Directional Adversarial Training for Robust Ownership-Based Recommendation System

ZHEFU WU, AGYEMANG PAUL, JUNZHUO CAO, AND LUPING FANG
College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, P.R.China
Corresponding author: Luping Fang (e-mail: flp@zjut.edu.cn).

ABSTRACT Machine learning algorithms are susceptible to cyberattacks, posing security problems in computer vision, speech recognition, and recommendation systems. So far, researchers have made great strides in adopting adversarial training as a defensive strategy. Single-step adversarial training methods have been proposed as viable solutions for improving model generality and resilience. However, there has been little study to address this issue in the context of ownership-based recommendations, which may fail to capture stable results. In this work, we adapt the single-step adversarial training for ownership recommendation systems. Our main technical contributions are as follows: (1) We propose Adversarial Consumption and Production Relationship (ACPR), a model that combines factorization machine and single-step adversarial training for ownership recommendations. It enables us to take advantage of modeling consumption-production interactions with a factorization machine instead of the conventional matrix factorization method for ownership recommendations. (2) We enrich the ACPR technique with directional adversarial training and call our technique Adversarial Consumption and Production Relationship-Aware Directional Adversarial Model (ACPR-ADAM). The idea behind our ACPR-ADAM is that instead of the worst perturbation direction, the perturbation direction in the embedding space is restricted to other examples in the current embedding space, allowing us to incorporate the collaborative signal into the training process. Lastly, through extensive evaluations on Reddit and Pinterest, we demonstrate that our proposed method outperforms state-of-the-art methods. Compared with CPR and ACPR on Reddit and Pinterest datasets, our proposed ACPR-ADAM achieves 93%, 88%, and 72%, 69% enhancement in terms of AUC and HR, respectively.

INDEX TERMS Adversarial attack, Directional adversarial training, Factorization machine, Ownership recommendation

I. INTRODUCTION

In an overloaded digital environment, recommendation systems are the most commonly used concepts for implementing personalization systems to provide information services to users. Even though recommendation systems cover e-commerce domains [1]–[3], they have effectively extended to cover a variety of scenarios, including e-learning [4], tourism [5], libraries [6], e-government [7], financial investment [8], and other application areas [9], [10]. Recommendation systems consist of different recommendation paradigms based on the information available to generate recommendations. Initially, demographic methods were the most relevant approaches because of the availability of such information. More recently, the collaborative filtering-based recommendation [11] and the content-based recommendation [12]–[14] have emerged as the two main recommendation paradigms. In addition to previous concepts, some studies have referred to other concepts such as social, knowledge-based, or hybrid filtering [15] depending on the methodology and information used for recommendation generation. Model-based collaborative filtering approaches are prominent in the recommendation system community and highly desired in industrial applications due to their interpretability and effectiveness. The fundamental tenet is that a consumer may prefer items similar to those with which he has previously interacted. The similarity is predicted based on previous interactions between users (see figure 1). Early model-based collaborative filtering methods primarily used linear approaches to construct their models. Matrix factorization [16]–[18], which transforms consumers and products into a low-dimensional
shared space and represents the consumer’s attachment for a product as the inner product of both the corresponding consumer/product feature vectors, is one of the powerful model-based techniques. Factorization Machines (FM) [19] have recently become one of the most popular model-based collaborative filtering models. FM provides a standardized factorization strategy for recommendations that can readily incorporate consumer and product side information. Given the recent resurgence of deep neural networks [20], [21], several deep learning methods for addressing the collaborative filtering problem in recommendation have been proposed. They have shown positive results and great potentials for further improvements [22]–[24].

Unlike traditional MF and deep neural network models, which focus solely on the user’s role as a consumer, ownership applications require additional features, such as the consumer’s production and consumption roles. In light of these features, Kang and McAuley [25] recently presented Consumer and Producer Recommendation (CPR), a matrix factorization-based technique [16], [26] for integrating users’ consumption and production roles. Although the CPR model contributes to better recommendation performance, we believe that one potential disadvantage is that the entire system is prone to misoperation or malicious operation, which leads to recommendation error. As demonstrated in previous work [27]–[32], many cutting-edge recommendation systems are vulnerable to adversarial attacks. To maximize the loss function, He et al. [27] compared the impact of adversarial attacks on model parameters using an adversarial personalized matrix factorization. We call this attack approach the maximal direction for clarity. They discovered that Adversarial Training (AT) improved model resilience and generalization. The idea behind AT is that for a given input \( x \), it maintains a similar label for \( x \) and an example of an attack \( x_{adv} \), perturbed by \( x \) during training. It facilitates the algorithm to consider additional unseen spaces in \( x \). As a result, it improves local smoothness between similar attack examples, moving the algorithm closer to better generalization [33]. Recently, Paul et al. [32] developed an ownership-based recommendation system that is defensible by using an iterative adversarial training approach on the ownership characteristics. However, while the iterative method is the most researched strategy, single-step adversarial training strategies that exhibited superior efficacy in recommendation systems [27]–[29] have been substantially under-examined in the ownership-based recommendation. Accordingly, ownership-based recommendation systems may be more vulnerable to single-step perturbations, potentially affecting downstream applications. In this work, we adapt the single-step adversarial training for ownership recommendations. We propose Adversarial Consumption and Production Relationship (ACPR), a model that combines factorization machine [34] and single-step adversarial training for ownership recommendation. It enables us to take advantage of modeling consumption-production interactions with a factorization machine rather than the usual matrix factorization [32] while also boosting the robustness of the ownership recommender model.

Furthermore, the maximal direction attack may cause the embedding vector of input \( x \) to be close to examples with different labels. As a result, the existing maximal direction technique cannot keep the original semantic information in consumption-production relationships. We address this issue by imposing appropriate constraints on the perturbation direction using directional adversarial training and term the method Adversarial Consumption and Production Relationship-Aware Directional Adversarial Model (ACPR-ADAM). We perturb an example \( x \) in the embedding space towards another existing example \( x' \), and a weight \( w \) is also set to control the distance \( x \) towards \( x' \). The ACPR-ADAM adds additional adversarial loss to make the training process a minimax game. The determination of weight \( w \) is through the maximization of the loss caused by the consumption and production relationship. Next, the model is tuned to minimize the loss. We also consider producing more effective embeddings by guiding the perturbation direction with the critical collaborative signal. The premise is that consumers who exhibit similar consumption patterns will have similar preferences for consumers in production. We compare our proposed method’s performance against numerous baselines. Extensive results reveal that our proposed solution is more effective than existing ownership-based recommendation methods.

