Airline Passenger Profiling Based on Fuzzy Deep Machine Learning

Yu-Jun Zheng, *Member, IEEE*, Wei-Guo Sheng, *Member, IEEE*,
Xing-Ming Sun, *Senior Member, IEEE*, and Sheng-Yong Chen, *Senior Member, IEEE*

***Abstract*— Passenger profiling plays a vital part of commercial aviation security, but classical methods become very inefficient in handling the rapidly increasing amounts of electronic records. This paper proposes a deep learning approach to passenger profiling. The center of our approach is a Pythagorean fuzzy deep Boltzmann machine (PFDBM), whose parameters are expressed by Pythagorean fuzzy numbers such that each neuron can learn how a feature affects the production of the correct output from both the positive and negative sides. We propose a hybrid algorithm combining a gradient-based method and an evolutionary algorithm for training the PFDBM. Based on the novel learning model, we develop a deep neural network (DNN) for classifying normal passengers and potential attackers, and further develop an integrated DNN for identifying group attackers whose individual features are insufficient to reveal the abnormality. Experiments on data sets from Air China show that our approach provides much higher learning ability and classification accuracy than existing profilers. It is expected that the fuzzy deep learning approach can be adapted for a variety of complex pattern analysis tasks.**

***Index Terms*— Biogeography-based optimization (BBO), deep Boltzmann machine (DBM), deep learning, evolutionary neural networks, passenger profiling, Pythagorean fuzzy set (PFS).**

I. Introduction

TERRORISM has threatened the freedom to travel throughout the history of commercial aviation [1]. In the last few years, there is an increasing number of serious air disasters around the world, many of which have been confirmed or suspected to be associated with terrorist attacks. Generally, in-depth screening and physical inspection can effectively identify or deter potential threats, and thus reduce the risk of being attacked. However, doing so on every passenger may be very costly and create delays and complaints, decreasing the welfare of the approver (airports and airlines) [2]. A tradeoff is to use passenger profiling to identify potential attackers who constitute only a very small fraction of passengers: if profiling can correctly identify likely attackers and likely normal passengers, in-depth screening and inspection efforts can be better tailored to target the likely attackers. Such a mechanism has been used in aviation security management for a long time in order to reduce the inconvenience caused to normal passengers and the overall cost of security in transportation systems [3].

Besides concerns about legality and privacy [4], one major criticism of passenger profiling systems is their low accuracy, which is bound to produce a high number of "false positives" (i.e., passengers wrongly identified as potential terrorists) and cause the costs of the systems to outweigh their benefits [5]–[7]. High inspection rates also bring much inconvenience to normal passengers, which in turn may further stimulate criticism on legality and privacy issues of profiling. Moreover, some critics note that most current profilers are vulnerable, that is, terrorists may be able to deceive the profiler through trial-and-error sampling and learning [8] (e.g., in weeks prior to the 9/11 attacks, Mohamed Atta and other hijackers practiced their attacks by boarding the same target flights [9]).

Profiling is a complicated pattern analysis task for almost all passengers (except very few passengers who have distinct criminal features). Typically, a profiler calculates a risk score for a passenger based on a set of key features and their abstractions. However, when designing a profiling system, identifying such key features from a large number of candidates is usually time-consuming and error-prone because of the unknown complex and probabilistic relationship between the features and the potential risk. The problem is even exacerbated by the emerging trend moving away from evidence-based observation and detection to knowledge-based discovery from large amounts of electronic records [10]–[13]. Furthermore, if we want to identify a group of attackers when the features of each individual attacker are insufficient to reveal the abnormality, the difficulty increases exponentially.

Recent advances in deep machine learning provide a powerful tool for modeling complex probability distributions by automatically discovering intermediate abstractions from a huge amount of basic features [14]–[20]. To the best of our best knowledge, however, there is no report on passenger profiling based on deep learning. This paper proposes a deep learning approach to passenger profiling, central to which is a new deep Boltzmann machine (DBM) model, called Pythagorean fuzzy DBM (PFDBM), which uses Pythagorean-type fuzzy sets [21], [22] to improve the representation ability and robustness. Based on PFDBM,

Manuscript received May 25, 2016; revised September 1, 2016; accepted September 12, 2016. Date of publication September 27, 2016; date of current version November 15, 2017. This work was supported by the National Natural Science Foundation of China under Grant 61325019, Grant 61472167, and Grant 61573316.

Y.-J. Zheng, W.-G. Sheng, and S.-Y. Chen are with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China (e-mail: yujun.zheng@computer.org; wsheng@zjut.edu.cn; sy@ieee.org).

X.-M. Sun is with the School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2016.2609437

we construct a deep neural network (DNN) for learning risk features and producing the likelihood of each individual passenger being an attacker, and further integrate a set of individual DNNs for identifying group attackers whose individual features are insufficient to reveal the abnormality. Experiments show that our approach achieves much higher classification accuracy than existing profilers.

The rest of this paper is structured as follows. Section II describes the related work. Section III introduces the preliminaries of DBM and Pythagorean fuzzy set (PFS). Section IV presents our PFDBM model and its evolutionary learning algorithm. Section V describes the PFDBM-based neural network for passenger profiling, the performance of which is validated by the experiments in Section VI. Finally, Section VII concludes with discussion.

II. Related Work

Aviation passenger profiling has a long history. The Federal Aviation Administration (FAA) has regulated passenger profiling as a fundamental part of airline security since 1960s. Its "Anti-Air Hijack Profile" developed around 25 characteristics empirically linked with those of past hijackers. Under a grant from the FAA, in 1996, Northwest Airlines developed the first computer airline passenger profiling system, named "computer-assisted passenger prescreening system" (CAPPS), which is a rule-based system utilizing about 39 pieces of preboarding data to identify high-risk passengers [23]. After the 9/11 attacks, the Transportation Security Administration (TSA) proposed CAPPS II, which utilizes an additional set of attributes (e.g., income and demography) retrieving from governmental databases (e.g., law enforcement and intelligence databases) to calculate a risk score for each passenger. Based on their risk scores, passengers are labeled as "green", "yellow", or "red" security risks and then subject to correspondingly intrusive scrutiny [24]. Due to aggravated privacy-related concerns, in 2004, the TSA replaced CAPPS II with a next-generation system "Secure Flight", which improves the selection process through a terrorist screening database and leaves responsibilities of comparing passenger identification with sensitive government data from the privatized airlines to the federal government [25]. Since 2010, the TSA has required airlines to share preboarding data with the government to perform centralized watch list matching.

A number of studies have investigated the effects of passenger profiling on aviation security. Virta *et al.* [26] developed a cost model to analyze the tradeoffs between screening all baggage and screening baggage of only those passengers selected by CAPPS. McLay *et al.* [27] developed a multilevel allocation problem to assign passengers differentiated by their perceived risk levels to different classes of screening such that the total security is maximized, and their results suggest that fewer security classes for passenger screening may be more effective. McLay *et al.* [8] further formulated the problem as a sequential stochastic assignment problem and used dynamic programming to assign passengers to security classes in real time. Considering a passenger classification problem under the assumption that the threat probability is known and identical for all passengers, Babu *et al.* [28] developed a model to determine the number of groups, the fractions of passengers and the assignment of check stations for each group, such that the number of false alarms is minimized. Their major conclusion was that passenger grouping can be beneficial even when the threat probability is assumed constant across all passengers. Cavusoglu *et al.* [3] analyzed how the addition of a profiler to a security setup that employs screening devices and physical inspections affects various performance measures, such as attacker detection rate, inconvenience caused to normal passengers, and security cost, and they proposed a two-screening-device setup where classified attackers and classified normal passengers are sent through separate screening devices to ensure the benefit of profiling. Interested readers can also refer to [29] for a review on profiling in general crime control.

Nevertheless, there is much skepticism and criticism. Besides legality (discrimination) and privacy, major concerns with profiling systems also include the effectiveness and cost-effectiveness. A 1999 report prepared by the Library of Congress for U.S. intelligence agencies wrote: "There seems to be general agreement among psychologists that there is no particular psychological attribute that can be used to describe the terrorist or any 'personality' that is distinctive of terrorists" [30]. Moreover, given that normal passengers constitute a much larger fraction, a small imperfection in classifying them will result in a large number of false positives due to the base rate fallacy [31]–[34]. Rosen [5] gave a case of identifying the 11 hijackers of 9/11 in a population of 300 million. Even assuming the profiler is 99% accurate, 3 million of those identified as potential terrorists would be wrongly accused, which would bring the nation's airports to a halt. Furthermore, it is unclear how to avoid and/or remedy profiling errors caused by mistaken identity, identity theft, fraud, or otherwise [4].

Critics also note that terrorists may be able to deceive the profiler through trial-and-error sampling and learning if the profiler heavily relies on individuals' attributes which can often be manipulated by attackers [8]. Once attackers figured out how the profiler could be defeated, it was a matter of identifying a set of attackers that fit the attributes or changing an attacker's attributes [31]. Raghunathan *et al.* [35] studied the effectiveness of profiling on intrusion detection when faking is possible, showing that if the profiling accuracy is higher than a threshold value, then it is optimal to degrade the detection rate of high-risk passengers and enhance the detection rate of low-risk ones. Cavusoglu *et al.* [31] examined how passenger profiling impacts airport security operations when the profiler is vulnerable to gaming by attackers. They found that, regardless of the profiling setup, adding a profiler to the current aviation security setup can benefit key performance measures. However, although accuracy in identifying attackers and accuracy in identifying normal passengers are both important, the profiling algorithm should first seek to maximize the latter.

