Pre-trained Behavioral Model for Malicious User Prediction on Social Platform

Meng Jiang[1], **Wenjie Wang**[2*], **Shaofeng Hu**[3], **Kaishen Ou**[3], **Zhenjing Zheng**[3], **Fuli Feng**[1*]
[1]University of Science and Technology of China
[2]National University of Singapore
[3]Weixin
jiangm@mail.ustc.edu.cn, wenjiewang96@gmail.com, hugohu@tencent.com, kaishenou@tencent.com,
cszjzheng@tencent.com, fulifeng93@gmail.com

Abstract

The proliferation of malicious users on social platforms poses significant financial and psychological threats, with activities ranging from scams to the dissemination of illicit content. Existing malicious user prediction comprises supervised and self-supervised learning methods. However, the former relies on extensive labeled malicious users for training, while the latter typically focuses on one form of malicious activity and depends heavily on manually crafted rules and features during pre-training. Moreover, existing pre-training methods fail to effectively capture the crucial repetitive and sporadic behavior patterns of malicious users. To address these limitations, we propose a **Ma**licious User Behavior **P**re-training framework (MaP) to build pre-trained behavior models. MaP integrates malicious pattern recognition with behavior consistency augmentation and local disruption augmentation strategies for contrastive learning to capture repetitive and sporadic malicious patterns, respectively. We instantiate MaP on a billion-level behavior pre-training scenario within an industry context. Both online and offline evaluations validate the superior performance of MaP in malicious user detection and classification.

1 Introduction

Malicious users have been rampant on social media platforms in recent years. Malicious users often engage in illegal activities including scams (Whitty and Buchanan 2012; Wash 2020), gambling (James and Bradley 2021), unethical marketing (Xu and Li 2021), and the distribution of illicit content on social platforms, posing financial and psychological harm to the victims. For example, global social media scams resulted in a staggering $1.02 trillion loss in 2023 according to Global Anti-Scam Alliance[1], while Weixin[2] has detected over 200 thousand cases of pornography dissemination within just three months in 2024. Consequently, predicting malicious users on social media platforms is a critical research challenge.

Existing work primarily focuses on predicting malicious users based on user behaviors, roughly falling into two folds.

- Supervised learning methods utilize labeled datasets of malicious users to learn the malicious user patterns. Among these, one line models users and behaviors as graphs, transforming into a node classification task (Hu et al. 2019; Liu et al. 2021). Another line composes user behaviors into sequences and leverages temporal models for prediction (Zhu et al. 2020; Xiao et al. 2024). However, these methods heavily rely on extensively labeled malicious users for training, while labeling malicious users is challenging (Hu et al. 2019).
- In contrast, many efforts utilize self-supervised learning for pre-training, followed by fine-tuning with a small number of labeled samples such as UB-PTM (Liu et al. 2022) and SAGE (Wang et al. 2023). However, malicious users on social platforms fall into various categories, each displaying distinct behavioral patterns. Previous research has often concentrated on identifying a single type of malicious activity, such as fraud (Liu et al. 2022) or cash-out schemes (Hu et al. 2019). Moreover, these studies depend heavily on manually crafted rules (Liu et al. 2022) and features (Wang et al. 2023), hindering the adaptability.

To overcome these limitations, we propose developing a pre-trained behavior model with two primary objectives: 1) capturing the behavior patterns of diverse malicious user categories to generate distinguishable user representations, and 2) eliminating reliance on manually crafted rules and features, thereby enhancing adaptability. The pre-trained behavior model can be fine-tuned for downstream tasks in malicious user prediction, including 1) *malicious user detection*, which classifies users as malicious or not, and 2) *malicious user classification*, which identifies different categories of malicious users.

However, capturing the heterogeneous behavior patterns across different categories of malicious users presents a significant challenge. Previous pre-training approaches based on reconstruction tasks (Sun et al. 2019) and contrastive learning (Wu et al. 2022; Franceschi, Dieuleveut, and Jaggi 2019; Fan, Zhang, and Gao 2020; Yang and Hong 2022) struggle to capture the unique patterns of different types of malicious users, especially 1) **repetitive malicious behavior patterns**, where malicious users repeat malicious activities across multiple periods. For example, users of malicious marketing will repeatedly post on social media feeds to promote their products; 2) **sporadic malicious behavior pat-**

*Corresponding authors.

[1]https://www.gasa.org/research.
[2]https://www.weixin.qq.com/.

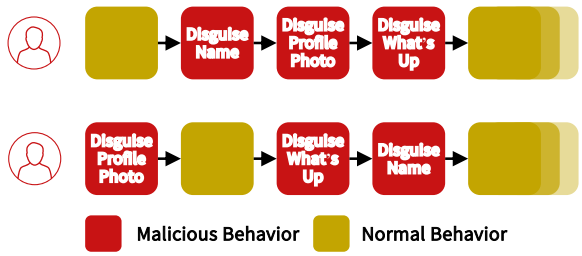


Figure 1: Illustration of the sporadic malicious behaviors, which are shuffled with normal activities.

terns, characterized by infrequent malicious behaviors that may be shuffled and hidden within normal behaviors, increasing the concealment of malicious behaviors. As shown in Figure 1, some malicious users may shuffle malicious behaviors with normal activities.

