Expert Systems With Applications 260 (2025) 125447

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Differentially private recommender framework with Dual semi-Autoencoder

Yang Deng [a], Wang Zhou [a,*], Amin Ul Haq [b], Sultan Ahmad [c], Alia Tabassum [d]

[a] *School of Computer and Software Engineering, Xihua University, Chengdu, China*
[b] *School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China*
[c] *Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Alkharj, Saudi Arabia*
[d] *Mohi ud Din Islamic University, Azad Kashmir, Pakistan*

ARTICLE INFO

Keywords:
Recommender system
Theoretical analysis
Differential privacy
Semi-Autoencoder

ABSTRACT

To provide much better recommendation service, traditional recommender systems collect a large amount of user information, which, if obtained and analyzed maliciously, can cause incalculable damage to users. Therefore, differential privacy techniques, such as noise injection, have been widely introduced into recommender systems to safeguard users' sensitive information. However, the introduction of privacy noise will lead to a degradation in recommendation quality. Hence, it is pragmatic to design a system that can furnish high quality recommendation and ensure privacy guarantee. In this article, we design a novel Differentially private recommender system with Dual Semi-Autoencoder recommender framework referred to as DP-DAE, which aims to improve the quality of recommendation while protecting user privacy. Specifically, DP-DAE is a hybrid framework of dual autoencoder and matrix factorization, which can effectively reduce data dimensionality to extract intricate features. In practice, to prevent potential privacy leaks, the differential privacy mechanism is incorporated into DP-DAE via introducing extra noise. Moreover, theoretical analysis certificates that DP-DAE satisfies ϵ-differential privacy. We do the experimental evaluation for DP-DAE over FilmTrust, Movielens-1M and Movielens-10M. The experimental results indicate that DP-DAE can provide privacy protection as well as high performance in recommendation tasks.

1. Introduction

Currently, the volume of information increases exponentially with the rapid development of the Internet. As a result, both information providers and consumers need to confront the issue of information overload. To alleviate the adverse effects of information overload on individuals, recommender systems (RS) have been created, moreover, RS are rapidly evolving. In past decades, Recommender systems are widely used in many fields, such as Taobao (Yang, 2017), Amazon (Smith & Linden, 2017) and Netflix. With the increasing number of Internet users, various service requirements are not short of being raised, such as online shopping, movie recommendation, travel recommendation and so on. Nowadays, recommender systems have become a crucial tool in our daily life. In particular, online service platforms generally leverage RS to provide much better service.

In practical application scenarios, there are two most widely used recommender systems (Chen et al., 2020). One type relies on users' historical behaviors, and integrates them with decisions made by similar users to collaboratively generate recommendations (Harrouche & Yamina, 2024). These historical behaviors encompass various actions, including product purchases, providing reviews, and adding items to the shopping cart. The other type of recommender system is built upon the discrete features of the items. This model extracts latent characteristics of items, and computes the items with similar properties for recommendation.

Since that recommender systems continually collect and analyze users' historical data and predict preferences through computational methods, users can search and select products quickly. A qualified recommender system can guarantee the recommendation quality due to that the user's private information will not be disclosed. In recent years, due to the frequent occurrence of information leakage, researchers pay more attention to the security on privacy. Therefore, recommender systems with privacy preservation are constantly proposed (Chen, Wang et al., 2021; Ermis & Cemgundefinedl, 2020).

In practice, recommender systems collect and process large amounts of user information in order to better understand user preferences, and improve the accuracy of the recommendations (Zhou et al., 2023). Nevertheless, user's personal privacy data has become publicly available. Various factors, including the accuracy and bias of data sources, as well

[*] Corresponding author.
E-mail addresses: Ljx_luz@stu.xhu.edu.cn (Y. Deng), dean_uestc@163.com (W. Zhou), Khan.amin50@yahoo.com (A.U. Haq), s.alisher@psau.edu.sa (S. Ahmad), alia.tabassum1085@gmail.com (A. Tabassum).

https://doi.org/10.1016/j.eswa.2024.125447
Received 23 April 2024; Received in revised form 20 August 2024; Accepted 22 September 2024
Available online 28 September 2024

as defects and vulnerabilities in algorithm implementation, can affect the recommendation performance. This, in turn, may lead to erroneous recommendations or the compromise of user data privacy.

The user data required during the recommendation formation process is highly sensitive, such as user age and location. Consequently, this raises concerns about privacy leakage (Gao et al., 2020; Jiang et al., 2023). A notable case involves the application of the K-anonymity anonymization model to safeguard privacy in medical data. Despite such measures, attackers can still leverage information from external public datasets to discern individual sensitive information.

Differential privacy stands out as a widely embraced privacy protection framework, particularly in its application in recommender systems based on collaborative filtering (Jiang et al., 2023; Liu et al., 2023). The core concept of Differential Privacy is to protect individual privacy by introducing noise during data processing, ensuring that the presence or specific value of any single data record cannot be inferred from the algorithm's output (Zheng et al., 2020). This approach safeguards data privacy while still allowing for effective analysis and statistical evaluation. Specifically, the goal of Differential Privacy is to ensure that even if an attacker has accessed most of the records in a database, they cannot accurately infer information about any individual record from the query results. By introducing appropriate noise, Differential Privacy minimizes the impact of the presence or absence of any single data point on the final query outcome, thereby effectively protecting individual privacy. Consequently, researches focused on developing recommender systems that incorporate robust privacy security measures. The developed methods can concurrently deliver high-quality and efficient recommendation services, which hold significant scholarly importance.

Moreover, in practical applications, the majority of entries in the matrix representing interactions between new users and items, are often set to zero. This implies the existence of a significant amount of redundant information in the raw data (Neera et al., 2021). To address privacy concerns and simultaneously introduce additional noise into available information for privacy preservation, the system may inadvertently result in the extraction of useless features with excessively high dimensions, hindering efficient computational predictions. These factors may lead to a degradation in system performance, compromising its ability to deliver high-quality recommendations (Yang et al., 2022).

In response to these challenges, this article designs a Differentially Private recommender system with Dual Semi-autoencoder referred to as DP-DAE. The objective of DP-DAE is to protect user privacy while ensuring that the system can still provide accurate and high quality recommendations. Specifically, to enhance the model's learning capability for latent features, DP-DAE adopts dual semi-autoencoder and matrix factorization techniques. While compared to traditional autoencoder-based recommendation systems, DP-DAE can simultaneously learn the latent features, thereby reducing the model's time complexity and improving computational efficiency. Moreover, Semi-autoencoder can overcome the conventional constraints of autoencoder models where the output must be consistent with the input layers. Thus, through Semi-autoencoder, additional auxiliary information can be flexibly utilized to address issues like data sparsity in traditional autoencoder. Additionally, to tackle the privacy leakage problem in recommendation systems, DP-DAE employs a differential privacy mechanism. Specifically, by adding extra noise to satisfy differential privacy, user information can be protected. Moreover, we provide rigorous theoretical analyses to demonstrate that DP-DAE satisfies ϵ-differential privacy, thereby offering users a certain level of privacy protection. Experiments on three commonly used datasets with appropriate privacy budgets indicate that DP-DAE can provide high-quality recommendations while ensuring privacy security.

In general, the contributions of this research are as follows:

- DP-DAE is designed to provide accurate recommendation service, which can also maintain privacy guarantees. Additionally, the model can accurately uncover the latent features of users and items, and enhances the overall performance of the recommendation system by integrating matrix factorization techniques.
- DP-DAE employs a Dual Semi-autoencoder to address data sparsity, which can reduce adverse impacts of privacy noise and improve recommendation quality.
- DP-DAE integrates an algorithm that reduces estimation errors and improves overall performance, which randomly selects an element and calculates gradient information after perturbation.
- Theoretical analysis suggests that DP-DAE can achieve high performance under strong differential privacy requirements for user information.
- Experimental results over FilmTrust, Movielens-1M and Movielens-10M indicate that DP-DAE can provide accurate item recommendations while satisfying a certain level of privacy protection. Moreover, DP-DAE outperforms other benchmark methods.

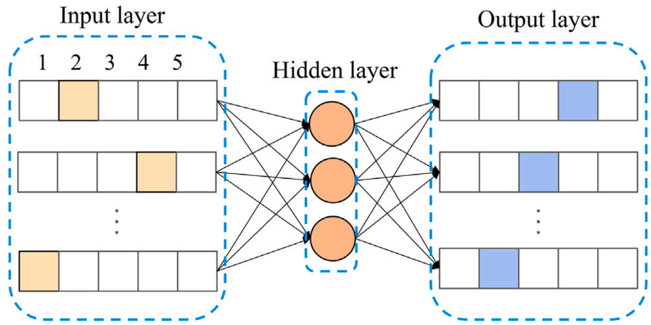


Fig. 1. The structure of user-specific Autoencoder based collaborative filtering (ACF). The orange squares in the input layer represent the corresponding user ratings.

The remaining parts are organized as follows. Section 2 is an overview of the relevant technical work. Section 3 is for the overall framework of the model. Section 4 is for experiments on three different datasets. Section 5 will present future work.

2. Related work

2.1. Autoencoder and semi-autoencoder

Autoencoders, proposed within the realm of machine learning, function as pivotal unsupervised learning models (Zhang et al., 2019). Essentially, the encoder transmutes input data into an abstract representation, with the decoder subsequently reconstructing the identical data within the output layer.

Autoencoders have a significant advantage in that they can effectively reduce dimensionality by representing data in a lower-dimensional latent feature space. Autoencoders have been widely applied across diverse domains, such as signal processing, image processing, and video analysis. In recent years, significant progress has been made as a result of in-depth research on deep learning (Song et al., 2024), and autoencoders also have shown remarkable effectiveness in recommender systems.