Typically, current approaches to profiling system design face two major difficulties: 1) determining a set of key features and their abstractions from a large number of candidates for risk

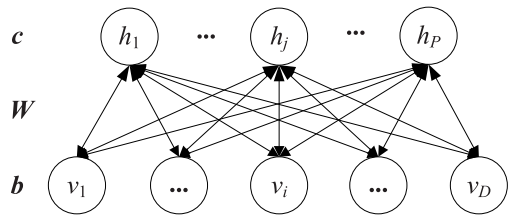


Fig. 1. RBM.

calculation and 2) tackling missing values and noise contained in the input data. For the first difficulty, emerging deep learning models combined with highly parallel computing [36], [37] have exhibited promising performance for feature exaction and abstraction, but their applications in aviation security management have rarely been reported. For the second difficulty, fuzzy set theory [38] provides a powerful tool for handling imperfect data [39]–[41] and has been applied in risk assessment and control in some works [42]–[44], but studies on fuzzy profiling are scarce. In particular, research on fuzzy deep learning is still in its infancy, for which we have only found the work of Chen *et al.* [45] on fuzzy restricted Boltzmann machines (RBMs) that exhibit much better representation ability than classical ones. Moreover, to the best of our knowledge, there is no study on passenger profiling methods for identifying potential attackers in group.

III. Preliminaries

A. Deep Boltzmann Machine

As shown in Fig. 1, RBM [46] is an energy-based probabilistic model which defines a joint probability distribution over $\mathbf{v} \in \{0, 1\}^D$ and $\mathbf{h} \in \{0, 1\}^P$ as

$$P(\mathbf{v}, \mathbf{h}, \theta) = \frac{1}{Z(\theta)} \exp(-E(\mathbf{v}, \mathbf{h}, \theta)) \tag{1}$$

where $\theta = [\mathbf{b}, \mathbf{c}, \mathbf{W}]$ is the parameter vector representing visible-to-hidden and hidden-to-hidden interaction terms, $E(\mathbf{v}, \mathbf{h}, \theta)$ is the energy function defined as

$$E(\mathbf{v}, \mathbf{h}, \theta) = -\mathbf{v}^T \mathbf{b}\mathbf{v} - \mathbf{h}^T \mathbf{c}\mathbf{h} - \mathbf{v}^T \mathbf{W}\mathbf{h} \tag{2}$$

and $Z(\theta)$ is the partition function defined as

$$Z(\theta) = \sum_{\mathbf{v}} \sum_{\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h}, \theta)). \tag{3}$$

A DBM is an extension of the RBM that has multiple layers of hidden units arranged in layers, where each layer captures complicated, higher-order correlations between the activities of hidden features in the layer below [16]. Considering a two-layer DBM shown in Fig. 2, the energy function of a state $\{\mathbf{v}, \mathbf{h}_1, \mathbf{h}_2\}$ is defined as

$$E(\mathbf{v}, \mathbf{h}_1, \mathbf{h}_2, \theta) = -\mathbf{v}^T \mathbf{W}_1 \mathbf{h}_1 - \mathbf{h}_1^T \mathbf{W}_2 \mathbf{h}_2 \tag{4}$$

where $\theta = [\mathbf{W}_1, \mathbf{W}_2]$ is the parameter vector. The probability that the DBM model assigns to the visible vector $\mathbf{v}$ is

$$P(\mathbf{v}, \theta) = \frac{1}{Z(\theta)} \sum_{\mathbf{h}_1} \sum_{\mathbf{h}_2} \exp(-E(\mathbf{v}, \mathbf{h}_1, \mathbf{h}_2, \theta)). \tag{5}$$

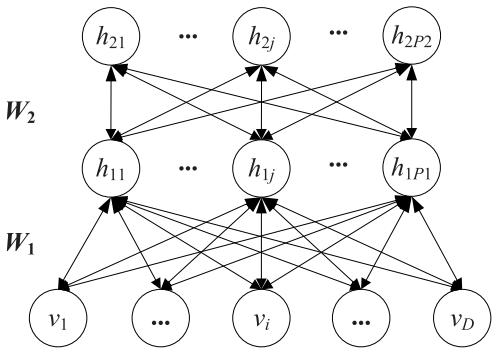


Fig. 2. DBM.

By extension, for a DBM with a single visible layer and L hidden layers parameterized by weights $\mathbf{W}_l$ between the lth layer and the $(l+1)$th layer $(1 \le l \le L)$, the energy function of a state $\{\mathbf{v}, \mathbf{h}_1, \ldots, \mathbf{h}_L\}$ is defined as

$$E(\mathbf{v}, \mathbf{h}_1, \ldots, \mathbf{h}_L, \theta) = -\mathbf{v}^T \mathbf{W}_1 \mathbf{h}_1 - \sum_{l=2}^{L} \mathbf{h}_{l-1}^T \mathbf{W}_l \mathbf{h}_l \tag{6}$$

where $\theta = [\mathbf{W}_1, \ldots, \mathbf{W}_L]$. The corresponding probability is

$$P(\mathbf{v}, \theta) = \frac{1}{Z(\theta)} \sum_{\mathbf{h}_1} \cdots \sum_{\mathbf{h}_L} \exp(-E(\mathbf{v}, \mathbf{h}_1, \ldots, \mathbf{h}_L, \theta)). \tag{7}$$

The basic RBM and DBM learn distributions over binary vectors. A popular approach to model real-valued data is first transforming a real vector into a binary one using Gaussian–Bernoulli RBM (GRBM) [14], which improves the RBM by replacing the binary-valued visible units with Gaussian ones, and then using DBM to learn distributions over the binary vector extracted. The energy function of GRBM is defined as

$$E(\mathbf{v}, \mathbf{h}, \theta) = \sum_{i=1}^{D} \frac{(v_i - b_i)^2}{2\sigma_i^2} - \sum_{i=1}^{D} \sum_{j=1}^{P} W_{ij} h_j \frac{v_i}{\sigma_i^2} - \sum_{j=1}^{P} c_j h_j \tag{8}$$

where σ_i is the standard deviation associated with Gaussian visible neuron v_i $(1 \le i \le D)$.

B. Pythagorean Fuzzy Set

The theory of fuzzy sets, initially proposed by Zadeh [38], is a generalization of the classical set theory in that the membership of an element to a set S (called the support set) is graded between 0 and 1 as opposed to being pure boolean. Intuitionistic fuzzy set (IFS) [47] extends the basic fuzzy set by utilizing a membership degree and a nonmembership degree whose sum is less than or equal to 1, to assess an element from the positive and negative aspects simultaneously.

More recently, Yager [21], [22] proposed PFS which is more general than IFS in that the sum of squares of the membership degree and the nonmembership degree is less than or equal to 1. Formally, let S be an arbitrary nonempty set, a PFS P is a mathematical object of the form

$$P = \{\langle x, P(\mu_P(x), \nu_P(x))\rangle | x \in S\} \tag{9}$$

where $\mu_P(x) : S \to [0,1]$ and $\nu_P(x) : S \to [0,1]$ are, respectively, the membership degree and the nonmembership degree of the element x to S in P, satisfying that $\mu_P^2(x) + \nu_P^2(x) \leq 1$. The hesitant degree of $x \in X$ is expressed as

$$\pi_P(x) = \sqrt{1 - \mu_P^2(x) - \nu_P^2(x)}. \tag{10}$$

For convenience, $\beta = P(\mu_\beta, \nu_\beta)$ is called a Pythagorean fuzzy number (PFN) [48], which satisfies $\mu_\beta, \nu_\beta \in [0,1]$ and $\mu_\beta^2 + \nu_\beta^2 \leq 1$. The following operations are defined on PFN:

$$\begin{aligned}
\beta^{\mathrm{C}} &= P(\nu_\beta, \mu_\beta) \\
\beta_1 + \beta_2 &= P\Big(\sqrt{\mu_{\beta_1}^2 + \mu_{\beta_2}^2 - \mu_{\beta_1}^2\mu_{\beta_2}^2}, \nu_{\beta_1}\nu_{\beta_2}\Big) \\
\beta_1 \times \beta_2 &= P\Big(\mu_{\beta_1}\mu_{\beta_2}, \sqrt{\nu_{\beta_1}^2 + \nu_{\beta_2}^2 - \nu_{\beta_1}^2\nu_{\beta_2}^2}\Big) \\
\lambda\beta &= P\Big(\sqrt{1 - \big(1 - \mu_\beta^2\big)^\lambda}, \nu_\beta^\lambda\Big) \\
\beta^\lambda &= P\Big(\mu_\beta^\lambda, \sqrt{1 - \big(1 - \nu_\beta^2\big)^\lambda}\Big).
\end{aligned}$$

And the following score function [48] and accuracy function [49] can be used for ranking a different PFN:

$$s(\beta) = \mu_\beta^2 - \nu_\beta^2 \tag{11}$$

$$h(\beta) = \mu_\beta^2 + \nu_\beta^2. \tag{12}$$

Based on the two functions, the ranking of any two PFN $\beta = P(\mu_{\beta_1}, \nu_{\beta_1})$ and $\beta = P(\mu_{\beta_2}, \nu_{\beta_2})$ is determined as [49] follows.

1) If $s(\beta_1) < s(\beta_2)$, then $\beta_1 < \beta_2$.
2) If $s(\beta_1) = s(\beta_2)$, then
   a) If $h(\beta_1) < h(\beta_2)$, then $\beta_1 < \beta_2$.
   b) If $h(\beta_1) = h(\beta_2)$, then $\beta_1 = \beta_2$.

IV. Pythagorean-Type Fuzzy Deep Boltzmann Machine

A. Model Description

The proposed PFDBM extends the DBM model by representing the governing parameters with PFN. The reason behind this fuzzified model is threefold.

