Evolutionary Ensemble Adversarial Learning for Identifying Terrorists among High-Speed Rail Passengers

Yu-Jun Zheng[a,*], Cong-Cong Gao[a], Yu-Jiao Huang[b], Wei-Guo Sheng[a], Zidong Wang[c]

[a]*School of Information Science and Technology, Hangzhou Normal University, Hangzhou 311121, China*
[b]*Zhijiang College, Zhejiang University of Technology, Hangzhou 312030, China*
[c]*Department of Computer Science, Brunel University London, Uxbridge, Middlesex, UB8 3PH, United Kingdom*

Abstract

As one of the most salient features of China's economic development, high-speed rail (HSR) is considered to be an attractive target and travel mode for terrorists. Distinguishing potential terrorists from normal passengers is of critical importance to public security, but very challenging because terrorists constitute only a very small fraction of HSR passengers, especially when they can disguise their attributes and behaviors to deceive the classifiers. For this extremely imbalanced classification problem, we propose a novel evolutionary generative adversarial network (GAN) ensemble method, where each GAN in the ensemble simultaneously trains a discriminator to identify abnormal samples from a large number of passenger profiles and trains a generator to produce abnormal samples that are disguised as normal ones in a subspace of the sample space, and the final classifier combines these GANs using an evolutionary fusion method. Experiments on benchmark problems demonstrate that the proposed ensemble adversarial learning method has very competitive performance compared to popular imbalanced classifiers. The successful applications in terrorist identi-

*Corresponding author. Tel.: +86-571-28866786.
Email address: yujun.zheng@computer.org (Yu-Jun Zheng), gaocongcong@compintell.cn (Cong-Cong Gao), huangyujiao@zjut.edu.cn (Yu-Jiao Huang), weiguouk@mail.com (Wei-Guo Sheng), zidong.wang@brunel.ac.uk (Zidong Wang) (Yu-Jun Zheng)

fication for China Railway also demonstrate the feasibility and effectiveness of our approach.

Keywords: Anti-terrorism, classification, deep learning, ensemble learning, evolutionary algorithm, generative adversarial network (GAN).

1. Introduction

Since its first high-speed rail (HSR) line opened between Beijing and Tianjin as one of the key infrastructures of the 2008 Beijing Olympic Games, China has developed a large-scale HSR network which is now more than 30,000 km long and carries over fiver million passengers per day, greatly shortening the time-space distances between megacities and stimulating the development of second- and third-tier cities (Zheng & Kahn., 2013). It is expected that HSR will not only continue to have significant macro effects on the national pattern of activity (Cao et al., 2013) but also become a key part of the "One Belt, One Road" strategy for Eurasian integration (Huang, 2016).

HSR not only greatly benefits normal travelers, but also helps malicious people in their actions including gathering, maneuvering, absconding, etc. In particular, the iconic status of HSR makes it an attractive potential target for terrorists due to the large number of potential victims, "high-value" passengers, significant investments, and other attractive elements (Maurillo, 2012). Although HSR has unique built-in safety and security features, performing an in-depth screening and/or physical inspection for every passenger would be unaffordable. A trade-off is to first try to identify potential terrorists from passengers, preferably by using data mining and machine learning techniques, and then better tailor in-depth inspection efforts to target terrorists and therefore deter threats. Such classification approaches have been used in critical areas such as aviation security management for years (Barnett, 2004, Babu et al., 2006, McLay et al., 2006, 2010, Majeske & Lauer, 2012, Cavusoglu et al., 2013, Rudner, 2015, Skorupski & Uchroński, 2016, Zheng et al., 2017, Feng & Huang, 2018). However, there are two major difficulties with existing classifiers. The

first is the base-rate fallacy, that is, given that normal passengers constitute a significantly larger fraction, a small imperfection in classification may result in a large number of wrongly accused passengers and thus cause the costs of the systems to outweigh their benefits (Cavusoglu et al., 2013, Rosen, 2007). The second is the vulnerability, i.e., terrorists may be able to deceive the classifiers through trial-and-error sampling and learning (McLay et al., 2010, Tutun et al., 2017).

Compared to airline security, HSR security can be much more difficult for the following reasons:

- HSR systems are much more open to the public and therefore more accessible to terrorists.
- HSR security is often not taken as seriously as airline security. As Barack Obama touted, one of the benefits of HSR is that "passengers wouldn't have to go through a security check that requires taking off their shoes..." (Gerstein, 2010)
- HSR addresses significantly larger numbers of passengers. For example, during the 2018 Chinese Spring Festival, the number of passengers carried by HSR was over 381 million, while that by airlines was only 65 million.
- As a consequence of the previous reason, the average inspection time in HSR stations is typically much shorter than that of airports (otherwise, crowds standing in long screening lines can also be vulnerable to attack (Maurillo, 2012)).

Consequently, terrorist identification in HSR security management should have a much higher feature learning ability as well as higher classification accuracy. Recent advances in deep neural networks provide a powerful tool for feature learning by automatically abstracting learned features from raw features layer by layer (Hinton & Salakhutdinov, 2006). Particularly, in circumstances where it is difficult or expensive to label sufficient training samples, deep generative models have been leveraged to synthesize labeled samples to improve the

classification accuracy (Goodfellow et al., 2014, Kingma et al., 2014, Alam et al., 2018, Fajardo et al., 2021). Motivated by these research advances, in this paper, we propose a generative adversarial network (GAN) ensemble approach for HSR passenger classification. A GAN is formulated as a minimax game between a discriminator model and a generative model (Goodfellow et al., 2014). The GAN ensemble consists of a set of individual GANs, each of which concurrently trains a generator to produce abnormal passenger profiles that are disguised as normal ones in a subspace of the sample space, and trains a discriminator to identify abnormal passenger profiles from normal ones. Our approach exhibits significant advantages in classification performance over state-of-the-art methods in experiments and has been successfully applied to improve the efficiency of anti-terrorism for the Chinese HSR. The main contributions of this paper are as follows:

- We propose a novel GAN ensemble method for imbalanced classification by iteratively constructing multiple GANs that have different classification accuracies on different training subsets and therefore are complementary to each other to improve the overall classification performance.
- We propose a new multi-rule method for fusing multiple individual classifiers of an ensemble, where the threshold parameters in the fusion rules are optimized by an evolutionary algorithm.
- Our approach has been successfully applied to identify potential terrorists from HSR passengers and effectively improved the security of China Railway.

The rest of this paper is organized as follows: Section 2 reviews related work on GAN-based methods and ensemble methods for imbalanced classification, Section 3 presents an overview of our decision-making process for passenger classification, Section 4 described the proposed GAN ensemble model, Section 5 presents the results of experiments and applications, and Section 6 concludes with a discussion.

2. Related Work

Classical machine learning methods for imbalanced classification can be divided into three groups: 1) modified classification algorithms that reinforce the learning towards the minority class; 2) cost-sensitive methods that reduce higher cost among different misclassification costs; 3) undersampling/oversampling methods that rebalance class distribution. Adversarial learning is a recently popular approach for training both generative and discriminative models in machine learning deployed in non-benign environments. It can be very useful in imbalanced classification by generating artificial data for the minority class, and its performance has been shown to be superior to various standard oversampling algorithms (Douzas & Bacao, 2018). Wang et al. (2017) found that convolutional neural networks (CNNs), although having human-level performance on image classification, often tend to be biased to large imbalanced classes. Thus, they proposed a GAN model to tackle the issue by adversatively learning discriminative features on minority class data. The performance of the model was validated on imbalanced plankton classification problems. Merdivan et al. (2017) proposed an energy-based adversarial model that minimizes the energy for a given data distribution while maximizing the energy for another distribution. They demonstrated the effectiveness of the model for positive and unlabeled learning with imbalanced data. Yin et al. (2018) referred the widely-used approach that simultaneously attack all features of the classifiers as a "dense feature attack", and they proposed a "sparse feature attack" approach that only manipulates a small subset of the features and minimize the manipulation cost at the same time. They also designed an algorithm to improve the robustness of a classifier against such attacks. Zheng et al. (2018) proposed an adversarial learning method based on deep denoising autoencoder for telecom fraud detection, which exhibited a high accuracy together with a low misclassification rate. Liu et al. (2018) proposed a semi-supervised method that combines GAN and CNN for image classification, the performance of which was demonstrated on a highly imbalanced traffic camera dataset. To improve

classification in credit card fraud detection, Fiore et al. (2019) trained a GAN to output mimicked minority class examples, which were then merged with training data into an augmented training set so as to improve the effectiveness of a classifier. Ren et al. (2019) proposed an oversampling strategy dubbed entropy-based Wasserstein GAN which, for each class, combines an entropy-weighted label vector with the original feature vector to train the generator; after being trained, the generator produces minority data samples from the concatenation of the entropy-weighted label vector with random noise feature vectors. In the GAN approach proposed by Salazar et al. (2021), the generator uses Markov random fields to synthesize surrogates by the graph Fourier transform, and the discriminator implements a linear discriminant on features measuring clique similarities between the synthesized and the original instances. Jo & Kim (2022) proposed a method of minority oversampling near the borderline with GAN, which trains a discriminator for each class to competitively affect the generator, such that the generator learns the minority class with a focus near the borderline.

