Overview of Federated Recommender Systems

Overview of Federated Recommender Systems (FedRec)

- **Importance of Recommender Systems**: They play a crucial role in managing information overload and are significant in various business models globally. These systems are more effective when they have access to substantial user data.
- **Data Privacy and Security Concerns**: Centralization of user data in Recommender Systems (RecSys) leads to privacy and security risks, and compliance with regulations like GDPR is necessary. Traditional methods for protecting user data, such as obfuscation or cryptography, often still require data to leave local repositories.
- **FedRec Introduction**: FedRec is introduced as a decentralized approach to RecSys that maintains user privacy and data security. It involves local data storage at each party, with only intermediate results like parameter updates being communicated.

Definition and Categorization

- **FedRec Definition**: Defined as a collaboration among multiple parties to train recommendation models without direct access to each other's private data. It involves optimizing a loss function representing the recommendation accuracy.
- **Performance Expectation**: FedRec aims to perform better than individual RecSys using their own data and close to the performance of aggregating all data without considering privacy.
- **Categorization**: FedRec is categorized based on data structure into Horizontal FedRec, Vertical FedRec, and Transfer FedRec, considering how users and items are shared among different parties.

Specific Types of FedRec

Horizontal FedRec: Involves **shared items but different users between parties**. A typical scenario includes personalized movie recommendations where user data is kept private and distributed locally on devices

- **Principle**: Items are shared across parties, but users are different. Each user device is considered a separate party.
- **Federated Collaborative Filter (FCF)**: Based on matrix factorization, FCF uses a central server for item latent factors and local storage for user latent factors. The server aggregates item latent factor updates from each party. This approach preserves privacy as only model updates are shared, not user data.
- **Decentralized Matrix Factorization**: Avoids central servers; parties directly communicate for model updates. Improves algorithm performance and maintains user privacy.
- **Federated Meta-Learning Framework**: Treats each user's recommendation as a separate task, designs a meta-learner to generate each task parameters.one framework utilizes a support set to generate the recommendation model on each party and computes the loss gradient on a query set.Another framework needs no separate support and
  query sets. The latter one performs relatively well within considerably fewer episodes in the experiments.
- **Obfuscation Methods**:

  The obfuscation methods contain the anonymization, the randomization, and the differential privacy techniques. Among them, the **differential privacy (DP) technique** is a popular method(e.x. private social recommendation (PrivSR)). It incorporates random noise to anonymize data and protect privacy. It offers low computation costs,These works involve a trade-off between performance and privacy.
- **Cryptography Techniques:**

  The cryptography methods contain homomorphic encryption (HE), secure multi-party computation (SMC) protocols,etc.They guarantee good security protection without the loss of accuracy.

  HE techniques have been widely utilized because it allows computing over encrypted data without access to the secret key.(federated matrix factorization with HE schemes(FedMF), Each user encrypts the item latent factor updates with HE before transmitting. Besides, the item latent factor is aggregated and maintained by the central server under the encrypted form.)

SMC protocol is designed to compute the summation of private values of each party without revealing them.Then with this protocol, the PrivateCosine and PrivatePearson algorithm are implemented to calculate the item correlations. Final recommendations are generated using the correlations without revealing privacy

Vertical Federated Recommender System:(類似CDR)

- **Principle**: Different recommenders or data providers share the same user set but have different item sets or feature spaces.
- **Method**: Utilizes side information from one party to enhance the recommender model of another, ensuring privacy during training. Example methods include …
- asynchronous stochastic gradient descent: Each party could use an arbitrary model to map its local features to a local prediction. Then local predictions from different parties are aggregated into a final output using linear and nonlinear transformations.The training procedure of each party is allowed to be at various iterations up to a bounded delay. This approach does not share any raw data and local models. Therefore, it has fewer privacy risks.
- DP technique
- cryptography techniques:

  secure gradient-tree boosting using homomorphic encryption:adopts HE methods to provide lossless performance as well as preserving privacy.

  secure linear regression with multi-party computation protocols(MPC) which are designed using garbled circuits to obtain a highly scalable solution.

Transfer FedRec: Neither users nor items are shared between parties. It involves transferring knowledge from a source-domain party to a target-domain party, using a limited set of co-occurrence samples as a bridge for knowledge transfer.

- **Principle**: Neither users nor items are shared between parties. It's typically used between different recommender systems.
- federated transfer learning：A limited set of co-occurrence samples is used as a “bridge” to transfer knowledge from the source domain to the target domain. At first, parties update their neural networks using local data. Then, they together optimize the loss on the co-occurrence samples.
- The secret sharing technique is adopted to design a secure and efficient algorithm. Similarly, this algorithm can be applied in the transfer FedRec scenario via co-occurrence users or items.

horizontal FedRec managing RecSys across individuals or user sets is important and attracts lots of research attention. Vertical
FedRec and transfer FedRec building RecSys among organizations are typical
tasks in recommendation businesses. Yet, vertical and transfer FedRec are still underexplored areas with a lot of opportuni