1) Fuzzified neural networks can handle inputs with fuzzy (labeled) and/or incomplete features [50], which are inevitable in passenger profilers.
2) Fuzzy parameters can also improve the representation ability of DBM by supporting fuzzy probability distribution [51], [52] over cross-layer units, as the principle of incompatibility asserts that high precision is incompatible with high complexity in dealing with complex systems [53], such as passenger profilers.
3) The parameter learning of fuzzy DBM has a larger space than its crisp counterpart, and thus will be more helpful in utilizing the merits of deep learning [45].

Moreover, in comparison with regular fuzzy parameters, PFN parameters enable that each neuron of the model can learn not only how a feature *encourages* the production of the correct output but also how it *discourages* the production of the output because they are characterized by both membership and nonmembership functions, and thus improve the classification accuracy and reduce the vulnerability.

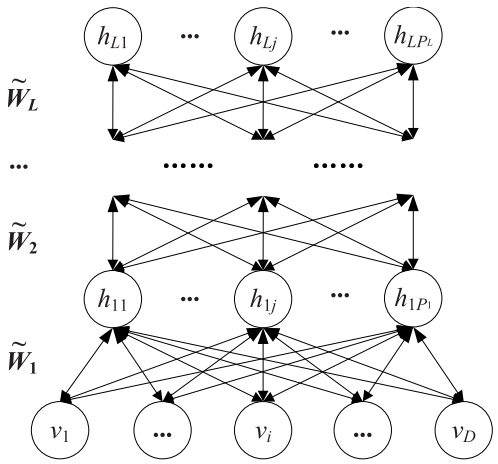


Fig. 3. PFDBM.

Let $\widetilde{\theta} = [\widetilde{\mathbf{W}}_1, \ldots, \widetilde{\mathbf{W}}_L]$ denote the fuzzy parameters of a PFDBM shown in Fig. 3, the fuzzy energy function of a state $\{\mathbf{v}, \mathbf{h}_1, \ldots, \mathbf{h}_L\}$ of the model is defined as

$$\widetilde{E}(\mathbf{v}, \mathbf{h}_1, \ldots, \mathbf{h}_L, \widetilde{\theta}) = -\mathbf{v}^T \widetilde{\mathbf{W}}_1 \mathbf{h}_1 - \sum_{l=2}^{L} \mathbf{h}_{l-1}^T \widetilde{\mathbf{W}}_l \mathbf{h}_l. \tag{13}$$

And the corresponding probability is

$$\widetilde{P}(\mathbf{v}, \widetilde{\theta}) = \frac{1}{\widetilde{Z}(\widetilde{\theta})} \sum_{\mathbf{h}_1} \cdots \sum_{\mathbf{h}_L} \exp(-\widetilde{E}(\mathbf{v}, \mathbf{h}_1, \ldots, \mathbf{h}_L, \widetilde{\theta})). \tag{14}$$

The objective of PFDBM learning is to maximize the likelihood as follows (where $\mathcal{D}$ is the training data set):

$$\max_{\widetilde{\theta}} \widetilde{\mathcal{L}}(\widetilde{\theta}, \mathcal{D}) = \sum_{\mathbf{v} \in \mathcal{D}} \log(\widetilde{P}(\mathbf{v}, \widetilde{\theta})). \tag{15}$$

However, here the function value is a PFN and thus the above optimization problem is inherently a fuzzy maximum problem. In general, such a problem is quite intractable and traditional learning methods may be very inefficient [45]. To tackle this issue, here we employ the score function defined by (11) to defuzzify the result PFN $\beta = P(\mu_\beta, \nu_\beta)$ as

$$c(\beta) = \sqrt{1 + \mu_\beta^2 - \nu_\beta^2}. \tag{16}$$

For example, an interval PFN $\beta = \langle (a,b), (a',b') \rangle$ can be defuzzified as

$$c(\beta) = \sqrt{1 + \frac{(a+b)^2 - (a'+b')^2}{4}}. \tag{17}$$

After defuzzification, the objective function to be optimized becomes

$$\max_{\widetilde{\theta}} \mathcal{L}_c(\widetilde{\theta}, \mathcal{D}) = \sum_{\mathbf{v} \in \mathcal{D}} \log(c(\widetilde{P}(\mathbf{v}, \widetilde{\theta}))). \tag{18}$$

Consequently, the fuzzy optimization problem is transformed into a crisp optimization problem, for which we can apply classical gradient-based optimization methods directly. Next, we propose a learning algorithm integrating a gradient-based method with a metaheuristic for PFDBM.

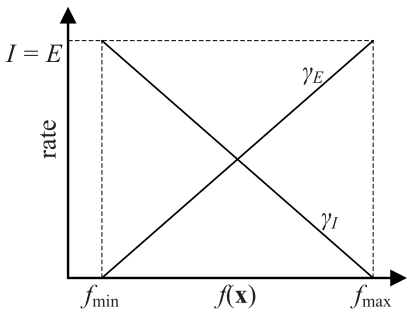


Fig. 4. Linear model of the emigration and immigration rates of BBO.

B. Hybrid Biogeography-Based Learning Algorithm

Although deep learning has achieved great success, its applications on new problems can still be difficult due to the issues, such as training speed, local optima, and manual selection of network structures [54]–[56]. Evolutionary algorithms have demonstrated their capability in optimizing artificial neural networks (ANNs) for decades [57]–[59], but their applications in DNN are very limited [56], [60]–[62].

For training the PFDBM to meet the high accuracy requirements of tasks, such as passenger profiling, we propose a hybrid algorithm integrating the greedy layerwise training method [63] with biogeography-based optimization (BBO) [64], a metaheuristic optimization method inspired by the science of biogeography. When solving an optimization problem, BBO first randomly generates a population of initial solutions to the problem, and then continually evolves the solutions toward the global optimum mainly by "migrating" components probably from more suitable (high-quality) solutions to low-quality ones. The equilibrium theory of BBO indicates that the high-quality solutions have high emigration rates and the low-quality solutions have high immigration rates. In a common linear migration model (shown in Fig. 4), the immigration rate $\gamma_I(\mathbf{x})$ and the emigration rate $\gamma_E(\mathbf{x})$ of a solution $\mathbf{x}$ are calculated as

$$\gamma_I(\mathbf{x}) = I\left(\frac{f_{\max} - f(\mathbf{x}) + \epsilon}{f_{\max} - f_{\min} + \epsilon}\right) \tag{19}$$

$$\gamma_E(\mathbf{x}) = E\left(\frac{f(\mathbf{x}) - f_{\min} + \epsilon}{f_{\max} - f_{\min} + \epsilon}\right) \tag{20}$$

where f denotes the fitness function, $f_{\max}$ and $f_{\min}$ are the maximum and minimum function values in the population, I and E are the maximum possible immigration rate and emigration rate which are typically set to 1, and ϵ is a small constant to avoid zero-division-error.

BBO also has a mutation operator for increasing the diversity of the population. In our algorithm, we use a constant mutation rate γ_M for the worst half of the population, i.e., each solution in the worst half has a probability of γ_M of being mutated, while solutions in the best half will not be modified.

We further modify the basic BBO in two aspects in order to improve its exploration ability. First, a local topology is employed for the population, where each solution is only directly connected to a subset of other solutions (called neighboring solutions) in the population. In contrast to the global topology where all the solutions are interconnected, the local topology can effectively maintain the diversity of the population and avoid premature convergence [65]. In the proposed algorithm, the topology is randomly generated such that each solution has probably k_N neighbors (where k_N is a predefined parameter). If the search efficiency decreases, i.e., no better solution has been found after k_P iterations (where k_P is another parameter), the topology will be randomly reset.

Second, we introduce a control parameter η to represent the "maturity" of the evolution process. The value of η increases with the iteration number t as [66]

$$\eta = \eta_{\min} + \frac{t}{t_{\max}}(\eta_{\max} - \eta_{\min}) \tag{21}$$

where $\eta_{\min}$ and $\eta_{\max}$ are the lower and upper limits of η, and $t_{\max}$ is the maximum number of iterations. When performing a migration on a solution $\mathbf{x}$, the emigrating solution $\mathbf{x}'$ has a probability of η of being selected from the neighbors of $\mathbf{x}$ and a probability of $(1-\eta)$ of being selected from the nonneighbors. This way, a migration is more likely to be performed between nonneighbors in early stages to diversify the search and between neighbors in later stages to enhance exploitation.

The motivation of our hybrid-training algorithm is using the greedy gradient-based method for efficiently locating an optimum in a small region of the solution space, while employing BBO to explore a wider area to avoid being trapped in local optima. The hybrid algorithm applies to the PFDBM layer by layer. At each hidden layer, Algorithm 1 is performed to search an optimal layer structure which includes the number of hidden units and their weight settings (where *rand*() generates a random number uniformly distributed in [0,1]).

Fig. 5 shows the migration operation (Line 17) of Algorithm 1. When training the lth layer, a solution $\mathbf{x}$ contains a number of components $\{x_1, x_2, \ldots\}$, and each component represents a unit of the layer together with the connection weights between the unit and the units of the $(l-1)$th layer. Suppose that the second unit of $\mathbf{x}$ is to be immigrated and $\mathbf{x}'$ is selected as the emigrating solution, and then x_2' will replace x_2 in $\mathbf{x}$ (if there is no corresponding unit in the emigrating solution, the migration will be abandoned).

V. Fuzzy DNN for Passenger Profiling

A. PFDBM-Based DNN for Individual Passenger Profiling

Based on PFDBM, we construct a DNN (the architecture of which is shown in Fig. 6) for identifying the risk of each individual passenger. The input to the DNN includes the following.

