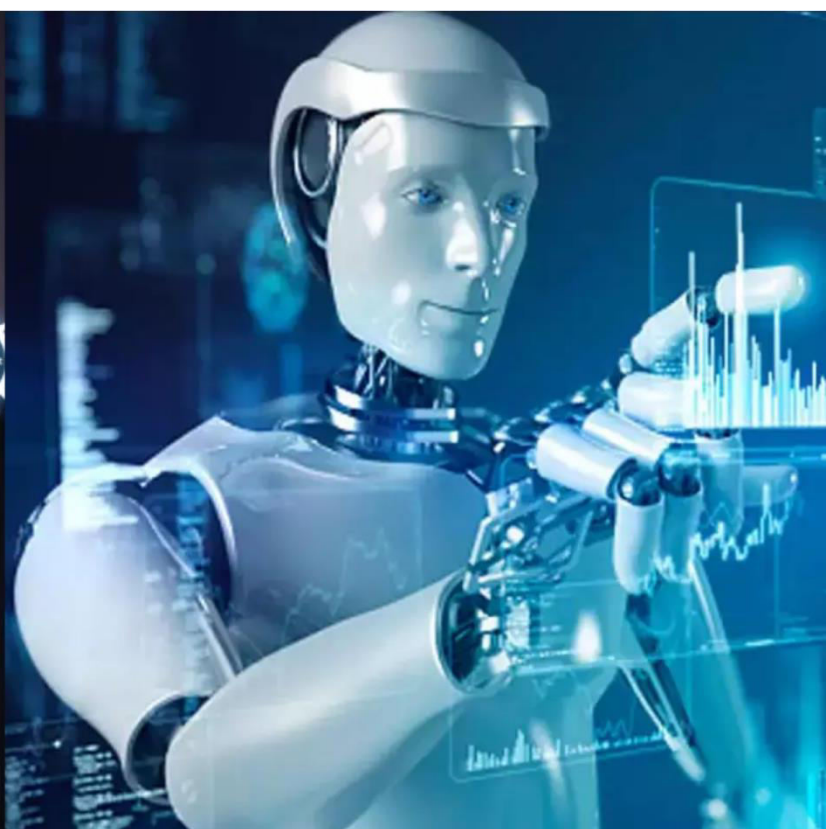




International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.935

Volume 15, Issue 10, October 2026

Adaptive Cross-Domain Recommendation Algorithms using Customer Interaction in Machine Learning: A Privacy-Aware LLM-Sequential Framework (PALS-CDR)

Dr R. Nithya, Maragathavalli K

Assistant Professor, School of Computing Science, KPR College of Arts Science and Research, Coimbatore, India

Research Scholar, KPR College of Arts Science and Research, Coimbatore, India

ABSTRACT: PALS-CDR is a cross-domain recommendation framework that transfers a customer's interaction patterns between domains while keeping raw behavioral data inside each domain. It combines three ideas: a sequential Transformer encoder that models each customer's ordered interactions, a large language model (LLM) that turns item text into a domain-agnostic semantic space, and a differentially private exchange of user representations in place of raw logs. An adaptive gate decides, for every customer and every step, how much source-domain evidence to trust when predicting the next item in the target domain. This paper formalizes the problem, specifies the architecture and training objective, states the privacy guarantee obtained from clipped and noised representation sharing, and compares the framework with representative cross-domain, sequential, LLM-based and privacy-aware recommenders on design-level criteria, including what each method exposes across domains.

KEYWORDS: cross-domain recommendation, sequential recommendation, large language models, differential privacy, federated learning, cold-start, customer interaction modeling

I. INTRODUCTION

Customers rarely stay inside one product category, yet most recommenders learn from one domain at a time. A shopper's book purchases, movie ratings and electronics searches each reveal part of the same taste profile, but each lives in a separate log, often owned by a separate team or company. Cross-domain recommendation (CDR) tries to use the richer domain to help the sparser one, easing the data-sparsity and cold-start problems that limit single-domain models [1, 2].

Three gaps keep existing CDR methods from being deployable in practice.