As is well known, making a decision based on the single best classifier may discards the valuable contributions of other classifiers (Zhou et al., 2002). In this regard, ensemble methods that try to combine the strengths of multiple classifiers into an ensemble have become popular approaches for improving classification performance (Galar et al., 2012). Recent advances along this direction include EUSBoost (Galar et al., 2013) that uses evolutionary undersampling to combine and improve random undersampling with Boosting algorithms, BMW-SMOTE (Gao et al., 2020) that calculates the weight of each minority class instance by the ratio between the majority class proportion in the neighborhood of the current instance and the sum of all these proportions, the cost-sensitive decision tree ensembles (Krawczyk et al., 2014) that trains cost-based classifiers on random feature subspaces to ensure diversity, ClusterBal/SplitBal (Sun et al., 2015) that converts an imbalanced dataset into multiple balanced ones and then builds base classifiers on these multiple data, the supervised clustering ensemble (Xiao et al., 2016) that partitions the samples of each class into a number of clusters and then pairwise combines the clusters to construct base classifiers,

and MSEA (Liu et al., 2020) that decouples model-training and meta-training to adaptively resample the training set in iterations to get multiple classifiers in a cascade ensemble model. These ensemble methods have been used in a number of highly imbalanced classification problems such as protein prediction (Zhang et al., 2012), fault detection (Amozegar & Khorasani, 2016), traffic surveillance (Liu et al., 2017), manufacturing quality assessment (Kim et al., 2018), credit classification (Yu et al., 2018), and cancer detection (Yuan et al., 2018).

Some research efforts have also been devoted to ensembles of GANs, using combination methods like ensemble of other machine learning models. Wang et al. (2016) investigated two ways to construct ensembles of GANs. In the first way, different GANs use the same initial network but take models trained with different amount of iterations. The second way redirects part of the training data which is badly modeled by the one GAN to another. Results showed that the second performs better. Hu et al. (2017) proposed a GAN ensemble method to generate organ motion models from patient images, where each GAN is trained separately with a pre-disjointed training data set. Tramèr et al. (2018) proposed a method that augments a model's training data with adversarial examples from other models; by decoupling the examples with the model, minimizing the training loss implies increased robustness to black-box attacks. Rezaei et al. (2020) proposed a framework composed of a single-generator and a multi-discriminator variant, where the generator analyzes the input image as a condition to predict a corresponding semantic segmentation image using feedback from the multi-discriminator. Nevertheless, to our knowledge, studies on ensembles of GANs, especially those fully utilizing multiple generators and multiple discriminators for imbalanced classification, are still few.

3. The Overall Decision-Making Process

For the considered HSR passenger classification problem, a large number of records from governmental and non-governmental databases can be utilized, but acquiring all relevant records could be very time-consuming and present privacy

issues. We have suggested the security department of China Railway to adopt a decision-making process (illustrated in Figure 1) that consists of multiple levels defined based on the range of records used as follows:

L1 that only queries passenger name records (PNR) that match criminal records of the public security department. Special proactive measures will be taken for those identified as wanted or suspected criminals.

L2 that queries a *core set* of records associated with each PNR (including the passenger's *short-term* history of travel by train, and current bookings of trains, flights, inter-city buses, hotels, and scenic spots), and then employs a decision-tree based inference tool to identify anomalies (e.g., a passenger booking multiple flights and trains from the same city in the same day).

L3 that queries a much wider range of records from internal and external databases, and then trains a more powerful classifier to discover anomalies in big data.

In other words, L1 and L2 aim to identify "obviously dangerous" passengers (including but not limited to terrorists), and L3 is expected to identify more potential terrorists that have not been detected at L1 and L2. To build a classifier for the organization, we have tested many popular imbalanced classification models (Liu et al., 2009, Błaszczyński et al., 2010, Galar et al., 2013, Krawczyk et al., 2014, Sun et al., 2015, Oh et al., 2019, Gao et al., 2020) but observed that their performance is far from satisfactory, as they often misclassify too many normal passengers and/or cannot detect terrorists that are well disguised. To resolve this issue, we developed a new classifier based on adversarial ensemble learning to simultaneously improve the accuracy of identifying disguised terrorists and decrease the misclassification rate of normal passengers.

Considering privacy issues, the organization requires that the use of the classifier at L3 should be further divided into two cases:

L3.1 When the current pressure of terrorism is low, the classifier uses a *basic set* of records, including the passenger's *short-term* history and current

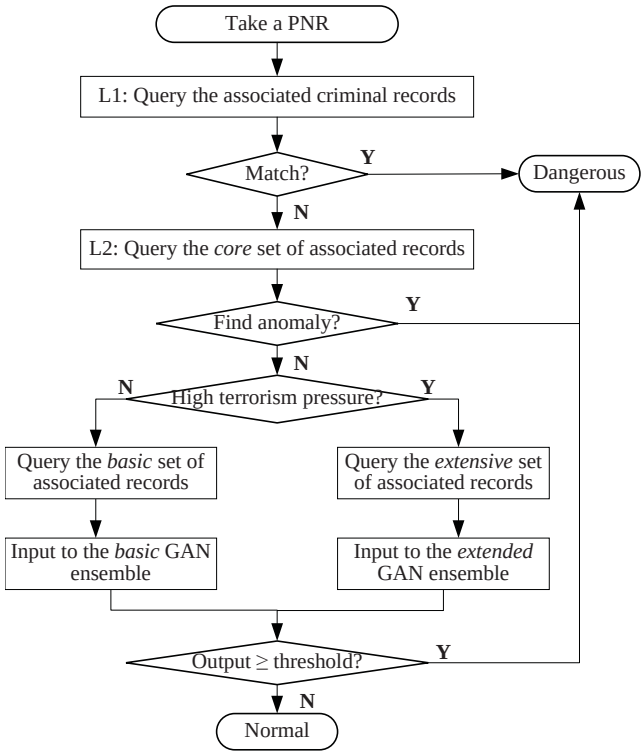


Figure 1: The flowchart of the decision-making process for HSR passenger classification.

bookings of trains, flights, inter-city buses, hotels and scenic spots, along with *cached* educational records and tax records;

L3.2 When the pressure is high, the classifier uses an *extensive set* of records, including *long-term* history and current bookings of travel services, *original* educational records (from educational departments), tax records (from tax departments and customs), fixed asset records (from housing departments), consumption records (from banks and e-commerce platforms), telecommunication records (from telecom operators), and social behavior records (provided by an Internet social network analysis tool).

For the above two cases we use input feature vectors with different lengths (currently the former is approximately 1800 while the latter is approximately 20000), which are to be processed by two different versions of GAN ensembles.

The two versions use the same underlying mechanism described in the next section.

4. Evolutionary GAN Ensemble Model

4.1. Building Block: Denoising Autoencoder

Because the input passenger profiles often contain noise and missing values, the GANS in our ensemble classifier use denoising autoencoders (Vincent et al., 2008, 2010) as the building block. As a stochastic machine, a denoising autoencoder takes an input vector $\mathbf{x} \in [0,1]^n$, corrupts it into $\widetilde{\mathbf{x}}$ by replacing a small portion of components with noise, and then transforms (encodes) it to a hidden representation $\mathbf{z}$:

$$\mathbf{z} = f_\beta(\widetilde{\mathbf{x}}) = s(\mathbf{W}\widetilde{\mathbf{x}} + \mathbf{b}) \tag{1}$$

where $\beta = [\mathbf{W}, \mathbf{b}]$, $\mathbf{W}$ is the matrix of connection weights between the input and output neurons, $\mathbf{b}$ is the vector of bias of the output neurons, and s is the mapping (which is the sigmoid function in our GAN).

Then, $\mathbf{z}$ is mapped back (decoded) to a reconstructed vector $\mathbf{x}'$:

$$\mathbf{x}' = g_{\beta'}(\mathbf{z}) = s(\mathbf{W}'\mathbf{z} + \mathbf{b}') \tag{2}$$

The training of denoising autoencoder aims to minimize the reconstruction error:

$$\arg\min_{\beta,\beta'} \mathbb{E}_{\mathbf{x}\sim\mathcal{X}}\big[-\log p\big(\mathbf{x}|g_{\beta'}(f_\beta(\widetilde{\mathbf{x}}))\big)\big] \tag{3}$$

where $\mathcal{X}$ is the empirical distribution of the input data space defined by the training set. Note that once the model has been trained, no corruption is applied to an input profile for classification.

4.2. Outlier Detectable GAN Based on Deep Denoising Autoencoder

As illustrated in Figure 2, each GAN in our model consists of a deep denoising autoencoder with two hidden layers and a Gaussian mixture model (GMM)

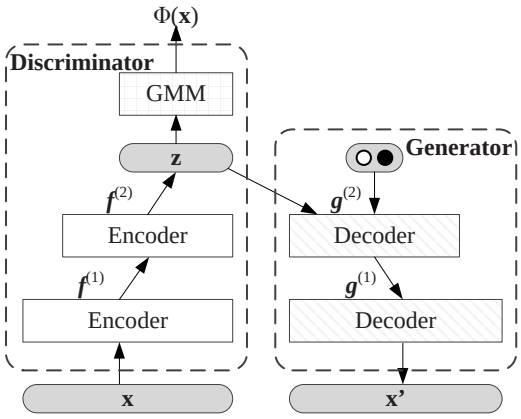


Figure 2: The architecture of the basic GAN classifier.