1) The passenger name record, including the identity information of the passenger and booking information of the flight.
2) Travel statistics of the passenger's flight history from the Aviation Administration (collected from different airlines).
3) Travel statistics from other transportation departments, such as railway and marine.

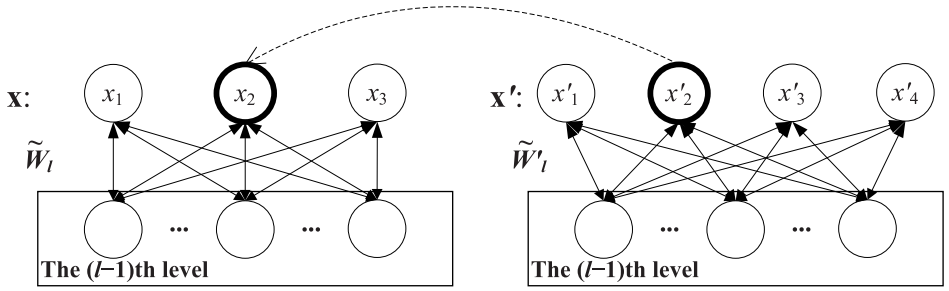


Fig. 5. Illustration of a BBO migration operation of the PFDBM learning algorithm.

Algorithm 1 Hybrid BBO and Greedy Layer-Wise Training Algorithm for PFDBM

```
1  Randomly generate a population of n solutions, each of which represents a design of the layer structure;
2  Randomly initialize a local topology of the population;
3  For each solution x in the population do
4    Use the gradient-based method to train the network constructed so far to maximize the objective Eq. (18);
5    Assign the objective function value calculated as the fitness value to x;
6  End For.
7  Compute the immigration and emigration rates of the solutions according to Eqs. (19) and (20);
8  Update the maturity index η according to Eq. (21);
9  For each solution x in the population do
10   For each component (unit) x_i of the layer do
11     IF rand() < γ_I(x) then
12       IF rand() < η then
13         Select an emigrating solution x′ from the neighbors of x with a probability proportional to γ_E(x′);
14       Else
15         Select an emigrating solution x′ from the non-neighbors of x with a probability proportional to γ_E(x′);
16       End IF.
17       Migrate x′_i to replace x_i of x;
18     End IF.
19   End For.
20   IF x is in the worst half of the population and rand() < γ_M then
21     Mutate x by randomly adding or removing a unit;
22   End IF.
23   Employ the gradient-based method to optimize the network newly constructed;
24   If the objective function value increases then
25     Replace the original solution with x in the population;
26   End If.
27 End For.
28 If the current best solution is not improved for k_P consecutive iterations then
29   Reset the local topology.
30 End If.
31 If the termination condition is not met then go to Line 7 End If.
32 Return the best solution found so far as the current layer structure.
```

4) Travel statistics from the Tourism Administration (collected from travel agencies).
5) Criminal records from the Public Security Department.
6) Educational records from the Education Department.
7) Tax records from the Tax Department.
8) Assert information from banks and the Housing Administration.
9) Consumption records or patterns from large retailers (the detailed records have typically been preprocessed, summarized, and/or mined by retailers' systems).
10) (Preprocessed) telecommunication behavior records or patterns from telecom operators.
11) (Authenticated and preprocessed) Internet behavior records or patterns from Internet operators.

Based on their data types, the input features can be divided into the following four classes.

1) Static binary-valued features, which can be input to the PFDBM directly.
2) Static real-valued features, which are first transformed into binary values using an additional GRBM and then input to the PFDBM.

3) Static labeled features, which are first transformed into real values using membership functions of fuzzy sets defined on the domain of the labels, and then transformed into binary values using GRBM.
4) Dynamic (temporal) features, which are first transformed into real values using an additional temporal filter, and then transformed into binary values using GRBM. Here, we employ the maximizing-discriminability-based recurrent fuzzy network [67] whose output is a function of past output or past input or both as the temporal filter.

In practice, a passenger profile rarely has all the above features filled (e.g., a department or company may refuse to provide some information), and some inputs can be contaminated by noise (e.g., Internet behaviors may be from others who use the same account of the passenger). This is an important reason why we use the PFN parameters to improve the robustness of the learning model (which will be validated in Section VI-B).

It should be noted that, if some input features satisfy specific conditions (e.g., the passenger has some information matching a suspect record from the Public Security Department, or has frequently bought contraband during a recent period), he/she can be directly marked as a potential attacker. Such activation rules have been implemented in our profiling system, but will not be discussed in detail in this paper.

The PFDBM is used for learning risk features, and on the top of the PFDBM we add a Gaussian mixture model (GMM) [68], [69] to produce the likelihood of the passenger being an attacker. That is, the output z of the DNN is calculated from the (implicit) feature vector $\mathbf{y}$ of the topmost layer of the PFDBM

$$z = \mathcal{L}(\mathbf{y}|\lambda_p) = \frac{1}{P}\sum_{i=1}^{P}\log\left(\sum_{j=1}^{N_G} w_j \mathcal{N}(y_i; \mu_j, \Sigma_j)\right) \quad (22)$$

where λ_p is the parameter set for passenger p, P is the dimension of vector $\mathbf{y}$, $\mathcal{N}(y_i; \mu_j, \Sigma_j)$ 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 $\sum_{j=1}^{N_G} w_j = 1$. The GMM is trained with the expectation maximization algorithm [70]. The higher the output value, the higher the likelihood of the passenger being an attacker is. The classification is made using a predefined threshold τ. The claim that the passenger is a (potential) attacker is accepted if $z \geq \tau$ and rejected otherwise. Here, τ can be tuned according to the inspection capability and the current pressure of terrorism, e.g., it can be turned down after the airport receives a threatening phone call.

Thus, the DNN has two preprocessing layers, the PFDBM, and the GMM, as shown in Fig. 6. In our current implementation, the PFDBM has four layers, and the dimension of the input to its first layer is about 7000 (which is expected to increase when more external information is available).

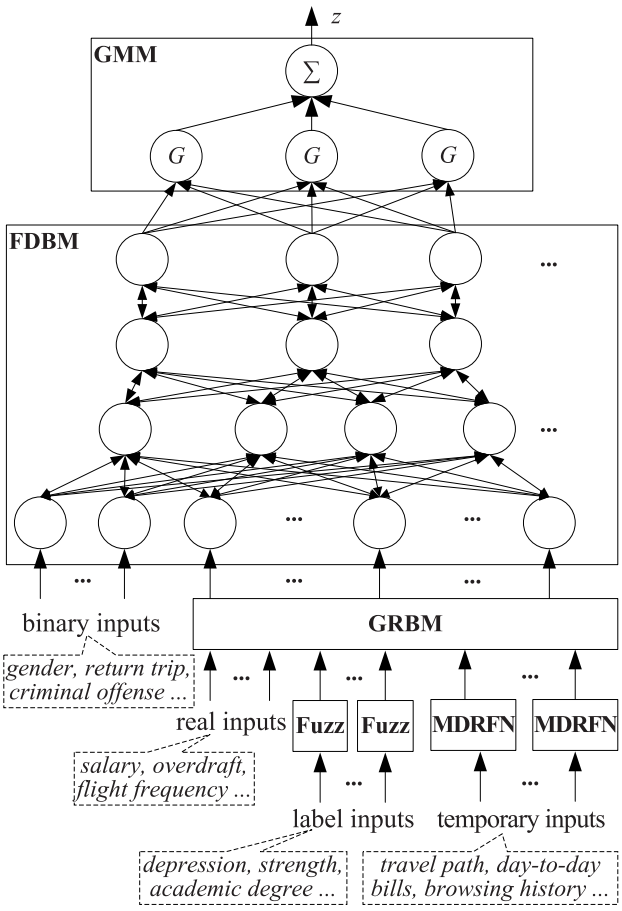


Fig. 6. Structure of the PFDBM-based DNN for individual passenger profiling.

B. Integrated DNN for Criminal Group Identification

An attack is often conducted by a group of attackers. However, in many cases, the features of an individual attacker in the group are insufficient to reveal the abnormality—only a conjoint analysis on the relationships among the group has a possibility. For this task, we construct an integrated DNN on top of a set of K individual profiling DNNs of passengers in the same flight. By experiments, we find that it is better to use the union of the third layers of the sub-DNNs as the input layer of the integrated DNN. If we use the topmost (fourth) layers of the sub-DNNs, the identification ability of the integrated DNN will decrease; if we use the lower layers, the computational cost will increase significantly (e.g., when using the first layers of the sub-DNNs, the dimension of the input to the integrated DNN will be around 700 000 for a flight with 100 passengers).

The integrated DNN consists of four layers (including the input layer from the sub-DNNs), as shown in Fig. 7. Here, we do not add an additional classifier on top of the integrated DNN. Instead, we directly use the state of its topmost layer which consists of K neurons as the output. For the kth component of the output vector, a value of 1 indicates that the kth passenger belongs to the group and 0 otherwise; if the output is an all-zero vector, we need only to inspect those passengers identified as attackers by individual sub-DNNs. The integrated DNN is trained by the same algorithm for the sub-DNN model, under the premise that the sub-DNNs have been fixed.