To capture these malicious patterns, we propose a new **Ma**licious User Behavior **P**re-training framework (MaP) to build a pre-trained behavior model. Specifically, MaP comprises three pre-training stages. 1) The first stage is traditional **masked behavior reconstruction** (Sun et al. 2019), which masks part of behavior sequences and then reconstructs them to learn the behavior transition relations. 2) The second stage is **feature learning for malicious behavior patterns**, which designs contrastive samples to capture the repetitive and sporadic malicious patterns. In particular, 1) MaP leverages *behavior consistency augmentation* to sample the behaviors of the same user across different periods as positive samples and different users' behaviors as negative samples for contrastive learning. This pushes the pre-trained behavior models to capture repetitive malicious behavior patterns across periods. Besides, 2) MaP adopts *local disruption augmentation* to disrupt local segments of a user's behavior sequence as positive samples while keeping different users' behaviors as negative ones. This aims to make the pre-trained behavior model less sensitive to local disruptions in user behaviors, thereby better capturing sporadic malicious behavior patterns. 3) As an extension, MaP conducts **feature learning for pseudo malicious behaviors**. Considering the low proportion of malicious users in the pre-training data, we improve the proportion of users being reported, blocked, or frequently deleted as pseudo-malicious users, facilitating malicious behavior pattern recognition in the pre-training.

We apply MaP to a large-scale industrial scenario: malicious user prediction on the Weixin platform. Initially, we pre-trained a BERT model (Devlin et al. 2019) with MaP on a high-quality dataset comprising 20 million users and 1 billion user behaviors. Subsequently, we fine-tuned the model using a downstream dataset containing 1 million users, including two specific tasks: malicious user detection and malicious user classification. Both online and offline evaluations demonstrate that MaP significantly outperforms previous approaches.

To summarize, our contributions are threefold:

- We investigate the limitations of existing user behavior pre-training methods on social platforms, with a focus on the repetitive and sporadic patterns of malicious users.
- We propose MaP, a novel three-stage user behavior pre-training framework designed to capture diverse behavior patterns of malicious users.
- We conduct large-scale pre-training, downstream fine-tuning, online and offline evaluations on the Weixin platform, validating the effectiveness of the proposed MaP.

2 Related Work

In this section, we review related work in two folds: malicious user prediction and pre-training behavior model.

2.1 Malicious Users Prediction

The proliferation of malicious users on social platforms compels companies to explore countermeasures. Early malicious user prediction often relies on user profiles (Jiang et al. 2023) or the content (Shu et al. 2017) they generate. Recently, a considerable amount of research has been dedicated to leveraging user behavior data to predict malicious users. These efforts can be categorized into two types.

Supervised learning methods use the labeled datasets of malicious users to learn patterns of malicious users.

- One line models users and behaviors as graphs, and graph neural networks (GNNs) have been employed in detecting fraudulent transactions (Hu et al. 2019; Liu et al. 2021; Sánchez-Corcuera, Zubiaga, and Almeida 2024). For example, HACUD (Hu et al. 2019) introduces a meta-path-based graph embedding method that extracts feature representations of users. IHGAT (Liu et al. 2021) devise a heterogeneous transaction-intention network and a GNN to elaborately model user intentions and leverage the transaction-level interactions. MINT (Xiao et al. 2023) build a time-aware behavior graph for each user, and use GNN to capture users' short-term, medium-term, and long-term intentions.
- Another line composes user behaviors into sequences and leverages temporal models for prediction (Graves and Graves 2012; Vaswani et al. 2017). For example, LIC Tree-LSTM (Liu et al. 2020) and VecAug (Xiao et al. 2024) use the pages that users visit to represent their behaviors. In HEN (Zhu et al. 2020), each behavior is represented using behavioral content such as IP address, Card ID, and behavior category.

However, these methods heavily rely on extensively labeled malicious users for training, while labeling malicious users is challenging (Hu et al. 2019).

Therefore, some work has focused on using unlabeled datasets to learn patterns of malicious users. For example, UB-PTM (Liu et al. 2022), SAGE (Wang et al. 2023) rely on manual rules or features to pre-train models, restricting their adaptability and hindering their transferability to the Weixin platform. Moreover, these methods have often concentrated on identifying a single type of malicious activity, such as fraud (Liu et al. 2022) or cash-out schemes (Hu et al. 2019), yet there are various types of malicious users on social platforms.

In this paper, our goal is to address these limitations by eliminating reliance on manually crafted rules and features

and capturing the behavior patterns of a wide range of malicious user categories.

2.2 Pre-training Behavior Model

Inspired by the success of pre-training methods in NLP tasks (Devlin et al. 2019), many researchers turn their attention to the application of pre-training methods in behavior modeling (Sun et al. 2019; Wu et al. 2022; Franceschi, Dieuleveut, and Jaggi 2019; Fan, Zhang, and Gao 2020; Yang and Hong 2022). Pre-trained behavior models can be broadly categorized into two folds. Firstly, some methods are based on reconstruction for pre-training. For example, BERT4Rec (Sun et al. 2019), UserBERT (Wu et al. 2022) all contain masked behavior reconstruction tasks to model the relations between behaviors. MSDP (Fu et al. 2023) replaces predicting the next behavior with predicting the distribution of multiple behaviors, reducing the impact of the stochastic with noise and randomness of behavior sequences. Another type of method involves utilizing contrastive learning (Wu et al. 2022) for pre-training, such as subsequence (Franceschi, Dieuleveut, and Jaggi 2019; Fan, Zhang, and Gao 2020) and contextual (Yang and Hong 2022) consistency. However, these methods fail to capture repetitive and sporadic patterns in user behavior sequences.