The main contributions of this paper can be summarized as follows:

- We adapt the single-step adversarial training for ownership recommendation systems.
- We propose Adversarial Consumption and Production Relationship (ACPR), a model that combines factorization machine and single-step adversarial training for ownership recommendation. It enables us to take advantage of modeling consumption-production interactions with a factorization machine rather than the usual matrix factorization while also boosting the robustness of the ownership recommender model.
- We improve the ACPR model with directional adversarial training and call our technique Adversarial Consumption and Production Relationship-Aware Directional Adversarial Model (ACPR-ADAM). The idea behind our ACPR-ADAM is that instead of the worst perturbation direction, the perturbation direction in the embedding space is restricted to other examples in the current embedding space, allowing us to incorporate the collaborative signal into the training process.
- To show the efficacy of our proposed strategy, we ran experiments on the Reddit and Pinterest datasets. ACPR-ADAM, in particular, achieves considerable gains in HR (Hit Ratio) and AUC measures when compared to cutting-edge models.

The following make up the remaining sections:

In Section II, we discussed some background knowledge regarding the current work and related studies. Section III contains a description of the proposed model and algorithm. Section IV discusses the proposed algorithm’s empirical
evaluation and comparison with state-of-the-art methods. Finally, Section V provides conclusions from the current work and prospects for future research on this topic.

II. RELATED WORK

This section will highlight the most current advancements in recommendation systems.

A. RECOMMENDATION SYSTEM

The three types of recommendation systems are collaborative filtering-based recommendation (CF), content-based recommendation, and hybrid-based recommendation [35].

The first class includes approaches such as User and Item-KNN [36] and Bayesian Personalized Ranking Matrix Factorization (BPR-MF) [26], which learns consumers’ preferences via consumer-product interactions. The second type presents previously unseen products based on their content-based similarity to consumer-consumed products [12]–[14]. The last type combines collaborative filtering and content-based techniques to offer additional signals to consumers or products [37]. In the last ten years, CF methods have attracted a lot of interest. Models were created and developed for collaborative filtering processes, which employ training examples to understand particular trends and then estimate real-life data. Collaborative filtering models make an effort to depict two forms of interactions in the real world: consumer-product interactions and product-product relations, which result in consumer-product and product-product techniques.

Models that employ consumer-product collaborative filtering assess a consumer’s decision on a target product to determine whether the consumer has a connection to it. Early algorithms based on collaborative filtering [34] generate recommendations based on explicit feedback. Explicit collaborative filtering methods use regression to generate recommendations and estimate the preference score. Several studies [38] later found that collaborative filtering strategies are ineffective in the Top-k recommendations. The current research trend in CF has steadily evolved away from explicit feedback and toward implicit feedback. Model-based CF solutions are prevalent in recommendation system communities. One of the most well-known examples is matrix factorization (MF) [25], [26]. To represent each consumer and product, MF employs a low-dimensional feature vector. The consumer ratings of the products can then be estimated using the inner product of the relevant consumer vector and product vector. Matrix factorization-based models, such as the social-aware MF [40], have been presented. Factorization Machines (FM) [19] have recently emerged as one of the popular models for recommendations, providing a flexible factorization technique that can easily integrate consumers and products side information.

Product recommendation, according to Rendle et al. [26], is a personalized pairwise ranking problem. The authors developed the BPR (bayesian personalized ranking), a method for model optimization using the preferences of pairs of products. BPR is now being used to optimize many recommendation models [41], [42]. Ding et al. [43] used negative sampling to improve Bayesian Personalized Ranking. Recently, Kang and McAuley [25] introduced a novel ownership-based recommendation approach that uses BPR [16], [26] to leverage consumers’ production and consumption roles. In this work, we used model-based CF (e.g., FM) as the primary recommendation technique. We consider both the consumption and production roles (see equation (7)), regard feedback as implicit feedback, and optimize with a multi-objective BPR approach.

B. DEEP LEARNING-BASE RECOMMENDATION

In recent years, collaborative filtering methods have begun to employ neural networks [22]. According to He et al. [22], the inner product feature restricts the expressive power of matrix factorization (MF) approaches and is even ineffective for uncovering complicated consumer-product interactions. The authors proposed neural collaborative filtering (NCF) paradigm to enhance MF’s nonlinearity expressiveness by utilizing a deep neural network to learn the consumer-product interaction function. The NCF was later extended to leverage the neighbor [44] and attribute information [45] of the consumers and products based on similarity [23]. In such scenarios, the recommendation techniques only model the consumer and product interaction. However, to solve the sparsity problems of these interactions, some techniques seek to leverage product and consumer similarities. The basic principle is that the consumer’s judgment on a target product is inferred based on similarities between the consumer and past interactions. The most popular approach is the factor item similarity model [46], which models the consumer representation as a mean aggregation of product embeddings. Following this research line, several techniques have been introduced in recent years using consumer information [25] and neural networks [24]. Despite advances in deep collaborative filtering, these approaches still rely solely on product-product interactions, likely to result in unsatisfactory recommendation accuracy. Unlike previous work that only leverages consumer-product and product-product interactions in their models, our proposed methods combine the advantages of consumption and production relationships to enable better recommendation performance.

C. ADVERSARIAL RECOMMENDATION

The failure of recommendation models is widely studied using adversarial machine learning strategies [30], [33]. Adversarial machine learning is the technique for generating optimal perturbations to reduce the performance of a recommender model and its defense strategies. Most recently, He et al. [27] demonstrated the weakness of Bayesian Personalized Matrix Factorization (BPR-MF) against adversarial perturbations obtained from the Fast Gradient Sign Method (FGSM) and suggested adversarial training technique as a defensive strategy. The proposed technique is tested and implemented in many algorithms for model robustness [28], [29], [32]. As we investigate enhancing the robustness of ownership
recommendations, this line of research is relevant to this research. ACPR [32] is an approach that applies to this study. The most significant distinctions between ACPR and ACPR-ADAM are as follows:

1) ACPR is a variant of ownership-based recommendation that incorporates consumption and production roles via a matrix factorization. In contrast, ACPR-ADAM is a variant of ownership-based recommendation that incorporates consumption and production roles through an extension of factorization machine.

2) ACPR applies the iterative perturbation as a technique for the attack, while ACPR-ADAM applies the single-step perturbation as the attack strategy.