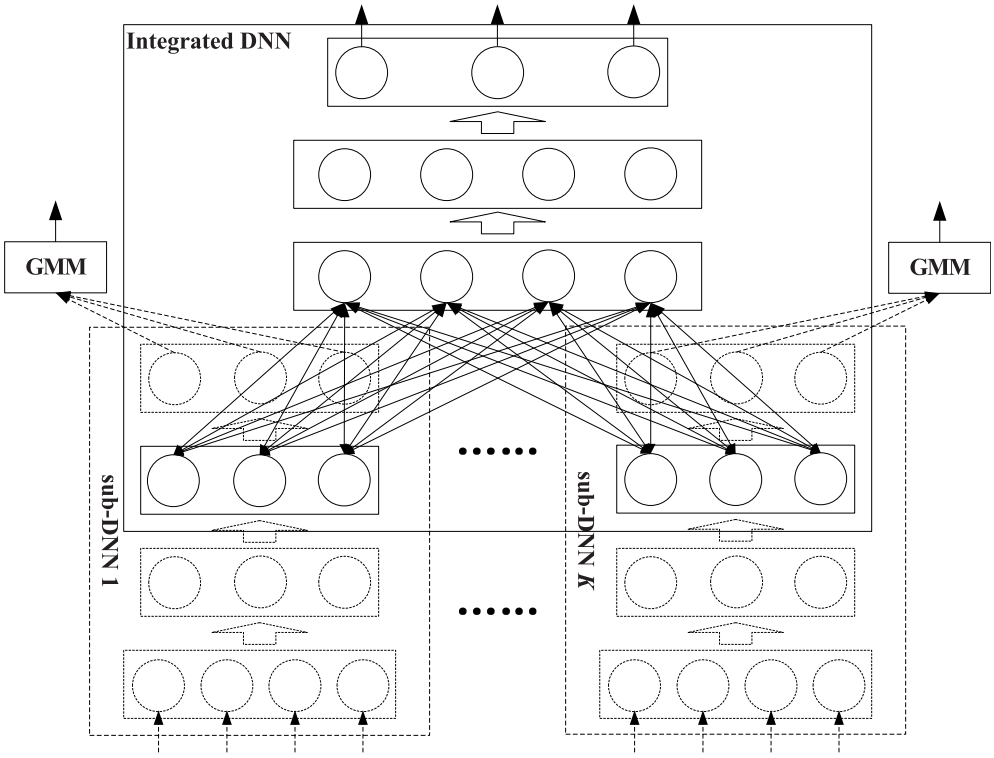


Fig. 7. Integrated DNN.

VI. Experiments

A. Experiments of the Individual Profiling DNN

We use a data set consisting of 12 000 passenger profiles for training and testing the proposed model. There are 11 000 normal passenger profiles and 162 attacker profiles taken from real-world flight records from Air China, while the other 838 attacker profiles are virtually constructed by experts in aviation security and crime.

Since, in real-world applications, the sample of attackers is often very small, here we do not use the normal k-fold cross validation. Instead, we partition the 11 000 normal passengers into five equal sized pieces (2200 per piece) but do not equally partition the attackers; the validation is run for 5 times, and at each time, we use four pieces of normal passengers together with randomly selected 995 attackers as the training set, and use the remaining 2200 normal passengers and five attackers as the test set. Consequently, the combination of the validations can be regarded as a task of identifying 25 attackers among 11 025 passengers (which is about the daily number of departing passengers of a medium-sized airport in China).

We have also implemented the following six models for comparison with our FPDBM-based DNN with hybrid gradient and *enhanced* BBO learning (denoted as FPDBM-EBO).

1) A standard three-layer feed-forward ANN previously constructed and used by an airport (denoted as ANN). Its first layer accepts a 296-D input vector.
2) The extension of the above ANN model whose first layer accepts a 7000-D input vector as our model (denoted as L-ANN).
3) A DNN which employs the preprocessing layers and output classifier as our model, but uses the classical DBM [16] as the central learning model (denoted as DBM).
4) A fuzzy DNN which employs the preprocessing layers and output classifier as our model, but uses a deep fuzzy RBM model whose parameters are represented by regular fuzzy numbers [45] as the central learning model (denoted as FDBM).
5) Our FPDBM-based DNN which is trained only by the greedy layer-wise method [63] (denoted as PFDBM-G).
6) Our FPDBM-based DNN with hybrid gradient and *basic* BBO learning, i.e., the local topology and the maturity parameter are not used (denoted as PFDBM-BBO).

For our enhanced BBO learning algorithm, we set $\gamma_M = 0.05$, $k_N = 3$, and $k_P = 5$. Both the basic BBO and our enhanced BBO use the same population size of 30. For a fair comparison, all the seven models use the same maximum number of training iterations of 3000. The experimental environment is a Windows 2003 server with 4× Intel Xeon 3430 CPU and 4×2 GB DDR3 memory.

In evaluation, we employ the *precision* (confidence), *recall* (sensitivity), and *fallout* measures [71], which denote what percentage of passengers identified as attackers are actually such, what percentage of attackers are identified as such, and what percentage of normal passengers are identified as attackers, respectively

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{23}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{24}$$

$$\text{fallout} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{25}$$

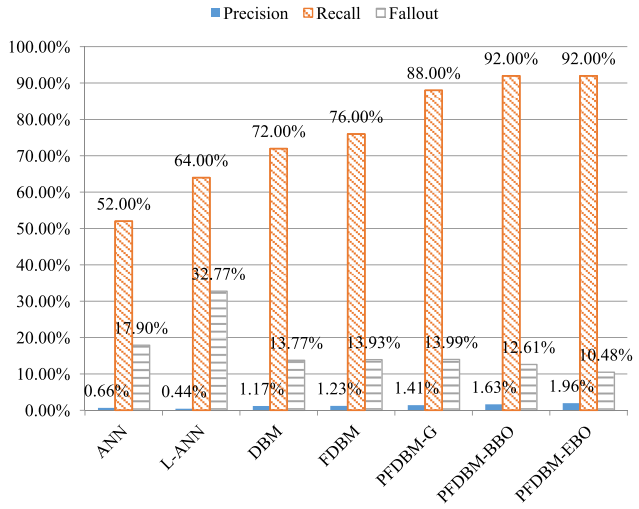


Fig. 8. *Precision*, *recall*, and *fallout* results of the seven models for individual passenger profiling.

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

Fig. 8 compares the performance of the seven models for individual passenger profiling. As we can see from the results, it has the following.

1) The old ANN can only identify 52% of the attackers mainly due to the limited number of input features and the poor feature learning ability of the model. Thus, it can only be utilized as a reference tool—If the decision maker completely relies on it, about half of the attackers will break through.
2) By using a much larger and more comprehensive set of input features, the L-ANN improves the percentage of attackers correctly identified (recall) to 64%. However, its precision becomes lower and its fallout becomes much higher than the ANN, mainly because the shallow learning model suffers from overfitting when the input is high-dimensional and the target function is highly complex [59]. A fallout nearly 33%, which means that one of every three normal passengers will be sent for in-depth inspection, is obviously unacceptable.
3) In comparison with the shallow model, the deep learning models can both improve precision and recall and reduce fallout significantly, demonstrating their abilities of representing and learning risk features for profiling.
4) In comparison with the classical DBM model, FDBM achieves better precision and recall (at the expense of a slightly higher fallout), which shows that using fuzzy parameters can actually improve the representation ability of the model.
5) Similarly, in comparison with FDBM, our PFDBM achieves much better precision and recall (the difference of fallout between FDBM and PFDBM is trivial), which shows that using PFN parameters can further improve the model ability by modeling the impacts of input features on the output from both the positive and negative sides.
6) When training our PFDBM, the single gradient-based method suffers premature convergence, whereas the proposed hybrid evolutionary learning algorithms are capable of jumping out of local optima by using a population of solutions to explore much wider areas. That is why PFDBM-BBO and PFDBM-EBO have higher precision and recall and lower fallout than PFDBM-G.
7) PFDBM-BBO and PFDBM-EBO achieve the same recall, but the latter has higher precision and lower fallout. This shows that the local topology and the maturity parameter used in PFDBM-EBO are useful in improving the effectiveness of learning.

TABLE I

AVERAGE CPU TIMES (IN SECONDS) FOR TRAINING AND CLASSIFYING 100 INDIVIDUAL TUPLES CONSUMED BY THE DIFFERENT MODELS

Model	L-ANN	DBM	FDBM	PFDBM-G	PFDBM -BBO	PFDBM -EBO
T_t	11.6	53.8	82.0	93.9	275.9	290.8
T_c	2.12	3.09	3.39	4.66	4.66	4.66

Table I presents the average training time (denoted by T_t) and classification time (denoted by T_c) for every 100 individual tuples (approximately the capacity of a medium-sized aircraft) of the six learning models using 7000-D inputs. As we can see, L-ANN uses the least training time, but its accuracy is unacceptable; introducing fuzzy parameters to DBM requires about 50% more training time, and the hybrid algorithm consumes about three times of the training time of the greedy layerwise algorithm, but the performance improvement is considerable. However, using PFN to replace regular fuzzy numbers in the model and using EBO to replace BBO in the hybrid algorithm do not significantly increase the training time. There are no significant differences among the classification times of the different models.

B. Experiments of the Individual Profiling DNN on Incomplete Data

To test the robustness of the models, we randomly select 5% normal passenger tuples and 5% attacker tuples, and for each tuple randomly set 5%–15% features to empty. We then reuse the above procedure to train and test the seven models on the incomplete data set.

Fig. 9 shows the experimental results of the seven models on the incomplete data set. As we can see, for the first four models (i.e., non-PFDBM models), precision, recall, and fallout all become worse than those on the original data set. However, the performance advantage of FDBM over DBM becomes more significant than that on the original data set, showing that the fuzzy deep learning model is more robust than the classical one. An interesting finding is that, for the PFDBM-G model, its fallout on the incomplete data set is better than that on the original data set, i.e., the percentage of normal passengers that are wrongly identified reduces on the incomplete data set. Although its precision and recall still deteriorate, its performance advantage over FDBM becomes more significant. This indicates that our PFDBM model has more robustness than FDBM in dealing with incompleteness.