Challenges and Solutions

- **Privacy Protection Techniques**: Include differential privacy (DP) and cryptography methods like homomorphic encryption (HE) and secure multi-party computation (SMC). DP adds noise to anonymize data, whereas HE and SMC provide secure computation without data disclosure.
- **Trade-offs**: These techniques involve balancing between privacy protection and algorithm performance. DP adds noise which might affect performance, while HE and SMC maintain performance but are more complex to implement
- **Challenges in Construction**: FedRec faces challenges at both the algorithm and system levels, including designing recommendation algorithms and comprehensive system design.
- **Specific Model Challenges**: Challenges exist in applying deep models, graph models, and reinforcement learning models to FedRec, due to limitations like non-linear function support and privacy protection of structure information.
- **Issue of Malicious Participants**: Addressing untrustworthy parties in the system is crucial. Solutions like DeepChain use Blockchain technology to incentivize correct

behavior and preserve privacy.

- **Algorithm-Level Challenges:**
  1. **Federated Deep Model for Recommendation**:
     - **Challenge**: Deep learning models often use non-linear activation functions like tanh and ReLU, which are not well supported by Homomorphic Encryption (HE). HE is crucial in FedRec for ensuring privacy during computation.
     - **Solution**: To address this, one approach uses low-degree polynomial approximations for these activation functions. This method provides a balance between maintaining model performance and the computational limitations of HE. The goal is to use polynomial approximations with the lowest possible degrees for common activation functions such as ReLU, Sigmoid, and Tanh, thereby optimizing both performance and privacy.
  2. **Federated Graph Model for Recommendation**:
     - **Challenge**: In graph-based recommendation models, the main difficulty lies in preserving the privacy of structural information within the graph. These models use user-item relationship data to enrich their recommendations, which is complex and sensitive.
     - **Solution**: A method uses graph sampling to enhance both efficiency and privacy in association rules mining. This approach allows users to control their participation in the sample and maintain privacy over their item sets. It represents users with common interests as groups, without exposing individual user's specific item sets to the recommender or other users.Efficient privacy-preserving recommendations based on social graphs | Proceedings of the 13th ACM Conference on Recommender Systems
  3. **Federated Reinforcement Learning Model for Recommendation**:
     - **Challenge**: Adapting reinforcement learning models for FedRec involves designing the state, action, and reward mechanisms to capture real-time user interests while deciding what information to share among parties. Its application in FedRec is less explored compared to other domains.
     - **Solution**: An example in a different context is the lifelong federated reinforcement learning architecture used for robots （not for

recommendation）. This approach involves a **knowledge fusion algorithm** and a **transfer learning approach**, allowing robots to combine prior knowledge and adapt quickly to new environments. It illustrates how federated learning can be applied to dynamically learn and adjust in response to changing data or environments, a principle that can be adapted for FedRec.

FedDSR: Daily Schedule *Recommendation* in
a *Federated* Deep *Reinforcement Learning* Framework

- **System Level Challenges**: These include designing privacy-preserving recall and ranking procedures with real-time feedback, managing communication costs, ensuring flexibility and scalability, and addressing the non-IID data problem.

  1. **Design of Recall and Ranking**:

     - **Challenge**: Implementing privacy-preserving recall and ranking procedures with real-time feedback is complex.RecSys sequentially adopts these two procedures to obtain the final recommendations. Traditionally, these processes are centralized, but FedRec requires a decentralized approach to protect user privacy.

     - **Solutions**:

       - **Server-Side Recall and Participant-Side Ranking**: Parties send encrypted model parameters to the server for recall, and ranking is done locally. Privacy stream searching techniques are used to prevent privacy leaks.

       - **Participant-Side Recall and Ranking**: All data processing is done locally, eliminating privacy leaks but increasing communication costs and computational needs. Advancements in technologies like 5G may help mitigate these costs.

  2. **Communication Cost**:

     - **Issue**: High communication cost is a significant challenge in FedRec due to high-dimensional features and real-time processing requirements.

     - **Solution**: Communication-mitigated federated learning (CMFL) techniques compress data and filter out irrelevant updates to reduce communication overhead. They provide feedback to clients about global model trends to ensure relevant updates.

3. **Flexibility and Scalability**:
   - **Problem**: The synchronous client-server architecture commonly used in federated learning systems is not scalable or flexible enough for systems with many participants.
   - **Solutions**:
     - **Asynchronous Federated Optimization**: Updates the global model immediately upon receiving local updates, allowing non-blocking communication.
     - **Gossip Learning Algorithm**: A fully decentralized approach where parties communicate directly with each other, enhancing scalability.
4. **Non-IID Data**:
   - **Challenge**: Non-identically distributed (non-IID) data across different parties leads to performance degradation in FedRec.
   - **Solution**: A data-sharing strategy that distributes a uniform global dataset to all parties for initial training, followed by local training with both shared and private data.
5. **Malicious Participant Cooperation**:
   - **Risk**: Parties in FedRec may not always be trustworthy, potentially leading to privacy leaks and incorrect model updates.
   - **Solution**: DeepChain uses blockchain technology to incentivize correct behavior. It ensures the privacy of local gradients and the auditability of the training process using smart contracts.