Differentially Private Knowledge Graph Neural Network for Recommeder System

Ce
Ningbo University
**** (lbs@nbu.edu.cn)
Ningbo University

Research Article

Keywords: recommender system, knowledge graph, differectial privacy, graph convolutional network, neural network

Posted Date: August 22nd, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3270487/v1

Additional Declarations: No competing interests reported.

Differentially Private Knowledge Graph Neural Network for Recommeder System

Ce Hu[1] and Baisong Liu[1*]

[1*]Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo, 315000, China.

*Corresponding author(s). E-mail(s): lbs@nbu.edu.cn;
Contributing authors: 2111082350@nbu.edu.cn;

Abstract

Knowledge Graphs (KGs) have been instrumental in mitigating the challenges of cold start and data scarcity in recommender systems, serving as a kind of auxiliary information. However, recent studies reveal that KGs, while beneficial, may also inadvertently furnish malicious attackers with more comprehensive background knowledge, escalating the risk of user privacy data breaches within these recommender systems. Current frameworks, which amalgamate KGs with privacy-preserving techniques, predominantly concentrate on enhancing privacy preservation, often at the expense of recommendation efficacy. In this research, we introduce a framework, named **D**ifferential **P**rivacy **K**nowledge graph **N**eural network for **Rec**ommender systems (**DKPNRec**), designed to harmonize the trade-off between differential privacy noise and recommendation performance. Firstly, we formulate an integrated KG convolutional neural network adept at mining interrelationships between KG entities and item correlations effectively. Secondly, we incorporate a differential privacy technique to impose uniform Laplacian noise on feature embeddings and obfuscate genuine user-item interactions through pseudo-interaction item sampling, thereby augmenting user data security. Lastly, we evaluate our model using three distinct datasets-movies, music, and books. Empirical results from three benchmarks indicate that DPKNRec markedly supersedes contemporary benchmarks.

Keywords: recommender system, knowledge graph, differectial privacy, graph convolutional network, neural network

1 Introduction

The exponential rise in the volume of information on the Internet presents both advantages and challenges [1]. On the one hand, these huge data have greatly enriched the lives of users, but on the other hand, the overly redundant data has also created difficulties for users to choose. This phenomenon is known as the information overload problem [2]. To counter this issue, researchers proposed the concept of recommender system (RS) [3]. The RS can analyze each user's historical behavior data to discover the user's interests, model the user's interest preferences, and match them with appropriate content to generate a recommendation list.

Presently, the majority of RS methodologies leverage deep learning techniques [4]. However, in practical recommendation process, this technique is often limited by data sparsity [5] and cold start problem [6]. To address the problem of declining recommendation quality, researchers often add additional auxiliary information to recommendation algorithms to improve recommendation performance. Given that recommendation information encapsulates numerous knowledge entities, the Knowledge Graph (KG) [7] is frequently employed to uncover underlying knowledge-level connections between items. KGs, laden with semantic data, have promising applications in augmenting recommendation accuracy and interpretability by accurately illustrating complex relational semantics and synergistic signals [8].

Contrasted with methods void of KG, integrating KG into recommendation offers three distinct benefits [9]. Firstly, the rich semantic associations between entities in the KG assist in unearthing potential connections, thereby enhancing recommendation accuracy. Secondly, the diverse relationships within the KG contribute to a reasonable extension of user interests, enriching recommendation diversity. Lastly, the KG bridges the gap between products users prefer in their interaction history and items suggested by the RS, thereby offering interpretability to the recommendations.

Graph Neural Networks (GNN) [10] are increasingly being adopted in knowledge-aware RSs. The primary methodology revolves around creating information aggregation schemes that adeptly propagate collaborative signals with multi-hop neighbors. For example, CKAN [11] employed different neighborhood aggregation schemes on the user-item graph and KG, and utilized user and item embeddings to make predictions. Wang [12] proposed an attentive neighborhood aggregation mechanism that combines KG with the user-item bipartite graph to generate user and item representations for recommendation. Subsequently, KGIN [13] was introduced to model each user intent as an attentive combination of KG relations, recursively integrating relation sequences of long-range connectivity.

The RS that integrates the KG can effectively improve the accuracy of the recommendation results through the association of more auxiliary information, but the introduced association will also provide more background knowledge for the attacker [14]. It will also face more serious privacy security issues. This is because when personalized recommendations are made for users in the RS, the historical user behavior data used for analysis can reflect users' interests and contain a large amount of sensitive information. Once the data is obtained by the attacker, it will cause unpredictable consequences to the users. As shown in Figure 1, for the server in the RS, the relation embedding r' can be obtained by head entity embedding h, rail entity embedding

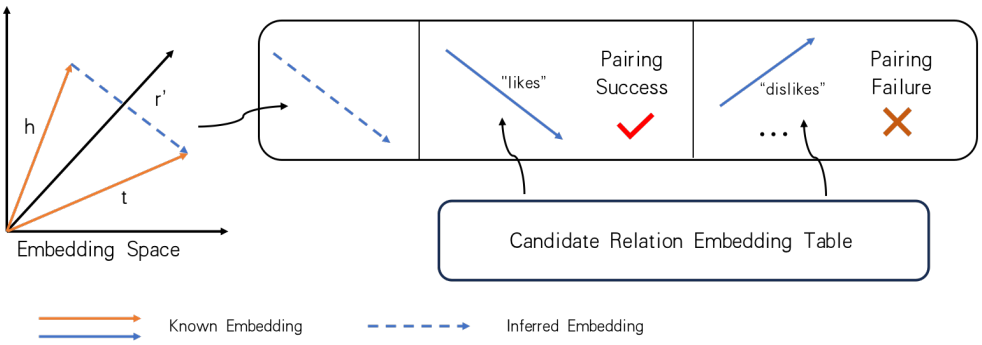


Fig. 1 Data Privacy leakage problem in knowledge graph based recommender system.

t, and the specific knowledge graph embedding function. Once the malicious attacker can access the name of entities and relations, the data of any target triples will be exposed. The malicious attacker will be able to determine whether the target head entity 'likes' or 'dislikes' the tail entity.