1. **Static transfer.** Embedding-mapping and shared-parameter methods such as EMCDCR, CoNet and DDTCDR learn one transfer function for all users [3, 4, 5]. Yet the value of source-domain evidence differs by customer and changes over time: a recent book purchase may predict a movie choice for one user and be irrelevant for another.
2. **Weak use of interaction order.** Many CDR models treat history as an unordered set, discarding the sequence in which customers actually act. Sequential models such as GRU4Rec, SASRec and BERT4Rec show that order carries strong signal [6, 7, 8], and recent cross-domain sequential work confirms it [9, 10].
3. **Privacy exposure.** Transfer normally requires pooling interaction logs or exposing user embeddings across domains. This conflicts with data-protection rules and with the business reality that domains are often held by different parties. Federated and differentially private learning address this in general [11, 12, 13], but are seldom combined with modern sequential and language-model recommenders.

Large language models add a fourth opportunity. Item titles and descriptions give every domain a common vocabulary, so an LLM encoder can place a novel in one domain and a film in another in the same semantic space, without needing overlapping items [14, 15, 16]. The risk is that sending customer histories to an LLM would itself leak private data.

Design questions.

- RQ1: Can per-user, per-step gating of source-domain evidence replace a single static transfer function?
- RQ2: Can an LLM-derived semantic item space link domains and handle cold-start items without exposing user data?
- RQ3: What privacy guarantee can be stated for the cross-domain channel, and what exactly does it cover?
- RQ4: How does this design differ from existing cross-domain, sequential, LLM-based and privacy-aware methods?

Contributions.

- We propose PALS-CDR, a framework that unites sequential modeling, LLM-based item semantics and adaptive cross-domain fusion under an explicit privacy mechanism.
- We keep the LLM on item metadata only, so no customer history or identifier ever reaches it.
- We state the privacy guarantee for the representation-sharing channel and describe how the cumulative budget is tracked.
- We compare PALS-CDR with representative methods on design-level criteria: interaction order, text semantics, adaptive transfer, what crosses domains, and privacy guarantee.

The rest of the paper is organized as follows. Section 2 reviews related work, Section 3 formalizes the problem, Section 4 presents PALS-CDR, Section 5 compares PALS-CDR with existing methods, Section 6 covers challenges and future work, and Section 7 concludes.

II. LITERATURE REVIEW

Prior work falls into four lines. None of them combines adaptive transfer, sequence modeling, LLM semantics and formal privacy in one design, which is the gap PALS-CDR targets.

2.1 Cross-domain recommendation

Early CDR methods pool domains in a joint factorization: collective matrix factorization shares latent factors across relations [2]. Embedding-and-mapping approaches such as EMCDR train a model per domain and then learn a function that maps source-domain user embeddings into the target space, which works for users who appear in both domains [3]. Neural methods share structure during training: CoNet connects hidden layers of two networks through cross-connections [4], and DDTCDR transfers preferences in both directions with a learned latent orthogonal mapping [5]. Personalized transfer methods generate the mapping per user from that user's characteristics rather than using one function for everyone [30]. The survey in [1] organizes these methods by the transfer task (single-target, dual-target, multi-target) and by overlap assumption. Most of these models take an unordered view of history.

2.2 Sequential and cross-domain sequential recommendation

Sequential recommenders predict the next item from an ordered history. GRU4Rec uses recurrent networks over sessions [6]. SASRec applies self-attention to let each position attend to earlier items [7], and BERT4Rec trains a bidirectional Transformer with a masked-item objective [8]. Cross-domain sequential recommendation extends this to histories that span domains. π -Net passes information between two parallel sequence encoders for shared accounts [9], and contrastive cross-domain sequential recommendation aligns single-domain and cross-domain sequence representations with contrastive losses [10]. These methods learn from IDs, so they depend on item overlap or on enough data to learn item embeddings, and they assume the logs can be combined.

2.3 LLMs and text semantics in recommendation

Text gives items a representation that does not depend on interaction counts. UniSRec learns universal sequence representations from item text with adapters, so a model pre-trained on several domains transfers to new ones [14]. P5 casts recommendation tasks as language tasks inside one pre-trained sequence-to-sequence model [15], and TALLRec tunes an LLM on recommendation instructions with parameter-efficient methods [16]. Surveys describe a growing set of such methods and their open problems, including cost, position bias and privacy [17, 32]. Low-rank adaptation keeps tuning cheap by training small update matrices over frozen weights [20], and sentence encoders provide fixed-size text embeddings at low cost [19]. A shared limitation is that user histories are often serialized into prompts, which creates a data-exposure path.