(Gauvain & Lee, 1994, Cardinaux et al., 2003) on the top of the second layer to produce the output $\Phi(\mathbf{x})$ from the latent vector $\mathbf{z}$:

$$\Phi(\mathbf{x}) = \Phi'(\mathbf{z}) = \frac{1}{|\mathbf{z}|} \sum_{i=1}^{|\mathbf{z}|} \log \big(\sum_{j=1}^{N_G} w_j \mathcal{N}(z_i; \mu_j, \sigma_j) \big) \tag{4}$$

where $\mathcal{N}(z_i; \mu_j, \sigma_j)$ is a high-dimensional Gaussian function with mean μ_j and diagonal covariance matrix σ_j, N_G is the number of Gaussians, and w_j is the weight for Gaussian j subject to $\big(\sum_{j=1}^{N_G} w_j \big) = 1$.

In the adversarial game, the two-layer decoder acts as the generator G for generating false samples from the prior distribution $\mathcal{Z}$ of the hidden space to the data space to deceive the discriminator, while the two-layer encoder together with GMM act as the discriminator D for discriminating positive samples (including terrorist profiles and generated samples) from normal ones. The discriminator and the generator are simultaneously trained using iterative gradient descent that alternates between D and G to optimize the following minimax objective function (Goodfellow et al., 2014):

$$\arg\min_G \arg\max_D \mathbb{E}_{\mathbf{x} \sim \mathcal{X}^+} \log D(\mathbf{x}) + \mathbb{E}_{\mathbf{z} \sim \mathcal{Z}} \log(1 - D(G(\mathbf{z}))) \tag{5}$$

Because the majority class of normal passengers can have some outliers that appear to be inconsistent with the rest of the data and therefore would degrade

the classification performance, we employ the outlier detectable GAN (OD-GAN) for adversarial learning (Oh et al., 2019), which considers a discriminator output value of around 0.5 as that the input is real data, and an output value far from 0.5 as that the input is artificial data. OD-GAN calculates the dissimilarity with the minority class, denoted by Ψ, as follows:

$$\Psi(\mathbf{x}) = 2|0.5 - \Phi(\mathbf{x})| \tag{6}$$

A Ψ value closer to 0 implies that the input is similar to the minority class. To detect and eliminate outliers from the majority data, we sort data in increasing order of Ψ values, and determine the outlier set based on the elbow point where the Ψ values changes rapidly from near 0 to near 1. After removing the outliers, the generator produces artificial data to fill insufficient data in the minority class.

4.3. Ensemble Construction

As a single classifier developed to minimize a global measure of error can be strongly biased towards the majority class in imbalanced classification, we adopt the classifier ensemble approach to mitigate the biases. Let N be the number of GANs used in the ensemble, X^+ be the set of terrorists and X^- be the set of normal passengers in the training set, our aim is to construct N GANs, each achieving high classification performance on a distinct training subset. A simple way is to divide the training set into N subset and then construct a GAN on each subset. However, a random or equal division would impair the complementarity among the individual GANs and the generality of the ensemble.

We propose a procedure that uses N iterations of training-and-testing. At each iteration, we tentatively train K GANs on K subsets, each of which consists of all samples of X^+ and $1/K$ samples of X^- (where K decrease from N to 1 with iteration), and select the GAN with the best test performance into the ensemble; the first $1/K$ of samples that are with the highest confidence in the current selected GAN are removed from X^- for the next iteration, such that the remaining GANs can focus on samples that are not well classified by the

Algorithm 1: The procedure for constructing an ensemble of N GANs on the minority (positive) set X^+ and the majority (negative) set X^-.

```
Initialize an empty ensemble, and let K = N;
while K > 0 do
    Equally divide X^- into K parts, and then construct K training subsets,
      each of which is the union of X^+ and one part of X^-;
    Construct K GANs, each being trained on one of the K training subsets
      (the training consists of two phases, one for minimizing the
      reconstruction error (3) and the other for minimizing the minimax
      function (5));
    Use X^- to test each GAN, and select the GAN with the best
      classification accuracy into the ensemble;
    Sort the samples in X^- in increasing order of the probability of being
      identified as positive by the selected GAN;
    Remove the first 1/K of samples from X^-;
    K = K - 1;
return the ensemble of N GANs.
```

selected GAN. Algorithm 1 presents the procedure for constructing the GAN ensemble. After N iterations, the ensemble has N GANs that have different classification performance on different training subsets.

4.4. Multi-Rule Fusion of Individual GANs

Based on the above iterative construction procedure, the ensemble consists of N GANs that are well complementary to each other. However, the popular voting or weighted voting approaches cannot fully utilize the complementarity among the individual GANs (Wozniak & Jackowski, 2009, Krawczyk et al., 2014). We propose a new multi-rule fusion method to promote the classification rate of the minority class while limiting the misclassification rate of the majority class. Given an input vector $\mathbf{x}$ to be classified, let $\Psi_1(\mathbf{x}), \Psi_2(\mathbf{x}), \ldots, \Psi_N(\mathbf{x})$ be the output probabilities of the N GANs *sorted in non-decreasing order*, the

ensemble makes the decision according to the rule defined as follows:

$$\begin{aligned} \text{IF} \quad & (\Psi_1(\mathbf{x}) \leq \theta_1) \vee (\Psi_2(\mathbf{x}) \leq \theta_2) \vee \ldots \vee (\Psi_{N/2}(\mathbf{x}) \leq \theta_{N/2}) \\ \text{THEN} \ & \mathbf{x} \in X^+ \\ \text{ELSE} \ & \mathbf{x} \in X^- \end{aligned}$$

where $\theta_1, \theta_2, \ldots, \theta_{N/2}$ are thresholds satisfying $0 \leq \theta_1 \leq \theta_2 \leq \ldots \leq \theta_{N/2} \leq \hat{\theta}$, where $\hat{\theta}$ is an upper limit such that, if $\Psi(\mathbf{x}) \leq \hat{\theta}$, then $\mathbf{x}$ is classified as the minority class. This rule can be interpreted as a set of sub-rules as follows:

R_1: If there is at least one GAN identifying $\mathbf{x}$ as a positive sample with the first-level (largest) confidence, then classify $\mathbf{x}$ as positive.

R_2: If there are at least two GANs identifying $\mathbf{x}$ as a positive sample with the second-level (second largest) confidence, then classify $\mathbf{x}$ as positive.

$\vdots$

$\mathrm{R}_{N/2}$: If there are at least $N/2$ GANs identifying $\mathbf{x}$ as a positive sample with the $(N/2)$st-level (smallest) confidence, then classify $\mathbf{x}$ as positive.

$\mathrm{R}_\perp$: Otherwise, classify $\mathbf{x}$ as negative.

The popular voting (or weighted voting) method can be interpreted as the special sub-rule $\mathrm{R}_{N/2}$ of our multi-rule fusion method.

4.5. Evolutionary Optimization of Fusion Parameters

Obviously, the effectiveness of the multi-rule fusion fusion depends on the threshold parameters $\theta_1, \theta_2, \ldots, \theta_{N/2}$. The parameter selection problem can be regarded as a high-dimensional global optimization problem, which is difficult to solve when N is relatively large. We employ an evolutionary optimization method to efficiently explore the $(N/2)$-dimensional parameter space to search for an optimal or near-optimal threshold setting. In the evolutionary algorithm, each chromosome represents a threshold parameter $\vec{\theta} = [\theta_1, \theta_2, \ldots, \theta_{N/2}]$. The algorithm randomly initializes a population of chromosomes, and then continually evolves them by crossover and mutation.

Given two parents $\vec{\theta} = [\theta_1, \theta_2, \ldots, \theta_{N/2}]$ and $\vec{\theta'} = [\theta'_1, \theta'_2, \ldots, \theta'_{N/2}]$, the crossover operation produces two offsprings $\vec{\theta}^{\mathrm{a}}$ and $\vec{\theta}^{\mathrm{b}}$. The components of the first offspring $\vec{\theta}^{\mathrm{a}}$ are iteratively calculated from front to back as follows:

$$\theta_1^{\mathrm{a}} = \alpha\theta_1 + (1-\alpha)\theta'_1 \tag{7}$$

$$\theta_j^{\mathrm{a}} = \begin{cases} \alpha\theta_j + (1-\alpha)\theta'_j, & \theta_{j-1}^{\mathrm{a}} \leq \min(\theta_j, \theta'_j) \\ \alpha\theta_{j-1}^{\mathrm{a}} + (1-\alpha)\max(\theta_j, \theta'_j), & \text{otherwise} \end{cases}$$

$$j = 2, \ldots, N/2 \tag{8}$$

where α is a random number between [0,1]. The components of the second offspring $\vec{\theta}^{\mathrm{b}}$ are iteratively calculated from back to front as follows:

$$\theta_{N/2}^{\mathrm{b}} = \alpha\theta_{N/2} + (1-\alpha)\theta'_{N/2} \tag{9}$$

$$\theta_j^{\mathrm{b}} = \begin{cases} \alpha\theta_j + (1-\alpha)\theta'_j, & \theta_{j+1}^{\mathrm{b}} \geq \max(\theta_j, \theta'_j) \\ \alpha\theta_{j+1}^{\mathrm{b}} + (1-\alpha)\min(\theta_j, \theta'_j), & \text{otherwise} \end{cases}$$

$$j = N/2-1, \ldots, 1 \tag{10}$$

The mutation operation modifies a chromosome $\vec{\theta} = [\theta_1, \theta_2, \ldots, \theta_{N/2}]$ by randomly selecting a dimension j and setting θ_j to be a random value. When $j = 1$, the random value is uniformly distributed in $[0, \theta_1)$; when $1 < j < N/2$, the random value is in $(\theta_{j-1}, \theta_{j+1})$; when $j = N/2$, the random value is in $(\theta_{N/2-1}, \hat{\theta}]$.