To address this concern, Differential Privacy (DP) [15] is proposed as an effective privacy protection method. It can effectively resist various attacks even when the attacker has strong background knowledge. Many studies propose to apply DP to collaborative filtering recommendation algorithm for privacy protection. For example, Boutet et al. [16] proposed a matrix factorization algorithm based on DP, which preserves privacy by adding noise satisfying different privacy budgets to user rating data and stochastic gradient descent. Yet, this method is only suitable for a single matrix factorization recommendation scenario. Moreover, Yu et al. [17] suggested a privacy-preserving multi-task recommendation method, adding noise to the gradient descent process and the correlation parameters of the multi-task model to bolster the efficiency and security of the RS. Nonetheless, this approach necessitates alternate learning of the KG feature learning module and the recommendation module, thereby significantly increasing the time overhead. Additionally, continuously adding Laplacian noise to the gradient during each training iteration leads to accumulated noise, impacting the final model's accuracy and unsuitability for single-task recommendation processes.

Therefore, this paper proposes a recommendation framework amalgamating KG and DP to address the inability of existing methodologies to effectively balance Laplacian noise and RS efficiency. This framework aims to enhance the privacy-preserving performance of RSs while preserving the high accuracy and interpretability of recommendation results. Notably, this paper introduces an innovative privacy protection mechanism, which simultaneously safeguards the information in the KG with DP and protects user privacy.

Experimental results indicate that the proposed recommendation method can significantly improve recommendation results' accuracy and diversity while safeguarding user privacy and maintaining good interpretability. Simultaneously, the method effectively resists various privacy attacks, including those based on statistical inference and background knowledge. Therefore, the KG and DP combined recommendation method proposed in this paper holds promising application prospects and practical value and can aid in addressing privacy protection and recommendation accuracy issues in RSs.

The contributions in this paper are summarized as follows:

1. We architect a privacy-preserving mechanism for RSs incorporating knowledge graphs based on differential privacy and pseudo-interaction item sampling. The privacy-preserving mechanism simultaneously balances recommendation accuracy and privacy security. It is worth noting that, unlike previous knowledge graph recommendation frameworks that use alternate training, this paper adopts a joint training model that effectively reduces the Laplace noise that is constantly stacked in the user's feature vector, ensures the accuracy of the recommendation results, and also protects the user's security.
2. In this paper, a novel knowledge graph convolutional network is used in RSs to automatically capture higher-order structural and semantic information in the knowledge graph. It automatically aggregates and merges biased domain information when computing the representation of a given entity in the knowledge graph to further improve recommendation accuracy.
3. We conducted comprehensive experiments on three datasets to validate the efficiency of DPKNRec. The experimental results show that DPKNRec outperforms state-of-the-art baseline models.

The remainder of this paper is organized as follows: The related works are introduced in Sect. 2. In Sect.3, we give the preliminary knowledge and problem definition. The implementation of proposed framework is specified in Sect. 4. The sufficient experiments are in Sect. 5. At last, the conclusions and the future works are discussed in Sect. 6.

2 RELATED WORK

Fusing KGs into RSs can effectively improve recommendation accuracy. However, the path to personalized recommendation often entails a trade-off between precision and privacy. Recurrently, user interaction history datasets, prediction models, and recommendation results exploited by RSs may inadvertently leak user privacy [18]. Moreover, such leakage can be exacerbated in scenarios necessitating the exchange of these datasets between users and servers. The misuse of this data by recommendation service providers or attackers could lead to substantial breaches of user privacy. Although the integration of a knowledge graph with RSs can augment model accuracy through auxiliary information, it simultaneously exposes more background knowledge to potential attackers, thereby raising further privacy concerns.

There are several broad approaches to privacy preservation in RSs: anonymization [19], adversarial learning [20], federated learning [21], and differential privacy [16].

2.1 Anonymization

Anonymization, the most straightforward privacy-preserving method, protects user privacy by obfuscating data origins. For instance, the Anonymous Random Wandering (ARW) algorithm [19] employs wandering sampling on user social network graph models. The end-users can anonymously decide to provide data or not, rendering the aggregated data incapable of accurately identifying corresponding users. However,

anonymization only provides limited privacy protection and is typically reserved for smaller, localized modules due to potential data leakage risks. In addition, although anonymization methods can prevent servers or attackers from knowing the user from whom the information originated, there is still a risk of leakage of the information itself. Therefore, anonymization-based methods have limited ability to protect the privacy of large-scale knowledge graph-based RSs.

2.2 Adversarial Learning

Moreover, the reconstruction of user representations by attackers can inadvertently lead to privacy breaches. To counter this, adversarial learning [19] introduces adversarial perturbations to input data, significantly reducing the accuracy of reverse predictions. Most existing adversarial learning research in privacy-preserving RSs focuses on Generative Adversarial Networks (GAN). Here, a generator creates data containing perturbations to deceive a discriminator, which determines the data's authenticity. This iterative adversarial training enhances both the recommendation effectiveness and privacy protection capabilities of the final model. However, adversarial learning is suitable for scenarios where the RS currently has known attack methods from malicious attackers. This privacy-preserving approach requires training for a specific attack model and has weak generalization capabilities.

2.3 Federated Learning

Federated learning [21] is a special type of distributed learning designed to build federated models based on distributed datasets. Federated learning allows each participant to train a local model individually, and then uploads the model parameters to a central server for aggregation. The scheme ensures that all participants' data are always kept locally. In federated learning, users' private data are only used to train their own local models without uploading to the server, thus enabling the protection of users' private data. Currently, many researchers in different fields have proposed approaches in federated settings.In most federated frameworks, there is a high amount of communication between the user and the server. The gradient information uploaded by the model can still be attacked. Existing research on recommendation algorithms based on federation learning is often combined with differential privacy algorithms. Because when each user uploads a local gradient to the server, this gradient information usually contains user privacy-sensitive information, and then differential privacy algorithms can be used to protect the gradient information.

2.4 Differential Privacy

Differential Privacy (DP) [16] provides privacy protection through data perturbation. This method has a rigorous mathematical underpinning and the most robust assumptions about background knowledge, making it a prevalent choice for contemporary RSs. Differential attack is a special and efficient method for attacking RSs. he principle behind differential attacks lies in inferring a user's private information based on differential values between two sets of information obtained from separate attack queries. DP thwarts these attacks by ensuring that the results of two adjacent datasets queries

are nearly indistinguishable, preventing user information inference based on query differences. Approaches like Uni-FedRec [22] and Associated FedNCF [23] incorporate DP for user interest representation learning and gradient data protection in a federated learning framework, respectively. Meng et al. [24] proposed a privacy-preserving social recommendation approach by adding noise of varying strengths to sensitive and non-sensitive training datasets. Yu et al.[17] introduced DP into a knowledge graph-enhanced multi-task recommendation model by adding Laplace noise during the gradient descent process and to relevant model training parameters. However, this method necessitates alternate learning of the knowledge graph feature learning module and the recommendation module, leading to accumulated noise and diminished accuracy of the final model. With rigorous mathematical definitions and comprehensive background knowledge assumptions, DP has become the mainstream privacy protection method in current RSs.