In this paper, we propose a novel three-stage user behavior pre-training framework to capture both repetitive and sporadic behavior patterns of malicious users.

3 Problem Formulation

Malicious user prediction aims to utilize user behavior to predict malicious users, involving two tasks: malicious user detection and malicious user classification. The goal of malicious user detection is to differentiate between normal users and malicious users. Malicious user classification aims to classify fine-grained categories of malicious users.

User behaviors are represented as behavior sequences in chronological order. Due to privacy concerns on social platforms, the inputs of behaviors mainly consist of behavior IDs without the actual behavior content (Fu et al. 2023). Let $\mathcal{X}$ denote the behavior sequence set, and we denote each sample as $x = \{\mathrm{ID}_1, \mathrm{ID}_2 \cdots\} \in \mathcal{X}$. In the pre-training phase, we aim to pre-train a behavior model with an encoder E to map x to user representation e by a self-supervised learning approach, leading to discriminative user representations. Afterward, we can fine-tune the behavior model on two downstream tasks using labeled data to enhance the model's performance in malicious user detection and classification.

4 Method

In this section, we first briefly present an overview of the malicious user prediction model and then elaborate on the MaP framework with three pre-training stages.

4.1 Overview

Figure 2 illustrates a representative pipeline for malicious user prediction. We use BERT (Devlin et al. 2019) as the backbone encoder with several considerations: 1) BERT's modest parameter size renders it suitable for online deployment; 2) it is a bidirectional transformer architecture that can capture more contextual information (Sun et al. 2019). Following the approach in ConSERT (Yan et al. 2021), we also employ average pooling to obtain the user representation. We consider further enhancing the encoder through a universal pre-training method to obtain distinguishable user representations for downstream tasks. To this end, we propose MaP with three pre-training stages to build a pre-trained behavior model. Next, we detail the three stages, respectively.

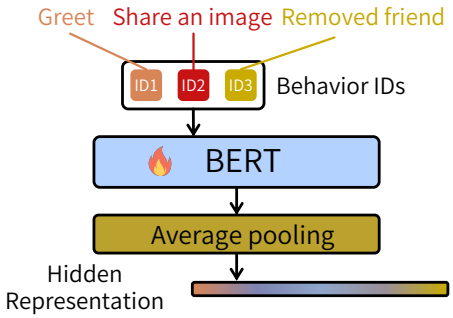


Figure 2: Illustration of malicious user prediction models. The blue box represents the backbone encoder, followed by the average pooling over behavior representations.

4.2 Masked Behavior Reconstruction

The first is **Masked Behavior Reconstruction** (MBR). Following the BERT4Rec (Sun et al. 2019), we use an MBR task to model the relations between behaviors. As shown in Figure 3a, we randomly mask partial behaviors within the behavior sequence and utilize the representations from the hidden layer at those positions to predict the masked behaviors. We randomly selected 15% of the behaviors for prediction, in which 80% are replaced with a "[MASK]" token, 10% are randomly substituted with other behaviors, and 10% remain unchanged. Subsequently, we employ Cross-Entropy loss for optimization. For each user behavior sequence, we have the following loss:

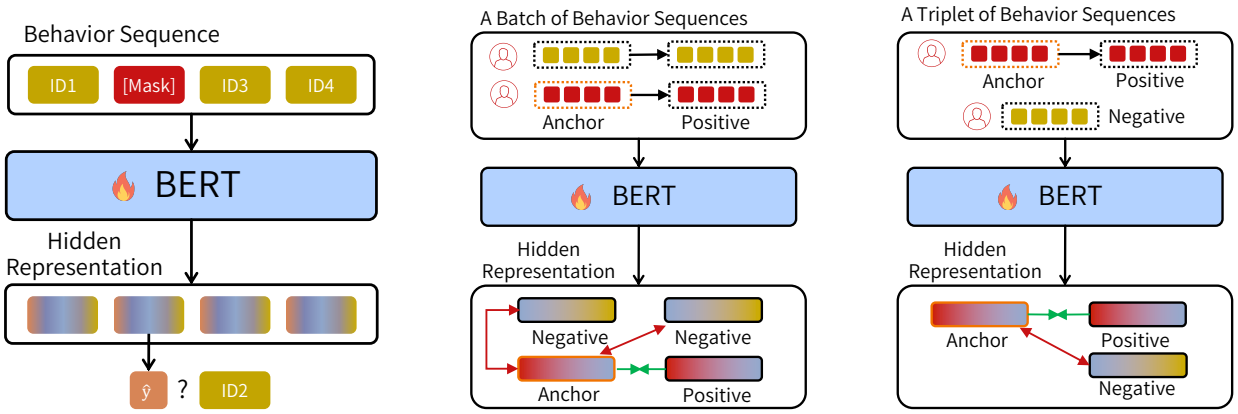
$$\mathcal{L}_{\mathrm{MBR}} = \frac{1}{N} \sum_{i}^{N} CrossEntropy(y_i, \hat{y}_i), \tag{1}$$

where N denotes the number of masked behaviors, and the y_i and $\hat{y}_i$ denote the ground truth and predicted i-th masked behavior, respectively.