2.4 Privacy-preserving recommendation

Federated learning trains a shared model while data stays on the clients [11], and federated collaborative filtering and secure federated matrix factorization apply it to recommendation [26, 27]. Differential privacy bounds how much any single record can change a released output [12]. Clipping and noising gradients gives a trainable guarantee for deep models [13], and Rényi differential privacy gives tighter accounting across many releases [22]. In CDR the sensitive object is the cross-domain channel itself, since user embeddings or gradients exchanged between domains can reveal behavior in the source domain.

2.5 Research gap

Line of work	Models order	Uses semantics text	Adaptive transfer per-user	Formal privacy on the transfer channel
Classical CDR [2, 3, 4, 5]	No	No	Limited	No
Cross-domain sequential [9, 10]	Yes	No	Limited	No
LLM and text-based [14, 15, 16]	Yes	Yes	No	No
Line of work	Models order	Uses semantics text	Adaptive transfer per-user	Formal privacy on the transfer channel
Federated recommendation [26, 27]	Varies	No	No	Partial
PALS-CDR (proposed)	Yes	Yes	Yes	Yes

The table summarizes the claim of this paper at the level of design goals; it is a design-level comparison, not an empirical one, and Section 5 extends it to named methods.

III. PROBLEM FORMULATION

The task is next-item prediction in a target domain, using the customer's target history and a privacy-protected summary of the same customer's source-domain history.

3.1 Notation

Two domains are considered, a source domain S and a target domain T ; the extension to more domains repeats the source side. Each domain $d \in \{S, T\}$ has its own user set U_d , item set I_d and interaction logs held by that domain's operator. The users present in both domains form the overlap set $U_o = U_S \cap U_T$. Every item i has a text description x_i (title, category, description).

For a user u , H_u^d denotes the chronologically ordered sequence of items the user interacted with in domain d , with n_d items in total.

3.2 Recommendation objective

Given H_u^T and the shared summary z_u^S released from the source domain, the model scores every candidate target item and ranks them. The training goal is to maximize the probability of the true next item at every position in every target user's history.

For users outside U_o no source summary exists, so z_u^S is replaced by a learned default vector. This makes cold-start users a first-class case, not an exception.

3.3 Privacy objective

The only information that crosses the domain boundary is the vector z_u^S . The mechanism M that produces it must satisfy (ϵ, δ) -differential privacy with respect to any single source-domain interaction record: meaning that changing one record in the source log changes the probability of any released output by at most a factor controlled by ϵ , plus a small slack δ .

Smaller ϵ means stronger protection. The design goal is to keep ranking quality high at a small ϵ , and the limits of that trade-off are discussed in Section 6.

3.4 Assumptions and threat model

- Each domain operator is honest-but-curious: it follows the protocol but may try to infer the other domain's raw data from what it receives.
- Overlapping users are linked through a pseudonymous identifier that both operators already hold. Private set intersection can replace this when no common identifier is allowed.
- Item text is non-sensitive and may be sent to the LLM encoder. Customer histories and identifiers are never sent to it.
- Attacks that need access to a domain's internal training process are out of scope.

3.5 Proposed Methodology: PALS-CDR Framework

PALS-CDR has four modules: an LLM-based item encoder, a sequential encoder per domain, a private sharing channel, and an adaptive fusion gate. Raw logs never leave their domain, and the LLM sees item text only.