In light of these issues, this paper proposes a novel privacy-preserving method, merging Knowledge Graphs (KG) and Differential Privacy (DP), for recommender systems.

3 Preliminary Knowledge And Problem Definition

3.1 Differential Privacy

Differential attacks represent a privacy-impinging technique based on group encryption. This approach infers a user's private information by contrasting the differential values from two separate attack queries. For instance, suppose a hospital has 100 patients, an attacker discovers that 10 are diabetic through one attack and finds out whether 99 patients have diabetes through another attack. The discrepancy between these attacks can reveal the health status of the 100th patient.

DP techniques were conceived to defend against such attacks. The objective of DP is to ensure that the query results from two adjacent datasets are nearly identical, thus preventing the inference of user privacy information based on query differences. An adjacent dataset is defined as two datasets differing by only a single data point. Existing DP implementations can be classified into centralized and localized DP based on the sequence of data encryption.

Centralized DP requires user data to be gathered at a central data repository where a uniform DP algorithm is applied before data public release. The formal definition of a DP algorithm is as follows:

Definition 1. (ϵ-Differential Privacy). Two databases differing by a single data record are referred to as 'neighboring databases', symbolized $D \sim D^{'}$. Any random mechanism preserves (ϵ, δ) - DP (differential privacy) if for any neighboring databases D and $D^{'}$ as well as for all sets of possible outputs S :

$$P\{M(D) \in S\} \leq e^{\epsilon} P\{M(D') \in S\} + \delta. \tag{1}$$

Here, ϵ represents the privacy budget, quantifying the privacy assurance of differentially private mechanisms. A smaller privacy budget signifies better privacy protection.

δ is an exceptionally small number, often viewed as the upper limit on the probability of a catastrophic privacy breach.

Definition 2. (Global Sensitivity). For a given database D and a query q, the global sensitivity denoted by ΔD is

$$\Delta D = \underset{D \sim D'}{max} \left| q\left(D\right) - q\left(D'\right) \right| \tag{2}$$

While any mechanism satisfying Definition 1 can be considered a differentially private mechanism, four mechanisms are particularly prevalent: the Laplacian mechanism, the Gaussian mechanism, the exponential mechanism, and the randomized response mechanism.

The Laplacian and Gaussian mechanisms specifically address numerical data. The Laplacian mechanism, in particular, distorts its output by adding noise from a Laplacian distribution with a location parameter of 0 and a scale parameter of $\Delta D/\epsilon$. Formally:

Definition 3. (Laplacian Mechanism). For a query q and a given database D, and a privacy budget ϵ. If the output result of algorithm A satisfies the following formula, it is said that algorithm a satisfies ϵ-differential privacy.

$$A\left(D\right) = q\left(D\right) + Lap\left(\Delta f/\epsilon\right) \tag{3}$$

Here, $Lap\left(\Delta f/\epsilon\right)$ represents added noise from Laplacian distribution. Notably, δ is 0 for the Laplacian mechanism.

3.2 Knowledge Graph

Knowledge graph is essentially a heterogeneous information network, commonly used to represent complex relations between entities. The nodes in the network represent entities, while the edges between the nodes represent the relationships between the entities. The knowledge graph G can be formally represented as a set of triples (h, r, t), where h represents the head entity, t represents the tail entity, and r represents the relationship between two entities.

3.3 Problem Definition

This paper aims to train a recommendation model to predict whether users will be interested in items that they have not interoperated with before according to the user-item interaction matrix Y and knowledge graph G. Then we can recommend items for users based on the results. In this paper, we give a user set containing m users $U = \{u_1, u_2, ..., u_m\}$, an item set containing n items $V = \{v_1, v_2, ..., v_n\}$, and an interaction matrix $Y \in R_{m \times n}$ for recording the interaction between users and items.

Where, if the element in the matrix Y is 1, namely $y_{u_i v_j} = 1$, it means that user u_i has clicked on item v_j before. And if the element in the matrix Y is empty, namely $y_{u_i v_j}$ $= 0$, it means that user $u_i$and item v_j are not related before. In addition, we also give a knowledge graph G which is expressed by triple (h, r, t), namely $(head, relation, tail)$.

For example, we predict whether user u_i is interested in item v_j . The purpose of this paper is to train a privacy-preserving prediction function $\hat{y} = F(u, v; G, Y)$ and calculate the probability of user u clicking on item v. Then make the final prediction result of this model closer to the real value y_{uv} through model training.

4 The Proposed Model

The DPKNRec model balances privacy protection of system data and prediction accuracy of recommendation outcomes. As depicted in Figure 2, the DPKNRec model comprises two modules: the recommendation module and the privacy protection module.

The recommendation module, a fusion of knowledge graph elements, incorporates abundant auxiliary information into the recommendation task. By employing a knowledge graph representation learning method, we not only retrieve the embedding information of entities but also gather comprehensive contextual information associated with the entities. Drawing inspiration from graph convolutional neural networks, a knowledge graph convolutional network is utilized for the DPKNRec to automatically apprehend higher-order structural and semantic data within the knowledge graph. This network calculates the representation of a given entity in the knowledge graph, aggregating and merging domain information with a bias. This design captures and preserves the local neighborhood structure within each entity via domain aggregation operations. Domains receive weighting via scores influenced by interconnected relationships and specific users, encapsulating the semantic data of the knowledge graph and the users' personalized interests in the relations. Additionally, the recommendation model leverages a joint training approach, thereby diminishing the adverse effects on data utility induced by theconstant stacking of Laplacian noise in alternate training.

Within the privacy-preservation module, considering that the model's uploaded gradient and embedding vectors harbor private information, we introduce a privacy-preserving model update method to safeguard user-item interaction information. This method combines local differential privacy (LDP) and a pseudo-interaction item sampling technique. Initially, the server employs a pseudo-interaction item sampling technique for obfuscation, making it impossible to discern which items the user has interacted with. Specifically, we blend randomly sampled embedding gradients of pseudo-interaction items into the embedding gradients of actual items uploaded by the client, creating random fake embedding gradients. Throughout the process, the introduced noise upholds ϵ -differential privacy. The privacy budget ϵ can be described by $\frac{2\delta}{\lambda}$; the higher the noise intensity λ, the smaller the privacy budget ϵ, leading to enhanced privacy protection but at the expense of the uploaded model gradient's accuracy.

