Self-Supervised Interest Transfer Network via Prototypical Contrastive Learning for Recommendation

Guoqiang Sun[1,2], Yibin Shen[2], Sijin Zhou[2], Xiang Chen[1], Hongyan Liu[1], Chunming Wu[1],
Chenyi Lei[2], Xianhui Wei[2], Fei Fang[2]

[1]College of Computer Science and Technology, Zhejiang University, China
[2]Alibaba Group, China
{sungq, wasdnsxchen, hyliu20, wuchunming}@zju.edu.cn,
{shilou.syb, zhousijin.zsj, chenyi.lcy, xianhui.wxh, mingyi.ff}@alibaba-inc.com

Abstract

Cross-domain recommendation has attracted increasing attention from industry and academia recently. However, most existing methods do not exploit the interest invariance between domains, which would yield sub-optimal solutions. In this paper, we propose a cross-domain recommendation method: **S**elf-supervised **I**nterest **T**ransfer **N**etwork (SITN), which can effectively transfer invariant knowledge between domains via prototypical contrastive learning. Specifically, we perform two levels of cross-domain contrastive learning: 1) instance-to-instance contrastive learning, 2) instance-to-cluster contrastive learning. Not only that, we also take into account users' multi-granularity and multi-view interests. With this paradigm, SITN can explicitly learn the invariant knowledge of interest clusters between domains and accurately capture users' intents and preferences. We conducted extensive experiments on a public dataset and a large-scale industrial dataset collected from one of the world's leading e-commerce corporations. The experimental results indicate that SITN achieves significant improvements over state-of-the-art recommendation methods. Additionally, SITN has been deployed on a micro-video recommendation platform, and the online A/B testing results further demonstrate its practical value. Supplement is available at: https://github.com/fanqieCoffee/SITN-Supplement.

Introduction

In single-domain recommender systems, only historical interaction data in one domain are utilized to predict users' behaviors. Earlier work like FM (Rendle 2010) considers both first-order and second-order feature interactions to boost performance. Some follow-up works such as DIN (Zhou et al. 2018) and DIEN (Zhou et al. 2019) model user behavior sequences with the attention mechanism.

However, there are some critical and non-trivial issues that the single-domain recommendation methods are difficult to address, such as cold-start and data sparsity. Therefore, cross-domain recommendation (CDR) methods emerged and tried to solve these issues. Specifically, these methods aim to achieve higher performance by efficiently utilizing and transferring knowledge among multiple domains. Many CDR methods are dedicated to establishing connections between domains. For instance, CoNet (Hu,

Zhang, and Yang 2018) establishes cross-connections between base networks to transfer knowledge across domains. With the rapid growth of contrastive learning in computer vision, some methods combine it with CDR. For example, SEMI (Lei et al. 2021) maximizes a lower bound of the mutual information between different domains. Nevertheless, most of existing methods do not explore and exploit the interest invariance between domains. MiNet (Ouyang et al. 2020) just use the attention mechanism to associate the interest of each domain. SEMI only focuses on the transfer of users' individual behaviors. They both ignore the significance of the interest invariance between domains. We argue that there exists a commonality in user interests between domains, which is beneficial for CDR.

For instance, Figure 1 shows the clicking sequences of a user from the source domain and the target domain of a real-life scenario. Here, the source domain is a e-commerce platform where users can purchase products while the target domain is a community where users can post their feelings about the products they purchased and share their lives. In the source domain, we can notice that the user has three interests: 1) cameras, 2) pet supplies (e.g. cat food), 3) home furnishing supplies (e.g. chairs, tables and sofas). Also, in the target domain, we can observe that the user has three similar interests corresponding to the above interests: 1) buying and using guides for cameras, 2) knowledge of pet raising, 3) home design and decoration. We can find that the user has the interest invariance between domains. Yet, existing methods do not explicitly learn such invariant knowledge and transfer it between domains. We believe that model performance will achieve significant improvement if the role of the interest invariance in cross-domain recommendation is fully exploited.

Nonetheless, exploiting such knowledge is non-trivial, which raises the following challenges: 1) how to extract the knowledge of interests from each domain; 2) how to transfer the interest invariance between domains; 3) how to make the behaviors and interests of the user work jointly on the model. To address such challenges, we propose **S**elf-supervised **I**nterest **T**ransfer **N**etwork (SITN), which can efficiently learn and transfer the invariance of users' interests from different domains and allow users' behaviors to fully interact with the interests to obtain higher model performance. Specifically, we treat users' sequential behavioral
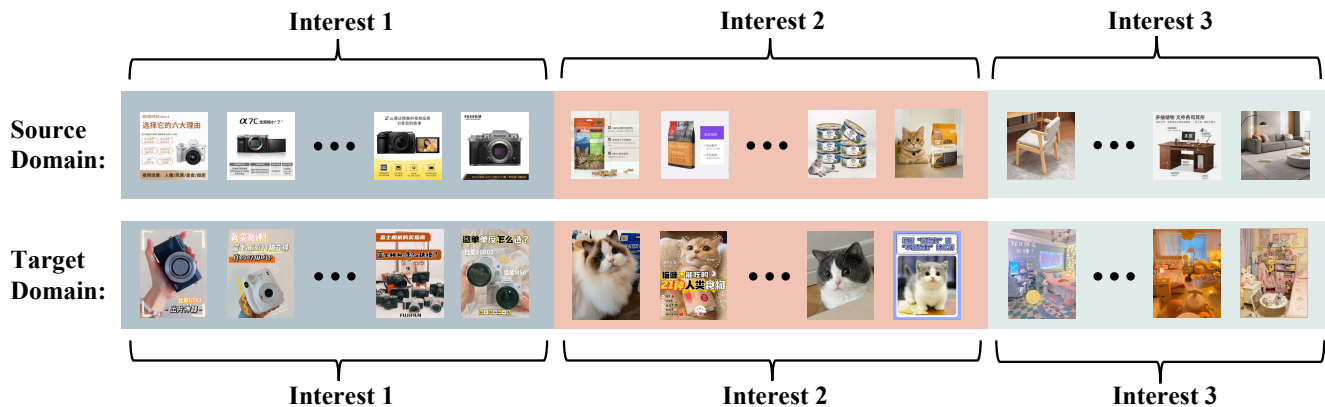
Figure 1: The user clicking sequences in a real-life scenario. The upper part is the user's clicking sequence in a product recommendation platform (source domain), and the lower part is the user's clicking sequence in a content/micro-video recommendation platform (target domain). The user has three interests in both domains and these interests correspond to each other.