To avoid premature convergence, we also employ a random local topology for the population, where each solution is randomly assigned with probably K_N neighbors (where K_N is a parameter for controlling the neighborhood size). When selecting two chromosomes for crossover, we have a probability of η of selecting two neighbors and a probability of $(1-\eta)$ of selecting two non-neighbors, where η is a parameter increasing from a lower limit $\eta_{\min}$ to an upper $\eta_{\max}$, such that crossover among non-neighbors is preferred to facilitate global exploration in early stages and crossover among neighbors is preferred to enhance local exploitation in late stages of the algorithm (Zheng et al., 2014).

Algorithm 2 presents the framework of the algorithm, where *rand*() produces

a random number uniformly distributed in [0,1], c_r is the crossover rate, m_r is the mutation rate, and $\widehat{g}$ is a control parameter for avoiding search stagnation.

5. Results

We first test the proposed OD-GAN ensemble method on selected benchmark classification problems, and then test it on the HSR passenger classification problem. Finally, we report a 30-week application of the GAN ensemble in China Railway.

5.1. Experiments on Benchmark Problems

We select 15 benchmark datasets, including 11 KEEL data sets from Alcala-Fdez et al. (2011) and 4 DNA microarray data sets from Bullinger et al. (2004) and Yang et al. (2006), which are summarized in Table 1. Because the proposed GAN ensemble classifier targets highly imbalanced problems, we remove some minority samples from the original data sets to increase the imbalance ratio.

We compare the proposed OD-GAN-Ensemble model with 11 comparative classification models, including four single-classifier models and seven ensemble models:

- The synthetic minority oversampling technique combined with neural network (SMOTE-NN) classifier (Jeatrakul et al., 2010);
- The single basic GAN model as described in Section 4.2 but without outlier detection;
- The entropy-based Wasserstein GAN (EWGAN) (Ren et al., 2019);
- The single OD-GAN model (Oh et al., 2019) as described in Section 4.2;
- EasyEnsemble (Liu et al., 2009), an ensemble-based undersampling that samples several subsets from the majority class and trains a learner on each of them.

Algorithm 2: The evolutionary algorithm for optimizing the fusion Parameters of the GAN ensemble.

1 Randomly initialize a population P of N_P solution vectors of thresholds;
2 **foreach** *solution vector* $\vec{\theta}$ *in the population* **do**
3 **foreach** *other solution vector* $\vec{\theta}'$ *in the population* **do**
4 **if** *rand*() $< K_N/(N_P - 1)$ **then** set $\vec{\theta}'$ as a neighbor of $\vec{\theta}$;
5 **while** *the stopping criterion is not satisfied* **do**
6 **foreach** *solution vector* $\vec{\theta}$ *in the population* **do**
7 Construct a set of subrules R_1, R_2,...,$R_{N/2}$ from $\vec{\theta}$;
8 Evaluate the solution fitness based on the accuracy of ensemble defined by the rule on the training set;
9 Update η and the best solution found so far;
10 **if** *the best solution has not been update for* $\widehat{g}$ *successive generations* **then**
11 Reset the neighborhood structure as Lines 2–4;
12 Create an empty population P′;
13 **while** |P′| $< N_P$ **do**
14 Select a solution $\vec{\theta}$ from P with a probability proportional to its fitness;
15 **if** *rand*() $< c_r$ **then**
16 **if** *rand*() $< \eta$ **then**
17 Select a neighboring solution $\vec{\theta}'$ with a probability proportional to its fitness;
18 **else**
19 Select a non-neighboring solution $\vec{\theta}'$ with a probability proportional to its fitness;
20 Perform crossover on $\vec{\theta}$ and $\vec{\theta}'$;
21 Add the two offsprings to P′;
22 **else**
23 **if** *rand*() $< m_r$ **then** perform mutation on $\vec{\theta}$;
24 Add $\vec{\theta}$ to P′;
25 **return** *the best solution found so far.*

Table 1: Summary of the selected benchmark imbalanced datasets.

Dataset	Number of samples	Number of attributes	Imbalance ratio
yeast-0-5-6-7-9_vs_4	498	8	22.71
yeast2vs8	474	8	38.50
yeast6	1474	8	57.96
abalone19	4174	8	129.44
vowel0	936	13	23.63
vehicle0	666	18	34.05
segment0	2008	19	68.24
autos	153	25	50.00
dermatology-6	344	34	56.33
kddcup-buffer_overflow_vs_back	2212	41	244.78
kddcup-rootkit-imap_vs_back	2205	41	1101.5
SRBCT	75	2308	24.00
LUNG2	187	3312	45.75
CAR	165	9182	81.50
BULL	92	17404	45.00

- IIvotes (Błaszczyński et al., 2010), a rule-based ensemble with selective data pre-processing.
- EUSBoost (Galar et al., 2013), an ensemble construction technique that improves RUSBoost by using evolutionary undersampling.
- An ensemble of cost-sensitive decision trees (CSTrees) which are trained on random feature subspaces (Krawczyk et al., 2014).
- Bal-Ensemble, an ensemble method that converts an imbalanced dataset into multiple balanced ones and builds multiple classifiers on them (Sun et al., 2015).
- BMW-SMOTE based on model dynamic selection in an ensemble driven by data partition hybrid sampling (Gao et al., 2020);
- GAN-Ensemble, the ensemble of basic GANs (instead of OD-GAN) but using the same ensemble construction and evolutionary fusion methods as

described in Section 4.3 and Section 4.4.

For each of the ensemble models, we fine tune the number of ensemble members between [5,30] on each benchmark problem. For our evolutionary algorithm for GAN ensemble fusion, we set $c_r = 0.95$, $m_r = 0.015$, $\eta_{\min} = 0.35$, $\eta_{\max} = 0.75$, $\widehat{g} = 12$, $N_P = 30$, and maximum number of generations to 200. Other control parameters of the comparative models are typically set as suggested in the literature and then fine tuned on the whole test set. We uses a five-fold cross-validation strategy on each data set. The datasets are stored in an IBM Storwize V7000 storage server (with 24×600G 15K SAS disk, a 300G STEC SSD, and 64GB cache), and the computational environment is a LenovoSystem x3850 X6 server (with 4×Intel Xeon 4830 CPU, 32GB DDR4 memory, and Windows Server NT 6.2 operating system).

The experimental results are evaluated based on the sensitivity measure that denotes what percentage of minority samples are identified as such and the specificity measure that denotes what percentage of majority samples are identified as such:

$$sensitivity = \frac{TP}{TP + FN} \tag{11}$$

$$specificity = \frac{TN}{FP + TN} \tag{12}$$

where TP, FP, TN and FN refer to true positives, false positives, true negatives and false negatives, respectively.

We also use a combined measure, the Area Under the receiver operating characteristic Curve (AUC) (Huang & Ling, 2005), which evidences that increasing the number of TP without also increasing the number of FP and thus is widely used in imbalanced problems:

$$\text{AUC} = \frac{sensitivity + specificity}{2} \tag{13}$$

Tables 2, 3, and 4 present the *sensitivity*, *specificity*, and AUC results of the comparative classification methods on the benchmark problems, respectively. On each benchmark problem, the best result(s) among the 12 methods is shown in boldface.

As we can observe From Table 2, among the 15 benchmark problems the proposed OD-GAN-Ensemble method achieves the best sensitivity values on 12 problems, which is the largest among the 12 methods; on the remaining three problems (which typically have relatively low dimensions and/or imbalance ratios), the sensitivity values of OD-GAN-Ensemble are always the second best. Note that all 12 methods obtains the same best sensitivity values on five problems including autos, kddcup-rootkit-imap_vs_back, SRBCT, LUNG2, and CAR. On the remaining ten problems, the overall sensitivity performance of the first four non-ensemble methods are significantly lower than that of the ensemble methods. Except OD-GAN-Ensemble, Bal-Ensemble and GAN-Ensemble obtain relatively good sensitivity performance, achieving the best sensitivity values on 11 and 10 problems, respectively.

From Table 3, we can observe that our OD-GAN-Ensemble method exhibits more promising performance in terms of low misclassification rate: Its specificity values are the best on 14 benchmark problems, and is only the second best on the vehicle0 problem. GAN-Ensemble obtains the best specificity values on two problems including vehicle0 and CAR; IIvotes and CSTrees obtain the same best specificity values as OD-GAN-Ensemble on the SRBCT problem; EUSBoost, BMW-SMOTE and GAN-Ensemble obtain the same best specificity values as OD-GAN-Ensemble on the CAR problem. OD-GAN-Ensemble uniquely obtain the best specificity values on 12 problems, demonstrating its high ability of accurately identifying majority class samples.

Regarding the combined AUC results shown in Table 4, the performance of our OD-GAN-Ensemble is also the best on 14 benchmark problems, and is only the second best on the dermatology-6 problem (where OD-GAN obtains the best AUC value). IIvotes and CSTrees obtain the same best AUC values as OD-GAN-Ensemble on the SRBCT problem; BMW-SMOTE obtains the same best AUC value as OD-GAN-Ensemble on the Lung2 problem; BMW-SMOTE and GAN-Ensemble obtain the same best AUC values as OD-GAN-Ensemble on the CAR problem. OD-GAN-Ensemble uniquely obtain the best AUC values on 11 problems.