4.1 The Recommendation Module

KGs typically undergo preprocessing through representation learning methods. This paper employs the renowned TransE method [25] for learning feature vectors of entities and relations. Most present-day knowledge graph representation learning techniques

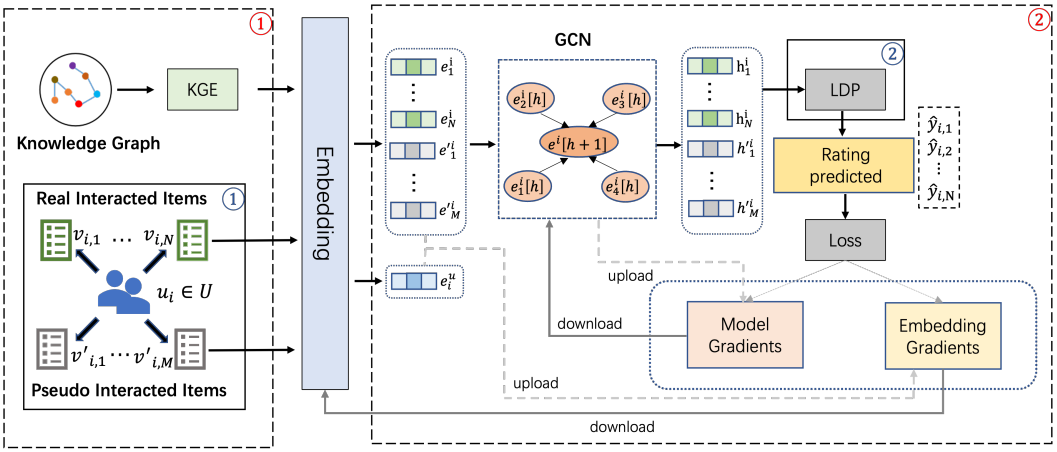


Fig. 2 The framework of the DPKNRec model. The red serial number represents step one and step two of the recommendation module, while the blue serial number represents the two steps of the privacy protection module. It is worth noting that the two privacy protection methods of the privacy protection module act on different parts of the recommendation module.

adeptly learn the structural information within the knowledge graph, thereby bolstering recommendation effectiveness. We postulate that correlations exist among multiple entities, and users may show interest in additional items that encompass related entities subsequent to engaging with a specific item. To integrate more auxiliary information, this paper extracts the contextual information-information pertaining to the one-hop neighbor nodes-from each entity within the knowledge graph in addition to amalgamating the structural information of the graph. The contextual information of an entity is formally expressed as:

$$con\,(e) = \{e_i \in e |\,(e, r, e_i) \in G \quad or \quad (e_i, r, e) \in G\} \tag{4}$$

Here, 'e' symbolizes an entity in the knowledge graph G, and 'r' indicates a relation between two entities. To exemplify the topological proximity structure of item v, we express the item's feature vector as the average of the product $v^u_{N(v)}$, which is the feature representations of all contextual entities of entity e and the relation score $\widetilde{w}^u_{r_{v,e}}$:

$$v^u_{N(v)} = \sum_{e \in N(v)} \widetilde{w}^u_{r_{v,e}} e \tag{5}$$

In this equation, $\widetilde{w}^u_{r_{v,e}}$ represents the normalized user relation score, $N\,(v)$ denotes the set of entities directly connected to item v,, and the relation score $\widetilde{w}^u_{r_{v,e}}$ can be calculated from the function g (such as the dot product) for the user and the relation:

$$w^u_r = g\,(u, r) \tag{6}$$

$$\widetilde{w}^{u}_{r_{v,e}} = \frac{exp\left(w^{u}_{r_{v,e}}\right)}{\sum_{e \in N(v)} exp\left(w_{r_{v,e}}\right)} \tag{7}$$

In these equations, $u \in U^d$ and $r \in R^d$ are the representations of user u and relation r, and 'd' denotes the dimension of representations. Generally, $w^u_r = g(u, r)$ describes the significance of relation r to user u. For instance, one user might be intrigued by the platform hosting the news items they have previously clicked on, while another user may focus more on the news type. 'e' represents the entity, and user-relation scores function as personalized filters when computing an entity's neighborhood representation as these scores aggregate the neighbors with a user-specific bias.

A crucial step of the model involves consolidating the entity representation and the contextual representation into a single vector. A summation aggregator is employed in this study to capture users' personalized preferences and the semantic information of the knowledge graph fully. It sums the two representation vectors and performs a nonlinear transformation:

$$aggregator = \sigma\left(W \cdot v^u_N(v) + b\right) \tag{8}$$

Here, W and b are the linear transformation weights and deviations, respectively, and σ stands for a nonlinear function such as ReLU.

Upon obtaining the potential feature vector of the user $e(i)$ a and the potential feature vector of the candidate item $e(c_j)$, they are concatenated and fed into an additional neural network H^1 to compute the click probability:

$$p_{u,c_j} = H^1\left(e(u), e(c_j)\right) \tag{9}$$

4.2 The Privacy Protection Module

Uploading the Graph Convolutional Network (GCN) model and item embedding gradients directly may raise privacy concerns due to two key reasons. Initially, regarding the embedding gradients, only those items that the user interacts with have non-zero gradients, updating their embedding. As such, the server can reconstruct the comprehensive history of all non-zero item-based embedding gradients for user-item interactions. Secondly, in addition to the embedding gradients, the gradients of the GCN model and rating predictor may inadvertently disclose users' private interaction histories and ratings as these gradients encode user preferences on items.

To mitigate these privacy risks, we suggest two strategies aimed at preserving user privacy during the model update process.

The first strategy employs pseudo-interaction item sampling. Specifically, we sample M items that the user u_i has not interacted with and randomly generate their gradients g^p_i using a Gaussian distribution, maintaining the same mean and covariance values as the actual item embedding gradients g^e. While there are various sampling methods available such as utilizing displayed items with no user interaction, in our experiments, we opted to randomly sample items from the full item set for simulation.

The actual embedding gradients g^e are amalgamated with the pseudo item embedding gradients g^p, and the unified gradient of the model g^m and embeddings g^e on the device is modified as:

$$g = (g^m, g^e, g^p) \tag{10}$$

The second strategy involves incorporating Laplace noise into the feature embeddings processed through the GCN model to ensure local difference privacy. Subsequently, we clip the embeddings based on their L1-norm with a threshold δ, and introduce a LDP module with zero-mean Laplacian noise to the unified embeddings to fortify user privacy protection. This is formulated as follows:

$$g_i = clip\left(g_i, \delta\right) + Laplace\left(0, \lambda\right) \tag{11}$$

Here, λ signifies the noise scale, and the privacy budget ϵ can be constrained by $\frac{2\delta}{\lambda}$. The safeguarded gradients g are uploaded to the server for aggregation.