Autoencoders primarily consist of encoding and decoding components. During the encoding process, the autoencoder takes the input vector R_u and passes it through the hidden layer H_u, as represented by Eq. (1), to obtain latent representation. Subsequently, in the decoding process, the autoencoder reconstructs the original information from the encoded H_u using Eq. (2). This latent representation is then utilized to predict ratings $\widehat{R}_u$.

$$H_u = g\left(W_h^{\mathrm{T}} R_u + \mathcal{B}_h\right), \tag{1}$$

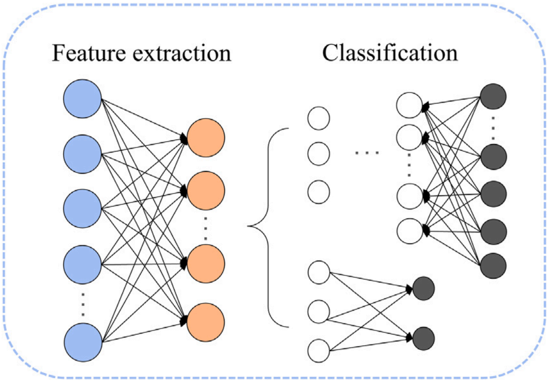


Fig. 2. The structure of supervised neural recommendation framework (SNR).

$$\widehat{R}_u = f\left(W_o^{\mathrm{T}} H_u + \mathcal{B}_o\right), \tag{2}$$

where W_h and W_o are weight matrices, and $\mathcal{B}_h$ and $\mathcal{B}_o$ are bias vectors.

Currently, there are several developed versions of autoencoders, including variational autoencoders (Li et al., 2023; Liang et al., 2018), denoising autoencoders, marginalized autoencoders, and so on (Zhang, Yao, & Xu, 2017), which are widely applied in recommender systems (Karamanolakis et al., 2018).

The semi-supervised autoencoder includes an input layer and an output layer. By harnessing the potential of previously unlabeled data for model training and effectively extracting valuable information, the semi-supervised autoencoder effectively mitigates data sparsity concerns, ultimately leading to remarkable enhancements in the model's precision and generalization capabilities.

The Hybrid Collaborative Recommendation via Semi-Autoencoder (Yang et al., 2022), can alleviate the inherent constraint of equal dimensions between the input and output layers in traditional autoencoder models. Moreover, the method has a low dimension in the output layer, enabling the model to flexibly learn the latent representation.

The hybrid semi-supervised autoencoder based recommendation model can flexibly handle diverse types of data, such as user behavior records and supplementary information (Ribero et al., 2022). By setting different dimensions for the input layer and output layer, the model can simultaneously leverage various types of data to learn more comprehensive representations of latent features.

2.2. Matrix factorization

Matrix factorization technique can decompose large-scale user–item rating matrices, aiming to better capture latent features or patterns within the data. In practice, the user–item rating matrix generally contains many blank entries. In practical terms, a recommender system involves m users and n items, forming a user–item rating matrix $R \in \mathbb{R}^{m\times n}$. Within the user–item matrix, each value represents an individual user rating for a specific item, denoted as R_i for the rating to item i, assigned by user u.

The core idea of matrix factorization is to decompose the user–item matrix to learn latent features, enabling the construction of lower-dimensional matrices to approximate the original user rating matrix. Jannach et al. introduced a model-based collaborative filtering algorithm that effectively addresses the issue of data sparsity in recommender systems (Jannach & Ludewig, 2017).

Item-based collaborative filtering recommender systems face several issues, such as cold start, data sparsity, and low scalability. Guo et al. incorporated additional attribute information into the process of computing item similarity, to improve the prediction accuracy (Guo et al., 2016).

Table 1
Key notations and meanings.

Notation	Description
$R \in \mathbb{R}^{m\times n}$	rating matrix
R'	prediction matrix
n	number of items
m	number of users
p^i	addition information
C	representation vector
r'^{I}	prediction item rating matrix
$\omega,\ \omega'$	weight metrices
h	dimensionality of the hidden layer
f_i	loss function
$\mathcal{N}$	noise matrix
V	item matrix
ϵ	privacy budget
δ	probability parameter
Δ_f	sensitivity function
ξ_i	predicted data matrix
q	perturbed vector
Δ_r	range of ratings
$D,\ D'$	adjacent datasets
$Range(\mathscr{A}),\ \mathcal{S}$	the set of output results

Aghdam et al. proposed an NMF-based approach to address issues in traditional recommender systems (Aghdam et al., 2017). Through experiments, it was certificated that the method effectively alleviated data sparsity and system scalability issues, successfully improving the quality of recommendations.

2.3. Autoencoder based recommendation

Autoencoder based collaborative filtering (ACF) is based on autoencoder, and incorporates collaborative filtering for recommendation. ACF transforms the original integer ratings into a vector composed of 0 and 1, treating this as an input vector.

Fig. 1 illustrates the structure of ACF. In this structure, each row in the input layer represents an item. The rating range is restricted to integers between 1 and 5. In the output layer, each cell represents the probability of an item being rated as k. The prediction values can be obtained through this network.

Wu et al. propose the Collaborative Denoising Autoencoder (CDAE) (Wu et al., 2016), which is designed for item ranking predictions. CDAE incorporates a preference vector to measure user's interesting level to an item, the value of which is 1 indicating residual interest, and 0 otherwise.

In Wu et al. (2016), a negative sampling technique is proposed, significantly reducing the time overhead of computing user ratings, while maintaining the quality of the model's ranking. Moreover, researchers find that using Frobenius as a regularization parameter can result in excessively smooth parameter curves.

Supervised neural recommendation algorithm (SNR) utilizes stacked autoencoder (SAE) for feature learning (Yi et al., 2016), which is solely based on autoencoder without incorporating traditional recommendation algorithms. Fig. 2 illustrates the structure of SNR. SAE firstly learns the latent features of items, and then undergoes reconstruction, with the obtained output serving as the predicted ratings.

2.4. Differential privacy recommender system

With the exponential growth of data, incidents of information leakage have become increasingly prevalent, prompting users to focus on information security. To address the potential risk of private information leakage during the collection of user data, privacy-preserving recommender systems are continuously being proposed (Beg et al., 2021; Gao et al., 2020; Jiang et al., 2023). Scholars in the field of machine learning have contributed numerous researches on privacy-preserving recommender systems.

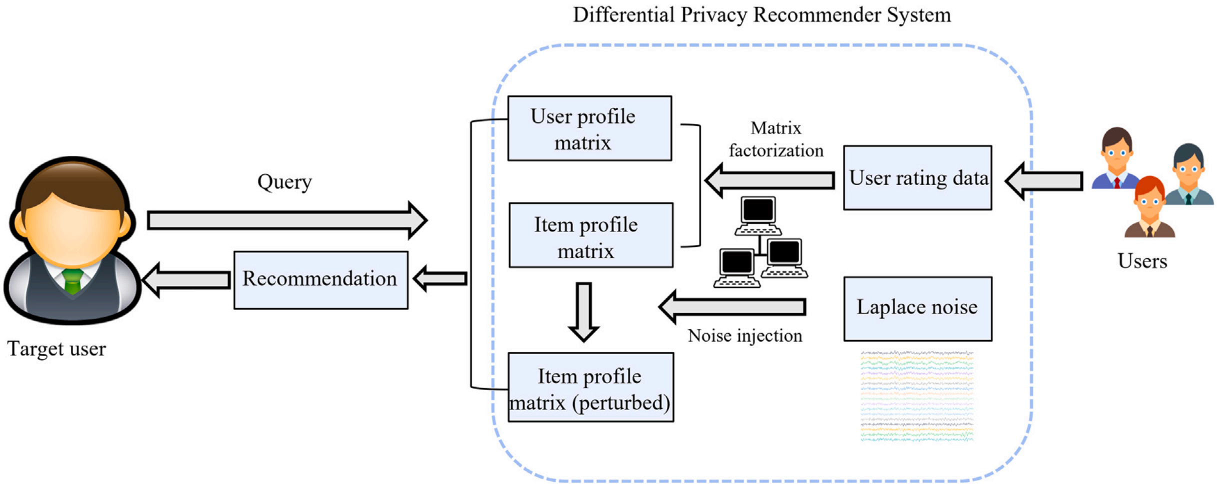


Fig. 3. Differential privacy matrix factorization for recommender system.

In recommendation systems, differential privacy effectively protects user privacy by introducing noise (such as Laplace noise or Gaussian noise) during data processing. This noise is added during the computation of recommendation results to ensure that even if some records in the dataset are leaked or accessed, attackers cannot accurately infer the personal information of any specific user. By doing so, differential privacy significantly reduces the impact of the presence or absence of any single data record on the final recommendation results, thereby lowering the risk of privacy breaches while maintaining the functionality and effectiveness of the recommendation system.

In Fang et al. (2022), a variant of autoencoder-based recommender system with differential privacy is introduced. Built upon a variant of autoencoder, the authors present a novel recommendation model that ensures privacy preservation without compromising the accuracy of recommendations for each user (Li & She, 2017). Furthermore, the model optimizes the autoencoder variant by incorporating user-level prior information from user metadata.

Matrix factorization-based differential privacy recommendation algorithms have also been widely applied (Zhou et al., 2023). Fig. 3 illustrates the differential privacy recommender framework. Ran et al. utilize objective perturbation and stochastic gradient descent to learn features, providing differential privacy guarantees for item attribute profiles (Ran et al., 2022a). The approach in this article ensures that user privacy is thoroughly protected while delivering personalized recommendation services.

3. The proposed DP-DAE

We will present the whole framework of differentially private recommender system with Dual semi-Autoencoder (DP-DAE), and present the relevant theoretical analysis to demonstrate that the framework satisfies ϵ-differential privacy. To better present the proposed DP-DAE, we first give the following definitions. We have organized and presented the pertinent symbols and meanings in Table 1.

Definition 1. ϵ-differential privacy (Chen, Liu et al., 2021). Algorithm $\mathscr{A}$ will satisfy ϵ-differential privacy ($\epsilon - DP$), if for all measurable sets $S \subseteq Range(\mathscr{A})$ and for all $D, D' \in D^n$ that differ by one element. We define that $\mathscr{A}$ is ϵ-differential, when $\delta = 0$.

$$\mathbb{P}(\mathscr{A}(D) \in S) \leq \exp(\epsilon)\mathbb{P}\left(\mathscr{A}\left(D'\right) \in S\right) + \delta, \tag{3}$$

where $\mathbb{P}(\cdot)$ measures the probability of the outputs. D and D' are adjacent datasets, indicating that they are almost identical except for one data record. δ is a small probability parameter for privacy leakage, which is typically set to a value close to zero. Parameter ϵ controls the strictness of privacy protection. $\mathscr{A}$ represents a random algorithm that satisfies ϵ-differential privacy, and $S \subseteq Range(\mathscr{A})$ is the set of all possible outputs of the algorithm.