3) ACPR seeks to maximize the impact of perturbation by using adversarial training to train a consumption and production recommender. Alternatively, ACPR-ADAM employs directional adversarial training by imposing appropriate constraints on the perturbation direction.

III. ACPR-ADAM ALGORITHM

We define the notations used throughout the study (see Table 1) and discuss several essential concepts in this section. Then we examine the shortcomings of the CPR algorithm and propose ACPR. In addition, to improve the ACPR algorithm, we are training it with directed adversarial training and proposing ACPR-ADAM as a solution.

A. MATRIX FACTORIZATION

Collaborative filtering (CF) [11] is the key technique for personalized recommendation systems. It exploits consumer-product interactions by assuming that similar consumers would have similar preference on items. Model-based CF solutions are prevalent in recommendation system communities. One of the simplest yet successful models for model-based CF is matrix factorization (MF) [16], which models consumer-product interactions:

\[
f_{MF} \left( u, i \mid \gamma_u, \gamma_i \right) = \gamma_u^T \gamma_i = \sum_{k=1}^{K} \gamma_{uk} \gamma_{ik} \tag{1}\]

where \( \gamma_u \in \mathbb{R}^k \) and \( \gamma_i \in \mathbb{R}^k \) are the parameters for the model denoting the latent features of consumer \( u \) and product \( i \), respectively.

B. CONSUMPTION AND PRODUCTION MODELING

The MF only models the item consumed by the consumer (i.e., the interaction between the consumer and the product). However, for ownership-based recommendation systems, we have additional features to model, particularly the relationship between production and consumption. Kang and McAuley [25] presented a Consumer and Producer Recommendation (CPR) method based on the consumers’ production and consumption roles. In CPR, each consumer has latent vectors \( \gamma^c_u \) and \( \gamma^p_u \) that represents the consumption and production roles. These features are designed based on \( \gamma_u \) as follows:

\[
\gamma^c_u = W^c \gamma_u, \quad \gamma^p_u = W^p \gamma_u \tag{2}\]

where \( W^c \) and \( W^p \) are utilized to transform the core vector of consumer \( u \) to \( \gamma^c_u \) and \( \gamma^p_u \). The consumer \( u \) preference toward product \( i \) is computed as follows.

\[
x_{ui} = \alpha + \beta_u + \beta_i + \langle \gamma^c_u, \gamma_i \rangle + \langle \gamma^p_u, \gamma^p_i \rangle \tag{3}\]

where \( \alpha \) denotes the global offset, \( \beta_u \) and \( \beta_i \) are the consumer/product bias terms, \( \gamma_i \) is a latent vector of \( i \), \( \gamma^c_u \) is a consumer of product \( i \), and \( \gamma^p_u \) is a producer of product \( i \). The parameters are learned based on Bayesian Personalized Ranking (BPR) [26]. We consider modeling the consumption and the production roles from a different perspective based on the factorization machine (FM) [34].
TABLE 1. Notations.

| Notations | Description |
|-----------|-------------|
| $\mathcal{U}$ | The consumer set |
| $Q^+_{c}$, $Q^-_{c}$ | Positive product set for consumer $c$, The producer of product $i$ |
| $p_{i} \in \mathcal{U}$ | Predicted score consumer $c$ gives to product $i$ produced by $p$ |
| $x_{c,i,p} \in \mathbb{R}$ | Latent factor dimension |
| $K \in \mathbb{N}$ | Core embedding for consumer in consumption $c$ |
| $\gamma^c_{i} \in \mathbb{R}^K$ | Core embedding for consumer in production $p$ |
| $Q_{p}$ | Unlabeled set |
| $W_{i}^{c} \in \mathbb{R}^{K \times K}$ | Role transformation for consumption |
| $\gamma_{i} \in \mathbb{R}^{K}$ | Role transformation for production |
| $Q$ | Product $i$’s embedding |
| $\Delta$ | The products set |
| $\epsilon$ | Adversarial perturbations |
| $\Theta$ | The magnitude of adversarial perturbations |
| $d$ | Model parameters |
| $\Delta_{adv}$ | The direction vector |
| $u_{dadv}$ | The adversarial perturbation direction |
| $w_{dadv}$ | The worst-case weights |

C. ADVERSARIAL ATTACK OF MODEL PARAMETERS

Adversarial recommendation [27] demonstrated the vulnerability of collaborative filtering models to adversarial perturbation [30]. The objective of adversarial recommendations is to maximize recommendation accuracy utilizing adversarial perturbations by either directly sampling from the product pool or by conducting perturbations on the embedding variables. It allows us to enhance their robustness against threats while training more accurate recommendation engines. It employs the maximum-norm constraint and approximates the objective of adversarial recommendations is to generate adversarial perturbation ($\Delta_{adv}$). Where the adversary is the maximization under the constraint:

$$\Delta_{adv} = \epsilon \frac{\Pi}{\|\Pi\|} \text{ where } \Pi = \frac{\partial \mathcal{L}(\Theta + \Delta)}{\partial \Delta} \quad (4)$$

where $\| \cdot \|$ represents the $L_2$-norm. After generating $\Delta_{adv}$, the attack is injected to the model parameters $\Theta_{adv} = \Theta + \Delta_{adv}$ and obtained the lists of recommendation with the perturbed parameter. The goal behind adversarial recommendation is to generate adversarial perturbation ($\Delta_{adv}$). Where the adversary is the maximization under the constraint:

$$\Delta_{adv} = \arg\max_{\Delta} \mathcal{L}(\Theta + \Delta) \quad (5)$$

where the initial attack $\Delta$ is fed into the parameters $\Theta$ and $\epsilon$ denotes the attack budget used to reduce noise. The minimax optimization is then formulated as:

$$\arg\min_{\Theta} \max_{\Delta : \|\Delta\| \leq \epsilon} \mathcal{L}(\Theta + \Delta) \quad (6)$$

where the adversary’s goal is to maximize the chances of success, and the training process is to minimize noise.

The adversarial training technique provided in equation (6) is being explored and implemented in ownership recommendation algorithms for model robustness and efficiency [32]. In this paper, we utilize single-step adversarial training to improve the robustness of an ownership recommendation system and hence its generalization performance. With CPR as the core recommender model, we incorporate an adversary that applies perturbations to model parameters to maximize the CPR loss function. The adversarial perturbations are optimized to maximize the CPR loss, while the model parameters are optimized to minimize the CPR loss and the adversary’s loss. In this manner, we can improve model robustness to adversarial perturbations so that the adversarial perturbations have a lower impact on the model’s prediction.