5 Experimental design and analysis

In this section, we evaluate model on three real-world scenarios: movie, book, and music.

5.1 Datasets analysis

- MovieLens-20M[1] : A commonly used benchmark dataset for movie recommendations, it comprises roughly 20 million explicit ratings (ranging from 1 to 5) provided on the MovieLens website.
- Book-Crossing[2] : This dataset houses 1 million ratings (ranging from 0 to 10) for various books within the Book-Crossing community.
- Last.FM[3] : This dataset consists of listening information for musicians provided by a set of 2 thousand users from the Last.fm online music system.

As all three datasets contain explicit feedback, we transform them into implicit feedback, with each entry marked as '1' to indicate that the user has positively rated the item. An unwatched set marked as '0' is sampled for each user. A positive rating threshold of 4 is set for the MovieLens-20M dataset. However, due to their sparsity, no such threshold is set for the BookCrossing and Last.FM datasets. The basic statistics of the three datasets are shown in Table 1.

5.2 Evaluation metrics analysis

This paper utilizes the AUC score, recall rate, precision rate, and F1-value as evaluation metrics to gauge the Click-Through Rate (CTR) prediction performance for users on items. The subsequent text will elaborate on the definitions of the evaluation metrics employed in the experiments. All samples are categorized into four groups

[1] https://grouplens.org/datasets/movielens/
[2] http://www2.informatik.uni-freiburg.de/ cziegler/BX/
[3] https://grouplens.org/datasets/hetrec-2011/

according to their actual classification and model prediction results, namely, True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). TP represents the positive samples correctly predicted by the model where users are interested in items with which they haven't interacted. TN represents the negative samples correctly predicted by the model where users aren't interested in items with which they haven't interacted. FP and FN respectively refer to the model mistakenly predicting negative samples and positive samples that users aren't and are interested in.

The AUC value, the area under the ROC curve typically ranging between 0.5 and 1.0, mirrors the accuracy of the classifier's outcomes. An AUC closer to 1.0 indicates better model classification. As this study predominantly applies to the CTR prediction scenario, we evaluate algorithm performance using AUC. Precision rate depicts how many items the model predicts users may be interested in when users haven't interacted with items. The precision formula is as follows,

$$precision = \frac{TP}{TP + FP} \tag{12}$$

Recall rate represents how many of the items that the user is actually interested in are correctly predicted by the model. The recall formula is as follows,

$$recall = \frac{TP}{TP + FN} \tag{13}$$

F1-value is an evaluation metric that considers both the precision rate and recall rate. An F1-value of 1 signifies the best precision and recall rate, whereas a value of 0 indicates the worst recommendation effect. The F1-value formula is,

$$F1 - value = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{14}$$

5.3 Comparison model

- PER: This model treats the KG as heterogeneous information networks, extracting metapath-based features to represent user-item connectivity.
- RippleNet: A joint training method of knowledge graphs and recommender systems, RippleNet propagates users' historical preferences across the knowledge graph to capture potential preferences.

Table 1 Basic statistics of the three datasets.

	MovieLens-20M	**Book-Crossing**	**Last.FM**
users	138,159	19,676	1,872
items	16,954	20,003	3,846
interactions	13,501,622	172,576	42,346
entities	102,569	25,787	9,366
relations	32	18	60
KG triples	499,474	60,787	15,518

- CKE: This model integrates Collaborative Filtering (CF) with structural, textual, and visual knowledge in a unified recommendation framework. For this study, we implemented CKE as CF with a structural knowledge module.
- DKN: Utilizing knowledge graph representation in news recommendation, DKN is a content-based deep model suitable for click-through rate prediction, particularly in highly time-sensitive news scenarios.
- KGAT: The model investigates the utility of knowledge graphs, breaks the independent interaction assumption by associating items with their attributes, and models higher-order relationships between users and items.
- PPMTF: This is a privacy-preserving multitasking framework, which is applied in this paper to a knowledge graph-enhanced recommender system as a baseline for verifying the degree of privacy security.

5.4 Experimental design

We randomly divided the dataset into training (60%), validation (20%), and test (20%) sets. For the MovieLens-20M dataset, the hyperparameters included K=2 (neighbor sampling size), d=32 (embedding dimension), H=2 (receptive field depth), η=0.02 (learning rate), and batch size=65536 (number of samples per training). The Book-Crossing and Last.FM datasets have unique settings described subsequently. For the Book-Crossing dataset, the hyperparameters included K=8 (neighbor sampling size), d=64 (embedding dimension), H=1 (receptive field depth), η=0.0002 (learning rate), and batch size=256 (number of samples per training). For the Last.FM dataset, the hyperparameters included K=8 (neighbor sampling size), d=16 (embedding dimension), H=1 (receptive field depth), η=0.0005 (learning rate), and batch size=128 (number of samples per training).

Comparison of different methods in terms of recommended performance. Performance comparison of various methods in terms of recommendation quality is presented. This section primarily assesses the recommendation performance of different methods across diverse datasets to confirm that our method can maintain recommendation accuracy despite privacy protection.

Experimental results comparison for DPKNRec, PER, CKE, RippleNet, DKN, KGAT, PPMTF across three different datasets is displayed in Table 2 and Figures 3 - 6, from which the following conclusions are drawn:

The comparison results are shown in Figure 3. We can see that DPKNRec largely outperforms the baseline on all evaluation metrics on all three datasets. Compared to PER, CKE, RippleNet, DKN, and KGAT, DPKNRec improves performance. Because DPKNRec considers the problem of decreasing data utility caused by the continuous superposition of Laplacian noise in alternating training, and uses joint training and knowledge graph convolutional neural networks to alleviate it. DPKNRec also performs much better than the existing privacy-preserving and knowledge graph recommendation model PPMTF under the same privacy budget (ϵ of 1.0). Notably, DPKNRec also utilizes pseudo-interaction item sampling to improve the model's privacy-preserving capabilities.