representations as instances and the interests encoded by the instances as clusters, which can be trained within batches in an end-to-end manner. Next, we design the instance-to-cluster contrastive learning module, which perform prototypical contrastive learning (Li et al. 2020) across the source domain and the target domain. It treats an instance and the corresponding cluster from the other domain as a positive pair, to efficiently learn and transfer the interest invariance between domains. Meanwhile, we perform contrastive learning on the representations of users' behaviors from different domains like SEMI and we called it instance-to-instance contrastive learning. By employing the two levels of contrastive learning simultaneously, the instance-to-instance and instance-to-cluster contrastive learning can fully interact with each other and work jointly on the model, therefore, improve the model performance. In addition to that, a user's interests are often diverse and can not be determined by a single interest alone. Therefore, we introduce the multi-view interest module which reflects the user's varied interests based on the instance-to-cluster contrastive learning. Last but not least, there are numerous distinct levels of granularity for a user's interests across different interest spaces. For instance, the granularity of a girl's fashion interest space will typically be much finer than that of the digital interest space. In order to remedy the problem, we propose the multi-granularity module, which concentrates on users' different granular interests. The offline experimental results suggest the effectiveness of SITN, and the online A/B testing results further highlight its usefulness.

In summary, we can conclude our main contributions as follows:

- We underline the significance of the interest invariance across domains and explore how it plays a role in cross-domain recommendation.

- We propose the model SITN, which efficiently learn and transfer the interest invariance between domains, and also take into account users' multi-view and multi-

granularity interests.

- We conduct extensive offline experiments on a large-scale industrial dataset and a public dataset. The results show significant improvements over state-of-the-art methods.

- We have deployed SITN on a micro-video recommendation platform and conducted online A/B tests to show its practical value. More details are in the supplement.

Related Work

Cross-domain Recommendation

The basic idea of cross-domain recommendation (CDR) is to introduce knowledge from other domains to alleviate the problems such as cold-start and data sparsity, not only that, also to strengthen user preferences. Therefore, the key to CDR is how to leverage knowledge from domains effectively and establish connections between them efficiently. In the early days, CDR was mainly based on some traditional methods. The collective matrix factorization model (Singh and Gordon 2008) factorizes rating matrices collectively of two domains into user representations and item representations. Besides, the factorization machines-based model named FM-MCMC (Loni et al. 2014) takes into account the user's behaviors in the source domain and target domain simultaneously during the training process. Over the last few years, with the rapid growth of deep learning, some methods based on it have sprung up. The collaborative cross-network (CoNet) (Hu, Zhang, and Yang 2018) establishes cross-connections between base networks to enable dual knowledge transfer. Compared with CoNet, DDTCDR (Li and Tuzhilin 2020) learns a latent orthogonal mapping function between domains, which not only preserves the similarity of users' preferences, but also effectively calculates the reverse mapping function. There is also a method called Mixed Interest Network (MiNet) (Ouyang et al. 2020) that transfers users' interests from source domain to tar-

get domain by using attention mechanisms. To remedy the cold-start problem, PTUPCDR (Zhu et al. 2022) utilizes a meta network to transfer the user preferences between domains. The method most similar to ours is SEMI (Lei et al. 2021), which applies contrastive learning to CDR, bridging the gap between domains by pulling the representations of user behavior sequences in different domains closer. However, SEMI only performs contrastive learning at the instance-to-instance level, neglecting their interests, which will introduce noise and lead to a sub-optimal solution. To this end, our method employs contrastive learning at both instance-to-instance and instance-to-cluster levels simultaneously, which could better leverage and transfer the knowledge between domains.

Contrastive Learning

Contrastive learning (CL) plays a significant role in the self-supervised learning paradigms, aiming to extract essential invariant knowledge by constructing the pretext tasks through the principle of mutual information maximization (Oord, Li, and Vinyals 2018; Velickovic et al. 2019), which achieved great success in computer vision (Chen et al. 2020; He et al. 2020), then extended to natural language processing (Gao, Yao, and Chen 2021) and graph representation learning (Sun et al. 2019; You et al. 2020; Zhu et al. 2021). Recently, more and more methods related to CL have emerged in the field of recommender systems (RS) (Zhou et al. 2020; Wu et al. 2021; Xie et al. 2022). Specifically, the main idea of contrastive learning is to construct pairs of positive and negative samples by augmenting the data, then pull the former closer and push the latter farther in the feature space. At early stages, a variety of works focused on constructing different pretext tasks through different ways of augmentation. For instance, one is to exploit feature correlations and augment data by feature masking (Yao et al. 2021). Additionally, a method known as prototypical contrastive learning (PCL) (Li et al. 2020), which was recently applied to RS, treats an instance and the cluster to which it belongs as a positive pair. One of the PCL-based methods in RS is to construct positive sample pairs by semantic neighbors referring to semantically similar neighbors which may not be directly reachable on graphs in graph collaborative filtering (Lin et al. 2022). The other method (Chen et al. 2022) in Sequential Recommendation, which is similar to our work, treats the clusters of behavior sequences as users' intents like prototypes. While their method belongs to single-domain recommendation, ours is used for cross-domain recommendation, and the interest clusters are trainable within batches. Not only that, we also consider the user's multi-granularity and multi-view interests.