D. CPR-BASED ON ADVERSARIAL TRAINING

In recommendation tasks, the factorization machine (FM) [34] has been quite successful. We use the FM as our basic model. However, FM considers all interactions between inputs, which is expensive and may generate undesirable interactions that negatively affect our purpose. Thus, we only keep the interactions that are useful to our task and remove the others. Given consumer $c$’s consumption role $\gamma^c_{i}$, the production role $\gamma^p_{i}$, and the product $\gamma_{i}$, we predict how likely $\gamma^c_{i}$ will like $\gamma_{i}$ as:

$$x_{c,i,p} = \gamma^c_{i}\gamma_{i} + \sum_{\gamma_{p}\in \gamma_{c}} \gamma^c_{p}\gamma^p_{i} \quad (7)$$

where $\gamma^c_{i}$ and $\gamma_{i}$ denote the consumer $c$ and product $i$, and $\gamma^p_{i}$ is the producer $\gamma_{p} \in \gamma_{c}$ embedding. The first term $\gamma^c_{i}\gamma_{i}$ models the general interest in the consumption of product. The second term $\Sigma_{\gamma^c_{p}\gamma^p_{i}}$ models the consumption and the production roles.

Next, we design a ranking objective. The positive sample set $D$ is depicted by:

$$D = \{(c, i, p) \mid x_{c,i,p} = 1\} \quad (8)$$

and the unlabeled sample set is:

$$Q^{-}_{c} = \{i \mid i \in Q \setminus (Q^{+}_{c} \cup Q^{+}_{p})\} \quad (9)$$

where the set of products is represented with $Q$, $Q^{+}_{c} = \{i \mid x_{c,i} = 1\}$ denotes the consumer $c$’s consumption set, and
where the model parameter $\Theta$ is the minimizing parameter, and the adversarial perturbation ($\Delta_{adv}$) is the maximizing parameter. The two processes rotate until convergence.

Next, we will go over the solution for the minimax optimization method in equation (15).

**Step 1. Constructing Perturbations.** Given a training example $(c, i, j, p)$, the problem of creating adversarial disturbances $\Delta_{adv}$ can be expressed as maximizing

$$l_{adv}(c, i, j, p \mid \Delta_{adv}) = -\ln \sigma(x_{c,i,j,p}(\Theta + \Delta_{adv}))$$

where $\Theta$ is the parameters of the model. By maximization of the objective function, $\Delta_{adv}$ is obtained as the representation for the perturbations $\Delta_c, \Delta_p,$ and $\Delta_i$, which can have the greatest impact on the loss on example $(c, i, j, p)$. Because of the non-linear nature of the objective function $l_{adv}$ and the constraints on optimization, the exact solution is challenging to obtain. As a corollary, we borrow the concept from method [27], linearizing the objective function to approximate it around $\Delta_c, \Delta_p$ and $\Delta_i$, after which the limited optimization problem on this approximated linear function is solved. The best solution for $\Delta_{adv}$ is to move the variable in the direction of its gradient, which can be deduced as:

$$\frac{\partial l_{adv}(c, i, j, p \mid \Delta_{adv})}{\partial \Delta_{adv}} = -(1 - \sigma(x_{c,i,j,p}(\Theta + \Delta_{adv}))) \frac{\partial x_{c,i,j,p}(\Theta + \Delta_{adv})}{\partial \Delta_{adv}}$$

where equation (4) is obtained as the best solution for $\Theta_{adv}$ using the constraint $\|\Delta_{adv}\| \leq \epsilon$.

**Step 2. Parameter Learning.** For training example $(c, i, j, p)$, the model parameters $\Theta$ is obtained by minimizing:

$$l_{ACPR}((c, i, j, p \mid \Theta) = -\ln \sigma(x_{c,i,j,p}(\Theta) - x_{c,j,p}(\Theta)) - \lambda \ln \sigma(x_{c,i,p}(\Theta + \Theta_{adv}) - x_{c,i,p}(\Theta + \Theta_{adv}))$$

where $\Theta_{adv}$ is a constant determined by equation (4). Through minimizing the objective function, we obtain the model parameters $\Theta$, which make the model resilient to adversarial perturbations on the example $(c, i, j, p)$.

Finally, we update with the SGD rule:

$$\Theta = \Theta - \eta \frac{\partial l_{ACPR}(\Theta)}{\partial \Theta}$$

where the parameters of the model is denoted by $\Theta$, and $\eta$ denotes the learning rate.

**E. CPR-BASED ON DIRECTIONAL ADVERSARIAL TRAINING**

The predictive model is effective at increasing the robustness. However, only taking into account the attack on the embedding space's maximum direction will introduce error. As a result, we present the Adversarial Consumption and Production Relationship-Aware Directional Adversarial Model (ACPR-ADAM) to address this issue and improve the
application of adversarial training for the ownership-based recommendation. The idea behind the proposed technique is that instead of moving in the worst-case direction, the attack can be limited to additional examples from the existing embeddings, allowing us to improve the training process with the collaborative signal. Playing the minimax game increases the value of the embedding layers. We define $\Delta_{c,t}^c$ as the direction vector from consumption $c_t$ to consumption $c_z$ in the embedding space:

$$d_{c_t}^c = \frac{d_{c_t}^c}{\|d_{c_t}^c\|_2}, \quad d_{c_z}^c = y_{c_z} - y_{c_t}$$  \quad (20)$$

where the direction vector $d_{c_t}^c$ is always a unit vector, $\|d_{c_t}^c\|_2 = 1$. Specifically, if the $z = t$ is defined as a zero vector, the $d_{c_z}^c$ is defined as a zero vector. The weight corresponding to the direction vector $d_{c_z}^c$ is then defined as $w_{c,z}^c$. We then represent $\Delta(w^c)$ as the direction of adversarial perturbation:

$$\Delta(w^c) = w_{c,z}^c d_{c_z}^c$$  \quad (21)$$

where the target set $U_{target}^c$ in our method refers to the consumption perturbed by the training example. As a result, $c'$ belongs to the $U_{target}^c$ group. Then, for consumption $c$, the directional adversarial embedding vector is written as follows:

$$\gamma_{c}^c \Delta_{dadv} = \gamma_{c}^c + \Delta(w^c)$$  \quad (22)$$

Since the original objective function minimized during the training process, we find the worst-case direction vectors capable of maximizing the loss:

$$w_{dadv}^c = \arg \max_{w^c, \|w^c\|_2 \leq \epsilon} \mathcal{L}(\Theta + \Delta(w^c))$$  \quad (23)$$

After restricting the direction of attacks, the technique in equation (4) is not applicable any longer to compute the weights. We borrow the idea from [33], where the attack strength $w_{dadv}^c$ can be estimated by using second-order Taylor $\mathcal{L}(\Theta + \Delta(w^c))$.