Definition 2. Sensitivity (Chen, Liu et al., 2021). For any two adjacent datasets D and D', function f can measure the maximum change in the output of the function when there is a slight variation in the input datasets. It can be defined as follows:

$$\Delta_f = \max_{D,D'} \left\| f(D) - f\left(D'\right) \right\|_1 . \tag{4}$$

Sensitivity quantifies the maximum difference in output between two neighboring datasets which are the same except only one element. It is a crucial parameter for determining the appropriate level of noise to be added. Accurate estimation of sensitivity can facilitate better protection of privacy information within the dataset.

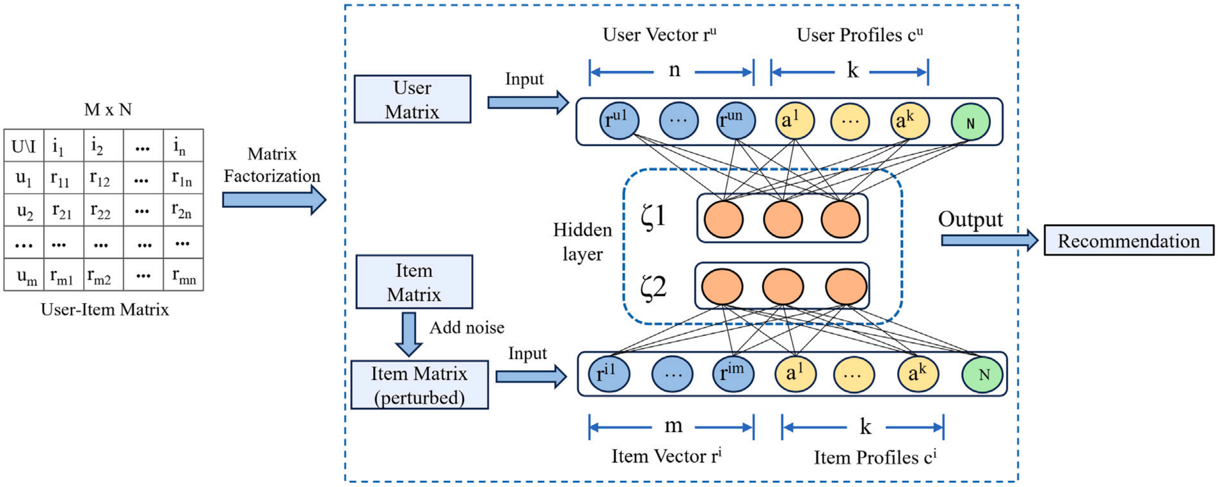
Definition 3. The Laplace mechanism (Zhao & Chen, 2022), also known as the Laplace perturbation mechanism, is employed to add noise, which conforms to the Laplace distribution. The noise has a mean of zero, and its scale is determined by the privacy parameter. For a given function f, there exists an algorithm $\mathscr{B}$ that ensures f to satisfy ϵ-differential privacy (ϵ-DP):

$$\mathscr{B} = f(D) + Lap(\frac{\Delta_f}{\epsilon})^d, \tag{5}$$

where Δ_f represents the sensitivity of the function, as stated in the equation above. $Lap(\cdot)$ represents a vector with a length of d, where each element is sampled randomly according to the Laplace distribution with a mean of zero.

3.1. Overview of DP-DAE

As illustrated in Fig. 4, we begin with a rating matrix $R \in \mathbb{R}^{m\times n}$, representing the ratings of all users for all items. Here, n and m respectively denote the quantities of items and users. To enhance the quality of recommendations, We extract the genre, release year, and release time of items from the Movielens dataset as additional item-related information, and represent it as $p^i \in P^I$, and incorporate it into the rating vector r^i. Consequently, an augmented item matrix C



U\I	i_1	i_2	…	i_n
u_1	r_{11}	r_{12}	…	r_{1n}
u_2	r_{21}	r_{22}	…	r_{2n}
…	…	…	…	…
u_m	r_{m1}	r_{m2}	…	r_{mn}

Fig. 4. An overview of the proposed DP-DAE, which can provide privacy protection as well as item recommendation.

is obtained, serving as the input to the autoencoder, facilitating more effective learning of hidden representations for items. Specifically, we use C to denote the concatenation of r^I and P^I, where r^I represents the overall ratings for all items, and P^i represents information related to all items.

By employing C as the input to the autoencoder, our objective is to acquire meaningful hidden representations of the items. According to Fang et al. (2022) and Hu et al. (2021), we incorporate special user noise to safeguard user privacy and security. Additionally, we will train the model using Stochastic Gradient Descent method (SGD). We apply a form of differential privacy, specifically targeting individual users, instead of employing global differential privacy across all users.

The role of the autoencoder is to maximize the restoration of the initial output, which equal to reconstruct the input rating data to the best extent possible. The primary function of the autoencoder is to maximize the fidelity of the reconstructed output, which involves restoring the input ratings to the greatest extent possible.

The autoencoder contains two general components: the encoder and the decoder. In the encoding phase, we utilize the concatenated representation C (as mentioned earlier, representing the connection between r^I and P^I) as the input to the encoder. The encoder's main objective is to map the data onto a lower-dimensional space, effectively reducing its dimensionality. This process enables the encoder to capture the essential features of the input data efficiently. The encoding process for the items can be mathematically represented as Eq. (6):

$$\xi_i = f(C\omega + b). \tag{6}$$

Here, ξ_i denotes the encoded representation of the input data obtained through the encoder. $C = r^I \oplus P^I$ is the concatenation of r^I and P^I. b is the bias vector.

By taking the encoded representation ξ_i as input, the decoder reconstructs the data back to its original form. This reconstruction process is expressed as follows:

$$r'^I = g(\omega' \xi_i + b'). \tag{7}$$

The ultimate goal of the autoencoder is to accurately reconstruct the original rating data. The specific formulation for the encoding process may vary, depending on the chosen architecture and design considerations of the autoencoder model. The reconstruction error of the autoencoder can be represented in Eq. (8).

$$f_i = \min_{\omega,\omega',b,b',r^{ui\in\omega}} \|r^I - r'^I\|^2. \tag{8}$$

Note that in the formulas above, $\omega \in R^{(m+y_i)\times h}$ and $\omega' \in R^{h\times m}$ represent the weight matrices within the autoencoder framework. The variables b and b' indicate the biases terms, and h is the dimensionality of the hidden layer.

Matrix factorization techniques offer the advantage of decomposing the user–item matrix, thereby reducing computational complexity. To enhance the efficiency of DP-DAE, we utilize the matrix factorization techniques. By lowering the matrix dimensions, the model can accurately capture the hidden representations. Following matrix factorization processing, we obtain a prediction matrix R'. The objective of DP-DAE is to minimize the prediction error, as formulated in Eq. (9).

$$f_r = \min_{\omega,\omega',b,b',\omega_u,\omega'_u,b_u,b'_u,r^{ui\in\omega}} \|R - R'\|^2. \tag{9}$$

To ensure model stability and bolster generalization capabilities, we apply L2 regularization to penalize parameters, which is shown in Eq. (10).

$$f_l = \frac{\kappa}{2}\left(\|\omega\|^2 + \|\omega'\|^2 + \|\omega_u\|^2 + \|\omega'_u\|^2\right). \tag{10}$$

Notice that κ is a trade-off parameter. Till now, the objective function for DP-DAE is expressed as shown in Eq. (11).

$$\mathcal{L} = f_i + f_u + f_r + f_l. \tag{11}$$

For the parameters optimization, we adopt SGD method. This approach has the merits of computational efficiency and faster parameter updates. By randomly selecting subsets of data for each iteration, SGD enables a fast convergence rate, reducing the computational cost.

3.2. Differentially private matrix factorization

We initially focus on a real world scenario of mutual trust, where users grant the recommender system permission to collect their rating data for personalized recommendations. Within the framework of trust, considering that users desire to safeguard their privacy, it will be crucial for privacy protection. Consequently, the recommender system must ensure privacy protection while providing recommendations.

Therefore, the recommender systems need to utilize appropriate techniques to the released item set V to ensure an adequate level of privacy protection, thereby meeting the requirements of ϵ-differential privacy. Failure to do so may pose risks to attackers, who generally infer user ratings by exploiting the dot product between any ratings and user information on the dataset. In practical implementation, recommender systems often adopt a privacy protection approach by separately handling users' personal information (such as gender and age) to ensure its confidentiality (Fredrikson et al., 2015). However, item-user ratings are typically made public and available for analysis and recommendation purposes.

By leveraging the publicly available item set V, the recommender system can effectively enhance the training of its model, providing much more accurate recommendations for new users, even in scenarios where the rating matrix is sparse and user rating data is limited. It is crucial to consider that in practical applications, the released V still contains some localized user information.

To address this issue, we employ a perturbation-based approach to achieve our objective. Specifically, we introduce random perturbation into the objective function to achieve differential privacy. It is worth noting that this approach differs from the method of adding perturbation at the output stage. The perturbed function is as follows:

$$\begin{aligned}\min_{V} \widetilde{\mathcal{L}}(V) = \frac{1}{m} \sum_{(i,j)\in\mathcal{R}} \left(r_{ij} - u_i^T v_j\right)^2 + \psi \sum_{i\in[n]} \|u_i\|_2^2 \\ + \phi \sum_{j\in[m]} \left\|v_j\right\|_2^2 + \frac{1}{m} \sum_{j=1}^{m} \eta_j^T v_j.\end{aligned} \tag{12}$$

We ensure user privacy by introducing noise, where $\mathcal{N} = [\eta_j]_{d\times m}$ represents the noise information. In a scenario of mutual trust, where user information is confidential, we treat the matrix U as a constant matrix. Consequently, we only need to handle a unique variable matrix U, regardless of f_u. Finally, the recommendation task is accomplished by combining V' and U. After the aforementioned processing, the objective function could be formulated as follows:

$$\mathcal{L} = f_i + f_r + f_l. \tag{13}$$

In summary, we utilize the user matrix obtained from Eq. (12) as a constant matrix. Subsequently, our focus shifts to the item matrix V. Therefore, our optimization objective is represented by Eq. (11). To preserve privacy attributes while performing matrix factorization, we introduce noise into the proposed DP-DAE. The calculation of sensitivity ensures that the added noise does not compromise the quality of recommendations. Consequently, we can obtain a perturbed objective function in Eq. (13), by introducing noise information that follows the Laplace distribution. **Algorithm** 1 provides further details on the process.