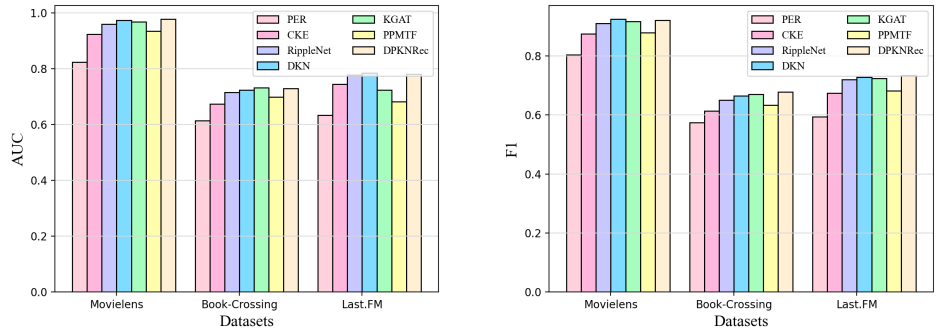
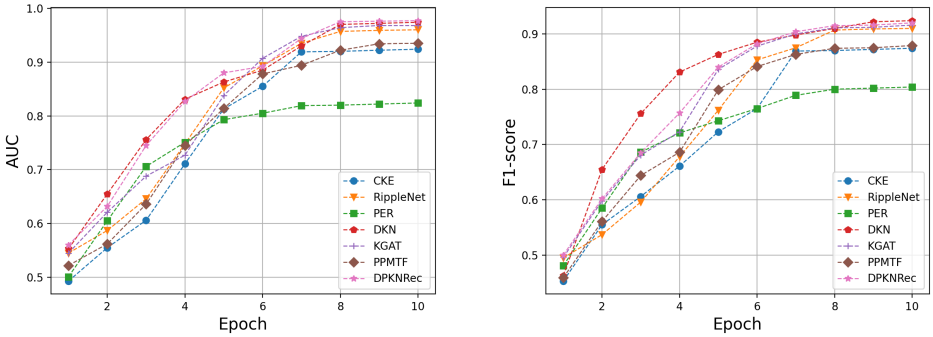


Fig. 3 Performance comparison of different methods on three datasets.

Figure 4 gives the comparison results of this paper's method DPKNRec (privacy budget ϵ of 1.0) with PER, CKE, RippleNet, DKN, KGAT, PPMTF (privacy budget ϵ of 1.0) when using Movielens-20M dataset in different epochs. As can be seen from figure 4, the model is gradually fitted by the 8th epoch of training, and the difference between the prediction accuracy, precision and AUC of this method and the recommendation method DKN without privacy-preserving fused knowledge graph is no more than 0. 01, so the performance of the two is generally comparable, although occasionally DKN has higher precision, but this is due to the data loss caused by the noise added to the results in this paper's method and the DKN method is a deep perception network for news recommendation, which is more suitable for news recommendation scenarios. Furthermore, it can be easily observed that the performance of the DPKNRec method far exceeds that of other knowledge graph recommendation methods (include PER, CKE and RippleNet). Although the perturbation of the data by the method in this paper causes some information loss, it can be seen from the

Table 2 Experimental contrast of results on three different datasets.

	MovieLens-20M		Book-Crossing		Last.FM	
Model	AUC	F1	AUC	F1	AUC	F1
PER	0.824	0.804	0.614	0.574	0.634	0.594
CKE	0.924	0.874	0.674	0.614	0.744	0.674
RippleNet	0.960	0.910	0.715	0.650	0.778	0.720
DKN	0.974	0.924	0.724	0.664	0.784	0.728
KGAT	0.968	0.916	0.720	0.670	0.780	0.724
PPMTF	0.935	0.879	0.698	0.633	0.755	0.682
DPKNRec	0.977	0.920	0.729	0.678	0.780	0.734
DPKNRec-TransD	0.975	0.917	0.722	0.670	0.773	0.731
DPKNRec-TransH	0.967	0.905	0.720	0.674	0.775	0.730
DPKNRec-TransR	0.970	0.912	0.712	0.675	0.768	0.724
DPKNRec-noGCN	0.872	0.830	0.628	0.594	0.688	0.620

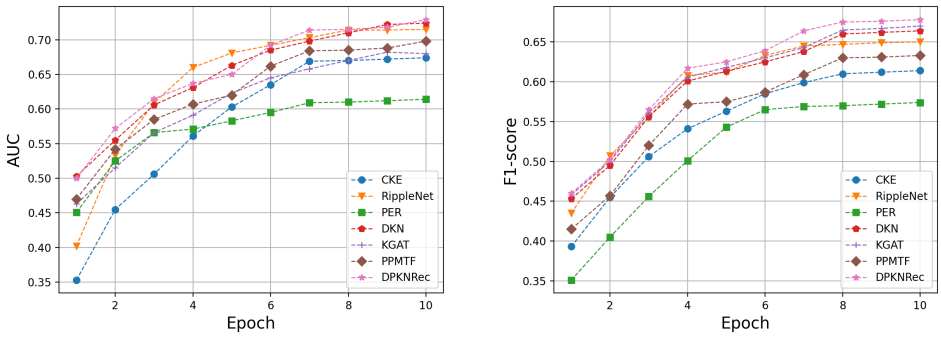


(a) (b)

Fig. 4 Performance comparison of different methods on Movielens-20M dataset.

results that the prediction accuracy of DPKNRec is still better compared to the DKN and the KGAT methods without privacy protection.

In addition, compared to the method Ripplenet, the prediction accuracy of DPKN-Rec improved more than 0. 017 when the model converged, the prediction accuracy of DPKNRec improved by 0. 034 compared to CKE, 0.153 compared to PER, 0.042 compared to PPMTF.

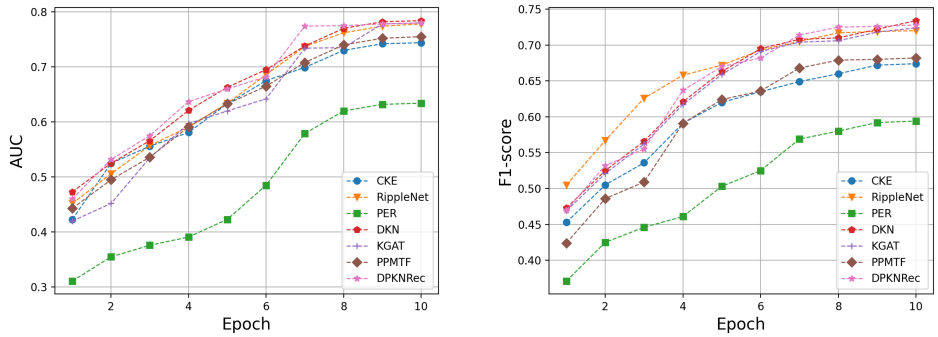


(a) (b)

Fig. 5 Performance comparison of different methods on Book-Crossing dataset.