The solution of estimating worst-case weight is formalized as follows:

$$w_{dadv}^c = \epsilon \Pi, \quad \Pi = \nabla_{w^c}\mathcal{L}(\Theta + \Delta(w^c))$$  \quad (24)$$

Alternatively, $w_{dadv}^p$ can be obtained by repeating the preceding procedure. Similarly to equation (15), the ACPR-ADAM is optimized as:

$$<\Theta> = \arg \min_{\Theta} \mathcal{L}(\Theta) + \lambda \mathcal{L}_{ACPR-Adam}$$  \quad (25)$$

To connect the two preceding processes, we express it as a minimax optimization problem. The model parameters optimization $<\Theta>$ denotes the minimization player, and the development of attacks weights $w_{dadv}$ denotes the maximization player:

$$<\Theta, w_{dadv}> = \arg \min_{\Theta} \arg \max_{w_{dadv}} \mathcal{L}(\Theta) + \lambda \mathcal{L}_{ACPR-Adam}$$  \quad (26)$$
Finally, we describe the minimax optimization and the SGD solution for ACPR in Algorithm 1 and ACPR-ADAM in Algorithm 2. The parameters of the model $\Theta$ is initialized via CPR optimization rather than being randomly initialized. The idea is that adversarial perturbations should only be applied when the model parameters overfit the data. When a model is underfitting, a standard training process is adequate to generate better parameters. In contrast to CPR pre-training, another potential technique is to dynamically regulate $\epsilon$, which modulates the degree of perturbations during training. A holdout validation set, for example, can be used to learn $\epsilon$. For each training process, we start by generating a random example $(c, i, j, p)$. The update rule for perturbations and model parameters is then executed sequentially.

### Algorithm 1: ACPR

Input: dataset $D$, learning rate $\eta$, adversarial noise level $\epsilon$, adversarial regularizer strength $\lambda$, $L_2$ regularizer strength $\lambda_{\Theta}$

Output: model parameters $\Theta$

Initialize $\Theta$ from CPR;

while Stopping criteria is not met do

Randomly draw $(c, i, j, p)$ from $D$;

//Developing single-step perturbations
$\Delta_{adv} = \epsilon \Pi$, where $\Pi = \frac{\partial L(\Theta + \Delta)}{\partial \Delta}$;

//Learning model parameters
$\Theta = \Theta - \eta \frac{\partial L(\Theta + \Delta)}{\partial \Theta}$;

return $\Theta$

### Algorithm 2: ACPR-ADAM

Input: dataset $D$, learning rate $\eta$, adversarial noise level $\epsilon$, adversarial regularizer strength $\lambda$, $L_2$ regularizer strength $\lambda_{\Theta}$

Output: model parameters $\Theta$

Initialize $\Theta$ from CPR by solving equation (7);

while Stopping criteria is not met do

Randomly sample examples $(c, i, j, p)$ from $D$;

//Computing the direction vector
$d \leftarrow$ equation (20);

//Computing the worst case weights
$w_{adv} \leftarrow$ equation (23);

//Constructing single-step perturbation direction
$\Delta (w) \leftarrow$ equation (21);

//Optimizing model parameters
$\Theta \leftarrow$ equation (25);

return $\Theta$

## IV. MATERIALS AND METHODS

In this section, we will first explain the evaluation technique, datasets, evaluation metrics, and baselines. Then, we compare the performance of ACPR and ACPR-ADAM to that of numerous state-of-the-art recommendation algorithms. Finally, we concentrated on examining the performance of ACPR and ACPR-ADAM across different datasets and evaluation methods.

### A. METHODOLOGY

We evaluate the proposed model to provide answers to the following questions:

- Is the ACPR-ADAM capable of outperforming traditional methods?
- What impact does directional adversarial training have on model performance?
- What impact does the size of the embedding have on model performance?
- What are the effects of the hyperparameters $\epsilon$ and $\lambda$ on model performance?

### B. EXPERIMENTAL DETAILS

#### 1) DATASETS

We perform experiments on two real-world datasets: Reddit and Pinterest [25], [32]. On both datasets, 1) each product is connected with a producer, and 2) the consumer-product interaction matrix is very sparse. Table 2 displays the dataset statistics. A prosumption in the table refers to a consumer who performs both the production and consumption of products.

- Pinterest is a content search application that focuses on images. Images can be browsed, uploaded (pinned), liked, and saved (repinned). The data included 0.89 million consumers, 2.4 million photos, and 56 million saved and appreciated activities. Activities such as appreciates and repins are considered implicit feedback. Each product has a producer connected with it. We sample a small subset of the data to verify our method. Specifically, we randomly select 134,747 consumers, 201,792 products, 690,506 activities, and then discard consumers who consume or produce fewer than ten products and products consumed by fewer than ten consumers.
- Reddit is a conversation website that offers a wide variety of topics such as news, science, and movies, to name a few. Consumers can post content and provide comments on other consumers’ posts. The data comprises 1.3 million consumers, 9.6 million posts, and 48 million feedbacks. Comments are considered implicit feedback, and posts associated with a single producer are considered products. We randomly select 52,654 consumers, 336,743 products, 1,786,032 activities, and then discard consumers who consume or produce fewer than ten products and products consumed by fewer than ten consumers.

#### 2) EVALUATION METRICS

To compare the proposed approach to baseline methods, we use the Hit Ratio and Area Under the Curve as metrics. The Hit Ratio is a ranking metric that calculates the average number of times a target product appears on the top-N
techniques: We compare the proposed method to the following standard

\[ \text{AUC} = \frac{1}{|\mathcal{U}|} \sum_{t \in D} \frac{1}{|\mathcal{U}|} \sum_{c \in |\mathcal{U}|} \text{hit} \circ N(t, c) \quad (28) \]

where hit \circ N(t, c) in equation (27) is 1 when the target product is in the top-N list of the consumer c, and \( \xi(\cdot) \) in equation (28) is the indicator function.