Algorithm 1 Updating process for perturbed objective function.

```
Input: Object function $f(q)$.
Output: Perturbed objective function $f(\bar{q})$.
for $i \leftarrow 0$ to $D$ do
    $f(q_i) = (-P_{ui}q_i)^T q_i + \frac{1}{2}(P_{u_i}^t + \lambda_Q E)q_i^t$;
    $\delta = \max ||p_u|| \cdot \Delta_r$;
    $f(q_i) = [-P_{ui}q_i + \mathrm{Lap}(\frac{\Delta}{\epsilon})^d]^T q_i + \frac{1}{2}(P_{u_i}^t + \lambda_Q E)q_i^t$;
end for
return $f(q_i)$.
```

3.3. Estimation error

In practical applications, we have identified estimation errors within the matrix V. To address this issue, we explore the randomized approach proposed by Shin et al. (2018), which has promising results. The research in Shin et al. (2018) underscores the effectiveness of this approach in diminishing estimation errors and enhancing overall performance. Therefore, we refer to this method to improve our method's performance.

To this end, we incorporate the algorithmic approach from Shin et al. (2018) into **Algorithm** 2, further enhancing the matrix factorization in our embedding model. In **Algorithm** 1, we select an element from the $m \times d$ matrix randomly, and then compute the gradient information after perturbation. This randomized sampling approach allows for a reduction in estimation errors with each iteration. According to the experimental analysis in Shin et al. (2018), the estimation error can decrease from $\mathcal{O}(m)$ to $\mathcal{O}(M)$ to $\mathcal{O}(\sqrt{MP})$.

Due to the server's requirements for users to randomly select a dimension to submit noise information, the obtained noise gradient can only be $T(= md\frac{e^{\epsilon^*}+1}{e^{\epsilon^*}-1})$. When facing noise information, attackers are unable to infer user privacy preferences and historical evaluations of a particular item, which can undoubtedly enhance the security of the system and protect user privacy preferences. Through theoretical analysis, we can also demonstrate that the proposed DP-DAE satisfies ϵ-differential privacy.

Algorithm 2 Stochastic Gradient Descent with Differential privacy.

```
Input: privacy budget $\epsilon$ iteration number $k$.
Output: item matrix $V \in \mathbb{R}^{m\times d}$.
Initialize $U, V$, itercount;
for $itercount <= k$ do
    $\nabla_V^* = \vec{0}$;
    for $i = 1$ to $n$ do
        $x_i^* = \vec{0}$;
        $j$= random.$[1, 2, ..., m]$;
        $l$ = random.$[1, 2, ..., d]$;
        $(x_i)_{l,j} = -2y_{ij}u_{il}(r_{ij} - u_i^T v_j)$;
        if $(x_i)_{l,j}$ not in [-1,1] then
            $(x_i)_{l,j} = [-1, 1]$;
        else
            $(x_i^*)_{j,l} = -T$;
        end if
        $DrawT \sim Bernouli(\frac{(x_i j)_{j,l}(e^{\epsilon^*}-1+e^{\epsilon^*}+1}{2(e^{\epsilon^*}+1)})$;
        $\nabla_V^* = \nabla_V^* + x_i^*$;
        itercount = itercount + 1;
        $V = V - \rho \cdot \nabla_V^* + 2\lambda_v V$;
        for $i = 1$ to n do
            $u_i = u_i - \rho \cdot \nabla_V^* + 2\lambda_u u_i$;
        end for
    end for
end for
return $V$.
```

Obviously, computing the gradient information for all users undoubtedly consumes a significant amount of computational resources and time. Traditional gradient descent algorithms involve calculating the gradient for individual users at each iteration to update V. As the computational load increases, the convergence speed of the algorithm tends to decrease. To address these challenges, we employ Differential Privacy Stochastic Gradient Descent (DPSGD) algorithm. Additionally, we reduce the training batch size at each iteration. This approach can accelerate the convergence rate and mitigate issues related to gradient rebounds caused by updates.

Furthermore, the gradient error of **Algorithm** 2 tends to increase as the privacy budget $\epsilon^*(\epsilon/k)$ decreases. With the increasing iterations, for a fixed value of ϵ, the privacy budget per iteration diminishes. This reduction in privacy budget, coupled with the presence of $\mathcal{O}(\epsilon^*) = \frac{e^{\epsilon^*}+1}{e^{\epsilon^*}-1}$, may lead to the noise gradient exploding, even causing the algorithm to fail to converge.

With the approach above, we can enhance the model's performance, prevent the occurrence of gradient explosions, and improve stability for DP-DAE. Even with the modification of a small portion of **Algorithm** 2, the computed V still maintains differential privacy, ensuring user privacy security.

3.4. Theoretical analysis

In the following, we present a rigorous mathematical definition for DP-DAE, to demonstrate the compliance of the noise added to the item matrix V with ϵ-DP.

Theorem 1. **Algorithm** *1 satisfies ϵ-DP.*

Proof. Here, η is sampled from a Laplace distribution, and λ is set as the perturbation vector. We can have

$$\begin{aligned}\frac{\Pr\left\{\bar{f}\left(q_i\right) \mid r_i\right\}}{\Pr\left\{\bar{f}\left(q_i\right) \mid r_i'\right\}} &= \frac{\Pr\left\{-P_{U_i}^T r_i + \eta = \lambda\right\}}{\Pr\left\{-P_{U_i}^T r_i' + \eta' = \lambda\right\}} \\ &= \frac{\prod_{j=1}^{l} \Pr\left\{\eta_j = \lambda_j - z_j\right\}}{\prod_{j=1}^{l} \Pr\left\{\eta_j' = \lambda_j - z_j'\right\}} \\ &= \frac{\prod_{j=1}^{l} \frac{\epsilon}{2\Delta} \exp\left(\frac{-\epsilon\left|\lambda_j - z_j\right|}{\Delta}\right)}{\prod_{j=1}^{l} \frac{\epsilon}{2\Delta} \exp\left(\frac{-\epsilon\left|\lambda_j - z_j'\right|}{\Delta}\right)} \\ &= \frac{\left(\frac{\epsilon}{2\Delta}\right)^l \exp\left(\frac{-\epsilon \sum_{j=1}^{l} |\lambda_j - z_j|}{\Delta}\right)}{\left(\frac{\epsilon}{2\Delta}\right)^l \exp\left(\frac{-\epsilon \sum_{j=1}^{l} |\lambda_j - z_j'|}{\Delta}\right)} \\ &= \frac{\exp\left(\frac{-\epsilon\left\|\lambda + P_{U_i}^T r_i\right\|}{\Delta}\right)}{\exp\left(\frac{-\epsilon\left\|\lambda + P_{U_i}^T r_i'\right\|_1}{\Delta}\right)} \\ &\leqslant \exp\left(\frac{\epsilon\left\|P_{U_i}^T r_i - P_{U_i}^T r_i'\right\|_1}{\Delta}\right) \\ &\leqslant \exp\left(\frac{\epsilon\left(\max_{u \in U_i}\left\|p_u\right\|_1\right) \cdot \Delta_r}{\Delta}\right) \\ &\leqslant \exp(\epsilon). \quad \square\end{aligned} \tag{14}$$

Therefore, the post-noise addition item matrix satisfies differential privacy. In **Algorithm** 1, we incorporate noise to the ratings of each item, ensuring their independent and non-overlapping nature. Consequently, **Algorithm** 1 satisfies the requirements of differential privacy.

Theorem 2. **Algorithm** *2 satisfies ϵ-DP.*

Proof. Suppose that D and D' are two adjacent datasets, and the two datasets differ at only one element. We can have

$$\begin{aligned}\Pr\left[\nabla_V^* = w \mid D\right] &= \Pr\left[\sum_{i=1}^{n} x_i^* = nw \mid D\right] \\ &= \sum_{\substack{\tilde{w}_1,\ldots,\tilde{w}_n \\ \tilde{w}_1+\cdots+\tilde{w}_n = nw}} \prod_{i=1}^{n} \Pr\left[x_i^* = \tilde{w}_i \mid D\right] \\ &= \sum_{\tilde{w}_1+\cdots,\tilde{w}_n = 1} \prod_{i=1}^{n} \Pr\left[x_i^* = \tilde{w}_i \mid x_i\right].\end{aligned} \tag{15}$$

To prevent overfitting, a quadratic randomization method is employed to obtain a probabilistic space. Therefore, we can infer that

$$\begin{aligned}\frac{\Pr\left[\nabla_V^* = w \mid D\right]}{\Pr\left[\nabla_V^* = w \mid D'\right]} &\leq \max_{\substack{\tilde{w}_1,\ldots,\tilde{w}_n \\ \tilde{w}_1+\cdots+\tilde{w}_n = nw}} \prod_{i=1}^{n} \frac{\Pr\left[x_i^* = \tilde{w}_i \mid x_i\right]}{\Pr\left[x_i^* = \tilde{w}_i \mid x_i'\right]} \\ &= \max_{\tilde{w}_p} \frac{\Pr\left[x_p^* = \tilde{w}_p \mid x_p\right]}{\Pr\left[x_p^* = \tilde{w}_p \mid x_p'\right]} \\ &\leq e^{\epsilon^*}.\end{aligned} \tag{16}$$

This indicates that each generated V during each iteration, satisfies ϵ/k differential privacy. According, we can infer that

$$\begin{aligned}\frac{\Pr[V = \bar{V} \mid D]}{\Pr\left[V = \bar{V} \mid D'\right]} &\leq \max_{a_1,\ldots,a_k} \prod_{t=1}^{k} \frac{\Pr\left[\nabla_V^{*,t} = a_t \mid D\right]}{\Pr\left[\nabla_V^{*,t} = a_t \mid D'\right]} \\ &\leq e^{\epsilon}.\end{aligned} \tag{17}$$

As a result, the final V obtained through **Algorithm** 2 also adheres to the principles of ϵ-differential privacy. □

In the following, we analyze the estimation error of the final project configuration, which is obtained through **Algorithm** 2.