Table 2: The sensitivity results of the comparative classifiers on the benchmark imbalanced datasets.

Dataset	SMOTE-NN	GAN	EWGAN	OD-GAN	Easy-Ensemble	IIvotes	EUSBoost	CSTrees	Bal-Ensemble	BMW-SMOTE	GAN-Ensemble	OD-GAN-Ensemble
yeast-0-5-6-7-9_vs_4	0.4762	0.4286	0.5238	0.5714	0.5714	0.6190	0.6667	0.6190	**0.7619**	0.7143	0.7143	**0.7619**
yeast2vs8	0.7500	0.6667	0.6667	0.7500	0.5833	0.5833	0.6667	**0.7500**	**0.7500**	0.6667	**0.7500**	**0.7500**
yeast6	0.6800	0.6400	0.6800	0.6800	0.6400	0.6800	0.8000	0.7200	**0.8400**	0.7600	0.8000	0.8000
abalone19	0.3750	0.3750	0.4063	0.4375	0.6250	0.5625	0.5938	0.5938	0.6250	0.6563	**0.6875**	**0.6875**
vowel0	**0.9737**	0.9474	0.9737	0.9474	0.8947	0.9211	0.9474	0.9211	**0.9737**	0.9474	**0.9737**	**0.9737**
vehicle0	0.7895	0.7895	0.7895	0.8421	0.8421	**0.8947**	0.8421	0.7895	0.8421	0.8421	0.7895	0.8421
segment0	0.6897	0.6207	0.6552	0.6897	0.6552	0.6552	0.6897	0.7241	0.7931	0.8276	0.8621	**0.8966**
autos	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
dermatology-6	0.8333	0.8333	0.8333	**1.0000**	0.8333	0.8333	0.8333	0.8333	0.8333	0.8333	0.8333	0.8333
kddcup-buffer_overflow_vs_back	0.6667	0.5556	0.6667	0.6667	**0.7778**	0.6667	**0.7778**	**0.7778**	**0.7778**	**0.7778**	**0.7778**	**0.7778**
kddcup-rootkit-imap_vs_back	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
SRBCT	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
LUNG2	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
CAR	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
BULL	0.5000	0.5000	0.5000	0.5000	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**

Table 3: The specificity results of the comparative classifiers on the benchmark imbalanced datasets.

Dataset	SMOTE-NN	GAN	EWGAN	OD-GAN	Easy-Ensemble	IIvotes	EUSBoost	CSTrees	Bal-Ensemble	BMW-SMOTE	GAN-Ensemble	OD-GAN-Ensemble
yeast-0-5-6-7-9_vs_4	0.8931	0.8973	0.8973	0.8931	0.9015	0.9078	0.8952	0.9057	0.8952	0.9078	0.9245	**0.9371**
yeast2vs8	0.9004	0.8961	0.9026	0.8961	0.9069	0.9091	0.9004	0.9134	0.9091	0.8961	0.9372	**0.9437**
yeast6	0.8951	0.9006	0.9068	0.9055	0.9041	0.9082	0.9048	0.9103	0.9096	0.9367	0.9406	**0.9793**
abalone19	0.7127	0.6907	0.6854	0.6644	0.7653	0.7593	0.7429	0.7675	0.7525	0.8411	0.9099	**0.9346**
vowel0	0.9955	0.9944	0.9944	0.9955	0.9911	0.9889	0.9900	0.9933	0.9911	0.9967	0.9933	**0.9978**
vehicle0	0.9567	0.8794	0.9536	0.9104	0.8903	0.8964	0.8887	0.8995	0.9073	0.9552	**0.9753**	0.9722
segment0	0.8388	0.8585	0.8706	0.8545	0.8999	0.9096	0.8964	0.9075	0.8449	0.8888	0.9520	**0.9859**
autos	0.5667	0.6200	0.6733	0.6800	0.6933	0.6733	0.6200	0.7400	0.7600	0.8800	0.9400	**0.9800**
dermatology-6	0.8580	0.8166	0.8550	0.8225	0.8787	0.8580	0.8314	0.8757	0.9142	0.9615	0.9645	**0.9734**
kddcup-buffer_overflow_vs_back	0.7653	0.8066	0.7921	0.7767	0.7844	0.8112	0.7726	0.8012	0.7190	0.8838	0.9142	**0.9719**
kddcup-rootkit-imap_vs_back	0.7340	0.7290	0.7631	0.7703	0.7649	0.7826	0.7594	0.7966	0.7726	0.8956	0.9383	**0.9646**
SRBCT	0.9861	0.9722	0.9861	0.9861	0.9722	**1.0000**	0.9583	**1.0000**	0.9722	0.9444	0.9861	**1.0000**
LUNG2	0.9727	0.9180	0.9672	0.9563	0.9781	0.9836	0.9891	0.9891	0.9727	0.9945	0.9891	**0.9945**
CAR	0.9141	0.8405	0.9325	0.9325	0.9264	0.9387	**0.9509**	0.9387	0.9080	**0.9509**	**0.9509**	**0.9509**
BULL	0.9000	0.8556	0.9111	0.9111	0.7778	0.8222	0.8333	0.8000	0.8222	0.9222	0.9444	**0.9556**

Table 4: The AUC results of the comparative classifiers on the benchmark imbalanced datasets.

Dataset	SMOTE-NN	GAN	EWGAN	OD-GAN	Easy-Ensemble	IIvotes	EUSBoost	CSTrees	Bal-Ensemble	BMW-SMOTE	GAN-Ensemble	OD-GAN-Ensemble
yeast-0-5-6-7-9_vs_4	0.6846	0.6629	0.7105	0.7323	0.7364	0.7634	0.7809	0.7624	0.8285	0.8110	0.8194	**0.8495**
yeast2vs8	0.8252	0.7814	0.7846	0.8231	0.7451	0.7462	0.7835	0.8317	0.8295	0.7814	0.8436	**0.8469**
yeast6	0.7876	0.7703	0.7934	0.7927	0.7720	0.7941	0.8524	0.8151	0.8748	0.8458	0.8703	**0.8896**
abalone19	0.5438	0.5329	0.5458	0.5510	0.6952	0.6609	0.6683	0.6806	0.6888	0.7487	0.7987	**0.8810**
vowel0	0.9846	0.9709	0.9841	0.9715	0.9429	0.9550	0.9687	0.9572	0.9824	0.9720	0.9835	**0.9857**
vehicle0	0.8731	0.8345	0.8716	0.8762	0.8662	0.8956	0.8654	0.8445	0.8747	0.8986	0.8824	**0.9071**
segment0	0.7642	0.7396	0.7629	0.7721	0.7776	0.7824	0.7930	0.8158	0.8190	0.8582	0.9070	**0.9412**
autos	0.7833	0.8100	0.8167	0.8400	0.8467	0.8367	0.8100	0.8700	0.8800	0.9400	0.9700	**0.9900**
dermatology-6	0.8457	0.8250	0.8442	**0.9112**	0.8560	0.8457	0.8323	0.8545	0.8738	0.8974	0.8989	0.9034
kddcup-buffer_overflow_vs_back	0.7160	0.6811	0.7294	0.7217	0.7811	0.7389	0.7752	0.7895	0.7484	0.8308	0.8460	**0.8748**
kddcup-rootkit-imap_vs_back	0.8670	0.8645	0.8815	0.8852	0.8824	0.8913	0.8797	0.8983	0.8863	0.9478	0.9691	**0.9823**
SRBCT	0.9931	0.9861	0.9931	0.9931	0.9861	**1.0000**	0.9792	**1.0000**	0.9861	0.9722	0.9931	**1.0000**
LUNG2	0.9863	0.9590	0.9836	0.9781	0.9891	0.9918	0.9945	0.9945	0.9863	**0.9973**	0.9945	**0.9973**
CAR	0.9571	0.9202	0.9663	0.9663	0.9632	0.9693	0.9755	0.9693	0.9540	**0.9755**	**0.9755**	**0.9755**
BULL	0.7000	0.6778	0.7056	0.7056	0.8889	0.9111	0.9167	0.9000	0.9111	0.9611	0.9722	**0.9778**

Among the three GAN models, EWGAN and OD-GAN perform better than the basic GAN in almost all cases; typically, EWGAN obtains higher sensitivity values than OD-GAN, while OD-GAN obtains higher specificity values than EWGAN, that is, by removing outliers from the majority samples, OD-GAN exhibits a higher accuracy in classifying the majority class. The AUC values obtained by OD-GAN are also higher than EWGAN in most cases. Therefore, the overall performance of OD-GAN is better than EWGAN, and therefore we choose OD-GAN as the underlying classification model in the ensemble. Among the eight ensemble classification methods, Bal-Ensemble and OD-GAN-Ensemble exhibit good performance in terms of sensitivity, while OD-GAN-Ensemble outperforms all other methods in terms of specificity and AUC. Particularly, on the last four high-dimensional problems, most methods can accurately identifying minority class samples, but OD-GAN-Ensemble exhibit significant lower misclassification rate than most other methods. Therefore, it is expected that the proposed OD-GAN-Ensemble method can be effective for the high-dimensional, extremely imbalanced HSR passenger classification problem.

5.2. Experiments on Terrorist Identification

Next, we focus on the terrorist identification problem. The dataset consists of 18,000,000 real passenger records from the online ticketing system of the China Railway in a period of 15 days. Note that the records identified as "dangerous" at level L1 and L2 (described in Section 2) have been excluded from the data set. The number of terrorist samples is 120, and thus the task is an extremely imbalanced classification problem with an imbalance ratio of 149,999, which is significantly higher than any benchmark problem used in the above subsection.