4.3 Feature Learning for Malicious Behavior Patterns

The second stage is **Feature Learning for Malicious Behavior Patterns** (FL4MBP), which utilizes contrastive learning methods to capture the special malicious user patterns: repetitive and sporadic malicious behavior patterns. Accordingly, we devise two behavior augmentation methods.

- **Behavior consistency augmentation** (BCA). To learn repetitive malicious behavior patterns, we sample the user's behavior across different periods as positive



(a) MBR (b) FL4MBP (c) FL4PMB

Figure 3: Illustration of three pre-training stages of MaP. Green arrows represent increasing the similarity between positive pairs while red arrows indicate the opposite. (a): Masked Behavior Reconstruction captures the temporal behavior transition patterns; (b): Malicious Behavior Pattern Recognition captures the repetitive and sporadic malicious behavior patterns; (c): Pseudo Malicious Behavior Recognition can better learn malicious patterns by sampling more pseudo-malicious users.

samples. Unlike behavior sequence matching in UserBERT (Wu et al. 2022), which predicts whether two sequences belong to the same user, we tighten the hypothesis by assuming that the user exhibits similar behavior patterns across different periods, to capture repetitive malicious behaviors. For instance, users engaged in unethical marketing may post promoting products daily in their moments, exhibiting repetitive behavior patterns.

- **Local disruption augmentation** (LDA). To learn sporadic malicious behavior patterns shown in Figure 1, we shuffle local segments of a user's behavior sequence as positive samples. The existing contrastive learning methods based on contextual consistency (Yang and Hong 2022) and subsequence consistency (Franceschi, Dieuleveut, and Jaggi 2019; Fan, Zhang, and Gao 2020) may potentially overlook sporadic malicious behavior patterns. By locally disrupting user behavior sequences, the pre-trained behavior model can capture infrequently occurring and shuffled user behavior patterns more effectively. Moreover, to enhance the robustness, we sample various time intervals for local disruption, such as 5 minutes, 10 minutes, 30 minutes, and 1 hour.

As depicted in Figure 3b, given a behavior sequence, we randomly choose one strategy from BCA and LDA to generate a positive sample, while considering behavior sequences of other users in the same batch as negative samples. We utilize InfoNCE (van den Oord, Li, and Vinyals 2018) for contrastive learning by following the specifications in SiT (Ahmed, Awais, and Kittler 2021). Formally, the loss for each sample in a batch is as follows:

$$\mathcal{L}_{\text{FL4MBP}}^{x_i,x_j} = \frac{e^{sim(e_i,e_j)/\tau}}{\sum_{k=1,k!=i}^{2N} e^{sim(e_i,e_k)/\tau}}, \tag{2}$$

where x_i and x_j form a pair consisting of the anchor and the positive sample; e_i and e_j denote the representation of x_i and x_j; $2N$ represents the double batch size, and we regard the anchor and positive samples of other users in the batch as negative samples for x_i. The function $sim(\cdot,\cdot)$ indicates the similarity function such as the Cosine function and τ represents the temperature coefficient, which is utilized to adjust the focus on challenging samples. Subsequently, the batch loss is as follows:

$$\mathcal{L}_{\text{FL4MBP}} = -\frac{1}{2N}\sum_{i=1}^{2N} log(L_{\text{FL4MBP}}^{x_i,x_j}), \tag{3}$$

where x_j is the positive sample of x_i. The reason for the length being $2N$ is that the augmented behavior sequence is also treated as an anchor to calculate $L_{\text{FL4MBP}}^{x_i,x_j}$.

4.4 Feature Learning for Pseudo Malicious Behaviors

The third stage is **Feature Learning for Pseudo Malicious Behaviors** (FL4PMB). Considering the low proportion of malicious users in the pre-training dataset, we propose a *Pseudo-malicious User Sampling* (PUS) strategy to increase the proportion of malicious users, facilitating the pre-trained behavior model to capture the patterns of malicious users. Specifically, we sample more users with abnormal activities for contrastive learning. The first screening criterion is users who have been reported on social platforms. Additionally, users who are blocked or unfriended multiple times are more likely to be malicious users. Then we treat these users as pseudo-malicious users and randomly select a normal user for contrastive learning.

As shown in Figure 3c, we construct triplets "anchor, positive sample, negative sample", where the anchor and negative sample are selected from samples with different pseudo labels, while the positive sample is augmented from the anchor by BCA or LDA in Section 4.3. A triplet loss (Schroff, Kalenichenko, and Philbin 2015) is applied to distinguish between pseudo-malicious and normal users. Following Sentence-BERT (Reimers and Gurevych 2019), we calculate a margin loss for a triplet $< x_a, x_p, x_n >$ by

$$\mathcal{L}_{\text{FL4PMB}}^{x_a,x_p,x_n} = max\left\{sim(e_a,e_p) - sim(e_a,e_n) + margin, 0\right\}, \tag{4}$$

where e_k denotes the representation of x_k extracted by the backbone encoder, $sim(\cdot, \cdot)$ denotes a similarity function, and $margin$ controls the minimum distance between the anchor and the negative sample.