Moreover, as on the original data set, the two PFDBM models with hybrid evolutionary learning exhibit better

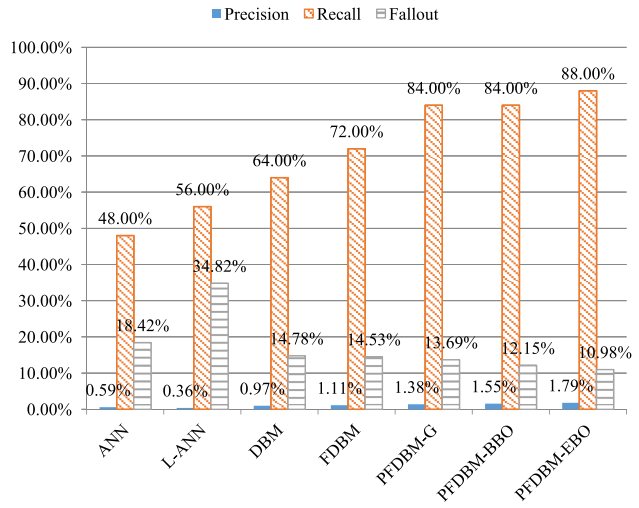


Fig. 9. *Precision*, *recall*, and *fallout* results of the seven models on the incomplete data set.

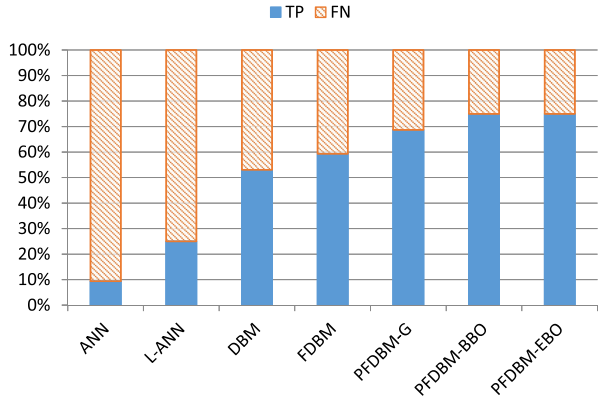


Fig. 10. Classification of faking attackers of the seven models.

performance than PFDBM-G, and PFDBM-EBO performs better than PFDBM-BBO on the incomplete data set, demonstrating that our hybrid learning algorithm is competitive on both the cases.

C. Experiments of Vulnerability to Faking

To validate the vulnerability of our model to faking, we invite police experts and reclaimed criminals to act as attackers that try to disguise their attributes and behaviors to defeat the profiler. Of course, they cannot directly access the profiles. Based on their experiences, the "attackers" use various deception measures, including simple frauds such as falsifying their occupations and asserts, and sophisticated tricks such as booking a hotel (that they would never check in) in the destination city of the flight and "honestly" declaring and paying taxes in a few days before the trip.

We construct 32 data tuples of the "faking attackers", and use them as inputs to the seven classification models described earlier. Fig. 10 shows the comparative results of the models on the tuples. As we can see, the two shallow ANN models are "defeated" by most attackers, while the DNN models can correctly identify a majority of faking attackers. Such a difference demonstrates that using more hidden layers can represent the probabilistic relationship between the input features and the risk not only more accurately but also more implicitly and holistically, and thus make the DNN less vulnerable to faking features. The performance advantage of FDBM over DBM further shows that using fuzzy parameters can improve the representation ability and reduce vulnerability of the deep model. Similarly, the comparative results of FDBM, the basic PFDMB, and PFDBM with hybrid evolutionary learning demonstrate the contributions of Pythagorean fuzzy parameters and evolutionary learning to the reduction of vulnerability.

D. Experiments of the Integrated DNN

As shown in Fig. 7, an input data tuple of the integrated DNN is a union of K profiles of passengers of one flight (which we called a "flight profile"). We use a data set of 150 tuples for training and testing the integrated DNN, where 125 tuples are union of the real-world tuples used in Section VI-A. The 150 flights have a total of 12 500 passengers *that are recognized as normal passengers by the sub-DNNs*, including 42 attackers belonging to 10 attacker groups (each with 3–5 attackers). Note here we assume that a flight can have at most one attacker group, and when training and testing the integrated DNN we only focus on the identification of attackers in the groups—It is the responsibility of the sub-DNNs to identify individual "acted-alone" attackers.

For the "group identification" problem, we extend the measures in (23)–(25) as [72]

$$\text{precision} = \frac{1}{m}\sum_{i=1}^{m}\frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i} \tag{26}$$

$$\text{recall} = \frac{1}{m}\sum_{i=1}^{m}\frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i} \tag{27}$$

$$\text{fallout} = \frac{1}{m}\sum_{i=1}^{m}\frac{\text{FP}_i}{\text{FP}_i + \text{TN}_i} \tag{28}$$

where m is the number of tuples (flight profiles), TP_i is the number of correctly identified group attackers *that have been recognized as normal passengers by the sub-DNNs* in the ith flight, and FP_i, FN_i, and TN_i have corresponding meanings.

In addition, we define the following four new measures for evaluating the effectiveness of "group identification:"

$$\text{sr}_A = \frac{G_A}{G_A + G_B + G_C + G_D} \tag{29}$$

$$\text{sr}_B = \frac{G_B}{G_A + G_B + G_C + G_D} \tag{30}$$

$$\text{sr}_C = \frac{G_C}{G_A + G_B + G_C + G_D} \tag{31}$$

$$\text{sr}_D = \frac{G_D}{G_A + G_B + G_C + G_D} \tag{32}$$

where G_A is the number of attacker groups where all the members are identified by the model, G_B is the number of groups where not all but at least half of the members are identified, G_C is the number of groups where at least one but

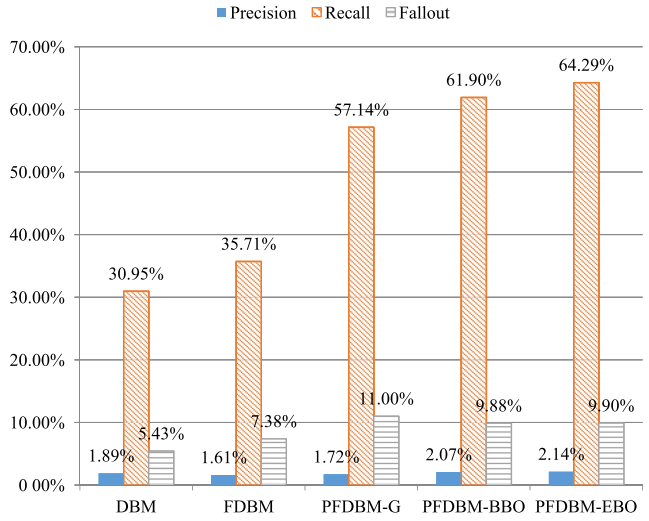


Fig. 11. *Precision*, *recall*, and *fallout* results of the five models for attacker group identification.

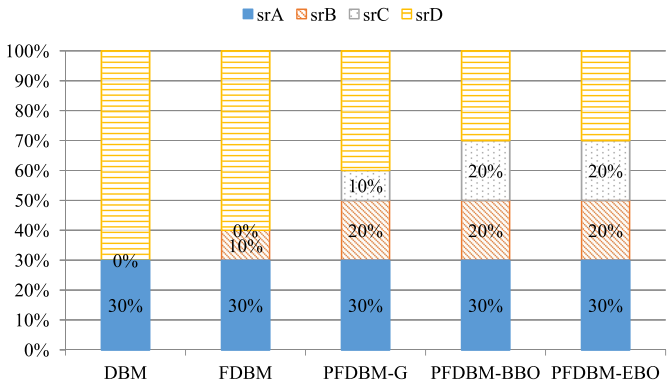


Fig. 12. sr_A, sr_A, sr_C, and sr_D results of the five models for attacker group identification.

TABLE II

AVERAGE CPU TIMES (IN SECONDS) FOR TRAINING AND CLASSIFYING A GROUP TUPLE CONSUMED BY THE DIFFERENT MODELS

Model	DBM	FDBM	PFDBM-G	PFDBM-BBO	PFDBM-EBO
T_t	21.6	38.9	51.4	128.2	138.0
T_c	1.96	2.43	2.75	2.75	2.75

less than half of the members are identified, and G_D is the number of groups where no member is identified.

For comparison with the PFDBM-EBO model, we have, respectively, implemented the DBM, FDBM, PFDBM-G, and PFDBM-BBO models for the integrated DNN, all using the same input layer (i.e., the union of the third layers of the sub-DNNs for individual passengers).

For each model, a standard fivefold cross validation is conducted on the data set. Fig. 11 comparatively shows the *precision*, *recall*, and *fallout* results of the five models for group identification. As we can see, the three PFDBM models achieve much higher *precision* than DBM and FDBM, showing that our Pythagorean-type fuzzy DBM model can mine group behaviors and features much more effectively than the crisp model and the regular fuzzy model (at the expense of a slightly higher fallout). Moreover, PFDBM-BBO outperforms PFDBM-G and PFDBM-EBO outperforms PFDBM-BBO on all three measures, which demonstrates that our hybrid evolutionary learning algorithm is also very effective for training the integrated DNN.