3) BASELINES

We compare the proposed method to the following standard techniques:

- Recommendation Based on Popularity (Pop). The popularity method assigns a ranking to items based on their popularity. It is a foundational model for evaluating the efficacy of personalized recommendations.
- Bayesian Personalized Ranking (BPR-MF) [26]. It is a recommendation strategy that uses implicit feedback. The underlying predictor in BPR-MF is standard matrix factorization.
- Adversarial Matrix Factorization (AMF) [27]. AMF is an adversarial recommendation model that makes recommendations via Adversarial Training (AT) on matrix factorization.
- Factorization Machine (FM) [34]. For the recommendations, FM mimics the user’s interaction with products and their features.
- Consumer and Producer Recommendation (CPR) [25]. For the ownership recommendation, the CPR takes advantage of the relationship between the consumption, the item, and the production.
- Adversarial Consumption and Production Relationship (ACPR). ACPR is our enhanced CPR model that includes a factorization machine and Adversarial Training (AT) for the recommendation. It uses the single-step Fast Gradient Sign Method (FGSM) as the attack strategy.
- Adversarial Consumption and Production Relationship-Aware Directional Adversarial Model (ACPR-ADAM). The ACPR-ADAM model is an upgraded ACPR model that employs DAT (directional adversarial training) by constraining the attack direction and explicitly incorporating the collaboration signal into the adversarial training process. Tensorflow is used to implement our approaches.

4) Hyper-Parameters Setting

We implemented all baseline method within the TensorFlow framework with Python. For a fair comparison, we set 20 as the embedding size. The adagrad optimizer [47], is used for optimization. We sequentially tune the ranges \{0.001, 0.01, 0.1, 1, 10\} for the \( L_2 \) regularizer, 0.01 is set as the learning rate, and 10000 is set as the batch size. After obtaining the optimal learning rate and \( L_2 \) regularizer, the obtained values are used for our methods and then tune \( \varepsilon \) in \{0.0001, 0.001, 0.01, 0.1, 1, 10\} and \( \lambda \) in \{0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000\}. Table 3 displays the set of hyperparameters.

C. EXPERIMENT 1

Here we compare the performance with baselines using the Area Under the Curve (AUC) and the Hit Ratio (HR). Table 4 displays the results. Our observations are as follows.

Firstly, in Table 4, BPR-MF performs the best compared to the popularity based Pop. This is because BPR-MF is more effective in settings where products are inherently ‘unpopular’, a characteristic which makes Pop less effective.

Secondly, among all methods, the performance of CPR, ACPR, and ACPR-ADAM consistently outperforms MF-BPR, FM, and even the AMF model on all dataset. Specifically, on the Reddit dataset CPR, ACPR, and ACPR-ADAM improves over baselines with larger margin than on the Pinterest dataset. The reason is that Reddit consumers consumes products produced by the same producer more repeatedly which is easy to predict.

Thirdly, the CPR approach surpasses FM on Reddit, but only by a small margin on the Pinterest dataset. This discovery can be attributed to FM models’ capacity to exploit production information. However, ACPR and ACPR-ADAM built on adversarial learning performs better in general. Thus, by doing adversarial training, ACPR can capture all information, leading to stable performance.

Finally, we discovered that while adversarial training can benefit AMF, it fails to capture consumption and production characteristics. Here, ACPR and ACPR-ADAM which in addition captures this factors achieve the best over AMF and
TABLE 3. Main Parameters

| Parameters          | Pinterest | Reddit |
|---------------------|-----------|--------|
| Learning rate       | 0.01      | 0.01   |
| Batch Size          | 10000     | 10000  |
| Embedding size      | 20        | 20     |
| Number of epochs    | 700       | 1200   |
| Regularizer weight  | 1         | 1000   |
| Adversarial noise   | 0.1       | 0.3    |
| Optimizer           | Adagrad   | Adagrad|

TABLE 4. Performance comparison in terms of AUC and HR with \( N = 10 \).

| Method   | Reddit AUC | Reddit HR@10 | Pinterest AUC | Pinterest HR@10 |
|----------|------------|--------------|--------------|-----------------|
| PopRec   | 0.6213     | 0.6867       | 0.5267       | 0.2327          |
| MF-BPR   | 0.7485     | 0.7054       | 0.5926       | 0.2587          |
| AMF      | 0.7909     | 0.7286       | 0.5954       | 0.2568          |
| FM       | 0.8346     | 0.7447       | 0.6133       | 0.2737          |
| CPR      | 0.8545     | 0.7490       | 0.6226       | 0.3103          |
| ACPR     | 0.8973     | 0.7748       | 0.6544       | 0.3577          |
| ACPR-Adam| 0.9305     | 0.8872       | 0.7249       | 0.6964          |

FIGURE 3. Impact of applying adversarial noise (gradient) and random noise (random) on model performance.

betters it on average. Compared to the AMF, ACPR-Adam reduces encoding error in training, resulting in an average improved performance of 0.9305 and 0.7249 versus 0.7909 and 0.5954 for the AMF. This significant improvement is because ACPR-ADAM incorporates the interaction of similar consumptions or productions in the adversarial training process. Moreover, we can observe that ACPR performs significantly worse than ACPR-ADAM, demonstrating the importance of modeling the collaborative signal.

D. EXPERIMENT 2

Here, we examine the influence of adversarial training on ACPR.

1) ROBUSTNESSS

To gain a quantitative understanding of the model’s robustness, we inject noise into the input parameters of the original CPR model and evaluate performance drop. A lower reduction ratio indicates higher robustness.

We first show the impact of noise on CPR. The performance on CPR is shown in figure 3(a) and figure 3(b). The horizontal green lines indicate the performance of unperturbed CPR on Reddit and Pinterest, respectively. The random perturbation (CPR-random) would reduce model performance. The adversarial perturbation (CPR-gradient) in equation (14), on the other hand, has a heinous effect on CPR. For example, on Reddit, the attack with \( \epsilon = 0.1 \) is steady, and the attack with \( \epsilon = 0.9 \) causes a drop. The results indicate how attacks affect model performance.