To further improve the model's ability to distinguish between normal and pseudo-malicious users, we introduce a strategy of selecting hard negative samples within the batch:

$$sim(e_a, e_n) = min\left\{sim(e_a, e_n^i), \forall e_n^i \in \mathcal{E}_n\right\}, \quad (5)$$

where $\mathcal{E}_n$ denotes the set of negative samples in the batch, and we select the hardest negative sample in the batch to optimize the margin in Equation (5).

4.5 Training

In this subsection, we summarize the whole pre-training process. Initially, we utilize $\mathcal{L}_{\text{MBR}}$ in MBR to learn the transition relationships between user behaviors. It is worth noting that we do not use the PUS strategy at this stage. Because the model benefits from exposure to a large number of samples during this phase. Subsequently, we jointly pre-train the behavior model via the FL4MBP and FL4PMB stages. The loss function is as follows:

$$\mathcal{L} = w_1 * \mathcal{L}_{\text{FL4MBP}} + w_2 * \mathcal{L}_{\text{FL4PMB}}, \quad (6)$$

where w_1 and w_2 are hyperparameters to balance the two losses. Finally, the pre-trained behavior model can be fine-tuned on downstream tasks for practical application.

5 Experiment

We apply MaP to the Weixin platform for malicious user prediction and conduct extensive offline and online experiments to answer the following questions.

- **RQ1**: Is MaP effective for malicious user detection and classification on the Weixin platform?
- **RQ2**: What are the impacts of each stage of the MaP?
- **RQ3**: Can MaP learn distinguishable representations between malicious and normal users?

5.1 Pre-training Datasets and Settings

We gather multiple large-scale user behavior datasets from the social platform Weixin, adhering to strict security and privacy regulations. All data is anonymized. For the sake of convenience in storage and online usage, the user behavior sequences in our datasets encompass the user's activities throughout one entire day. Our datasets comprise 274 distinct behaviors in user behavior sequences, which can be categorized into 4 classes: user social behavior, friendship interaction behavior, group behavior, and account information behavior. For the MBR task, we sample 20 million users on May 28, 2024. For FL4MBP and FL4PMB, we sample all eligible pseudo-malicious users (611 thousand in total) on the same day, and pair them with an equal number of normal users. To facilitate the execution of BCA, we also sample the behavior data from May 29, 2024 as positive samples. Specifically, May 28 and 29, 2024 fall on Tuesday and Wednesday respectively, avoiding potential harm to the BCA due to differences between workdays and weekends.

	#Normal	#Malicious	#Malicious Rate
Train	344,397	670	0.19%
Test	348,555	691	0.20%
Total	692,952	1,361	0.20%

Table 1: The statistical information of malicious user detection dataset.

We use 8 V100 GPUs for pre-training. We set the number of BERT (Devlin et al. 2019) encoder blocks to 2, the number of the self-attention heads to 12, and the hidden size to 768, resulting in 15 million trainable parameters. We perform dropout at each layer with ratio 0.1. Adam (Kingma and Ba 2015) is selected as the optimizer with the learning rate of $1e-5$. The batch size is set to 32.

5.2 Downstream Tasks and Evaluation Metrics

Malicious User Detection Task. The *malicious user detection* task aims to predict the probability of users being malicious based on their behavior sequences. The labels are derived from manual reviews of potential malicious users reported by others.

To prevent data leakage, we sample the training and test sets 20 days apart. The training set is sampled from June 5, 2024, and the test set from June 25, 2024. The statistical information of the dataset is presented in Table 1. As shown in Table 1, the proportion of malicious users is very small, accounting for only 0.19%. This poses a challenge to the performance of MaP and other baselines.

We use the Area Under Curve (AUC) and Kolmogorov-Smirnov (KS) (Massey Jr 1951) to evaluate the methods. The AUC is a performance metric derived from the ROC curve. The KS statistic for two classes is simply the largest distance between their two cumulative distribution functions. Both AUC and KS reflect the discriminative power of the model.

Malicious User Classification Task. The *malicious user classification* task aims to further classify malicious users. We sample the reported users who underwent manual review from June 1st to June 20th, 2024, as the training set, and from July 1st to July 5th, 2024, as the testing set. The 10-day interval is to prevent data leakage that may occur when the same user is reported multiple times. All data is anonymized. The dataset contains 6 categories: gambling, pornography, fake reviews, unethical marketing, predatory relationships, and white-labeling. The training set consists of 470,621 samples, while the test set contains 134,028 samples. The malicious user classification task is also characterized by class imbalance, with the highest category in the training set containing 140,000 samples, while the smallest category has only 1,505 samples. We use the Macro Average Precision, Recall, and F1-Score to evaluate methods on the malicious user classification task.