Preliminaries and Notations

We have a set of users and items which can be denoted by $\mathcal{U}$ and $\mathcal{V}$ respectively. Let $\mathcal{S}^u = \left\{ s_1^u, s_2^u, \cdots, s_{|\mathcal{S}^u|}^u \right\}$ and $\mathcal{T}^u = \left\{ t_1^u, t_2^u, \cdots, t_{|\mathcal{T}^u|}^u \right\}$ be clicking sequences sorted in chronological order from source domain and target domain respectively for each user $u \in \mathcal{U}$, where $|\mathcal{S}^u|$ and $|\mathcal{T}^u|$ de-

note the length of each sequence. Here, s_i^u and t_j^u are the i-th and j-th items clicked by the user u in the source and target domain respectively. Additionally, we define interest cluster matrix $C_{K \times D}^s$ and $C_{K \times D}^t$ for source domain and target domain respectively, where K and D denote the number of clusters and the dimension of each cluster respectively which are hyper-parameters. Note that $C_{K \times D}^s$ and $C_{K \times D}^t$ are randomly initialized and trainable. The task in our scenario is to predict the probability of user $u \in \mathcal{U}$ potentially clicking the item $v \in \mathcal{V}$ from target domain given sequences $\mathcal{S}^u$ and $\mathcal{T}^u$.

Method

In this section, we first introduce the significant basic modules of **S**elf-supervised **I**nterest **T**ransfer **N**etwork (SITN) (shown in Figure 2) , which incorporated three main modules: 1) *Sequence encoder and aggregation layer*, which captures users' sequential behavior patterns and obtains the representations of users' encoded clicking sequences (instances); 2) *Instance-to-instance contrastive learning*, which performs contrastive learning on the instances from different domains; 3) *Instance-to-cluster contrastive learning*, which performs prototypical contrastive learning on the instances and the interest clusters from different domains. Next, we elaborate our two-stage training paradigm. In the first stage, we employ InfoNCE (Oord, Li, and Vinyals 2018) in a self-supervised learning manner. In the second stage, instead of training from scratch, we use and fine-tune the pre-trained sequence encoders from the previous stage to predict the probability of a user would click a specific item in the target domain.

Sequence Encoder and Aggregation Layer

We employ multi-head attention mechanism (Vaswani et al. 2017), analogous to many other approaches (Chen et al. 2019; de Souza Pereira Moreira et al. 2021), to encode sequences in the source domain and the target domain, enabling the model to concentrate on critical information and thoroughly learn to assimilate it. In this work, we get two clicking sequences from source domain and target domain for each user. Here, we use the self-attention mechanism, which means that Query, Key and Value are all the same (the clicking sequence). Then, we perform scaled dot-product attention and compute the matrix of outputs as follows:

$$Attention(Q, K, V) = softmax(\frac{Q^T K}{\sqrt{d_k}})V, \quad (1)$$

where $Q \in \mathcal{R}^d, K \in \mathcal{R}^d$ and $V \in \mathcal{R}^d$ represent Query, Key and Value respectively, in addition, d_k is the dimension of K. Multi-head attention mechanism is beneficial to learn relevant information in different representation subspaces. Therefore we linearly project Query, Key and Value h times with different trainable weight matrices and then perform attention function on them to compute outputs. Next, the all outputs are concatenated and projected again to get the final outputs. After the sequence encoder layer, we perform aggregation layer (e.g., mean pooling function) to get the representations of encoded sequences.
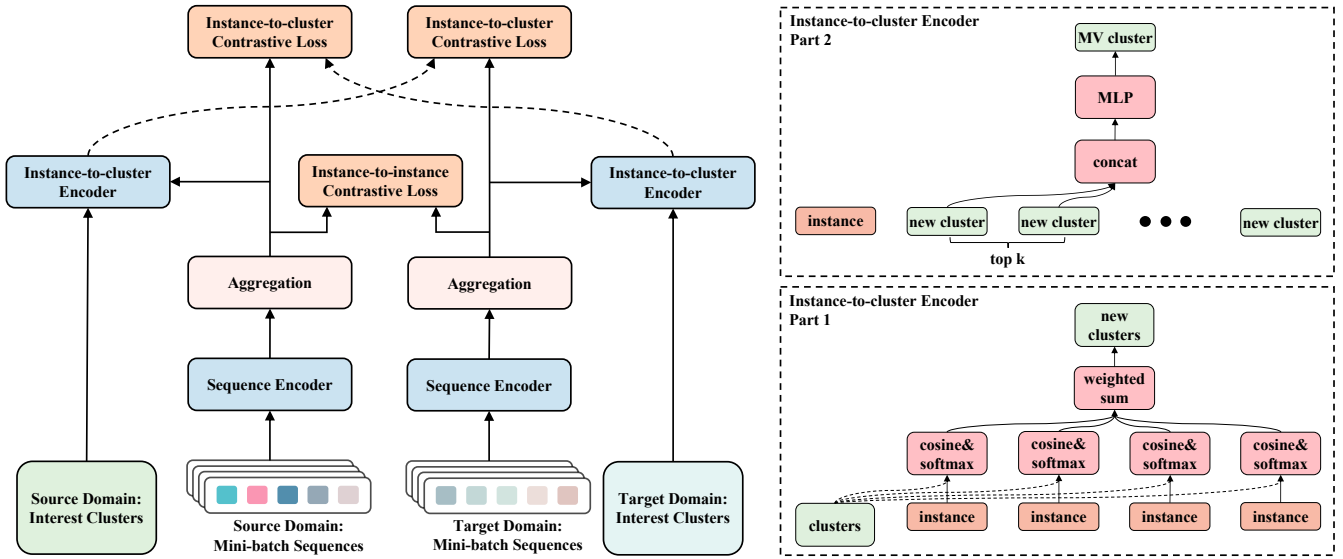
Figure 2: The overall framework of SITN's contrastive learning module. The left part shows the contrastive learning module at a high level. The right part shows the instance-to-cluster encoder. The role of the first part of instance-to-cluster encoder is to generate new clusters from instances and original clusters. The lower right part shows a example of clusters and four instances. The role of the second part of instance-to-cluster encoder is to generate MV clusters from instances and new clusters. The upper right part shows a example of new clusters and one instance, and the value of k is two.