Fig. 12 shows the results of sr_A, sr_A, sr_C, and sr_D of the five models. An interesting finding is that all five models have the same sr_A value of 3, i.e., they all "completely defeat" three attacker groups. However, DBM cannot identify any more attackers; FDBM "heavily beats" another group but cannot identify any more attackers; the three PFDBM models "heavily beat" two groups and also identify some attackers in other groups. This further demonstrates the advantages of fuzzy deep learning models, in particular, our Pythagorean-type models, in identifying attackers in group—typically, a criminal group will crumble if it has lost half or more of the members; even if only one member has been caught, the attack operation is very likely to be canceled because of the disruption of the plan, the psychological blow to the other members, and the possible confession of the arrestee.

Table II presents the average training time and classification time for each group tuple of the five models. As we can see, the time differences are similar to that among the models for individual profiling, and the computational costs are generally acceptable for practical applications.

VII. CONCLUSION

This paper presents a deep learning approach to passenger profiling that is of critical importance in aviation security. The key component of the DNN is a novel PFDBM model whose governing parameters are expressed based on PFSs. We propose for the PFDBM a hybrid learning algorithm that employs an evolutionary metaheuristic for facilitating exploration and a gradient-based method for enhancing exploitation. Experimental results show that the proposed DNN with evolutionary learning exhibits competitive performance on the training data sets.

To the best of our knowledge, it is the first study on Pythagorean-type DBM as well as the first work on deep learning for passenger profiling. We believe that our PFDBM model can be adapted and applied to many other deep learning tasks in computer science and management science. Now, we are employing and testing some other state-of-the-art evolutionary algorithms (such as particle swarm optimization [73] and water wave optimization [74]) in training the fuzzy deep models.

This paper does not study the impacts of the passenger profiling on privacy, which we believe should be seriously addressed by relevant laws and regulations. Technologies, however, can provide assistance in privacy protection. Currently, we are developing a multigrade DNN that integrates a low-grade DNN that only uses nonsensitive features or a very small number of sensitive features and a high-grade DNN that relies on sensitive features. Only when the risk index inferred by the low-grade DNN exceeds a threshold, can sensitive features be acquired and used for the input to the high-grade DNN for a more accurate classification. We firmly

believe that scientific achievements, if appropriately delivered, will make our traveling more convenient and efficient, not less.

REFERENCES

[1] H. E. Reser, "Airline terrorism: The effect of tightened security on the right to travel," *J. Air Law Commerce*, vol. 63, no. 4, pp. 819–848, 1998.

[2] X. Wang and J. Zhuang, "Balancing congestion and security in the presence of strategic applicants with private information," *Eur. J. Oper. Res.*, vol. 212, no. 1, pp. 100–111, Jul. 2011.

[3] H. Cavusoglu, B. Koh, and S. Raghunathan, "An analysis of the impact of passenger profiling for transportation security," *Oper. Res.*, vol. 58, no. 5, pp. 1287–1302, 2010.

[4] T. M. Ravich, "Airline passenger profiling systems after 9/11: Personal privacy versus national security," *J. Transp. Res. Forum*, vol. 44, no. 2, pp. 127–141, 2010.

[5] J. Rosen, "The silver bullet: Protecting privacy and security through law and technology," *Proc. Amer. Philos. Soc.*, vol. 151, no. 3, pp. 291–299, 2007.

[6] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1263–1284, Sep. 2009.

[7] B. Gu, X. Sun, and V. S. Sheng, "Structural minimax probability machine," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2016.2544779.

[8] L. A. McLay, A. J. Lee, and S. H. Jacobson, "Risk-based policies for airport security checkpoint screening," *Transp. Sci.*, vol. 44, no. 3, pp. 333–349, Aug. 2010.

[9] S. Chakrabarti and A. Strauss, "Carnival booth: An algorithm for defeating the computer-assisted passenger screening system," *First Monday*, vol. 10, no. 7, Oct. 2002. [Online]. Available: http://firstmonday.org/

[10] S. Koc-Menard, "Trends in terrorist detection systems," *J. Homeland Secur. Emerg. Manage.*, vol. 6, no. 1, pp. 16–21, 2009.

[11] H. He, S. Chen, K. Li, and X. Xu, "Incremental learning from stream data," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 1901–1914, Dec. 2011.

[12] Z. Xia, X. Wang, X. Sun, and Q. Wang, "A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 2, pp. 340–352, Feb. 2016.

[13] Y.-J. Zheng, Q.-Z. Chen, H.-F. Ling, and J.-Y. Xue, "Rescue wings: Mobile computing and active services support for disaster rescue," *IEEE Trans. Serv. Comput.*, vol. 9, no. 4, pp. 594–607, Jul./Aug. 2016.

[14] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[15] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[16] R. Salakhutdinov and G. Hinton, "Deep Boltzmann machines," in *Proc. AISTATS*, Clearwater Beach, FL, USA, 2009, pp. 448–455.

[17] A. Stuhlsatz, J. Lippel, and T. Zielke, "Feature extraction with deep neural networks by a generalized discriminant analysis," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 4, pp. 596–608, Apr. 2012.

[18] M. Bianchini and F. Scarselli, "On the complexity of neural network classifiers: A comparison between shallow and deep architectures," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 8, pp. 1553–1565, Aug. 2014.

[19] L. Shao, D. Wu, and X. Li, "Learning deep and wide: A spectral method for learning deep networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 12, pp. 2303–2308, Dec. 2014.

[20] C. Yan, Y. Zhang, F. Dai, X. Wang, L. Li, and Q. Dai, "Parallel deblocking filter for HEVC on many-core processor," *Electron. Lett.*, vol. 50, no. 5, pp. 367–368, Feb. 2014.

[21] R. R. Yager, "Pythagorean fuzzy subsets," in *Proc. IFSA/NAFIPS*, Edmonton, AB, Canada, Jun. 2013, pp. 57–61.

[22] R. R. Yager, "Pythagorean membership grades in multicriteria decision making," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 4, pp. 958–965, Aug. 2014.

[23] R. W. Hahn, "The economics of airline safety and security: An analysis of the White House Commission's recommendations," *Harvard J. Law Public Policy*, vol. 20, no. 3, pp. 791–827, 1997.

[24] A. Barnett, "CAPPS II: The foundation of aviation security?" *Risk Anal.*, vol. 24, no. 4, pp. 909–916, Aug. 2004.

[25] S. W. Dummer, "False positives and secure flight using dataveillance when viewed through the ever increasing likelihood of identity theft," *J. Technol. Law Policy*, vol. 11, no. 2, pp. 259–285, 2006.

[26] J. L. Virta, S. H. Jacobson, and J. E. Kobza, "Analyzing the cost of screening selectee and non-selectee baggage," *Risk Anal.*, vol. 23, no. 5, pp. 897–908, Oct. 2003.

[27] L. A. McLay, S. H. Jacobson, and J. E. Kobza, "A multilevel passenger screening problem for aviation security," *Naval Res. Logistics*, vol. 53, no. 3, pp. 183–197, Apr. 2006.

[28] V. L. L. Babu, R. Batta, and L. Lin, "Passenger grouping under constant threat probability in an airport security system," *Eur. J. Oper. Res.*, vol. 168, no. 2, pp. 633–644, Jan. 2006.

[29] C. F. Manski, "Profiling: Introduction to the feature," *Econ. J.*, vol. 116, no. 515, pp. F347–F350, Nov. 2006.

[30] R. A. Hudson, "Who becomes a terrorist and why: The 1999 government report on profiling terrorists," Guilford, CT, USA: The Lyons Press, 2002.

[31] H. Cavusoglu, Y. Kwark, B. Mai, and S. Raghunathan, "Passenger profiling and screening for aviation security in the presence of strategic attackers," *Decision Anal.*, vol. 10, no. 1, pp. 63–81, 2013.

[32] Y.-J. Zheng, H.-F. Ling, J.-Y. Xue, and S.-Y. Chen, "Population classification in fire evacuation: A multiobjective particle swarm optimization approach," *IEEE Trans. Evol. Comput.*, vol. 18, no. 1, pp. 70–81, Feb. 2014.

[33] X. Wen, L. Shao, Y. Xue, and W. Fang, "A rapid learning algorithm for vehicle classification," *Inf. Sci.*, vol. 295, pp. 395–406, Feb. 2015.

[34] B. Gu and V. S. Sheng, "A robust regularization path algorithm for ν-support vector classification," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2016.2527796.

[35] B. Mai, S. Raghunathan, H. Cavusoglu, and B. Koh, "Economics of user segmentation, profiling, and detection in security," in *Proc. Workshop Econ. Inf. Secur. (WEIS)*, Pittsburgh, PA, USA, June 2007.

[36] C. Yan *et al.*, "A highly parallel framework for HEVC coding unit partitioning tree decision on many-core processors," *IEEE Signal Process. Lett.*, vol. 21, no. 5, pp. 573–576, May 2014.

[37] C. Yan *et al.*, "Efficient parallel framework for HEVC motion estimation on many-core processors," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 12, pp. 2077–2089, Dec. 2014.

[38] L. A. Zadeh, "Fuzzy sets," *Inf. Control*, vol. 8, no. 3, pp. 338–353, Jun. 1965.

[39] B. Kosko, "Counting with fuzzy sets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 4, pp. 556–557, Jul. 1986.

[40] Y. Y. Lin, J. Y. Chang, and C. T. Lin, "Identification and prediction of dynamic systems using an interactively recurrent self-evolving fuzzy neural network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 2, pp. 310–321, Feb. 2013.

[41] Y.-J. Zheng, H.-F. Ling, S.-Y. Chen, and J.-Y. Xue, "A hybrid neuro-fuzzy network based on differential biogeography-based optimization for online population classification in earthquakes," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 4, pp. 1070–1083, Aug. 2015.

