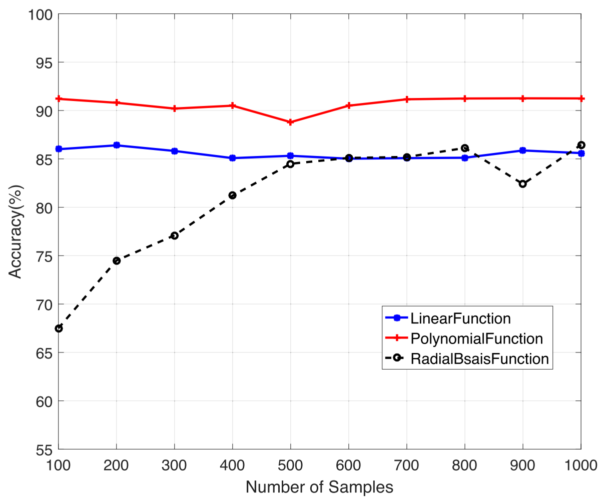


FIGURE 14. Accuracy with different kernel functions.

the polynomial kernel function has superior performance under different numbers of samples. While the performance improvement of the linear kernel function and the polynomial kernel function is not obvious with the increase of samples, which also proves that the SVM classification algorithm is insensitive to the number of samples. However, the performance of the classification model based on Gaussian kernel function fluctuates with the change of the sample. Since the classification model based on Gaussian kernel function is affected by the parameters, the classification model based on Gaussian kernel function is performed in this paper.

For the SVM classification model based on the Gaussian kernel functions, the important parameters are the penalty parameter and the nuclear parameter. Since there is no prior knowledge on the parameter selection, it is necessary to search within a certain range to find a value with higher classification accuracy. Specifically, the scope of the penalty parameter is set to be $2^{-8} \leq C \leq 2^8$ and the scope of the nuclear parameter is set to be $2^{-8} \leq \gamma \leq 2^8$. Fig. 15 shows the process for finding the optimal parameters g and c.

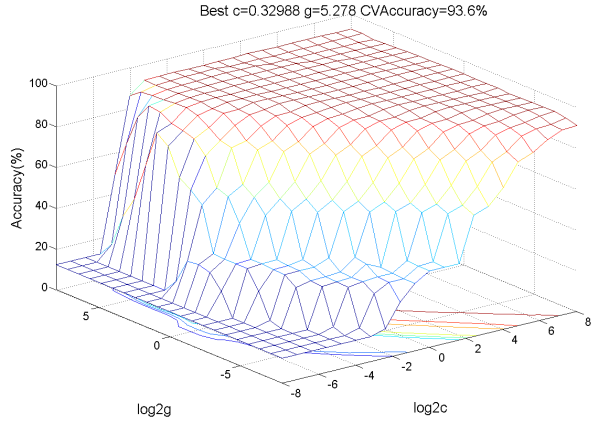


FIGURE 15. Accuracy with different parameters.

For each combination of g and c, we performed a simulation to test the classification accuracy. As shown in Figure 15, when the number of training set samples is 500, it can be seen that the best classification accuracy of 93.6 can be achieved.

VIII. CONCLUSION

In this paper, we have considered the self-healing in cellular networks and firstly proposed an ensemble learning based fault diagnosis system. And the imbalanced practical cellular network data set is preprocessed using SMOTE. In order to achieve higher accuracy, traditional classification methods are abandoned and ensemble of base classifiers is adopted. Also, we consider different costs for different misclassifications and have proposed a cost-sensitive scheme and a minimal total cost of misdiagnosis can be achieved. Furthermore, considering the sparse data and dense deployment for small cells in heterogeneous networks, we provide a distributed fault diagnosis system which significantly reduces the communications cost, which is much more practical than only maximizing the detection accuracy. Through extensive simulations and comparisons with several existing methods, we have shown the effectiveness of our proposed schemes.