Instance-to-instance Contrastive Learning

In this module, We perform instance-to-instance contrastive learning (CL-I2I), similar to SEMI (Lei et al. 2021), on the representations of encoded clicking sequences between domains. By pulling the representations of encoded clicking sequences of the same user closer in different domains, the goal of CL-I2I is to extract invariant behavior patterns and preferences. Specifically, as shown in Figure 2, let $\mathcal{P}^s = \{p_1^s, p_2^s, \cdots, p_n^s\}$ and $\mathcal{P}^t = \{p_1^t, p_2^t, \cdots, p_n^t\}$ denote the representations of encoded sequences in the source domain and target domain respectively within a mini-batch, where n is the size of mini-batch. For same user, the subscripts of p^s and p^t will be the same. Therefore, we could obtain the outputs (the representations of encoded sequences) as follows:

$$p_i^s = g(f_s(x_i^s)), \qquad p_i^t = g(f_t(x_i^t)), \qquad (2)$$

where $f^s(\cdot)$ and $f^t(\cdot)$ are the encoders and $g(\cdot)$ is the mean pooling function described in the previous subsection, x_i^s and x_i^t are the identical user's clicking sequences in the source domain and target domain respectively as inputs of the encoders. Next, we perform CL-I2I with infoNCE loss,

$$\mathcal{L}_{i2i}^{s2t} = -\sum_{k=1}^{n} log \frac{exp(sim(p_k^s, p_k^t)/\tau)}{\sum_{i=1}^{n} exp(sim(p_k^s, p_i^t)/\tau)}, \qquad (3)$$

where $sim(\cdot)$ is a dot product and τ denotes the the hyperparameter, known as *temperature* in softmax. CL loss is supposed to be symmetrical, so analogously, we could compute $\mathcal{L}_{i2i}^{t2s}$ as follows:

$$\mathcal{L}_{i2i}^{t2s} = -\sum_{k=1}^{n} log \frac{exp(sim(p_k^t, p_k^s)/\tau)}{\sum_{i=1}^{n} exp(sim(p_k^t, p_i^s)/\tau)}. \qquad (4)$$

Combining the two losses, we could obtain instance-to-instance contrastive loss as $\mathcal{L}_{i2i} = \mathcal{L}_{i2i}^{s2t} + \mathcal{L}_{i2i}^{t2s}$.

Instance-to-cluster Contrastive Learning

Considering solely CL-I2I without the user's interests adds noise and yields sub-optimal solution. Motivated by such issue, we propose instance-to-cluster contrastive learning (CL-I2C), which aims to transfer knowledge more effectively and boost performance by exploiting the same user's interest invariance across different domains. Furthermore, CL-I2I and CL-I2C could interact with each other to jointly enhance robustness of the model and achieve higher performance. In this module, we first introduce the instance-to-cluster encoder, which could obtain multi-view clusters from instances. Then we elaborate the instance-to-cluster contrastive loss, which is the key component in our method.

Instance-to-cluster Encoder We define and randomly initialize the interest cluster matrix $C_{K \times D}^s$ and $C_{K \times D}^t$ for source domain and target domain where each row represents a interest cluster. Given $C_{K \times D}^s$, $C_{K \times D}^t$ and the representations of encoded sequences $\mathcal{P}^s$ and $\mathcal{P}^t$, similar to TCC (Shen et al. 2021), as shown in the lower right part of Figure 2, we can compute the new interest clusters in the source domain:

$$u_k^s = \sum_{i=1}^{n} \pi_i^s(k) \cdot p_i^s,$$

$$\pi_i^s(k) = \frac{exp(sim(c_k^s, p_i^s))}{\sum_{k'=1}^{K} exp(sim(c_{k'}^s, p_i^s))}, \qquad (5)$$

where u_k^s and c_k^s denote the k-th new and original interest cluster in the source domain respectively. As a result,