5.3 Compared Methods

We compare our proposed pre-training framework MaP with the following methods:

BERT without pre-training (BERT w/o p). We directly train BERT with randomly initialized parameters on the

	Malicious User Detection		Malicious User Classification		
	AUC	KS	M.A. Precision	M.A. Recall	M.A. F1-Score
BERT w/o p	0.7480	0.4961	0.7319	0.6262	0.6675
BERT4Rec	0.8305	0.6610	0.7629	0.6466	0.6902
UserBERT	0.8398	0.6797	0.7627	0.6531	0.6968
MSDP	0.8210	0.6420	0.7401	0.6575	0.6919
SubSeq	0.8190	0.6380	0.7504	0.6571	0.6954
TS2Vec	0.8392	0.6784	**0.7653**	0.6583	0.6999
MaP	**0.8600**	**0.7201**	0.7608	**0.6678**	**0.7056**

Table 2: Performance comparison. The best results of all methods are indicated in boldface. M.A. denotes the Macro Average.

downstream task, skipping the pre-training phase.

BERT4Rec (Sun et al. 2019). The BERT4Rec uses the masked behaviors reconstruction task to model the relations between behaviors of the same user.

UserBERT (Wu et al. 2022). UserBERT learns the relations between user behaviors and the invariant behavior patterns of users at different time periods.

MSDP (Fu et al. 2023). MSDP aims to mitigate the impact of the disorderliness of user behaviors, by modifying the task from predicting a single behavior to predicting a distribution of multiple behaviors.

SubSeq (Franceschi, Dieuleveut, and Jaggi 2019). SubSeq generates positive samples by assuming that a sequence and its subsequence have similar representations.

TS2Vec (Yang and Hong 2022). TS2Vec utilizes timestamp masking and random cropping for augmented contexts.

In particular, contrastive learning alone shows limited effectiveness. To better compare the differences between various contrastive learning methods, we perform contrastive learning based on the model obtained from BERT4Rec, while also utilizing the PUS strategy.

5.4 Performance Comparison

To answer **RQ1**, we compare the proposed MaP with competitive baselines, and the result is demonstrated in Table 2.

Malicious User Detection. As shown in Table 2, MaP demonstrates superiority in both AUC and KS metrics for malicious user detection task. Specifically, MaP demonstrates the improvements of 2.4% in AUC and 5.9% in KS compared to UserBERT, and the improvements of 2.4% in AUC and 6.1% in KS compared to TS2Vec. As PUS is not employed in UserBERT, its unsatisfactory performance implies the significance of a higher proportion of malicious users in the pre-training data. Moreover, both improvements demonstrate the superiority of BCA and LDA strategies in learning repetitive malicious behavior patterns and sporadic malicious behavior patterns, compared to other contrastive learning methods.

Malicious User Classification. As shown in Table 2, MaP achieves the highest F1-Score in malicious user classification. Due to the subsequent manual review phase, we aim for the model to recall more malicious users while maintaining precision to reduce manual effort. Compared to TS2Vec, MaP achieves an increase of 0.0095 in recall with a decrease in precision of 0.005, indicating the compromise in some precision for improved recall. However, MaP is able to enhance recall significantly while maintaining precision without a substantial decline.

In-depth Analysis. Training BERT only with downstream data (BERT w/o p) demonstrates catastrophic performance. The reasons can be attributed to the scarcity of malicious users in the fine-tuning phase, which highlights the necessity of the pre-training stage for improving performance on downstream tasks.

Moreover, the MSDP shows the least pleasing performance among pre-training methods in both malicious user detection and classification. We speculate that predicting the distribution of multiple behaviors in MSDP could reduce the impact of noise and randomness of sequences, yet it may not effectively capture the relationships between behaviors.

Interestingly, while both SubSeq and TS2Vec are based on BERT4Rec for additional pre-training, SubSeq shows a performance decline, TS2Vec demonstrates some improvement in malicious user detection. SubSeq merely selects a subsequence of the anchor, potentially causing the model to forget sporadic malicious behavior patterns. Despite that TS2Vec incorporates random masking and cropping to capture low-frequency behavior patterns, it falls short in learning disrupted behavior patterns.

5.5 Ablation

To answer **RQ2**, we perform ablation studies to demonstrate the effectiveness of each stage in our pre-training framework MaP. The result is shown in Table 3.

Initially, we discover a significant performance decrease when pre-training the behavior model using a dataset with the PUS strategy at the MBR phase. This indicates that the MBR stage needs more samples to model the relations between user behaviors robustly, thus bringing a solid foundation for the next stage.

Subsequently, we evaluate the performance of the model after removing different modules and observed a decrease in F1-Score across the board. This indicates that enhancing the concentration of malicious users, capturing different malicious behavior patterns, and learning the differences between normal and malicious users can all contribute to malicious user prediction.

In particular, we attempt to use only one augmentation strategy in the FL4MBP phase and observe a significant decrease in model performance. This suggests that malicious user behavior patterns contain repetitive malicious behavior patterns and sporadic malicious behavior patterns, and the two augmentation strategies can effectively learn different types of malicious user behavior patterns.