We compare our OD-GAN-Ensemble method with the other 11 popular classifiers as used in the above subsection. Figure 3 presents the sensitivity (predictive accuracy on the minority class) and specificity (predictive accuracy on the majority class) results of the classifiers using the data set under low pressure of terrorism. Similar to the test results on the benchmark problems in the

above subsection, the ensemble methods exhibit significant performance advantages over the non-ensemble methods. As we can observe, only the last four classifiers can identify over 75% (90 among 120) of the terrorists, and the sensitivity value of 79.17% obtained by OD-GAN-Ensemble is the second best, only smaller than 80% obtained by BMW-SMOTE; however, the specificity value of BMW-SMOTE is much lower than that of OD-GAN-Ensemble. In general, it is expected that the classification specificity should be over 99%, i.e., the number of normal passengers misclassified as terrorists should be at most 180,000 (so that the average number of passengers to be specifically checked at a large station per day is not much larger than 200, given that there are more than 80% of passengers departing from nearly 50 large stations). In this sense, only CSTrees, GAN-Ensemble, and OD-GAN-Ensemble meet the requirement, where OD-GAN-Ensemble obtains the highest sensitivity value of 99.16%; the sensitivity values of these three classifiers are 62.5%, 75%, and 79.17%, respectively. The results indicate that our OD-GAN-Ensemble obtains the most satisfying classification results in this case.

Under high pressure of terrorism, by using much more extensive features (and more time for data processing), all the classifiers improve their sensitivity values, among which EUSBoost, BalEnsemble, BMW-SMOTE, GAN-Ensemble and OD-GAN-Ensemble can identify over 90% (108 among 120) of the terrorists, as shown in Figure 4. It is expected that, under high pressure, the average number of passengers to be specifically checked per station per day should be at most 800, and thus the specificity should be at least 96%, which is also only satisfied by CSTrees, GAN-Ensemble, and OD-GAN-Ensemble, where the specificity of OD-GAN-Ensemble is the largest. Therefore, in this case, OD-GAN-Ensemble also exhibits the best classification performance among all comparative classifiers.

Another observation is that, for most classifiers, the AUC values under high pressure are larger than those under low pressure; there are four classifiers including EUSBoost, CSTrees, GAN-Ensemble, and OD-GAN-Ensemble whose AUC values under high pressure increase over 8% compared to those under

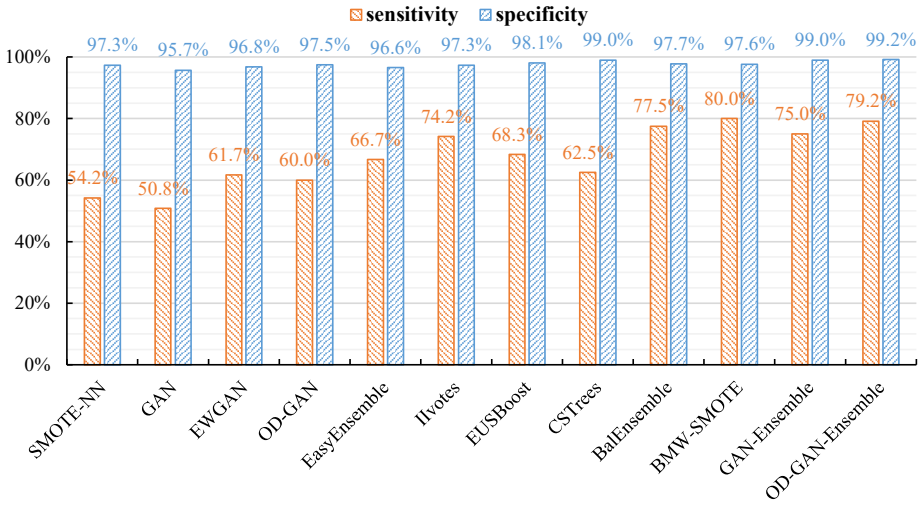


Figure 3: The classification results of the classifiers on the China Railway data set under low pressure of terrorism.

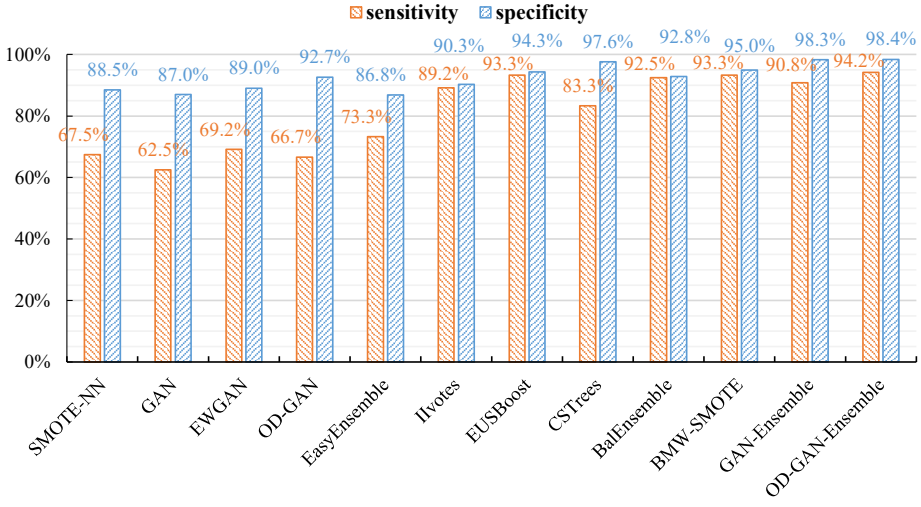


Figure 4: The classification results of the classifiers on the China Railway data set under high pressure of terrorism.

low pressure, which indicates that the they can better utilize the extensive data set to discover key characteristics that differentiate terrorists from normal passengers.

In summary, under either low pressure or high pressure of terrorism, OD-GAN-Ensemble is the only classifier that satisfies the basic requirements of predictive accuracies on both the minority class and the majority class. The results convinced the security department of the organization to take OD-GAN-Ensemble as the main tool for passenger profiling and terrorist identification.

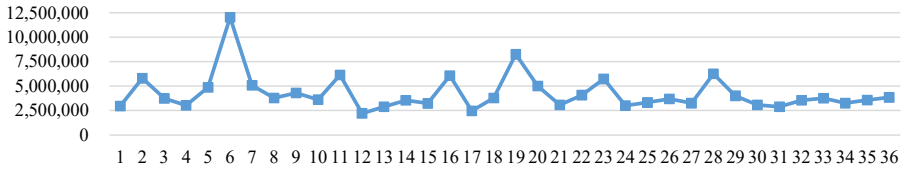
5.3. Applications

After being trained on the history data set, the proposed method has been applied to classify passengers at 24 HSR stations of the China Railway since Dec 2018. The classification is done once a week: At the first week, all passengers departing from the stations in the week are classified; since the second week, only those passengers that have not been classified in the previous month are classified, i.e., a passenger is classified at most once in a month. Here, we present the application results of the first 36 weeks, which can be divided into two stages: GAN-Ensemble method was used in 22 weeks from Dec 2018 to May 2019, and the updated OD-GAN-Ensemble was used in 14 weeks from May 2019 to Aug 2019. The 19th, 20th and 21th weeks were under high pressure of terrorism, and the other 33 weeks were under low pressure. The passengers that were identified as dangerous would be sent for special inspection when entering the station, and those confirmed as terrorists would then be controlled by the security department.

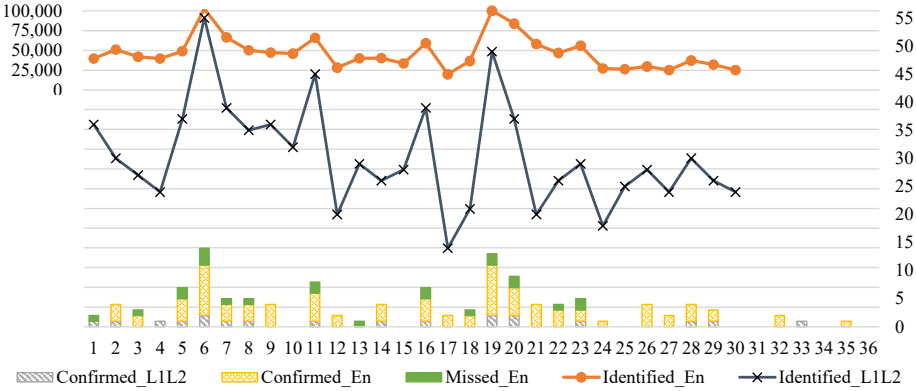
During the 36 weeks, there were a total of 153,131,146 passengers classified, and the number of each week is shown in Figure 5(a). Moreover, Figure 5(b) presents the number of dangerous passengers identified at L1 and L2 and that identified by our ensemble adversarial learning method (denoted by the suffix"En"), the number of terrorists confirmed, and the number of terrorists missed by the classifier but recognized by other means (e.g., betrayed by companions or caught by the police in other places). As we can observe, at L1

and L2, there were 1,031 passengers identified as dangerous, among which 19 were confirmed as terrorists. Our method further identified 1,536,435 dangerous passengers, among which 84 were confirmed. The number of known terrorists missed by our method was 22. Of course, there might be other unrecognized terrorists, and thus it was impossible to know the exactly accurate sensitivity. Nevertheless, the results showed that the tools at L1 and L2 only identified 19 (15.2%) of 125 known terrorists, while our method identified 84 (79.2%) of the remaining 106. Actually, there was only an attempted, small-scale attack launched by two terrorists that were missed by the classifier, and no other attack had been observed during the period. Thus, we believed that unrecognized terrorists were few, and concluded that our method successfully detected the majority of terrorists that had no obviously dangerous features and could not be identified by the traditional tools.