Theorem 3. *Assume that V^* represents the project matrix obtained through* **Algorithm** *2, and V is the project matrix obtained after k non-private iterations with a learning rate that decreases with the increasing iterations. The estimation error between V^* and V can be inferred as follows, with a probability of* $1 - \eta$,

$$\left\|V^* - V\right\|_{\max} = O\left(\frac{\sqrt{md\log(md/\eta)}}{\epsilon^*\sqrt{n}}\right). \tag{18}$$

Proof. Here we can set $c_t = \prod_{j=t+1}^{K}(1 - 2\lambda_v\rho_j)$, and the estimation error can be calculated according to **Algorithm** 2 via the formulation as

$$V^* - V = -\sum_{t=1}^{k} \rho c_t \nabla_V^{*,t} - \nabla_V^t, \tag{19}$$

where $\nabla^{*,t} j, l = o(\frac{md}{(\epsilon^*)^2 n})$.

Let $Pr\left[\left\|\sum_{t=1}^{k} \gamma_t c_t \left(\nabla_V^{*,t} - \nabla_V^t\right)\right\|_{\max} > \lambda \mid D\right] = p$. Then we can infer that

$$\begin{aligned}&\Pr\left[\left\|\sum_{t=1}^{k} \gamma_t c_t \left(\nabla_V^{*,t} - \nabla_V^t\right)\right\|_{\max} > \lambda \mid D\right] \\ &\leq \sum_{j=1}^{m}\sum_{l=1}^{d} p \\ &\leq 2\ md \cdot \exp\left(-\frac{\lambda^2}{\sum_{t=1}^{k} \gamma_t^2 c_t^2 \operatorname{Var}\left(\left(\nabla_V^{*,t}\right)_{j,l}\right) + \frac{2}{3}\lambda md \frac{e^{\epsilon^*}+1}{e^{\epsilon^*}-1}}\right) \\ &= O\left(md \cdot \exp\left(-n\lambda^2(\epsilon^*)^2/md\right)\right). \quad \square\end{aligned} \tag{20}$$

Theorem 3 constrains the prediction error for DP-DAE in practice, which can be applied to privacy protection recommendation tasks.

4. Evaluation

We will employ three datasets to evaluate the performance of DP-DAE, including FilmTrust, Movielens-1M, and Movielens-10M.

4.1. Datasets and metrics

The FilmTrust is a small-scale dataset, which contains ratings and social relations. The dataset size is relatively small, containing only 35,497 movie rating entries and 1853 social connection records.

The Movielens is a widely applied dataset composed of movie ratings, which is extensively employed in research on recommender systems and machine learning. It is collected and released by the MovieLens project team, with the primary objective of investigating the performance of collaborative filtering recommendation algorithms. The Movielens dataset contains a substantial amount of user ratings for movies, and also provides additional information related to both movies and users. The dataset includes rich metadata, such as movie titles, genres, release years, and user IDs. Researchers and developers can employ the Movielens dataset for various tasks, including evaluating recommendation algorithms, analyzing user behavior, performing feature engineering, and training models. The Movielens dataset has gained

Table 2
Statistic of FilmTrust, MovieLens-1M and MovieLens-10M.

Dataset	#users	#items	#Density	#Rating
FilmTrust	1,508	2,071	0.011	35,497
MovieLens-1M	6,040	3,796	0.042	1,000,209
MovieLens-10M	1,508	2,071	0.011	9,992,554

widespread popularity in both academic and industrial settings. In this article, we employ two datasets: Movielens-1M and Movielens-10M. Table 2 shows the details of each dataset.

We employ two primary metrics to assess performance: Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE), which are defined as:

$$\text{RMSE} = \sqrt{\frac{\sum_{(u,i)\in R^{\text{test}}} \left(r_{ui} - \hat{r}_{ui}\right)^2}{|N(R^{\text{test}})|}}, \tag{21}$$

$$\text{MAE} = \frac{\sum_{(u,i)\in R^{\text{test}}} |R - R'|}{|N\left(R^{\text{test}}\ \right)|}. \tag{22}$$

Here $N(R^{\text{test}})$ represents the pairs in the test dataset. $\hat{r}_{ui}$ represents the predicted rating in each user–item pair, and r_{ui} represents the true rating.

4.2. Baselines and implementation

In order to enhance the performance of DP-DAE and better capture the implicit relationships between users and items, we combine matrix factorization techniques and semi-Autoencoder together for model training. Unlike previous approaches employing matrix factorization and autoencoder, we introduce some novel modifications for DP-DAE. In following, we conduct performance analysis for DP-DAE and the following benchmark methods.

- **HRSA** is a novel semi-autoencoder structure (Zhang et al., 2017b), forming the foundation for the development of a hybrid collaborative filtering recommendation model. HRSA performs both rating prediction and ranking prediction concurrently. Experimental evaluations are carried out to provide empirical evidence of the model's superior performance in both rating and ranking prediction tasks.
- **SVD** (Guo et al., 2015) is a traditional matrix factorization technique used extensively in linear algebra and numerical analysis. It serves the purpose of matrix analysis and data dimensionality reduction. Regarded as a potent mathematical tool, SVD is instrumental in data analysis, dimensionality reduction, feature extraction, and problem-solving within multiple fields. Moreover, SVD is applied across various domains, including recommender systems, dimensionality reduction analysis, signal processing, and data compression. It can unveil essential information and inherent structures within data, and provide invaluable support to a multitude of applications.
- Differential Privacy Matrix Factorization (**DP-MF**) (Hua et al., 2015) is an approach that integrates matrix factorization techniques with the principles of differential privacy. In recommendation systems, where user preferences and individual behavior data are highly sensitive, DP-MF adds noise to obscure sensitive information for each user. Moreover, the application of differential privacy serves as a preventive measure against potential attacks, thereby enhancing the overall security of the system.
- **DP-CF** leverages the analysis of user historical behavior data to calculate user or item similarity, to achieve personalized recommendations. By seamlessly integrating differential privacy technologies with collaborative filtering together, DP-CF ensures the provision of personalized recommendation service, and prioritizes user privacy (Yang et al., 2017).
- **Item-Agrec** (Dong et al., 2021) leverages a semi-autoencoder to extract features of items. Moreover, it extracts the mathematical and graphical features for items in order to enhance the recommendation accuracy.

We further report the details of the model structures and the parameter configurations used for training, which are shown in Table 3. We utilize PyTorch 2.0 on a PC for model training, including an Nvidia GTX3080 GPU, an Intel i7-12700K CPU with 12 cores, and 64 GB of memory. In the experimental phase, we systematically adjust the training set to analyze the performance of several models.

4.3. Experimental results

During the course of the experiments, we employ distinct models for FilmTrust, Movielens-1M and Movielens-10M. We randomly select the specific proportion of the complete datasets as the training set, while the remaining portion is allocated as the testing set to assess model performance. All experimental results are derived from five repeated experiments, and the average results are computed accordingly. Tables 4–6 report the RMSE values of different models across each dataset, while Tables 7–9 present the MAE values of different models on each dataset. Additionally, corresponding experimental results are visually depicted in Figs. 5–7. These experimental findings offer valuable insights:

- Obviously, for HRSA, SVD, DP-MF, DP-CF, Item-Agrec and DP-DAE, values of RMSE and MAE over FilmTrust, Movielens-1M and Movielens-10M will decrease gradually with the increasing volume of training data (70%, 80%, 90%). For DP-MF and SVD, the performance improvements are relatively small.
- In most cases, autoencoder-based methods, such as HRSA, Item-Agrec, and DP-DAE, outperform other comparative models, indicating the powerful learning capabilities of autoencoder models in capturing the latent features.
- DP-DAE significantly outperforms matrix factorization-based models, such as DP-MF and SVD. This indicates that the proposed deep learning-based model yields significant advantages in learning the latent features for users and items.
- In contrast to HRSA and Item-Agrec, which are also autoencoder-based, DP-DAE can yield significant improvements. This indicates that the hybrid method of matrix factorization and autoencoder, as employed in DP-DAE, effectively captures and learns features between users and items, thereby enhancing the quality of recommendations.
- Under the same privacy budget, privacy-preserving models, such as DP-MF, DP-CF, and DP-DAE, exhibit different levels of performance. Notably, DP-DAE outperforms the other two privacy-preserving models, indicating the superiority of DP-DAE.

Overall, traditional matrix factorization methods perform well when the training data volume is small. However, as the training data volume increases, DP-DAE demonstrates the most outstanding performance across each dataset. For example, values of RMSE and MAE for DP-DAE are 0.8130 and 0.6221 respectively over FilmTrust, when 90% of the dataset is selected for training. Additionally, while compared to the advanced Item-Agrec model, DP-DAE yields the most significant performance improvement of 2.21% in terms of RMSE when the training set proportion is set to 80%. This further highlights the superiority of integrating matrix factorization and autoencoder in learning latent representation for DP-DAE.

Table 3
The structure of benchmark models and parameter settings.

Model	Model structure	Optimizer	Parameters settings
HRSA (Zhang et al., 2017b)	Hybrid recommendation model integrating semi-autoencoder and collaborative filtering together.	SGD	Learning rate $\eta = 0.005$. Parameter for regularization term is 0.02.
SVD (Guo et al., 2015)	A mathematical method for matrix decomposition.	SGD	Learning rate $\eta = 0.005$. Parameter for regularization term is 0.02.
DP-MF (Hua et al., 2015)	Combines matrix factorization techniques with differential privacy technology.	SGD	Learning rate $\eta = 0.005$. Parameter for regularization term is 0.02.
DP-CF (Yang et al., 2017)	Injecting differential privacy-preserving noise into collaborative filtering model.	SGD	Learning rate $\eta = 0.005$. Parameter for regularization term is 0.02.
Item-Agrec (Dong et al., 2021)	Hybrid collaborative recommendation of co-embedded item attributes and graph features.	SGD	Learning rate $\eta = 0.005$. Parameter for regularization term is 0.02.
DP-DAE	The dual semi-autoencoder model is shown in Fig. 4.	DPSGD	Learning rate $\eta = 0.005$. Parameter for regularization term is 0.02.