	Malicious User Detection		Malicious User Classification		
	AUC	KS	M.A. Precision	M.A. Recall	M.A. F1-Score
MaP_{MBR}+PUS	0.8384	0.6767	0.7556	0.6592	0.6989
MaP w/o FL4PMB	0.8427	0.6855	0.7615	0.6568	0.6994
MaP w/o FL4MBP	0.8377	0.6754	0.7537	0.6565	0.6945
MaP w/o PUS	0.8449	0.6898	**0.7725**	0.6377	0.6863
MaP w/o BCA	0.8216	0.6433	0.7540	0.6478	0.6894
MaP w/o LDA	0.8269	0.6538	0.7660	0.6423	0.6867
MaP	**0.8600**	**0.7201**	0.7608	**0.6678**	**0.7056**

Table 3: Ablation studies of every component in MaP. w/o denotes the *without*.

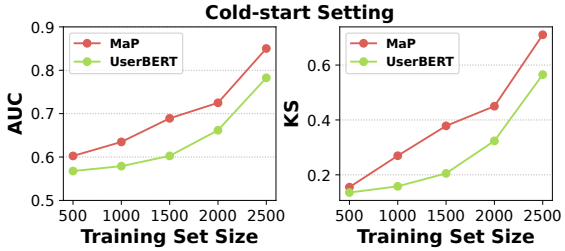


Figure 4: Cold-start setting. The AUC and KS performances with different sizes of training sets on the malicious user detection task.

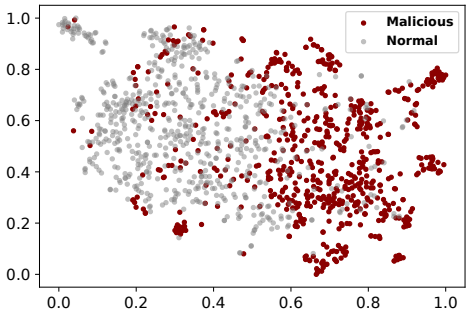


Figure 5: Visualization of malicious and normal representations on the malicious user detection task.

5.6 Cold-start Evaluation

To evaluate the model's performance on downstream tasks with limited data, we simulate a cold-start scenario. Sampling is conducted on the training set from malicious user detection task, with sample sizes ranging from 500 to 2,500, and the proportions of malicious users are all approximately 10%. As shown in Figure 4, MaP surpasses UserBERT across all sample sizes. Notably, with the sample size increasing from 2,000 to 2,500, we can observe significant improvements of 17.26% in AUC and 57.86% in KS. The performance improvement across all sample sizes indicates the superiority of MaP in the cold-start scenario.

5.7 Visualization

To answer **RQ3**, we explore the effectiveness of MaP in malicious user prediction. We employ the PCA method to reduce the dimensions of user representations. We randomly select 670 users from normal users and 670 users from malicious users and calculate their representations by our pre-trained behavior model. As depicted in Figure 5, despite some users being intertwined, a distinct boundary between normal users and malicious users is noticeable. This suggests that MaP can generate distinctive user representations, enhancing the model's performance on downstream tasks.

	AUC	KS
UserBERT	0.8231	0.6463
MaP	**0.8591**	**0.7183**
Improvement	4.4%	11.1%

Table 4: Online A/B testing results.

5.8 Online Experiment

We assess the pre-trained behavior model using the malicious user detection task in Weixin's online environment. We sample one billion users for this evaluation on June 27th. Subsequently, we conduct an online A/B test to compare MaP with UserBERT (Wu et al. 2022), a model already deployed on Weixin's platform. From the results of AUC and KS in Table 4, our pre-trained behavior model by MaP demonstrates significant enhancements in both metrics, with AUC increasing by 4.4% and KS by 11.1%. The superior performance validates that MaP can effectively help capture malicious behavior patterns on Weixin's platform, leading to distinguishable user representations for malicious user detection.

6 Conclusion

In this work, we introduced a novel three-stage pre-training framework, MaP, to regulate the pre-trained behavior model to capture both repetitive and sporadic malicious behavior patterns. Moreover, we perform comprehensive experiments on the Weixin platform through billion-level behavior pre-training, downstream fine-tuning, and online and offline evaluations. Extensive results exhibit the efficacy of MaP in building superior pre-trained behavior models for malicious user detection and classification.

This work leaves many future directions for in-depth exploration. 1) MaP solely relies on user behavior sequences to capture behavior patterns. However, leveraging user social networks and public content in either pre-training or fine-tuning can enhance malicious user prediction. 2) We will also apply the pre-trained behavior model to more malicious behavior issues on social platforms, such as the detection of malicious chat groups or specialized social security for different supervised groups, striving to create a better social environment on the Internet.

Acknowledgements

This work is supported by the National Key Research and Development Program of China (2022YFB3104701), the National Natural Science Foundation of China (62272437).

References

Ahmed, S. A. A.; Awais, M.; and Kittler, J. 2021. SiT: Self-supervised vIsion Transformer. *CoRR*, abs/2104.03602.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*, 4171–4186. ACL.

Fan, H.; Zhang, F.; and Gao, Y. 2020. Self-Supervised Time Series Representation Learning by Inter-Intra Relational Reasoning. *CoRR*, abs/2011.13548.

Franceschi, J.; Dieuleveut, A.; and Jaggi, M. 2019. Unsupervised Scalable Representation Learning for Multivariate Time Series. In *NeurIPS*, 4652–4663.