In Table 5, we list the three scenarios in the Reddit and Pinterest dataset for ACPR. As shown in Table 5 and figure 3(c), we can observe that ACPR can better solve the robustness problem. For example, in Table 5 when the perturbation scale is set to 0.6 in reddit, the performance of the CPR algorithm drops by 20.66%, while the ACPR algorithm only drops by 0.42%. This result shows the effectiveness of ACPR toward the robustness objective.
2) GENERALIZATION

To demonstrate the value of adversarial training in increasing overall model generalization, we first train CPR on Reddit and Pinterest for 600 and 100 epochs (mainly converged) and then compare it to CPR trained using adversarial training (i.e., ACPR and ACPR-ADAM techniques). With CPR as pre-training, ACPR and ACPR-ADAM achieve good performance with iterations higher than 600 and 100. Therefore, we report the performance in these settings. The training procedure of CPR and ACPR on Reddit and Pinterest datasets are shown in figure 4(a) and figure 4(b), respectively, for 1200 and 700 epochs. As can be observed, by training on CPR, the performance is not enhanced or reduced due to noise. On the other hand, by performing adversarial training can
FIGURE 6. (a) Optimal performance of ACPR with varying noise level $\epsilon$. ACPR achieves the best value when $\epsilon = 0.1$ and $\epsilon = 0.3$, respectively. (b) Optimal performance of ACPR with varying adversarial regularizer $\lambda$. ACPR achieves the best optimal value when $\lambda = 1000$ and $\lambda = 1$.

FIGURE 7. Performance of ACPR and ACPR-ADAM evaluated per 1200 and 700 epochs on Reddit and Pinterest datasets.

FIGURE 8. Performance of ACPR-ADAM with respect to different embedding sizes on Reddit and Pinterest datasets.
improve performance. As can be observed, on Reddit and Pinterest (figure 4(a) and figure 4(b) the best AUC of CPR are 0.854 and 0.622, which are increased to 0.897 and 0.654 by adversarial training (ACPR). These results verify the effect of adversarial training for improved generalization.

3) EMBEDDING SIZE
To explore the benefits of adversarial training on models of various sizes, we set embedding $K$ in [10, 20, 40, 60, 80]. Then under the different embedding sizes, we choose the optimal embedding value that increases performance. Figure 5(a) and figure 5(b) depict the results of CPR and ACPR with different embedding sizes. As can be observed, the outcomes of both strategies improve as the embedding size increases. As a result of the enhanced modeling power, a larger model is advantageous to the recommendation. Secondly, we find that ACPR outperforms CPR on models of all embedding sizes.

Notably, ACPR with an embedding size of 60 surpasses CPR with an embedding size of 80. It adds to the evidence that adversarial training has a good impact on model performance.

4) HYPERPARAMETER EXPLORATION
Hyperparameters turning are crucial for recommendation performance. However, too large values will make the model resilient to adversarial perturbations at the risk of disrupting the training process. On the other hand, too small values would reduce the adversary’s influence while making no advances to the model’s resilience and generalization. Here, using ACPR as an example, we examine the influence of hyperparameter selection on adversarial training, i.e., we tune $\epsilon$ and $\lambda$, which control the scale of attack and the weight of adversary, respectively. Note that we show the results of CPR and ACPR only.

Figure 6(a) shows the performance with varying $\epsilon$. As can be observed, the curve drops sharply when the perturbation scale $\epsilon$ is between 0.1 and 1. For example, we set $\lambda$ to 1 and varying $\epsilon$ in both dataset and achieved an optimal value of 0.1 and 0.2. This indicates that the improvement of ACPR is small when $\epsilon$ is not large enough. However, the performance of ACPR drops when $\epsilon$ is large. Similarly, in figure 6(b), we set the default value of $\epsilon$ to 0.2 and 0.3 for Reddit and Pinterest with varying $\lambda$. For example, in the Reddit dataset it can be observed that increasing $\lambda$ gradually improves the performance when $\lambda$ is small. In the Reddit and Pinterest dataset we set $\epsilon$ to the threshold value of 0.2 and 0.3 and obtained an optimal value of $\lambda$ to be 1000 and 1. Thus, the proposed ACPR algorithm attains good performance when $\epsilon = 0.1$ and 0.2, and $\lambda = 1$ and 1000 on Pinterest and Reddit, respectively. These results show the importance of hyperparameter turning on model performance.

E. EXPERIMENT 3
We explore the impact of directional adversarial training on ACPR-ADAM.

1) GENERALIZATION
This part only shows the results of ACPR and ACPR-ADAM on the Reddit and Pinterest datasets at 1200 and 700 epochs. In figure 7, we first show the ACPR and ACPR-ADAM training processes. We can observe that doing directional adversarial training on ACPR leads to better results. For example, the best AUC of ACPR in Reddit (figure 7(a)) is 0.8973, which improves to 0.9305 with directional adversarial training (ACPR-ADAM). Similarly, the best AUC of ACPR in Pinterest (figure 7(b)) is 0.6544, which improves to 0.7249 with directional adversarial training (ACPR-ADAM). These findings confirm the efficacy of ACPR-ADAM while emphasizing the importance of using directional adversarial training for improving model robustness.

2) EMBEDDING SIZE
We investigate how the size of the embedding influences the performance of the model. Figure 8 shows the performance. As can be observed, our ACPR-ADAM consistently outperforms the ACPR under different embedding sizes. It is worth noting that when the embedding size is less than 10, the performance improvement of ACPR-ADAM becomes less pronounced. The most likely explanation is that the limited capacity of the algorithm constrains the information contained in the embedding layer.

As a result, the benefit of directional adversarial training is unavailable. However, with an embedding size of 80 on all datasets, the performance of ACPR-ADAM improves. It supports the positive impact of directional adversarial learning in our ACPR-ADAM method, as well as the significance of model size in performance.

V. CONCLUSION
In this work, we adapt the single-step adversarial training for ownership-based recommendation systems. We propose Adversarial Consumption and Production Relationship (ACPR), a model that combines factorization machine and single-step adversarial training for ownership recommendation. It enables us to take advantage of modeling consumption-production interactions with a factorization machine rather than the usual matrix factorization while also boosting the robustness of the ownership recommendation model. Furthermore, we improve the ACPR model with directional adversarial training and call our technique Adversarial Consumption and Production Relationship-Aware Directional Adversarial Model (ACPR-ADAM). The idea behind our ACPR-ADAM is that instead of the worst perturbation direction, the perturbation direction in the embedding space is restricted to other examples in the current embedding space, allowing us to incorporate the collaborative signal into the training process. We compare the performance of our proposed method to several baseline methods. Extensive results demonstrate the effectiveness of our ACPR-ADAM model.