we could get the new interest cluster matrix $U_{K \times D}^s$ in the source domain. Meanwhile, the new interest cluster matrix $U_{K \times D}^t$ in the target domain could be obtained by the same way. We further consider the diversity of user interests, thus we build positive sample pairs with the k interests that are most related to the user's behaviors to generate soft labels, a method we refer to as multi-view CL-I2C. In particular, as shown in the upper right part of Figure 2, we obtain the multi-view clusters (MV clusters) $\mathcal{Q}^s = \{q_1^s, q_2^s, \cdots, q_n^s\}$ and $\mathcal{Q}^t = \{q_1^t, q_2^t, \cdots, q_n^t\}$ which represent the users' multi interests from the source and target domain respectively:

$$
\begin{aligned}
q_i^s &= MLP(Concat(u_{i1}^s, u_{i2}^s, \cdots, u_{ik}^s)), \\
q_i^t &= MLP(Concat(u_{i1}^t, u_{i2}^t, \cdots, u_{ik}^t)),
\end{aligned} \quad (6)
$$

where $\{u_{i1}^s, u_{i2}^s, \cdots, u_{ik}^s\}$ and $\{u_{i1}^t, u_{i2}^t, \cdots, u_{ik}^t\}$ denote the top k interests which are most similar to the user's encoded clicking sequences in the source domain and target domain respectively. In this manner, we could train the cluster prototypes end-to-end, requiring no alternating steps.

Instance-to-cluster Contrastive Loss Since users' behavior patterns are consistent with their interests, it is necessary to perform contrastive learning on the encoded clicking sequences and interest clusters. In our method, for the purpose of fully exploiting the interest invariance and transferring the knowledge efficiently between domains, we perform CL-I2C across domains. Each instance forms a positive pair with its corresponding MV cluster, and negative pairs with all new clusters. Here, the instance and the clusters belong to different domains. Therefore, comparable to instance-to-instance contrastive loss, we can compute instance-to-cluster contrastive loss in one interest subspace as:

$$
\begin{aligned}
e_{i2c}^{s2t} &= -\sum_{k=1}^n log \frac{exp(sim(p_k^s, q_k^t)/\tau)}{exp(sim(p_k^s, q_k^t)/\tau) + \sum_{i=1}^K exp(sim(p_k^s, u_i^t)/\tau)}, \\
e_{i2c}^{t2s} &= -\sum_{k=1}^n log \frac{exp(sim(p_k^t, q_k^s)/\tau)}{exp(sim(p_k^t, q_k^s)/\tau) + \sum_{i=1}^K exp(sim(p_k^t, u_i^s)/\tau)}, \\
e_{i2c} &= e_{i2c}^{s2t} + e_{i2c}^{t2s}.
\end{aligned} \quad (7)
$$

In addition to that, users' interests in different subspaces are multi-granular. For instance, a girl may have a finer granularity in the fashion interest space than in the digital interest space. Therefore, in order to take into account users' interests in different subspaces, we also perform multi-granularity CL-I2C. For interests in different subspaces, we can obtain the instance-to-cluster contrastive loss $\mathcal{E}_{i2c} = \{e_{i2c}^1, e_{i2c}^2, \cdots, e_{i2c}^l\}$ where l is the number of different subspaces which is a hyper-parameter. Finally, we can get the instance-to-cluster contrastive loss as $\mathcal{L}_{i2c} = \sum_{i=1}^l e_{i2c}^i$.

Two-stage Training Paradigm

In the first stage, we train the model in a self-supervised learning manner using the contrastive loss $\mathcal{L}_{ssl} = \mathcal{L}_{i2i} + \mathcal{L}_{i2c}$. Our task is to predict whether the user clicks a specific item in the target domain. Thus, our goal is to extract the user's interests based on the user's historical behaviors in the two domains and explore the connections between the interests and the specific item in the target domain. However,

the user's interactions in the target domain are often much more sparse than in the source domain. Nonetheless, we can effectively address such problem with the first stage of self-supervised learning of SITN. In the first stage, we learn and transfer the user's interest invariance between domains via contrastive learning, which allows us to accurately capture the user's preferences in the subsequent training. Instead of training from scratch, in the second stage, we initialize sequence encoders with the weights obtained in the previous stage.

To be specific, in the second stage, we first obtain the user's behavioral sequence representations $\mathcal{Z}^s = \{z_1^s, z_2^s, \cdots, z_{|\mathcal{Z}^s|}^s\}$ and $\mathcal{Z}^t = \{z_1^t, z_2^t, \cdots, z_{|\mathcal{Z}^t|}^t\}$ encoded by the sequence encoders which are pre-trained in the first stage from the source domain and target domain respectively, where $|\mathcal{Z}^s|$ and $|\mathcal{Z}^t|$ denote the length of each sequence. Then, we can obtain the user representation u_f^s in the source domain by feeding the target item v and $\mathcal{Z}^s$ into a target attention block to capture the connection between the target item and user behavior sequence, where Query is v, and key and value are the same, that is $\mathcal{Z}^s$. Similarly, we can get the user representation u_f^t in the target domain. Next, we concatenate u_f^s, u_f^t and v, and feed them into a MLP block to predict the probability $\hat{y}$ whether the user would like to click the item in the target domain as follows:

$$
\hat{y} = MLP(Concat(u_f^s, u_f^t, v)), \quad (8)
$$

where MLP denotes multi-layer perceptron. Finally, the cross-entropy loss can be computed as:

$$
\mathcal{L}(y, \hat{y}) = -(ylog\hat{y} + (1-y)log(1-\hat{y})), \quad (9)
$$

where $y \in \{0, 1\}$ is the ground-truth label indicating whether the user clicks the item in the target domain or not.

Experiments

Datasets

To verify the effectiveness of the proposed method, we conduct extensive experiments on two datasets. The statistics of the datsets are shown in Table 1.