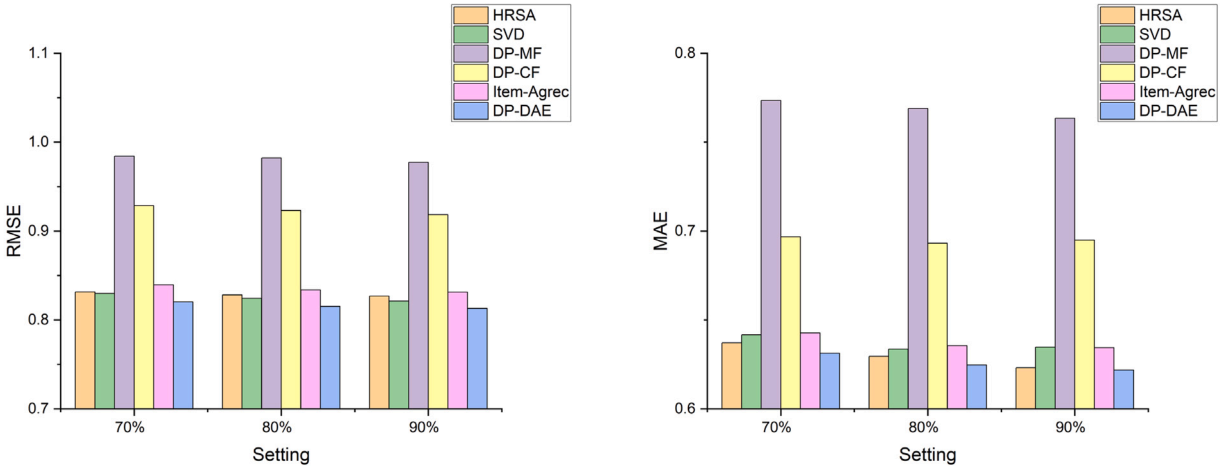


Fig. 5. Experimental results for HRSA, SVD, DP-MF, DP-CF, Item-Agrec, DP-DAE on FilmTrust.

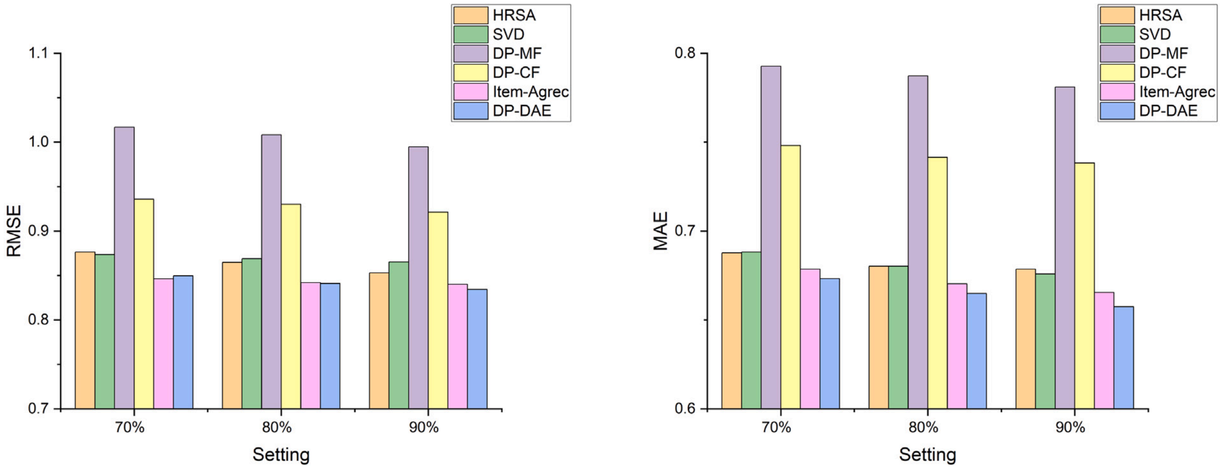


Fig. 6. Experimental results for HRSA, SVD, DP-MF, DP-CF, Item-Agrec, DP-DAE on Movielens-1M.

Table 4
Values of RSME for HRSA, SVD, DP-MF, DP-CF, Item-Agrec and DP-DAE over FilmTrust.

Setting	HRSA	SVD	DP-MF	DP-CF	Item-Agrec	DP-DAE
70%	0.8314	0.8296	0.9842	0.9286	0.8395	0.8203
80%	0.8281	0.8243	0.9822	0.9232	0.8337	0.8152
90%	0.8267	0.8211	0.9771	0.9184	0.8313	0.8130

Table 5
Values of RSME for HRSA, SVD, DP-MF, DP-CF, Item-Agrec and DP-DAE over Movielens-1M.

Setting	HRSA	SVD	DP-MF	DP-CF	Item-Agrec	DP-DAE
70%	0.8762	0.8735	1.0167	0.9358	0.8463	0.8497
80%	0.8647	0.8688	1.0083	0.9301	0.8418	0.8411
90%	0.8531	0.8653	0.9946	0.9213	0.8401	0.8343

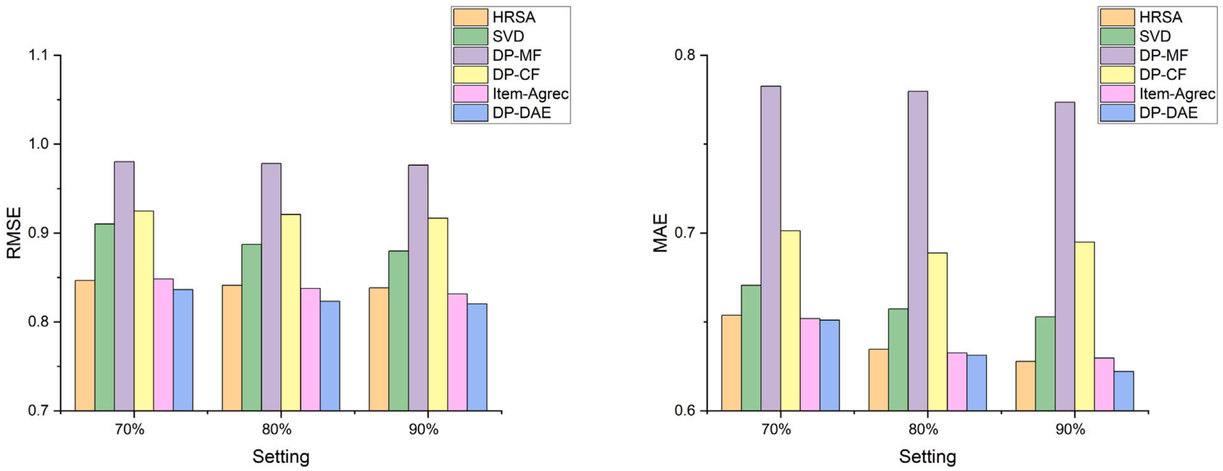


Fig. 7. Experimental results for HRSA, SVD, DP-MF, DP-CF, Item-Agrec, DP-DAE on Movielens-10M.

Table 6
Values of RSME for HRSA, SVD, DP-MF, DP-CF, Item-Agrec and DP-DAE over Movielens-10M.

Setting	HRSA	SVD	DP-MF	DP-CF	Item-Agrec	DP-DAE
70%	0.8466	0.9103	0.9801	0.9247	0.8483	0.8363
80%	0.8412	0.8872	0.9781	0.9208	0.8377	0.8231
90%	0.8383	0.8798	0.9763	0.9168	0.8316	0.8203

Table 7
Values of MAE for HRSA, SVD, DP-MF, DP-CF, Item-Agrec, DP-DAE on FilmTrust.

Setting	HRSA	SVD	DP-MF	DP-CF	Item-Agrec	DP-DAE
70%	0.6371	0.6417	0.7734	0.6967	0.6427	0.6313
80%	0.6296	0.6336	0.7689	0.6931	0.6356	0.6247
90%	0.6231	0.6348	0.7634	0.6949	0.6344	0.6219

Table 8
Values of MAE for HRSA, SVD, DP-MF, DP-CF, Item-Agrec, DP-DAE on Movielens-1M.

Setting	HRSA	SVD	DP-MF	DP-CF	Item-Agrec	DP-DAE
70%	0.6877	0.6882	0.7927	0.7481	0.6785	0.6732
80%	0.6802	0.6802	0.7873	0.7414	0.6703	0.6648
90%	0.6785	0.6758	0.7810	0.7383	0.6654	0.6574

Table 9
Values of MAE for HRSA, SVD, DP-MF, DP-CF, Item-Agrec, DP-DAE on Movielens-10M.

Setting	HRSA	SVD	DP-MF	DP-CF	Item-Agrec	DP-DAE
70%	0.6538	0.6706	0.7826	0.7013	0.6518	0.6509
80%	0.6347	0.6573	0.7797	0.6887	0.6326	0.6313
90%	0.6279	0.6528	0.7736	0.6949	0.6297	0.6221

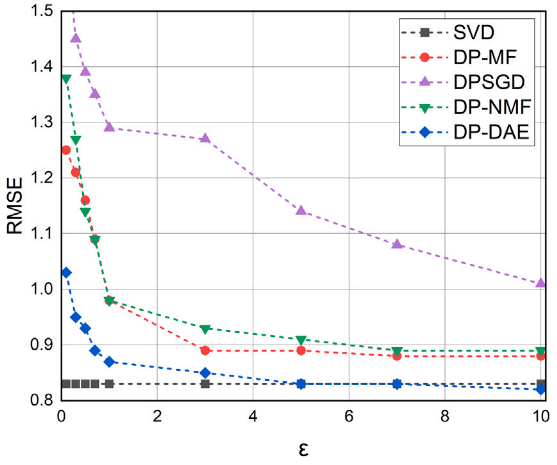


Fig. 8. RMSE of SVD, DP-MF, DPSGD, DP-NMF, DP-DAE on FilmTrust with varying privacy budgets.