The number of normal passengers misclassified by our method was much larger than those misclassified at L1 and L2, because L1 and L2 aimed to identify obviously dangerous passengers, while our method was used to identify terrorists without obvious features. The overall specificity of our method was 99% under low pressure and 98.5% under high pressure, and the average number of passengers to be specially inspected per station per day was 289 under low pressure and 481 under high pressure, which were generally acceptable. Moreover, as seen in Figure 5, by incrementally learning from real-world samples, Our ensemble classifier gradually increased its classification accuracy (the average specificity is approximately 98.7% during the first ten weeks and is approximately 99.2% during the last ten weeks). Finally, the ratio of terrorists to passengers decreased with the application of our method, which demonstrated the effectiveness of our method not only in preventing attacks but also in deterring terrorists.



(a) The number of passengers



(b) The numbers of dangerous passengers identified, terrorists confirmed, and terrorists missed by the classifier but recognized by other means

Figure 5: The classification results during the 36 weeks of application at the 24 HSR stations in China Railway.

6. Conclusion and Discussion

This paper proposes an ensemble generative adversarial learning approach for an extremely imbalanced classification problem, identifying terrorists among HSR passengers. The ensemble iteratively constructs multiple GANs that have different classification accuracies on different training subsets, and then uses a multi-rule fusion method, whose parameters are optimized by an evolutionary algorithm, to effectively combine the classification capabilities of individual GANs. The proposed method exhibits significant performance advantages over a number of popular classifiers on the benchmark problems as well as the real-world terrorist identification problem.

The experimental and application results show that, based on an extensive set of records associated with PNR, the proposed model can effectively learn

feature abstractions for detecting terrorists with high accuracy and an acceptable false alarm rate. Nevertheless, acquiring extensive data about passengers may have a significant impact on privacy concerns, which is partially addressed in our approach by using different ranges of data under different pressures of terrorism. To better address the issue, our ongoing work will extend the approach to a multi-level classification system, where low-level classifiers are first employed to classify passengers using less sensitive data, and only those passengers identified as potentially dangerous are sent to high-level classifiers for a more thorough classification based on more sensitive data. It is also expected that the proposed model can be adapted or extended for many other extremely imbalanced classification problems.

Acknowledgment

Funding: This work was supported by National Natural Science Foundation of China under Grant 61872123 and Zhejiang Provincial Natural Science Foundation of China under Grant No. LR20F030002.

References

Alam, M., Vidyaratne, L., & Iftekharuddin, K. (2018). Novel deep generative simultaneous recurrent model for efficient representation learning. *Neural Netw.*, *107*, 12–22. doi:10.1016/j.neunet.2018.04.020.

Alcala-Fdez, J., Ferndedez, A., Luengo, J., Derrac, J., Garcia, S., Sanchez, L., & Herrera, F. (2011). KEEL data-mining software tool: data set repository integration of algorithms and experimental analysis framework. *J. Multi-Valued Logic Soft Comput.*, (pp. 255–287).

Amozegar, M., & Khorasani, K. (2016). An ensemble of dynamic neural network identifiers for fault detection and isolation of gas turbine engines. *Neural Netw.*, *76*, 106–121. doi:10.1016/j.neunet.2016.01.003.

Babu, V. L. L., Batta, R., & Lin, L. (2006). Passenger grouping under constant threat probability in an airport security system. *Eur. J. Oper. Res.*, *168*, 633–644. doi:10.1016/j.ejor.2004.06.007.

Barnett, A. (2004). CAPPS II: The foundation of aviation security? *Risk Anal.*, *24*, 909–916.

Błaszczyński, J., Deckert, M., Stefanowski, J., & Wilk, S. (2010). Integrating selective pre-processing of imbalanced data with ivotes ensemble. In M. Szczuka, M. Kryszkiewicz, S. Ramanna, R. Jensen, & Q. Hu (Eds.), *Rough Sets and Current Trends in Computing* (pp. 148–157). Berlin, Heidelberg: Springer. doi:10.1007/978-3-642-13529-3_17.

Bullinger, L., Döhner, K., Bair, E., Fröhling, S., Schlenk, R. F., Tibshirani, R., Döhner, H., & Pollack, J. R. (2004). Use of gene-expression profiling to identify prognostic subclasses in adult acute myeloid leukemia. *New England J. Med.*, *350*, 1605–1616. doi:10.1056/NEJMoa031046.

Cao, J., Liu, X. C., Wang, Y., & Li., Q. (2013). Accessibility impacts of China's high-speed rail network. *J. Transp. Geogr.*, *28*, 12–21.

Cardinaux, F., Sanderson, C., & Marcel, S. (2003). Comparison of MLP and GMM classifiers for face verification on XM2VTS. In J. Kittler, & M. S. Nixon (Eds.), *Audio- and Video-Based Biometric Person Authentication* (pp. 911–920). Springer Berlin Heidelberg. doi:10.1007/3-540-44887-X_106.

Cavusoglu, H., Kwark, Y., Mai, B., & Raghunathan, S. (2013). Passenger profiling and screening for aviation security in the presence of strategic attackers. *Decision Anal.*, *10*, 63–81. doi:10.1287/deca.1120.0258.

Chawla, N., Hall, L., Bowyer, K., & Kegelmeyer, W. (2002). SMOTE: synthetic minority oversampling technique. *J. Artif. Intell. Res.*, (pp. 321–357).

Douzas, G., & Bacao, F. (2018). Effective data generation for imbalanced learning using conditional generative adversarial networks. *Expert Syst. Appl.*, *91*, 464–471. doi:10.1016/j.eswa.2017.09.030.

Fajardo, V. A., Findlay, D., Jaiswal, C., Yin, X., Houmanfar, R., Xie, H., Liang, J., She, X., & Emerson, D. (2021). On oversampling imbalanced data with deep

conditional generative models. *Expert Syst. Appl.*, *169*, 114463. doi:10.1016/j.eswa.2020.114463.

Feng, W., & Huang, J. (2018). Early warning for civil aviation security checks based on deep learning. *Data Analy. Knowl. Discovery*, *2*, 46. doi:10.11925/infotech.2096-3467.2018.0812.

Fiore, U., De Santis, A., Perla, F., Zanetti, P., & Palmieri, F. (2019). Using generative adversarial networks for improving classification effectiveness in credit card fraud detection. *Inf. Sci.*, *479*, 448–455. doi:10.1016/j.ins.2017.12.030.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2012). A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. *IEEE Trans. Syst. Man Cybern. Part C*, *42*, 463–484. doi:10.1109/TSMCC.2011.2161285.

Galar, M., Fernández, A., Barrenechea, E., & Herrera, F. (2013). EUSBoost: Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling. *Pattern Recog.*, *46*, 3460–3471. doi:10.1016/j.patcog.2013.05.006.

Gao, X., Ren, B., Zhang, H., Sun, B., Li, J., Xu, J., He, Y., & Li, K. (2020). An ensemble imbalanced classification method based on model dynamic selection driven by data partition hybrid sampling. *Expert Syst. Appl.*, *160*, 113660. doi:10.1016/j.eswa.2020.113660.

Gauvain, J. L., & Lee, C.-H. (1994). Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. *IEEE Trans. Speech Audio Process.*, *2*, 291–298. doi:10.1109/89.279278.

Gerstein, J. (2010). Obama: No shoe checks on high-speed rail.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems 27* (pp. 2672–2680). Curran Associates, Inc.

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *Proc. IJCNN* (pp. 1322–1328). doi:10.1109/IJCNN.2008.4633969.

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, *313*, 504–507.

Hu, Y., Gibson, E., Vercauteren, T., Ahmed, H. U., Emberton, M., Moore, C. M., Noble, J. A., & Barratt, D. C. (2017). Intraoperative organ motion models with an ensemble of conditional generative adversarial networks. In M. Descoteaux, L. Maier-Hein, A. Franz, P. Jannin, D. L. Collins, & S. Duchesne (Eds.), *Medical Image Computing and Computer-Assisted Intervention* (pp. 368–376). Cham: Springer. doi:10.1007/978-3-319-66185-8_42.

Huang, J., & Ling, C. X. (2005). Using AUC and accuracy in evaluating learning algorithms. *IEEE Trans. Knowl. Data Eng.*, *17*, 299–310. doi:10.1109/TKDE.2005.50.

Huang, Y. (2016). Understanding China's Belt & Road initiative: Motivation, framework and assessment. *China Eco. Rev.*, *40*, 314–321. doi:10.1016/j.chieco.2016.07.007.

Jeatrakul, P., Wong, K. W., & Fung, C. C. (2010). Classification of imbalanced data by combining the complementary neural network and smote algorithm. In K. W. Wong, B. S. U. Mendis, & A. Bouzerdoum (Eds.), *Neural Information Processing. Models and Applications* (pp. 152–159). Berlin, Heidelberg: Springer. doi:10.1007/978-3-642-17534-3_19.

Jo, W., & Kim, D. (2022). OBGAN: Minority oversampling near borderline with generative adversarial networks. *Expert Syst. Appl.*, *197*, 116694. doi:10.1016/j.eswa.2022.116694.