Fu, C.; Wu, W.; Zhang, X.; Hu, J.; Wang, J.; and Zhou, J. 2023. Robust User Behavioral Sequence Representation via Multi-scale Stochastic Distribution Prediction. In *CIKM*, 4567–4573. ACM.

Graves, A.; and Graves, A. 2012. Long short-term memory. *Supervised sequence labelling with recurrent neural networks*, 37–45.

Hu, B.; Zhang, Z.; Shi, C.; Zhou, J.; Li, X.; and Qi, Y. 2019. Cash-Out User Detection Based on Attributed Heterogeneous Information Network with a Hierarchical Attention Mechanism. In *AAAI*, 946–953. AAAI Press.

James, R. J.; and Bradley, A. 2021. The use of social media in research on gambling: A systematic review. *Current Addiction Reports*, 235–245.

Jiang, M.; Zhang, Y.; Gao, Y.; Wang, Y.; Feng, F.; and He, X. 2023. LightMIRM: Light Meta-learned Invariant Risk Minimization for Trustworthy Loan Default Prediction. In *ICDE*, 3494–3507. IEEE.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

Liu, C.; Gao, Y.; Sun, L.; Feng, J.; Yang, H.; and Ao, X. 2022. User Behavior Pre-training for Online Fraud Detection. In *KDD*, 3357–3365. ACM.

Liu, C.; Sun, L.; Ao, X.; Feng, J.; He, Q.; and Yang, H. 2021. Intention-aware Heterogeneous Graph Attention Networks for Fraud Transactions Detection. In *KDD*, 3280–3288. ACM.

Liu, C.; Zhong, Q.; Ao, X.; Sun, L.; Lin, W.; Feng, J.; He, Q.; and Tang, J. 2020. Fraud Transactions Detection via Behavior Tree with Local Intention Calibration. In *KDD*, 3035–3043. ACM.

Massey Jr, F. J. 1951. The Kolmogorov-Smirnov test for goodness of fit. *J AM STAT ASSOC*.

Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP*, 3980–3990. ACL.

Sánchez-Corcuera, R.; Zubiaga, A.; and Almeida, A. 2024. Early Detection and Prevention of Malicious User Behavior on Twitter Using Deep Learning Techniques. *IEEE Trans. Comput. Social Syst.*

Schroff, F.; Kalenichenko, D.; and Philbin, J. 2015. FaceNet: A unified embedding for face recognition and clustering. In *CVPR*, 815–823. IEEE.

Shu, K.; Sliva, A.; Wang, S.; Tang, J.; and Liu, H. 2017. Fake News Detection on Social Media: A Data Mining Perspective. *KDD*, 22–36.

Sun, F.; Liu, J.; Wu, J.; Pei, C.; Lin, X.; Ou, W.; and Jiang, P. 2019. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. In *CIKM*, 1441–1450. ACM.

van den Oord, A.; Li, Y.; and Vinyals, O. 2018. Representation Learning with Contrastive Predictive Coding. *CoRR*, abs/1807.03748.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In *NIPS*, 5998–6008.

Wang, Z.; Wu, Q.; Zheng, B.; Wang, J.; Huang, K.; and Shi, Y. 2023. Sequence As Genes: An User Behavior Modeling Framework for Fraud Transaction Detection in E-commerce. In *KDD*, 5194–5203. ACM.

Wash, R. 2020. How Experts Detect Phishing Scam Emails. *Proc. ACM Hum.-Comput. Interact.*

Whitty, M. T.; and Buchanan, T. 2012. The Online Romance Scam: A Serious Cybercrime. *Cyberpsychology, Behavior, and Social Networking*, 181–183.

Wu, C.; Wu, F.; Qi, T.; and Huang, Y. 2022. UserBERT: Pre-training User Model with Contrastive Self-supervision. In *SIGIR*, 2087–2092. ACM.

Xiao, F.; Cai, S.; Chen, G.; Jagadish, H. V.; Ooi, B. C.; and Zhang, M. 2024. VecAug: Unveiling Camouflaged Frauds with Cohort Augmentation for Enhanced Detection. arXiv:2408.00513.

Xiao, F.; Wu, Y.; Zhang, M.; Chen, G.; and Ooi, B. C. 2023. MINT: Detecting Fraudulent Behaviors from Time-series Relational Data. *VLDB*, 3610–3623.

Xu, P.; and Li, Z. 2021. Research on the Chaos, Causes and Countermeasures of Information Dissemination in WeChat Moments. In *ICLCCS*, 147–152. Atlantis Press.

Yan, Y.; Li, R.; Wang, S.; Zhang, F.; Wu, W.; and Xu, W. 2021. ConSERT: A Contrastive Framework for Self-Supervised Sentence Representation Transfer. In *IJCNLP*, 5065–5075. ACL.

Yang, L.; and Hong, S. 2022. Unsupervised Time-Series Representation Learning with Iterative Bilinear Temporal-Spectral Fusion. In *ICML*, 25038–25054. PMLR.

Zhu, Y.; Xi, D.; Song, B.; Zhuang, F.; Chen, S.; Gu, X.; and He, Q. 2020. Modeling Users' Behavior Sequences with Hierarchical Explainable Network for Cross-domain Fraud Detection. In *WWW*, 928–938. ACM.