REFERENCES

[1] O. G. Aliu, A. Imran, M. A. Imran, and B. Evans, "A survey of self organisation in future cellular networks," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 336–361, 2nd Quart., 2013.
[2] R. Barco, P. Lazaro, and P. Munoz, "A unified framework for self-healing in wireless networks," *IEEE Commun. Mag.*, vol. 50, no. 12, pp. 134–142, Dec. 2012.
[3] P. Szilagyi and S. Novaczki, "An automatic detection and diagnosis framework for mobile communication systems," *IEEE Trans. Netw. Service Manage.*, vol. 9, no. 2, pp. 184–197, Jun. 2012.
[4] A. Asghar, H. Farooq, and A. Imran, "Self-healing in emerging cellular networks: Review, challenges, and research directions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 1682–1709, 3rd Quart., 2018.
[5] D. Palacios, S. Fortes, I. de-la-Bandera, and R. Barco, "Self-healing framework for next-generation networks through dimensionality reduction," *IEEE Commun. Mag.*, vol. 56, no. 7, pp. 170–176, Jul. 2018.
[6] D. Palacios, E. J. Khatib, and R. Barco, "Combination of multiple diagnosis systems in self-healing networks," *Expert Syst. Appl.*, vol. 64, pp. 56–68, Dec. 2016.
[7] J. Wu, B. Cheng, M. Wang, and J. Chen, "Quality-aware energy optimization in wireless video communication with multipath TCP," *IEEE/ACM Trans. Netw.*, vol. 25, no. 5, pp. 2701–2718, Oct. 2017.

[8] J. Wu, B. Cheng, M. Wang, and J. Chen, "Energy-efficient bandwidth aggregation for delay-constrained video over heterogeneous wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 1, pp. 30–49, Jan. 2017.

[9] J. Wu, B. Cheng, and M. Wang, "Improving multipath video transmission with raptor codes in heterogeneous wireless networks," *IEEE Trans. Multimedia*, vol. 20, no. 2, pp. 457–472, Feb. 2018.

[10] R. M. Khanafer, B. Solana, J. Triola, R. Barco, L. Moltsen, Z. Altman, and P. Lazaro, "Automated diagnosis for UMTS networks using Bayesian network approach," *IEEE Trans. Veh. Technol.*, vol. 57, no. 4, pp. 2451–2461, Jul. 2008.

[11] L. Flores-Martos, A. Gomez-Andrades, R. Barco, and I. Serrano, "Unsupervised system for diagnosis in LTE networks using Bayesian networks," in *Proc. IEEE Veh. Technol. Conf. (VTC Spring)*, May 2015, pp. 1–5.

[12] M.-C. Tsai, F. Y.-S. Lin, and Y.-F. Wen, "Lagrangian-relaxation-based self-repairing mechanism for Wi-Fi networks," *IEEE Access*, vol. 7, no. 2, pp. 15868–15883, 2019.

[13] E. J. Khatib, R. Barco, A. Gómez-Andrades, P. Muñoz, and I. Serrano, "Data mining for fuzzy diagnosis systems in LTE networks," *Expert Syst. Appl.*, vol. 42, no. 21, pp. 7549–7559, Jun. 2015.

[14] E. J. Khatib, R. Barco, A. Gómez-Andrades, and I. Serrano, "Diagnosis based on genetic fuzzy algorithms for LTE self-healing," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1639–1651, Mar. 2016.

[15] A. Gómez-Andrades, P. Muñoz, I. Serrano, and R. Barco, "Automatic root cause analysis for LTE networks based on unsupervised techniques," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 2369–2386, Apr. 2016.

[16] W. Wang, J. Zhang, and Q. Zhang, "Cooperative cell outage detection in Self-Organizing femtocell networks," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 782–790.

[17] W. Wang, Q. Liao, and Q. Zhang, "COD: A cooperative cell outage detection architecture for self-organizing femtocell networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 11, pp. 6007–6014, Nov. 2014.

[18] I. de-la-Bandera, R. Barco, P. Muñoz, and I. Serrano, "Cell outage detection based on handover statistics," *IEEE Commun. Lett.*, vol. 19, no. 7, pp. 1189–1192, Jul. 2015.

[19] M. Qin, Q. Yang, N. Cheng, H. Zhou, R. R. Rao, and X. Shen, "Machine learning aided context-aware self-healing management for ultra dense networks with QoS provisions," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 12339–12351, Dec. 2018.

[20] P. Zhou, X. Fang, Y. Fang, R. He, Y. Long, and G. Huang, "Beam management and self-healing for mmWave UAV mesh networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1718–1732, Feb. 2019.

[21] U. Masood, A. Asghar, A. Imran, and A. N. Mian, "Deep learning based detection of sleeping cells in next generation cellular networks," in *Proc. IEEE GLOBECOM*, Dec. 2018, pp. 206–212.