In the future, we hope to incorporate user information such as demographics into the framework to improve the quality of recommendations, particularly for new users. Additionally,
though the adversarial technique is effective, it is plagued by the issue of interpretability that may impede its use in real-world scenarios. To enhance this approach, we are interested in designing an interpretable training method that enforces the reconstruction of adversarial attacks.

REFERENCES

[1] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, J. Riedl, “GroupLens: an open architecture for collaborative filtering of netnews,” In Proceedings of the Conference on Computer Supported Cooperative Work, pp. 175-186, 1994.
[2] J. B. Schafer, J. A. Konstan, J. Riedl, “Ecommerce recommendation applications,” Data Min. Knowl. Disc., vol. 5, no. 1, pp. 115-133, 2001.
[3] B. Smith and G. Linden, “Two decades of recommender systems at Amazon.com.” IEEE Internet Comput., vol. 21, no. 3, pp. 12-18, 2017.
[4] R. Y. Toledo, M. Y. Caballero, “An e-learning collaborative filtering approach to suggest problems to solve in programming online judges,” Int. J. Distance Educ. Technol., vol. 12, no. 2, pp. 51-65, 2014.
[5] J. M. Nogueira, M. J. Barranco, R. J. Segura, L. Martínez, “A mobile 3D-GIS hybrid recommender system for tourism,” Inf. Sci., pp. 37-52, 2012.
[6] W. S. Yang, Y. R. Lin, “A task-focused literature recommender system for digital libraries,” Online Inf. Rev., vol. 37, no. 4, pp. 581-601, 2013.
[7] J. Lu, Q. Shambour, X. Yuan, Q. Lin, G. Zhang, “A Web-Based Personalized Business Partner Recommendation System Using Fuzzy Semantic Techniques,” Comput. Intell., vol. 29, no. 1, pp. 37-69, 2013.
[8] C. Musto, G. Semeraro, P. Lops, M. de Gemmis, G. Lekkas, “Personalized finance advisory through case-based recommender systems and diversification strategies,” Decis. Support Syst., pp. 100-111, 2015.
[9] J. Lu, D. Wu, M. Mao, W. Wang, G. Zhang, “Recommender system application developments: a survey,” Decis. Support Syst., pp. 12-32, 2015.
[10] D. C. Liu, S. Rogers, R. Shiau, D. Kislyuk, K. C. Ma, Z. Zhong, J. Liu, Y. Jing, “Related pins at pinterset: The evolution of a real-world recommender system,” In Proceedings of the 26th International Conference on World Wide Web, pp. 585-592, 2017.
[11] X. Ning, C. Desrosiers, G. Karpysh, “A comprehensive survey of neighborhood-based recommendation methods,” In Recommender Systems Handbook. Springer, pp. 37-76, 2015.
[12] M. de Gemmis, P. Cops, M. Custo, F. Narducci, G. Semeraro, “Semantics-aware content-based recommender systems,” In Recommender Systems Handbook. Springer, pp. 119-159, 2015.
[13] H. Zhang, Y. Ji, J. Li, Y. Ye, “A Triple Wing Harmonium Model for Movie Recommendation,” IEEE Trans. Ind. Informatics, vol. 12, no. 1, pp. 231-239, 2016.
[14] H. Zhang, Y. Sun, M. Zhao, T. W. Chow, Q. M. J. Wu, “Bridging User Interest to Item Content for Recommender Systems: An Optimization Model,” IEEE Trans. Cybern., vol. 50, no. 10, pp. 4268-4280, 2020.
[15] J. Bobadilla, F. Ortega, A. Hernandez, A. Gutierrez, “Recommender systems survey,” Knowledge-Based Systems, vol. 46, pp. 109-132, 2013.
[16] Y. Koren, R. Bell, C. Volinsky, “Matrix factorization techniques for recommender systems,” Computer, vol. 42, no. 8, pp. 30-37, 2009.
[17] A. Minh, R. R. Salakhutdinov, “Probabilistic matrix factorization,” In Proceedings of the 21st Advances in Neural Information Processing Systems, pp. 1257-1264, 2008.
[18] Y. Shi, M. Larson, A. Hanjalic, “List-wise learning to rank with matrix factorization for collaborative filtering,” In Proceedings of the 4th ACM Conference on Recommender Systems, pp. 269-272, 2010.
[19] S. Rendle, “Scaling Factorization Machines to Relational Data,” In Proceedings of the 21st International Conference on Machine Learning, pp. 160-167, 2008.
[20] A. Graves and N. Jaitly, “Towards end-to-end speech recognition with recurrent neural networks,” In Proceedings of the 31st International Conference on Machine Learning, pp. 1764-1772, 2014.
[21] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, T. S. Chua, “Neural collaborative filtering,” In Proceedings of the 26th International Conference on World Wide Web, pp. 173-182, 2017.
[22] H. Zhang, Y. Ji, J. Song, Z. Liu, Y. G. Jiang, T. S. Chua, “Nais: Neural attentive item similarity model for recommendation,” IEEE Trans. on Knowl. and Data Eng., vol. 30, no. 12, pp. 2354-2366, 2018.
ZHEFU WU received the PhD degree from the College of Information Science and Electronic Engineering, Zhejiang University, China, in 2000. He is currently an associate professor with the College of Information Engineering, Zhejiang University of Technology, China. His research interests include data mining, machine learning, wireless communication and sensing.

AGYEMANG PAUL received B.Ed. Science degree from University of Education, Ghana, in 2011. He received MS Degree in Information Engineering from Zhejiang University of Technology, China, in 2018. He is currently working toward the PhD degree in the School of Information Engineering, Zhejiang University of Technology, China. His research interests include data mining, machine learning, and recommendation systems.

JUNZHUO CAO received the B.E. degree from the Renai College, Tianjin University, China, in 2020. He is currently working toward the M.E degree in the College of Information Engineering, Zhejiang University of Technology, China. His research interests include data mining, machine learning, and recommendation systems.

LUPING FANG received the B.E. and M.E. degrees from the College of Biomedical Engineering and Instrument Science, Southeast University and Zhejiang University, China, in 1990 and 1993, respectively. He is currently a Professor with the College of Information Engineering, Zhejiang University of Technology, China. His research interests include data mining, machine learning, medical signal processing and applications.