Figures 5 and 6 show performance comparisons of our DPKNRec (privacy budget ϵ of 1.0) with PER, CKE, RippleNet, DKN, KGAT, PPMTF (privacy budget ϵ of 1.0) on the Book-Crossing and Last.FM datasets in different epochs. The overall performance trend on these datasets is similar to that on the MovieLens-20M dataset, suggesting superior performance of our method across diverse datasets.

From Figure 5 and Figure 6, it can be seen that the overall performance trend of these four methods on this dataset is generally similar to that on the dataset

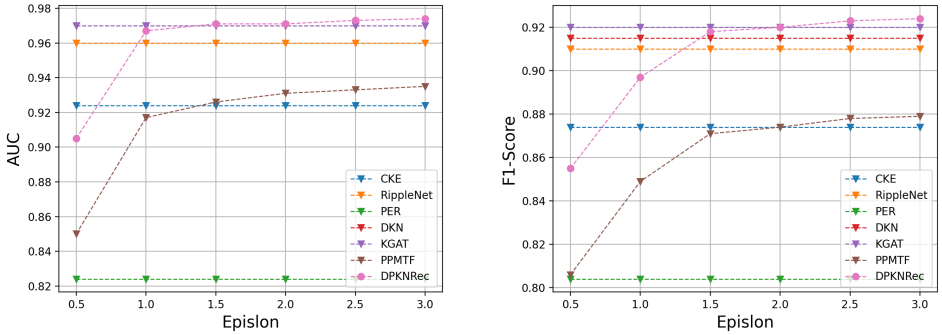


(a) (b)

Fig. 6 Performance comparison of different methods on Last.FM dataset.

MovieLens-20M, which also shows that the performance of the method in this paper is superior on different datasets.

The impact of privacy budget on recommendation results. This section examines the impact of different privacy budgets ϵ on recommendation accuracy and training model performance. In the experiments, the privacy budget ϵ was set to $\{0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5\}$. Figure 7 shows the impact of different privacy budgets on the prediction accuracy and precision of DPKNRec in the MovieLens-20M dataset.



(a) (b)

Fig. 7 Performance comparison of different methods under different privacy budgets.

From Figure 7, even at the lowest privacy budget ϵ of 0.5 (maximum noise), our method's prediction accuracy on the test set exceeds that of the DKN method. As the privacy budget increases, model noise decreases and accuracy progressively improves. Beyond a privacy budget ϵ of 2.0, the model's prediction accuracy plateaus, with

DPKNRec achieving similar maximum prediction accuracy to the DKN method (non-privacy-preserving). This implies effective prediction performance can be ensured while maintaining privacy.

From Figure 7, even at the lowest privacy budget ϵ of 0.5 (maximum noise), our method's prediction accuracy on the test set exceeds that of the DKN method. As the privacy budget increases, model noise decreases and accuracy progressively improves. Beyond a privacy budget ϵ of 1.0, the model's prediction accuracy plateaus, with DPKNRec achieving similar maximum prediction accuracy to the DKN method (non-privacy-preserving). This implies effective prediction performance can be ensured while maintaining privacy. Moreover, with a privacy budget ϵ greater than 1.5, DPKNRec's prediction accuracy surpasses that of DKN, indicating superior performance when the model is subjected to less noise.

The impact of variables in the model on recommendation results. To further validate the impact of variables in the model on the recommendation results, this study conducts experimental comparisons for different variants of DPKNRec under the assumption that the privacy budget ϵ is 1.0. Specifically, DPKNRec-noGCN refers to the method that excludes the use of knowledge graph convolutional neural networks for capturing higher-order structural and semantic data in the knowledge graph. On the other hand, DPKNRec-TransD [26], DPKNRec-TransH [27], and DPKNRec-TransR [28] denote the utilization of diverse learning methods of the knowledge graph representations in KGPNRec for obtaining the entity vectors. The experimental outcomes are presented in Figure 8 and Table 2.

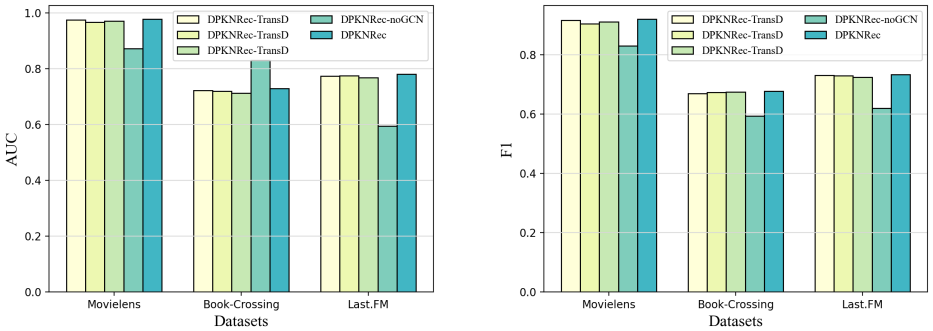


Fig. 8 Performance comparison of different variants of DPKNRec on Movielens-20M dataset.

The results depicted in Figure 8 and Table 2 reveal that the inclusion of knowledge graph convolutional neural networks for automatically capturing higher-order structural and semantic data in the knowledge graph can enhance the utility of the data, thereby leading to a significant improvement in the performance of the RS. Furthermore, when employing distinct knowledge graph representation learning methods

to generate feature vectors, the recommendation based on the TransE method outperforms the other methods. This observation can be attributed to the fact that the TransE method is more compatible with knowledge graph convolutional neural networks than the other three knowledge graph representation learning methods, enabling it to learn a superior nonlinear relationship in the privacy-preserving recommendation scenarios of knowledge graphs.

In summary, the proposed recommendation method in this paper demonstrates enhanced security and reliability compared to existing methods such as DKN, PPMTF, and KGAT. This assertion is based on the following analysis: firstly, the method in this paper leverages knowledge graph convolutional neural networks to capture the semantic data encapsulated in the knowledge graph and the personalized interests of users, thereby effectively improving the accuracy of the recommendation. Secondly, the joint design of differential privacy and pseudo-interaction sampling protection mechanism in this paper mitigates the impact on data while ensuring privacy security.

6 Conclusions

This study proposes an effective knowledge graph privacy-preserving recommendation method to model users' personalized interests and prevent their private data from being violated. We introduce a method known as DPKNRec that integrates a privacy protection algorithm with a knowledge graph convolutional neural network, applying differential privacy and pseudo-interaction item sampling. DPKNRec allows the neural network to capture and preserve the local neighborhood structure within each entity through domain aggregation operations, capturing knowledge higher-order structure and semantic data in the graph. Then, a pseudo-interaction item sampling technique is introduced to fuzzy the user embedding gradient so that it is not possible to identify which items the user has interacted with. To further ensure that the recommendation process does not leak user privacy, this paper employs a differential privacy approach by adding random noise obeying the Laplace distribution into the uniform embedding to enhance user privacy protection. Experimental validation confirms that DPKNRec maintains the utility of recommendation results while preserving data privacy. Future work will aim to optimize the recommendation algorithm to reduce computational overhead while providing privacy protection. The goal is to strike a balance between recommendation accuracy, algorithm performance, and privacy protection.