[22] M. Sun, H. Qian, K. Zhu, D. Guan, and R. Wang, "Ensemble learning and SMOTE based fault diagnosis system in self-organizing cellular networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2017, pp. 1–6.

[23] W. Wang and Q. Zhang, "Local cooperation architecture for self-healing femtocell networks," *IEEE Wireless Commun.*, vol. 21, no. 2, pp. 42–49, Apr. 2014.

[24] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 1, pp. 321–357, 2002.

[25] N. V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "SMOTEBoost: Improving prediction of the minority class in boosting," in *Knowledge Discovery in Databases: PKDD* (Lecture Notes in Computer Science), vol. 28, no. 3. Berlin, Germany: Springer, 2003, pp. 107–119.

[26] A. Agrawal, H. L. Viktor, and E. Paquet, "SCUT: Multi-class imbalanced data classification using SMOTE and cluster-based undersampling," in *Proc. 7th Int. Joint Conf. Knowl. Discovery, Knowl. Eng. Knowl. Manage. (ICK)*, Nov. 2015, pp. 226–234.

[27] M. Galar, A. Fernández, E. Barrenechea, and F. Herrera, "DRCW-OVO: Distance-based relative competence weighting combination for one-vs-one strategy in multi-class problems," *Pattern Recognit.*, vol. 48, no. 1, pp. 28–42, 2015.

[28] A. Gómez-Andrades, P. Muñoz, E. J. Khatib, I. de-la-Bandera, I. Serrano, and R. Barco, "Methodology for the design and evaluation of self-healing LTE networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 8, pp. 6468–6486, Aug. 2016.

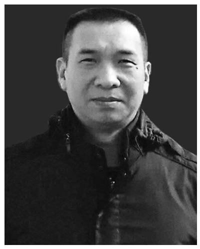
[29] X.-Y. Liu and Z.-H. Zhou, "The influence of class imbalance on cost-sensitive learning: An empirical study," in *Proc. 6th IEEE Int. Conf. Data Mining*, Dec. 2006, pp. 970–974.

[30] M. Ciraco, M. Rogalewski, and G. Weiss, "Improving classifier utility by altering the misclassification cost ratio," in *Proc. 1st Int. Workshop Utility-Based Data Mining*, 2005, pp. 46–52.

[31] P. Munoz, I. de la Bandera, E. J. Khatib, A. Gomez-Andrades, I. Serrano, and R. Barco, "Root cause analysis based on temporal analysis of metrics toward self-organizing 5G networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2811–2824, 2016.

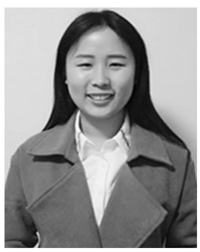
[32] J. Moysen and L. Giupponi, "A reinforcement learning based solution for self-healing in LTE networks," in *Proc. IEEE 80th Veh. Technol. Conf.*, Sep. 2014, pp. 1–6.

[33] S. Fortes, A. A. Garcia, J. A. Fernandez-Luque, A. Garrido, and R. Barco, "Context-aware self-healing: User equipment as the main source of information for small-cell indoor networks," *IEEE Veh. Technol. Mag.*, vol. 11, no. 1, pp. 76–85, Mar. 2016.



YUEPING WANG is currently pursuing the Ph.D. degree with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics. His research interests include self-organization networks, self-healing, outage detection and fault diagnosis for cellular networks, machine learning for wireless networks, resource allocation, and optimization.

KUN ZHU received the Ph.D. degree in computer engineering from Nanyang Technological University, Singapore, in 2012. He is currently a Professor with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics. He is also a Jiangsu Specially Appointed Professor. His research interests include 5G, resource management, applied game theory and optimization, and self-organization networks. He received the ACM Rising Star Chapter Award, in 2018. He has been serving as the Program Chair for the 4th MLICOM Conference and a TPC Member for the IEEE INFOCOM, GLOBECOM, and ICC. He is a PI of more than ten research projects. He is an Editor of *Computer and Communications* and has served as a Guest Editor for several journals.



MENGYUN SUN received the M.S. degree in computer science and technology from the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics. She is currently a Researcher with Huawei Company. Her research interests include self-organization networks, self-healing, outage detection, and fault diagnosis for cellular networks.

YUEYU DENG is currently pursuing the M.S. degree with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics. His research interests include self-organization networks, machine learning for networks, and cost-sensitive learning.