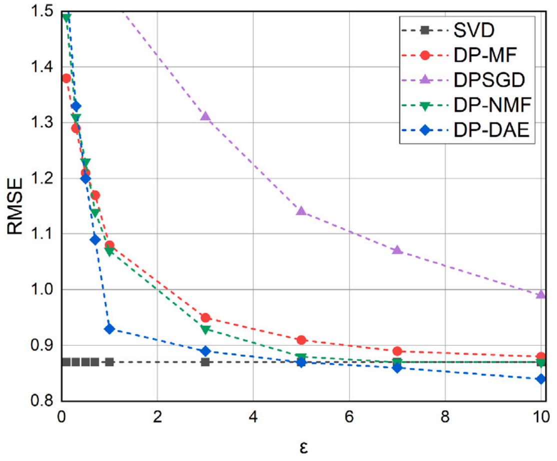


Fig. 9. RMSE of SVD, DP-MF, DPSGD, DP-NMF, DP-DAE on Movielens-1M with varying privacy budgets.

4.4. The impacts of privacy budget

When the privacy budget is small, the probability difference between the differential privacy algorithms on two datasets D and D' decreases, indicating that the outputs will be difficult to distinguish. This suggests that a small privacy budget will result in strong privacy protection. It is particularly important to note that when $\epsilon = 0$, the probability distributions of D and D' are completely indistinguishable, indicating the highest level of privacy strength but also rendering the original data unusable. Therefore, selecting an appropriate ϵ value is crucial.

We will investigate the recommendation performance of each model by setting different privacy budgets and dimensions of latent space. We use SVD, DPSGD, DP-MF, and DP-NMF (Ran et al., 2022b) as benchmark models for comparison. We set the privacy budget from 0.1 to 10 and conduct experiments for each recommendation algorithm under different privacy budgets.

With the training data sample proportion of 70%, the experimental results in Figs. 8 and 9 indicate that:

Table 10
The impacts of varying hidden layer dimensions on the FilmTrust (in terms of RMSE).

Dimensions	AE	DP-AE	DP-DAE
50	0.91	0.98	0.96
150	0.89	0.91	0.89
250	0.86	0.87	0.85
350	0.86	0.84	0.82
450	0.85	0.84	0.82

Table 11
The impacts of varying hidden layer dimensions on the Movielens-1M (in terms of RMSE).

Dimensions	AE	DP-AE	DP-DAE
50	0.93	0.97	0.93
150	0.89	0.94	0.91
250	0.88	0.90	0.87
350	0.88	0.87	0.86
450	0.87	0.86	0.85

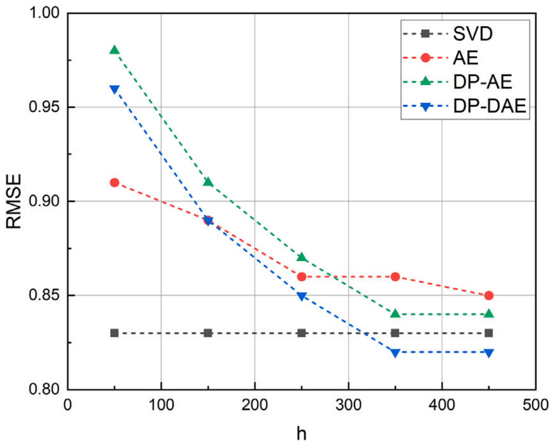


Fig. 10. Investigation on impacts of dimensionality h on FilmTrust.

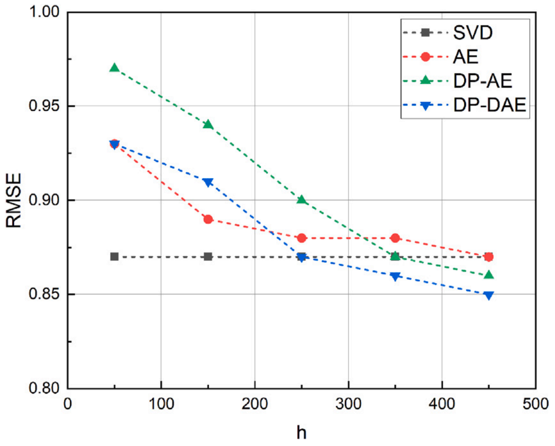


Fig. 11. Investigation on impacts of dimensionality h on Movielens-1M.

- A privacy budget of 0.1 leads to significant interference when introducing a large amount of noise into the matrix, resulting in poor recommendations for all models with added noise.
- Obviously, the prediction errors of SVD, DPSGD, DP-MF, DP-NMF and DP-DAE decrease as ϵ increasing. Lower privacy budgets imply stronger privacy protection, albeit at the cost of accuracy. Since DPSGD injects noise at each iteration, its RMSE remains relatively high, and it decreases with the increasing privacy budget. However, the curve of RMSE for DPSGD does not stabilize.
- Across different privacy budgets, all methods yield a little bigger RMSE values on FilmTrust, due to the small size of training data, which prevents the models from fully learning the latent features. When the dataset is the larger Movielens-1M, the convergence speed of each model noticeably increases.
- Notably, DP-DAE, which combines deep learning and matrix factorization, performs exceptionally well, further underscoring the superiority of DP-DAE in latent feature learning.

In summary, DP-DAE outperforms other comparative models, highlighting the advantage of deep learning models in latent feature learning. Additionally, noise injected at lower privacy budgets significantly affects the performance of all differentially private models.

4.5. The impacts of hidden layer

The hidden layer is one crucial factor affecting the performance of autoencoder models. The size of the layers directly determines the complexity of the model and its ability to extract information. The model's ability to extract information may be limited with several layers, however, excessively deep dimensions may lead to overfitting.

To further investigate the impact of hidden layer on model performance, we select SVD, DP-AE (Liu et al., 2019) and traditional autoencoder-based recommendation systems (AE) as the compared models. The dimensionality of hidden layers varies from 50 to 450 for datasets Filmtrust and Movielens-1M. The curves of experimental results are illustrated in Figs. 10 and 11, and the experimental results are presented in Tables 10 and 11. Note that the proportion of training samples is set to 70%. It is noteworthy that in Tables 10 and 11, due to no impacts on the experimental results of SVD, the experimental results are not presented in the tables. The experimental data for SVD over FilmTrust and Movielens-1M are 0.87 and 0.83, respectively.

Traditional autoencoder-based recommendation systems perform excellently in contrast to privacy-protected recommendation systems, when the hidden layer dimensions are low. As the number of hidden layers increasing, privacy-protected recommendation systems like DP-AE and DP-DAE, gradually improve the performance. DP-DAE, integrating matrix factorization techniques into autoencoders, exhibits significantly powerful latent representation learning capabilities while compared to traditional autoencoder-based recommendation systems and the aforementioned DP-AE. Moreover, the performance of AE, DP-AE and DP-DAE tends to stabilize with the increasing hidden layer depths. When the dimensionality of the hidden layer is set to 450, the RMSE values for DP-DAE could reach 0.82 and 0.85 over FilmTrust and Movielens-1M respectively, which are a little smaller than that of other methods.

In summary, DP-DAE can outperform other privacy-protected recommendation systems and traditional autoencoder methods. This further demonstrates that DP-DAE exhibits excellent recommendation performance while providing sufficient privacy protection.

5. Conclusion and future work

In order to enhance the ability to suggest relevant items to users for recommender systems, it is essential to acquire latent representations of user–item interactions. In comparison to conventional matrix factorization methods, the proposed framework DP-DAE owns superior advantages in feature extraction. We integrate matrix factorization and Dual Semi-autoencoder architecture together, enabling the extraction of two distinct matrices representing user and item information. Subsequently, we incorporate pertinent user and item data into the respective matrices, serving as inputs to two separate autoencoders. This innovative approach effectively mitigates the challenges associated with data sparsity. By employing matrix factorization for dimensionality reduction, DP-DAE not only captures hidden features between users and items more effectively but also reduces computational complexity, enabling the model to perform item recommendation admirably while ensuring user privacy protection.

Furthermore, to evaluate DP-DAE on datasets FilmTrust, Movielens-1M and Movielens-10M, we partition each dataset based on the number of item ratings, and employ HRSA, SVD, DP-MF, DP-CF and Item-Agrec as benchmark methods for comparison. The experimental results conclusively reveal the proposed model's exceptional performance in each dataset, thereby showcasing its capability in handling real-life datasets.

As future work, we will explore the adjustment of additional information volume and privacy budget, to make the model applicable to large-scale recommendation applications, moreover, we will also try to further reduce the computational cost for the model.

CRediT authorship contribution statement

Yang Deng: Investigation, Data curation, Formal analysis, Writing – original draft. **Wang Zhou:** Conceptualization, Resources, Software, Project administration. **Amin Ul Haq:** Methodology, Conceptualization, Formal analysis. **Sultan Ahmad:** Validation, Formal analysis. **Alia Tabassum:** Formal analysis, Resources.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work is partially supported by Guangdong Provincial Key Laboratory of Novel Security Intelligence Technologies, China (2022B1212010005), Intelligent Policing Key Laboratory of Sichuan Province, China (No. ZNJW2024KFQN008), and Sichuan Science and Technology Program, China (2023YFH0058).

References

Aghdam, M. H., Analoui, M., & Kabiri, P. (2017). Collaborative filtering using non-negative matrix factorisation. *Journal of Information Science*, *43*(4), 567–579. http://dx.doi.org/10.1177/0165551516654354.

Beg, S., Anjum, A., Ahmad, M., Hussain, S., Ahmad, G., Khan, S., & Choo, K.-K. R. (2021). A privacy-preserving protocol for continuous and dynamic data collection in IoT enabled mobile app recommendation system (MARS). *Journal of Network and Computer Applications*, *174*, Article 102874. http://dx.doi.org/10.1016/j.jnca.2020.102874.