Acknowledgments. This work was supported in part by the National Science Foundation of China (No. 61472194), Natural Science Foundation of Zhejiang Province (No. LZ20F020001), Science and Technology Innovation 2025 Major Project of Ningbo (No. 20211ZDYF020036), and the Natural Science Foundation of Ningbo (No. 2021J091).

References

[1] Raza, S., Ding, C.: News recommender system: a review of recent progress, challenges, and opportunities. Artificial Intelligence Review, 1–52 (2022)

[2] Amir, N., Jabeen, F., Ali, Z., Ullah, I., Jan, A.U., Kefalas, P.: On the current state of deep learning for news recommendation. Artificial Intelligence Review **56**(2), 1101–1144 (2023)

[3] Tang, H., Liu, B., Qian, J.: Content-based and knowledge graph-based paper recommendation: Exploring user preferences with the knowledge graphs for scientific paper recommendation. Concurrency and Computation: Practice and Experience **33**(13), 6227 (2021)

[4] Amir, N., Jabeen, F., Ali, Z., Ullah, I., Jan, A.U., Kefalas, P.: On the current state of deep learning for news recommendation. Artificial Intelligence Review **56**(2), 1101–1144 (2023)

[5] Lee, J., Kim, S., Lebanon, G., Singer, Y.: Local low-rank matrix approximation. In: International Conference on Machine Learning, pp. 82–90 (2013). PMLR

[6] Wang, F., Zhong, W., Xu, X., Rafique, W., Zhou, Z., Qi, L.: Privacy-aware cold-start recommendation based on collaborative filtering and enhanced trust. In: 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA), pp. 655–662 (2020). IEEE

[7] Hu, C., Liu, B., Zhang, X., Wang, Z., Lin, C., Luo, L.: A federated multi-server knowledge graph embedding framework for link prediction. In: 2022 IEEE 34th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 366–371 (2022). IEEE

[8] Wang, H., Zhang, F., Xie, X., Guo, M.: Dkn: Deep knowledge-aware network for news recommendation. In: Proceedings of the 2018 World Wide Web Conference, pp. 1835–1844 (2018)

[9] Wang, H., Zhang, F., Wang, J., Zhao, M., Li, W., Xie, X., Guo, M.: Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, pp. 417–426 (2018)

[10] Waikhom, L., Patgiri, R.: A survey of graph neural networks in various learning paradigms: methods, applications, and challenges. Artificial Intelligence Review, 1–70 (2022)

[11] Wang, Z., Lin, G., Tan, H., Chen, Q., Liu, X.: Ckan: collaborative knowledge-aware attentive network for recommender systems. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 219–228 (2020)

[12] Wang, X., He, X., Cao, Y., Liu, M., Chua, T.-S.: Kgat: Knowledge graph attention network for recommendation. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 950–958

(2019)

[13] Wang, X., Huang, T., Wang, D., Yuan, Y., Liu, Z., He, X., Chua, T.-S.: Learning intents behind interactions with knowledge graph for recommendation. In: Proceedings of the Web Conference 2021, pp. 878–887 (2021)

[14] Lin, C., Liu, B., Zhang, X., Wang, Z., Hu, C., Luo, L.: Privacy-preserving recommendation with debiased obfuscaiton. In: 2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 590–597 (2022). IEEE

[15] Hou, D., Zhang, J., Ma, J., Zhu, X., Man, K.L.: Application of differential privacy for collaborative filtering based recommendation system: A survey. In: 2021 12th International Symposium on Parallel Architectures, Algorithms and Programming (PAAP), pp. 97–101 (2021). IEEE

[16] Boutet, A., Frey, D., Guerraoui, R., Jégou, A., Kermarrec, A.-M.: Privacy-preserving distributed collaborative filtering. Computing **98**, 827–846 (2016)

[17] Yu, B., Zhou, C., Zhang, C., Wang, G., Fan, Y.: A privacy-preserving multi-task framework for knowledge graph enhanced recommendation. IEEE Access **8**, 115717–115727 (2020)

[18] Zhang, Y., Liu, B., Qian, J., Qin, J., Zhang, X., Jiang, X.: An explainable person-job fit model incorporating structured information. In: 2021 IEEE International Conference on Big Data (Big Data), pp. 3571–3579 (2021). IEEE

[19] Wainakh, A., Grube, T., Daubert, J., Mühlhäuser, M.: Efficient privacy-preserving recommendations based on social graphs. In: Proceedings of the 13th ACM Conference on Recommender Systems, pp. 78–86 (2019)

[20] Deldjoo, Y., Di Noia, T., Merra, F.A.: Adversarial machine learning in recommender systems (aml-recsys). In: Proceedings of the 13th International Conference on Web Search and Data Mining, pp. 869–872 (2020)

[21] Qammar, A., Karim, A., Ning, H., Ding, J.: Securing federated learning with blockchain: a systematic literature review. Artificial Intelligence Review **56**(5), 3951–3985 (2023)

[22] Qi, T., Wu, F., Wu, C., Huang, Y., Xie, X.: Uni-fedrec: A unified privacy-preserving news recommendation framework for model training and online serving. arXiv preprint arXiv:2109.05236 (2021)

[23] Jiang, X., Liu, B., Qin, J., Zhang, Y., Qian, J.: Fedncf: Federated neural collaborative filtering for privacy-preserving recommender system. In: 2022 International Joint Conference on Neural Networks (IJCNN), pp. 1–8 (2022). IEEE

[24] Meng, X., Wang, S., Shu, K., Li, J., Chen, B., Liu, H., Zhang, Y.: Personalized privacy-preserving social recommendation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32 (2018)

[25] Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. Advances in neural information processing systems **26** (2013)

[26] Ji, G., He, S., Xu, L., Liu, K., Zhao, J.: Knowledge graph embedding via dynamic mapping matrix. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (volume 1: Long Papers), pp. 687–696 (2015)

[27] Wang, Z., Zhang, J., Feng, J., Chen, Z.: Knowledge graph embedding by translating on hyperplanes. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 28 (2014)

[28] Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 29 (2015)