[42] J. T. Luxhoj and M. Hadjimichael, "A hybrid fuzzy-belief network (HFBN) for modelling aviation safety risk factors," *Human Fact. Aerosp. Safety*, vol. 6, no. 3, pp. 191–215, 2006.

[43] M. Hadjimichael, "A fuzzy expert system for aviation risk assessment," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 6512–6519, Apr. 2009.

[44] J. Skorupski and P. Uchroński, "Fuzzy inference system for the efficiency assessment of hold baggage security control at the airport," *Safety Sci.*, vol. 79, pp. 314–323, Nov. 2015.

[45] C. L. P. Chen, C.-Y. Zhang, L. Chen, and M. Gan, "Fuzzy restricted Boltzmann machine for the enhancement of deep learning," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 6, pp. 2163–2173, Dec. 2015.

[46] P. Smolensky, "Information processing in dynamical systems: Foundations of harmony theory," in *Parallel Distributed Processing: Foundations*, vol. 1, D. E. Rumelhart and J. L. McClelland, Eds. Cambridge, MA, USA: MIT Press, 1986.

[47] K. T. Atanassov, "Intuitionistic fuzzy sets," *Fuzzy Sets Syst.*, vol. 20, no. 1, pp. 87–96, Aug. 1986.

[48] X. Zhang and Z. Xu, "Extension of TOPSIS to multiple criteria decision making with pythagorean fuzzy sets," *Int. J. Intell. Syst.*, vol. 29, no. 12, pp. 1061–1078, Dec. 2014.

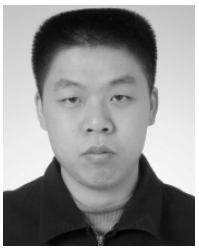
[49] P. Ren, Z. Xu, and X. Gou, "Pythagorean fuzzy TODIM approach to multi-criteria decision making," *Appl. Soft Comput.*, vol. 42, pp. 246–259, May 2016.

[50] H. Ishibuchi, K. Morioka, and I. B. Turksen, "Learning by fuzzified neural networks," *Int. J. Approx. Reason.*, vol. 13, no. 4, pp. 327–358, Nov. 1995.

[51] E. P. Klement, W. Schwyhla, and R. Lowen, "Fuzzy probability measures," *Fuzzy Sets Syst.*, vol. 5, no. 1, pp. 21–30, Jan. 1981.

[52] R. R. Yager, "Decision making with fuzzy probability assessments," *IEEE Trans. Fuzzy Syst.*, vol. 7, no. 4, pp. 462–467, Aug. 1999.

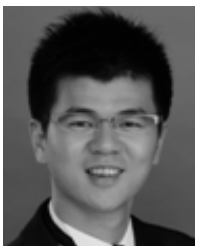
[53] H. Ishibuchi, H. Tanaka, and H. Okada, "Fuzzy neural networks with fuzzy weights and fuzzy biases," in *Proc. IEEE Int. Conf. Neural Netw.*, vol. 3. San Francisco, CA, USA, Mar./Apr. 1993, pp. 1650–1655.
[54] S. Chen and Z. Wang, "Acceleration strategies in generalized belief propagation," *IEEE Trans. Ind. Informat.*, vol. 8, no. 1, pp. 41–48, Feb. 2012.
[55] W. Hu, W. Li, X. Zhang, and S. Maybank, "Single and multiple object tracking using a multi-feature joint sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 4, pp. 816–833, Apr. 2015.
[56] S. Lander and Y. Shang, "EvoAE—A new evolutionary method for training autoencoders for deep learning networks," in *Proc. COMPSAC*, vol. 2. Taichung, Taiwan, Jul. 2015, pp. 790–795.
[57] X. Yao, "A review of evolutionary artificial neural networks," *Int. J. Intell. Syst.*, vol. 8, no. 4, pp. 539–567, 1993.
[58] A. G. B. Tettamanzi and M. Tomassini, *Soft Computing: Integrating Evolutionary, Neural, and Fuzzy Systems*. Berlin, Germany: Springer-Verlag, 2013.
[59] J. Tian, M. Li, F. Chen, and N. Feng, "Learning subspace-based RBFNN using coevolutionary algorithm for complex classification tasks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 1, pp. 47–61, Jan. 2016.
[60] O. E. David and I. Greental, "Genetic algorithms for evolving deep neural networks," in *Proc. GECCO*. Vancouver, BC, Canada, 2014, pp. 1451–1452.
[61] T. Shinozaki and S. Watanabe, "Structure discovery of deep neural network based on evolutionary algorithms," in *Proc. ICASSP*, Brisbane, QLD, Australia, Apr. 2015, pp. 4979–4983.
[62] J. P. Papa, W. Scheirer, and D. D. Cox, "Fine-tuning deep belief networks using harmony search," *Appl. Soft Comput.*, vol. 46, pp. 875–885, Sep. 2016.
[63] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Advances in Neural Information Processing Systems (NIPS)*, vol. 19, J. P. B. Schölkopf and T. Hoffman, Eds. Cambridge, MA, USA: MIT Press, 2007, pp. 153–160.
[64] D. Simon, "Biogeography-based optimization," *IEEE Trans. Evol. Comput.*, vol. 12, no. 6, pp. 702–713, Dec. 2008.
[65] Y.-J. Zheng, H.-F. Ling, X.-B. Wu, and J.-Y. Xue, "Localized biogeography-based optimization," *Soft Comput.*, vol. 18, no. 11, pp. 2323–2334, 2014.
[66] Y.-J. Zheng, H.-F. Ling, and J.-Y. Xue, "Ecogeography-based optimization: Enhancing biogeography-based optimization with ecogeographic barriers and differentiations," *Comput. Oper. Res.*, vol. 50, pp. 115–127, Oct. 2014.
[67] G.-D. Wu, Z.-W. Zhu, and P.-H. Huang, "A TS-type maximizing-discriminability-based recurrent fuzzy network for classification problems," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 2, pp. 339–352, Apr. 2011.
[68] J. Gauvain and C.-H. Lee, "Maximum *a posteriori* estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Trans. Speech Audio Process.*, vol. 2, no. 2, pp. 291–298, Apr. 1994.
[69] F. Cardinaux, C. Sanderson, and S. Marcel, "Comparison of MLP and GMM classifiers for face verification on XM2VTS," in *Audio- and Video-Based Biometric Person Authentication*, J. Kittler and M. S. Nixon, Eds. Berlin, Germany: Springer, 2003, pp. 911–920.
[70] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. Soc., B (Methodol.)*, vol. 39, no. 1, pp. 1–38, 1977.
[71] A. R. Webb and K. D. Copsey, *Statistical Pattern Recognition*, 3rd ed. New York, NY, USA: Wiley, 2011.
[72] S. Godbole and S. Sarawagi, "Discriminative methods for multi-labeled classification," in *Advances in Knowledge Discovery and Data Mining*, H. Dai, R. Srikant, and C. Zhang, Eds. Berlin, Germany: Springer, 2004, pp. 22–30.
[73] A. A. A. Esmin, R. A. Coelho, and S. Matwin, "A review on particle swarm optimization algorithm and its variants to clustering high-dimensional data," *Artif. Intell. Rev.*, vol. 44, no. 1, pp. 23–45, Jun. 2015.
[74] Y.-J. Zheng, "Water wave optimization: A new nature-inspired metaheuristic," *Comput. Oper. Res.*, vol. 55, no. 1, pp. 1–11, Mar. 2015.



Yu-Jun Zheng (M'06) received the Ph.D. degree from the Institute of Software, Chinese Academy of Sciences, Beijing, China, in 2010.

He is currently an Associate Professor with the Zhejiang University of Technology, Hangzhou, China. He has authored over 50 papers in famous journals, such as the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, the IEEE TRANSACTIONS ON FUZZY SYSTEMS, and the IEEE TRANSACTIONS ON SERVICES COMPUTING. His current research interests include bioinspired computing and operations research.

Dr. Zheng is an ACM member. In 2014, he received the runner-up of IFORS Prize for development due to the work of evolutionary optimization of emergency engineering rescue tasks scheduling in disaster relief operations in China.

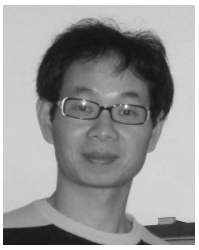


Wei-Guo Sheng (M'13) received the M.S. degree in information technology from The University of Nottingham, Nottingham, U.K., in 2002, and the Ph.D. degree in computer science from Brunel University London, Uxbridge, U.K., in 2005.

He was a Researcher with the University of Kent, Canterbury, U.K., and Royal Holloway, University of London, Egham, U.K. In 2011, he was with the Zhejiang University of Technology, Hangzhou, China, where he is currently a Professor in computer science. His current research interests include evolutionary computation, data mining, and machine learning.

Xing-Ming Sun (SM'07) received the M.S. degree in computing science from the Dalian University of Science and Technology, Dalian, China, in 1988, and the Ph.D. degree in computer science from Fudan University, Shanghai, China, in 2001.

He is currently a Professor with the College of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, China. His current research interests include artificial neural networks and cloud computing.



Sheng-Yong Chen (SM'10) received the Ph.D. degree from the City University of Hong Kong, Hong Kong, in 2003.

He is currently a Professor and a Ph.D. Advisor with the Zhejiang University of Technology, Hangzhou, China. He has authored over 100 scientific papers in international journals and conferences. His current research interests include evolutionary computation and intelligent systems.

Dr. Chen is an IET Fellow. In 2013, he achieved the National Outstanding Youth Fund of China.