Industrial Dataset We collect a large-scale dataset from one of the world's leading e-commerce corporations. In the dataset, the source domain is a e-commerce platform where users can purchase products while the target domain is a community where users can post their feelings about the products they purchased and share their lives. We collect the logs on the corporation from seven consecutive days during May 2022. The dataset contains user ID, item ID, item title, item category and clicking sequences from the source domain and target domain. Additionally, we only reserve those logs whose clicking sequence lengths in the two domains are both larger than 10 and truncate the clicking sequence length to 100.

Amazon Dataset The Amazon review dataset (Ni, Li, and McAuley 2019) is a widely used public dataset for recommender systems. The dataset contains users' review data and

products' metadata from many domains (Books, Movies and TV, Digital music, Electronics, etc.). Here, we take the book domain as the source domain and the movie & TV domain as the target domain. For each domain, we consider the ratings of 4 or more as clicking behaviors and form the clicking sequences by sorting the logs in chronological order. In addition to the clicking sequences, the features used include user ID, item ID, item title, item brand, item main and secondary category. Moreover, we only keep those logs with book clicking sequence length of 5 or more and truncate the clicking sequence length to 100.

Dataset	Industrial	Amazon
# Shared Users	23,706,610	87,896
# Items(Source)	54,829,040	673,826
# Items(Target)	13,149,185	100,164
# Instances	445,821,389	1,290,358
Avg. # clicked(Source)	94	39
Avg. # clicked(Target)	60	36

Table 1: Statistics of two utilized datasets. (avg. - average)

Overall Performance

We choose six single-domain methods and four cross-domain methods to compare with ours as follows:

Single-Domain Methods

- **FM** (Rendle 2010): This method considers both first-order and second-order feature interactions.

- **Wide&Deep** (Cheng et al. 2016): This method uses both logistic regression and deep neural networks (DNN).

- **DeepFM**(Guo et al. 2017): The method combines FM and DNN.

- **DIN**(Zhou et al. 2018): This work uses the attention mechanism in sequential recommendation.

- **DIEN** (Zhou et al. 2019): This work extends DIN with interest extractor layer and interest evolution layer.

- **CLSR** (Zheng et al. 2022): This work models the long and short term sequences of users by using the idea of contrastive learning.

Cross-Domain Methods

- **MV-DNN** (Elkahky, Song, and He 2015): This work is based on Deep Structured Semantic Model (Huang et al. 2013) and combines the features from multiple domains.

- **CoNet** (Hu, Zhang, and Yang 2018): This work establishes cross-connections between base networks to transfer knowledge across domains.

- **MiNet**(Ouyang et al. 2020): The model uses the attention mechanism to extract and transfer the knowledge of interests between domains.

- **SEMI**(Lei et al. 2021): This work employs contrastive learning to encourage the representations of sequences in the source domain and target domain to be similar.

	Model	Industrial		Amazon	
		AUC	Logloss	AUC	Logloss
Single-domain	FM	0.6501	0.4486	0.6215	0.4624
	Wide&Deep	0.6939	0.3701	0.7154	0.3411
	DeepFM	0.7040	0.3672	0.7201	0.3358
	DIN	0.7173	0.3408	0.7370	0.3183
	DIEN	0.7185	0.3392	0.7397	0.3062
	CLSR	0.6935	0.3740	0.6752	0.4061
Cross-domain	MV-DNN	0.7169	0.3391	0.7293	0.3285
	CoNet	0.7188	0.3382	0.7275	0.3278
	MiNet	0.7218	0.3313	0.7305	0.3260
	SEMI	0.7304	0.3276	0.7439	0.3036
	SITN	**0.7351***	**0.3207***	**0.7525***	**0.2857***

Table 2: Test AUC and Logloss. * indicates the statistical significance for $p <= 0.01$ compared with the best baseline method based on the paired t-test.

We perform extensive experiments on both datasets and use the Area Under the ROC Curve (AUC) and the value of Eq. 9 as the evaluation metrics for the methods. According to Table 2, the following three conclusions can be drawn:

Our proposed method achieves the best performance on both datasets. We can notice that SITN outperforms all of the baselines on both datasets. The next best method is SEMI, which is a method similar to ours. SITN exceeds it by 0.64% and 1.16% on the industrial dataset and amazon dataset respectively in terms of AUC. The results indicate the effectiveness of SITN, thanks to the fact that it fully learns and efficiently transfers the knowledge of interests between domains.

The cross-domain methods perform better overall than the single-domain methods on both datasets. Based on the experimental results, the cross-domain methods are significantly better than the single-domain methods on the industrial dataset, and are also better overall on the amazon dataset except for DIN and DIEN. We argue that the cross-domain methods can effectively transfer some valid knowledge between domains, especially in real complicated industrial-level scenarios. Among these cross-domain methods, our proposed method is still the most efficient one.

Contrastive learning is beneficial for cross-domain recommendation. We can observe that our method and SEMI give the best results on the two datasets, which are both based on contrastive learning. The experimental results further demonstrate that contrastive learning does play a crucial role in cross-domain recommendation. We assume that is due to the fact that contrastive learning can extract and transfer some valid knowledge between domains, which is useful for capturing users' preferences.

Ablation Study

We conduct two ablation studies to further explore the effect of instance-to-cluster contrastive learning and the effects of multi-granularity and multi-view modules.