Kim, A., Oh, K., Jung, J.-Y., & Kim, B. (2018). Imbalanced classification of manufacturing quality conditions using cost-sensitive decision tree ensembles. *Int. J. Comput. Integr. Manuf.*, *31*, 701–717. doi:10.1080/0951192X.2017.1407447.

Kingma, D. P., Mohamed, S., Jimenez Rezende, D., & Welling, M. (2014). Semi-supervised learning with deep generative models. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems 27* (pp. 3581–3589). Curran Associates, Inc.

Krawczyk, B., Wozniak, M., & Schaefer, G. (2014). Cost-sensitive decision tree ensembles for effective imbalanced classification. *Appl. Soft Comput.*, *14*, 554–562. doi:10.1016/j.asoc.2013.08.014.

Liu, W., Luo, Z., & Li, S. (2018). Improving deep ensemble vehicle classification by using selected adversarial samples. *Knowl. Based Syst.*, *160*, 167–175. doi:10.1016/j.knosys.2018.06.035.

Liu, W., Zhang, M., Luo, Z., & Cai, Y. (2017). An ensemble deep learning method for vehicle type classification on visual traffic surveillance sensors. *IEEE Access*, *5*, 24417–24425. doi:10.1109/ACCESS.2017.2766203.

Liu, X. Y., Wu, J., & Zhou, Z. H. (2009). Exploratory undersampling for class-imbalance learning. *IEEE Trans. Syst. Man Cybern. Part B*, *39*, 539–550. doi:10.1109/TSMCB.2008.2007853.

Liu, Z., Wei, P., Jiang, J., Cao, W., Bian, J., & Chang, Y. (2020). MESA: Boost ensemble imbalanced learning with meta-sampler. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems* (pp. 14463–14474). Curran Associates, Inc. volume 33. URL: https://proceedings.neurips.cc/paper/2020/file/a64bd53139f71961c5c31a9af03d775e-Paper.pdf.

López, V., Triguero, I., Carmona, C. J., García, S., & Herrera, F. (2014). Addressing imbalanced classification with instance generation techniques: IPADE-ID. *Neurocomputing*, *126*, 15–28. doi:10.1016/j.neucom.2013.01.050.

Majeske, K. D., & Lauer, T. W. (2012). Optimizing airline passenger prescreening systems with bayesian decision models. *Comput. Oper. Res.*, *39*, 1827–1836. doi:10.1016/j.cor.2011.04.008.

Maurillo, D. R. (2012). *High-Speed Rail in the US: Will it be a More Attractive Terror Target than Inter-City Rail?*. Master's thesis.

McLay, L. A., Jacobson, S. H., & Kobza, J. E. (2006). A multilevel passenger screening problem for aviation security. *Naval Res. Logis.*, *53*, 183–197. doi:10.1002/nav.20131.

McLay, L. A., Lee, A. J., & Jacobson, S. H. (2010). Risk-based policies for airport security checkpoint screening. *Transp. Sci.*, *44*, 333–349.

Merdivan, E., Loghmani, M. R., & Geist, M. (2017). Reconstruct & crush network. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (pp. 4548–4556). Curran Associates, Inc. volume 30.

Oh, J.-H., Hong, J. Y., & Baek, J.-G. (2019). Oversampling method using outlier detectable generative adversarial network. *Expert Syst. Appl.*, *133*, 1–8. doi:10.1016/j.eswa.2019.05.006.

Ren, J., Liu, Y., & Liu, J. (2019). EWGAN: Entropy-based wasserstein GAN for imbalanced learning. In *Proc. 33rd AAAI Conf. Artificial Intelligence* (pp. 10011–10022).

Rezaei, M., Näppi, J. J., Lippert, C., Meinel, C., & Yoshida, H. (2020). Generative multi-adversarial network for striking the right balance in abdominal image segmentation. *Int. J. Comput. Assist. Radiol. Surgery*, *15*, 1847–1858. doi:10.1007/s11548-020-02254-4.

Rosen, J. (2007). The silver bullet: protecting privacy and security through law and technology. *Proc. Amer. Philos. Soc.*, *151*, 291–299.

Rudner, M. (2015). Intelligence-led air transport security: Pre-screening for watch-lists, no-fly lists to forestall terrorist threats. *Int. J. Intell. CounterIntell.*, *28*, 38–63. doi:10.1080/08850607.2014.962352.

Salazar, A., Vergara, L., & Safont, G. (2021). Generative adversarial networks and markov random fields for oversampling very small training sets. *Expert Syst. Appl.*, *163*, 113819. doi:10.1016/j.eswa.2020.113819.

Skorupski, J., & Uchroński, P. (2016). Managing the process of passenger security control at an airport using the fuzzy inference system. *Expert Syst. Appl.*, *54*, 284–293. doi:10.1016/j.eswa.2015.11.014.

Sun, Z., Song, Q., Zhu, X., Sun, H., Xu, B., & Zhou, Y. (2015). A novel ensemble method for classifying imbalanced data. *Pattern Recog.*, *48*, 1623–1637. doi:10.1016/j.patcog.2014.11.014.

Tang, Y., Zhang, Y. Q., Chawla, N. V., & Krasser, S. (2009). SVMs modeling for highly imbalanced classification. *IEEE Trans. Syst. Man, Cybern. Part B*, *39*, 281–288. doi:10.1109/TSMCB.2008.2002909.

Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., & McDaniel, P. (2018). Ensemble adversarial training: Attacks and defenses. URL: arXiv:1705.07204.

Tutun, S., Khasawneh, M. T., & Zhuang, J. (2017). New framework that uses patterns and relations to understand terrorist behaviors. *Expert Syst. Appl.*, *78*, 358–375. doi:10.1016/j.eswa.2017.02.029.

Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. In *Proc. ICML* (pp. 1096–1103). New York, NY, USA: ACM. doi:10.1145/1390156.1390294.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P.-A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.*, *11*, 3371–3408.

Wang, C., Yu, Z., Zheng, H., Wang, N., & Zheng, B. (2017). CGAN-plankton: Towards large-scale imbalanced class generation and fine-grained classification. In *IEEE Int'l Conf. Image Processing* (pp. 855–859).

Wang, Y., Zhang, L., & van de Weijer, J. (2016). Ensembles of generative adversarial networks. *arXiv preprint*, . URL: 1612.00991.

Wozniak, M., & Jackowski, K. (2009). Some remarks on chosen methods of classifier fusion based on weighted voting. In E. Corchado, X. Wu, E. Oja, A. Herrero, & B. Baruque (Eds.), *Hybrid Artificial Intelligence Systems* (pp. 541–548). Springer.

Xiao, H., Xiao, Z., & Wang, Y. (2016). Ensemble classification based on supervised clustering for credit scoring. *Appl. Soft Comput.*, *43*, 73–86. doi:10.1016/j.asoc.2016.02.022.

Yang, K., Cai, Z., Li, J., & Lin, G. (2006). A stable gene selection in microarray data analysis. *BMC Bioinform*, (p. 228).

Yin, Z., Wang, F., Liu, W., & Chawla, S. (2018). Sparse feature attacks in adversarial learning. *IEEE Trans. Knowl. Data Eng.*, *30*, 1164–1177. doi:10.1109/TKDE.2018.2790928.

Yu, L., Zhou, R., Tang, L., & Chen, R. (2018). A DBN-based resampling SVM ensemble learning paradigm for credit classification with imbalanced data. *Appl. Soft Comput.*, *69*, 192–202. doi:10.1016/j.asoc.2018.04.049.

Yuan, X., Xie, L., & Abouelenien, M. (2018). A regularized ensemble framework of deep learning for cancer detection from multi-class, imbalanced training data. *Pattern Recognition*, *77*, 160–172. doi:10.1016/j.patcog.2017.12.017.

Zhang, Y., Zhang, D., Mi, G., Ma, D., Li, G., Guo, Y., Li, M., & Zhu, M. (2012). Using ensemble methods to deal with imbalanced data in predicting protein–protein interactions. (pp. 36–41). volume 36. doi:10.1016/j.compbiolchem.2011.12.003.

Zheng, S., & Kahn., M. E. (2013). China's bullet trains facilitate market integration and mitigate the cost of megacity growth. *Proc. National Academy Sci*, *110*, E1248–E1253.

Zheng, Y.-J., Ling, H.-F., & Xue, J.-Y. (2014). Ecogeography-based optimization: Enhancing biogeography-based optimization with ecogeographic barriers and differentiations. *Comput. Oper. Res.*, *50*, 115–127. doi:10.1016/j.cor.2014.04.013.

Zheng, Y. J., Sheng, W. G., Sun, X. M., & Chen, S. Y. (2017). Airline passenger profiling based on fuzzy deep machine learning. *IEEE Trans. Neural Netw. Learn. Syst.*, *28*, 2911–2923. doi:10.1109/TNNLS.2016.2609437.

Zheng, Y.-J., Zhou, X.-H., Sheng, W.-G., Xue, Y., & Chen, S.-Y. (2018). Generative adversarial network based telecom fraud detection at the receiving bank. *Neural Netw.*, *102*, 78–86. doi:10.1016/j.neunet.2018.02.015.

Zhou, Z.-H., Wu, J., & Tang, W. (2002). Ensembling neural networks: Many could be better than all. *Artif. Intell.*, *137*, 239–263. doi:10.1016/S0004-3702(02)00190-X.