Chen, J., Liu, L., Chen, R., Peng, W., & Huang, X. (2021). SecRec: A privacy-preserving method for the context-aware recommendation system. *IEEE Transactions on Dependable and Secure Computing*, *19*(5), 3168–3182. http://dx.doi.org/10.1109/TDSC.2021.3085562.

Chen, Z., Wang, Y., Zhang, S., Zhong, H., & Chen, L. (2021). Differentially private user-based collaborative filtering recommendation based on k-means clustering. *Expert Systems with Applications*, *168*, Article 114366. http://dx.doi.org/10.1016/j.eswa.2020.114366.

Chen, T., Yin, H., Ye, G., Huang, Z., Wang, Y., & Wang, M. (2020). Try this instead: Personalized and interpretable substitute recommendation. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval* (pp. 891–900). http://dx.doi.org/10.1145/3397271.3401042.

Dong, B., Zhu, Y., Li, L., & Wu, X. (2021). Hybrid collaborative recommendation of co-embedded item attributes and graph features. *Neurocomputing*, *442*, 307–316. http://dx.doi.org/10.1016/j.neucom.2021.01.129.

Ermis, B., & Cemgundefinedl, A. T. (2020). Data sharing via differentially private coupled matrix factorization. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *14*(3), 1–27. http://dx.doi.org/10.1145/3372408.

Fang, L., Du, B., & Wu, C. (2022). Differentially private recommender system with variational autoencoders. *Knowledge-Based Systems*, *250*, Article 109044. http://dx.doi.org/10.1016/j.knosys.2022.109044.

Fredrikson, M., Jha, S., & Ristenpart, T. (2015). Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security* (pp. 1322–1333). http://dx.doi.org/10.1145/2810103.2813677.

Gao, C., Huang, C., Lin, D., Jin, D., & Li, Y. (2020). DPLCF: Differentially private local collaborative filtering. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval* (pp. 961–970). http://dx.doi.org/10.1145/3397271.3401053.

Guo, G., Zhang, J., & Yorke-Smith, N. (2016). A novel recommendation model regularized with user trust and item ratings. *IEEE Transactions on Knowledge and Data Engineering*, *28*(7), 1607–1620. http://dx.doi.org/10.1109/TKDE.2016.2528249.

Guo, Q., Zhang, C., Zhang, Y., & Liu, H. (2015). An efficient SVD-based method for image denoising. *IEEE Transactions on Circuits and Systems for Video Technology*, *26*(5), 868–880. http://dx.doi.org/10.1109/TCSVT.2015.2416631.

Harrouche, O., & Yamina, M. B. A. (2024). Recommender system based on convolutional recurrent deep learning for protein-drug interaction prediction. *Expert Systems with Applications*, *245*, http://dx.doi.org/10.1016/j.eswa.2023.123090.

Hu, H., Dobbie, G., Salcic, Z., Liu, M., & Zhang, X. (2021). Differentially private locality sensitive hashing based federated recommender system. *Concurrency and Computation Practice and Experience*, *35*, http://dx.doi.org/10.1002/cpe.6233.

Hua, J., Xia, C., & Zhong, S. (2015). Differentially private matrix factorization. In *Proceedings of the 24th international conference on artificial intelligence* (pp. 1763–1770).

Jannach, D., & Ludewig, M. (2017). When recurrent neural networks meet the neighborhood for session-based recommendation. In *Proceedings of the eleventh ACM conference on recommender systems* (pp. 306–310). http://dx.doi.org/10.1145/3109859.3109872.

Jiang, H., Yu, H., Cheng, X., Pei, J., Pless, R., & Yu, J. (2023). Dp2-pub: Differentially private high-dimensional data publication with invariant post randomization. *IEEE Transactions on Knowledge and Data Engineering*, *35*(10), 10831–10844. http://dx.doi.org/10.1109/TKDE.2023.3265605.

Karamanolakis, G., Cherian, K. R., Narayan, A. R., Yuan, J., Tang, D., & Jebara, T. (2018). Item recommendation with variational autoencoders and heterogeneous priors. In *Proceedings of the 3rd workshop on deep learning for recommender systems* (pp. 10–14). http://dx.doi.org/10.1145/3270323.3270329.

Li, X., & She, J. (2017). Collaborative variational autoencoder for recommender systems. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 305–314). http://dx.doi.org/10.1145/3097983.3098077.

Li, L., Xiahou, J., Lin, F., & Su, S. (2023). DistVAE: Distributed variational autoencoder for sequential recommendation. *Knowledge-Based Systems*, *264*, Article 110313. http://dx.doi.org/10.1016/j.knosys.2023.110313.

Liang, D., Krishnan, R. G., Hoffman, M. D., & Jebara, T. (2018). Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 world wide web conference* (pp. 689–698). http://dx.doi.org/10.1145/3178876.3186150.

Liu, R., Cao, Y., Wang, Y., Lyu, L., Chen, Y., & Chen, H. (2023). Privaterec: Differentially private model training and online serving for federated news recommendation. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining* (pp. 4539–4548). http://dx.doi.org/10.1145/3580305.3599889.

Liu, X., Li, Q., Ni, Z., & Hou, J. (2019). Differentially private recommender system with autoencoders. In *2019 international conference on internet of things (iThings) and IEEE green computing and communications (greenCom) and IEEE cyber, physical and social computing (cPSCom) and IEEE smart data (smartData)* (pp. 450–457).

Neera, J., Chen, X., Aslam, N., Wang, K., & Shu, Z. (2021). Private and utility enhanced recommendations with local differential privacy and Gaussian mixture model. *IEEE Transactions on Knowledge and Data Engineering*, *35*(4), 4151–4163. http://dx.doi.org/10.1109/TKDE.2021.3126577.

Ran, X., Wang, Y., Zhang, L. Y., & Ma, J. (2022a). A differentially private matrix factorization based on vector perturbation for recommender system. *Neurocomputing*, *483*, 32–41. http://dx.doi.org/10.1016/j.neucom.2022.01.079.

Ran, X., Wang, Y., Zhang, L. Y., & Ma, J. (2022b). A differentially private nonnegative matrix factorization for recommender system. *Information Sciences*, *592*, 21–35. http://dx.doi.org/10.1016/j.ins.2022.01.050.

Ribero, M., Henderson, J., Williamson, S., & Vikalo, H. (2022). Federating recommendations using differentially private prototypes. *Pattern Recognition*, *129*, Article 108746. http://dx.doi.org/10.1016/j.patcog.2022.108746.

Shin, H., Kim, S., Shin, J., & Xiao, X. (2018). Privacy enhanced matrix factorization for recommendation with local differential privacy. *IEEE Transactions on Knowledge and Data Engineering*, *30*(9), 1770–1782. http://dx.doi.org/10.1109/TKDE.2018.2805356.

Smith, B., & Linden, G. (2017). Two decades of recommender systems at Amazon.com. *IEEE Internet Computing*, *21*(3), 12–18. http://dx.doi.org/10.1109/MIC.2017.72.

Song, X., Peng, Z., Song, S., & Stojanovic, V. (2024). Anti-disturbance state estimation for PDT-switched RDNNs utilizing time-sampling and space-splitting measurements. *Communications in Nonlinear Science and Numerical Simulation*, *132*, Article 107945. http://dx.doi.org/10.1016/j.cnsns.2024.107945.

Wu, Y., DuBois, C., Zheng, A. X., & Ester, M. (2016). Collaborative denoising auto-encoders for top-n recommender systems. In *Proceedings of the 9th ACM international conference on web search and data mining* (pp. 153–162). http://dx.doi.org/10.1145/2835776.2835837.

Yang, H. (2017). Bayesian heteroscedastic matrix factorization for conversion rate prediction. In *Proceedings of the 2017 ACM on conference on information and knowledge management* (pp. 2407–2410). http://dx.doi.org/10.1145/3132847.3133076.

Yang, Y., Zhu, Y., & Li, Y. (2022). Personalized recommendation with knowledge graph via dual-autoencoder. *Applied Intelligence: The International Journal of Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies*, *52*(6), 6196–6207. http://dx.doi.org/10.1007/s10489-021-02647-1.

Yang, M., Zhu, T., Ma, L., Xiang, Y., & Zhou, W. (2017). Privacy preserving collaborative filtering via the Johnson-Lindenstrauss transform. In *2017 IEEE trustcom/bigDataSE/ICESS* (pp. 417–424). http://dx.doi.org/10.1109/Trustcom/BigDataSE/ICESS.2017.266.

Yi, B., Shen, X., Zhang, Z., Shu, J., & Liu, H. (2016). Expanded autoencoder recommendation framework and its application in movie recommendation. In *2016 10th international conference on software, knowledge, information management & applications* (pp. 298–303). http://dx.doi.org/10.1109/SKIMA.2016.7916236.

Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys*, *52*(1), 1–38. http://dx.doi.org/10.1145/3285029.

Zhang, S., Yao, L., & Xu, X. (2017). Autosvd++ an efficient hybrid collaborative filtering model via contractive auto-encoders. In *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval* (pp. 957–960). http://dx.doi.org/10.1145/3077136.3080689.

Zhang, S., Yao, L., Xu, X., Wang, S., & Zhu, L. (2017). Hybrid collaborative recommendation via semi-autoencoder. In *24th international conference on neural information processing* (pp. 185–193). http://dx.doi.org/10.1007/978-3-319-70087-8_20.

Zhao, Y., & Chen, J. (2022). A survey on differential privacy for unstructured data content. *ACM Computing Surveys*, *54*, 1–28. http://dx.doi.org/10.1145/3490237.

Zheng, K., Cai, T., Huang, W., Li, Z., & Wang, L. (2020). Locally differentially private (contextual) bandits learning. *Advances in Neural Information Processing Systems*, *33*, 12300–12310. http://dx.doi.org/10.1016/j.eswa.2023.121687.

Zhou, W., Haq, A. U., Qiu, L., & Akbar, J. (2023). Multi-view social recommendation via matrix factorization with sub-linear convergence rate. *Expert Systems with Applications*, *237*, http://dx.doi.org/10.1016/j.eswa.2023.121687.