	Industrial		Amazon	
Method	AUC	Logloss	AUC	Logloss
w/o $\mathcal{L}_{i2c}$&$\mathcal{L}_{i2i}$	0.7203	0.3325	0.7328	0.3242
w/o $\mathcal{L}_{i2c}$	0.7295	0.3304	0.7456	0.3020
w/o $\mathcal{L}_{i2i}$	0.7324	0.3259	0.7489	0.2959
SITN	**0.7351**	**0.3207**	**0.7525**	**0.2857**

Table 3: Effect of instance-to-cluster contrastive learning.

Effect of Instance-to-cluster Contrastive Learning   Motivated by the desire to exploit the interest invariance between domains, we introduce instance-to-cluster contrastive learning. In this module, we investigate whether the design of CL-I2C fully utilizes of its role and jointly enhance robustness of the model with CL-I2I. To be specific, we remove CL-I2C, CL-I2I and both of them from SITN respectively and conduct extensive experiments on the two datasets. As shown in Table 3, we can find that removing either CL-I2I or CL-I2C will result in performance degradation. And if they are both removed, performance degradation is even greater. We can conclude that utilizing solely CL-I2I is somewhat effective in capturing user preferences and enhancing the model's performance, but not as effective as using only CL-I2C. Further, we can obtain the best performance by exploiting both CL-I2I and CL-I2C. We argue that, with the full interaction and co-action of CL-I2I and CL-I2C, which allows the interests between domains to be learned and transferred efficiently, the model will get higher performance.

	Industrial		Amazon	
Method	AUC	Logloss	AUC	Logloss
w/o MG&MV	0.7310	0.3343	0.7462	0.3013
w/o MG	0.7335	0.3236	0.7493	0.2953
w/o MV	0.7327	0.3305	0.7475	0.2965
SITN	**0.7351**	**0.3207**	**0.7525**	**0.2857**

Table 4: Effects of multi-granularity and multi-view modules.

Effects of Multi-granularity and Multi-view Modules The user preference is typically closely tied to multiple of the most similar interest clusters rather than being solely defined by one. Therefore, we design the multi-view module (MV) to be utilized on CL-I2C, which takes into account the multiple interests of the user. In addition to that, the user's interests in different spaces are multi-granular. Thus, we introduce the multi-granularity module (MG) on CL-I2C, which focuses on the user's both fine- and coarse-grained interests. To study the effects of multi-granularity and multi-view modules, we conduct extensive experiments on the datasets. According to Table 4, we can see that both MV and MG have a positive effect on the model. And if they are combined together, the model performance will be improved even more.

(a) 32 interest clusters. (b) 64 interest clusters. (c) 128 interest clusters. (d) 256 interest clusters.
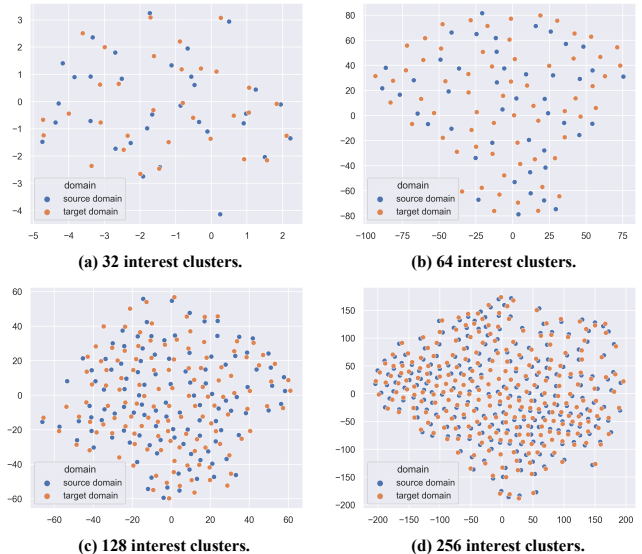
Figure 3: Visualization of the interest clusters in the source domain and the target domain. For (a), (b), (c) and (d), the numbers of clusters in both domains are the same, 32, 64, 128 and 256, respectively.

Case Study

We study whether SITN can learn and transfer the interest invariance between domains. We initialize four interest spaces with different granularity and conduct experiments on the industrial dataset. These four interest spaces have the same numbers of clusters in the source domain and the target domain, 32, 64, 128 and 256, respectively. Then, we visualize the interest clusters in the both domains via t-SNE (Van der Maaten and Hinton 2008). As shown in Figure 3, we can find that the interest clusters are evenly distributed, and almost each interest cluster in the source domain has an interest cluster in the target domain that is very close to it. Moreover, the distributions of the interest clusters in the two domains are also relatively close. It reflects the fact that SITN does learn the invariant knowledge of interests between domains, and thus improves the model performance.

Conclusion

In this paper, we identify the significance of the interest invariance between domains in cross-domain recommendation. We propose a novel method called Self-supervised Interest Transfer Network (SITN), which employs instance-to-instance and instance-to-cluster contrastive learning simultaneously. Moreover, SITN also considers the user's multi-granularity and multi-view interests. With this paradigm, SITN can efficiently learn and transfer the interest invariance between domains. Extensive offline experiments on a large-scale industrial dataset and a public dataset indicate the effectiveness of the proposed method. And the online A/B testing results further demonstrate its practical value.

Acknowledgments

References

Chen, Q.; Zhao, H.; Li, W.; Huang, P.; and Ou, W. 2019. Behavior sequence transformer for e-commerce recommendation in alibaba. In *Proceedings of the 1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data*, 1–4.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.

Chen, Y.; Liu, Z.; Li, J.; McAuley, J.; and Xiong, C. 2022. Intent Contrastive Learning for Sequential Recommendation. In *Proceedings of the ACM Web Conference 2022*, 2172–2182.

Cheng, H.-T.; Koc, L.; Harmsen, J.; Shaked, T.; Chandra, T.; Aradhye, H.; Anderson, G.; Corrado, G.; Chai, W.; Ispir, M.; et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*, 7–10.

de Souza Pereira Moreira, G.; Rabhi, S.; Lee, J. M.; Ak, R.; and Oldridge, E. 2021. Transformers4Rec: Bridging the Gap between NLP and Sequential/Session-Based Recommendation. In *Fifteenth ACM Conference on Recommender Systems*, 143–153.

Elkahky, A. M.; Song, Y.; and He, X. 2015. A multi-view deep learning approach for cross domain user modeling in recommendation systems. In *Proceedings of the 24th international conference on world wide web*, 278–288.

Gao, T.; Yao, X.; and Chen, D. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Guo, H.; Tang, R.; Ye, Y.; Li, Z.; and He, X. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. *arXiv preprint arXiv:1703.04247*.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9729–9738.

Hu, G.; Zhang, Y.; and Yang, Q. 2018. Conet: Collaborative cross networks for cross-domain recommendation. In *Proceedings of the 27th ACM international conference on information and knowledge management*, 667–676.

Huang, P.-S.; He, X.; Gao, J.; Deng, L.; Acero, A.; and Heck, L. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, 2333–2338.

Lei, C.; Liu, Y.; Zhang, L.; Wang, G.; Tang, H.; Li, H.; and Miao, C. 2021. Semi: a sequential multi-modal information transfer network for e-commerce micro-video recommendations. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 3161–3171.

Li, J.; Zhou, P.; Xiong, C.; and Hoi, S. C. 2020. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*.

Li, P.; and Tuzhilin, A. 2020. Ddtcdr: Deep dual transfer cross domain recommendation. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, 331–339.

Lin, Z.; Tian, C.; Hou, Y.; and Zhao, W. X. 2022. Improving Graph Collaborative Filtering with Neighborhood-enriched Contrastive Learning. In *Proceedings of the ACM Web Conference 2022*, 2320–2329.

Loni, B.; Shi, Y.; Larson, M.; and Hanjalic, A. 2014. Cross-domain collaborative filtering with factorization machines. In *European conference on information retrieval*, 656–661. Springer.

Ni, J.; Li, J.; and McAuley, J. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, 188–197.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Ouyang, W.; Zhang, X.; Zhao, L.; Luo, J.; Zhang, Y.; Zou, H.; Liu, Z.; and Du, Y. 2020. Minet: Mixed interest network for cross-domain click-through rate prediction. In *Proceedings of the 29th ACM international conference on information & knowledge management*, 2669–2676.

Rendle, S. 2010. Factorization machines. In *2010 IEEE International conference on data mining*, 995–1000. IEEE.

Shen, Y.; Shen, Z.; Wang, M.; Qin, J.; Torr, P.; and Shao, L. 2021. You never cluster alone. *Advances in Neural Information Processing Systems*, 34: 27734–27746.

Singh, A. P.; and Gordon, G. J. 2008. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 650–658.

Sun, F.-Y.; Hoffmann, J.; Verma, V.; and Tang, J. 2019. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. *arXiv preprint arXiv:1908.01000*.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Velickovic, P.; Fedus, W.; Hamilton, W. L.; Liò, P.; Bengio, Y.; and Hjelm, R. D. 2019. Deep Graph Infomax. *ICLR (Poster)*, 2(3): 4.

Wu, J.; Wang, X.; Feng, F.; He, X.; Chen, L.; Lian, J.; and Xie, X. 2021. Self-supervised graph learning for recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, 726–735.

Xie, R.; Liu, Q.; Wang, L.; Liu, S.; Zhang, B.; and Lin, L. 2022. Contrastive cross-domain recommendation in matching. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 4226–4236.

Yao, T.; Yi, X.; Cheng, D. Z.; Yu, F.; Chen, T.; Menon, A.; Hong, L.; Chi, E. H.; Tjoa, S.; Kang, J.; et al. 2021. Self-supervised learning for large-scale item recommendations. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 4321–4330.

You, Y.; Chen, T.; Sui, Y.; Chen, T.; Wang, Z.; and Shen, Y. 2020. Graph contrastive learning with augmentations. *Advances in Neural Information Processing Systems*, 33: 5812–5823.

Zheng, Y.; Gao, C.; Chang, J.; Niu, Y.; Song, Y.; Jin, D.; and Li, Y. 2022. Disentangling Long and Short-Term Interests for Recommendation. In *Proceedings of the ACM Web Conference 2022*, 2256–2267.

Zhou, G.; Mou, N.; Fan, Y.; Pi, Q.; Bian, W.; Zhou, C.; Zhu, X.; and Gai, K. 2019. Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 5941–5948.

Zhou, G.; Zhu, X.; Song, C.; Fan, Y.; Zhu, H.; Ma, X.; Yan, Y.; Jin, J.; Li, H.; and Gai, K. 2018. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 1059–1068.

Zhou, K.; Wang, H.; Zhao, W. X.; Zhu, Y.; Wang, S.; Zhang, F.; Wang, Z.; and Wen, J.-R. 2020. S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 1893–1902.

Zhu, Y.; Tang, Z.; Liu, Y.; Zhuang, F.; Xie, R.; Zhang, X.; Lin, L.; and He, Q. 2022. Personalized transfer of user preferences for cross-domain recommendation. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, 1507–1515.

Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; and Wang, L. 2021. Graph contrastive learning with adaptive augmentation. In *Proceedings of the Web Conference 2021*, 2069–2080.