Only a clipped, noised vector crosses between domains

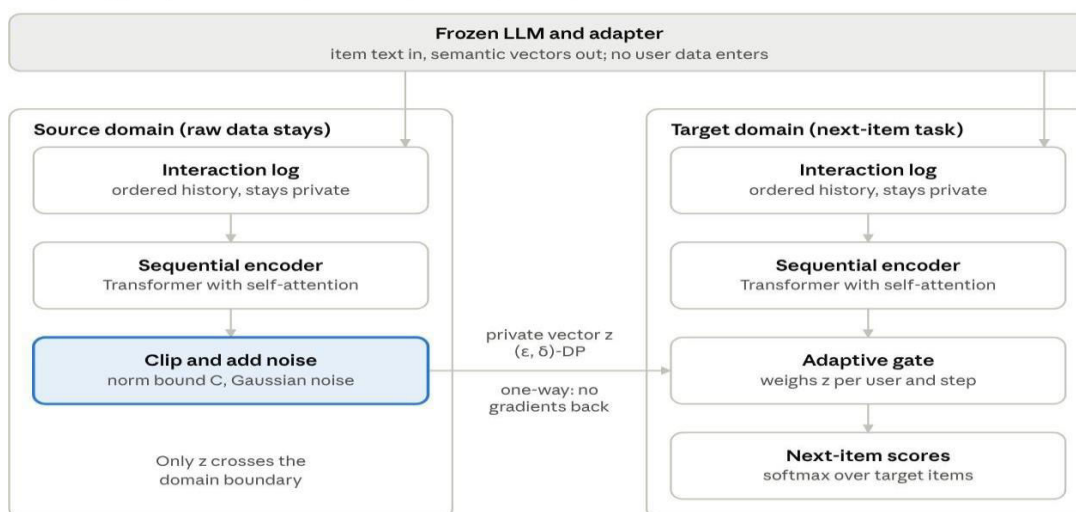


Figure 1 · PALS-CDR architecture, source to target

Figure 1 shows the flow: item text is embedded by the frozen LLM for both domains, the source domain releases one clipped and noised vector per overlapping user, and the target domain's gate decides how much of it to use.

3.6 LLM semantic item encoder

Each item description x_i is embedded once, offline, by a frozen text encoder or LLM (for example a BERT-style encoder [31]), giving a vector f_i . A small trainable adapter A maps it to the model dimension d , so the item vector is the adapter output plus the ID embedding. An optional ID embedding p_i captures collaborative signal that text cannot; it is set to zero for items with no interactions, so new items are scored from text alone.

Because the same encoder embeds items from both domains, the item spaces are comparable without any item overlap. The adapter can be a two-layer MLP or low-rank update matrices [20]; the LLM itself stays frozen, which keeps cost low and removes any need to send behavioral data to it.

3.7 Sequential encoder

Each domain runs a causal Transformer encoder [18] over its user's history, in the style of SASRec [7]. The input at step t adds a position embedding and a domain embedding to the item vector v_i , and the causal Transformer returns a state h_t at every step.

The state h_t summarizes everything the user did up to step t in that domain.

3.8 Private sharing channel

The source operator takes the user's final state as a summary r_u , clips its L2 norm to a bound C , and adds Gaussian noise before release. The released vector is therefore the clipped summary plus random noise.

Clipping bounds how much any one user's vector can differ, so the sensitivity of a release is at most twice the clipping bound when one user's vector is replaced by another. Gaussian noise calibrated to that sensitivity, to ϵ and to δ then gives (ϵ, δ) -differential privacy for the release, for $\epsilon < 1$ [12]. For larger ϵ , the analytic Gaussian calibration or a Rényi accountant is used instead [22]. Everything the target operator does with z_u afterwards is post-processing and cannot weaken the guarantee.

Two scope notes keep the claim honest. First, the guarantee covers the released vector for a fixed source encoder; to extend it to the training data of the encoder itself, the source encoder can be trained with DP-SGD [13], at some cost in accuracy, and Section 5 compares the two approaches. Second, if z_u is refreshed R times, the budgets compose, and the cumulative ϵ is tracked with a Rényi accountant and reported.

3.9 Adaptive fusion gate

The target encoder produces a state at every step. A gate, computed from that state and the received vector, decides per user, per step and per feature how much of the received vector to add to it.

The addition is applied only to overlapping users, so users without a source history fall back to the target-only model. The gate is the mechanism for adaptivity (RQ1), and it has a second job: because z_u is noisy, the gate can learn to lower its weight when the noise makes the source evidence unreliable. Whether it does so is an empirical question left to future work.

Candidate items are scored by the inner product of the fused state with their semantic vectors.

3.10 Training objective

The target loss is the next-item cross-entropy over all target items (a pairwise BPR loss [28] is a drop-in alternative). An alignment loss encourages the user's target state to agree with their received source vector, relative to other users in the batch, using InfoNCE [21] with a temperature parameter and a projection layer. The training loss is the target loss plus a weighted alignment loss, with the weight tuned as a hyperparameter.

The source encoder is trained by its own operator on a source next-item loss of the same form. Since z_u is treated as a constant in the target's training, no gradient flows back to the source, so no gradients need to be exchanged, and the alignment loss uses only already-released vectors, so it costs no extra privacy budget.

3.11 Training and inference procedure

- Embed all item texts once with the frozen LLM, in both domains.
- The source operator trains its sequential encoder locally.
- For each overlapping user, the source operator computes r_u , clips it, adds noise and releases z_u .
- The target operator trains its encoder, adapter and gate on the total loss L using the released vectors.
- At inference, the target scores all candidates for a user from the user's target history and, if present, the stored z_u .
- For bidirectional transfer, the roles are swapped and the budgets of both channels are reported.
- The LLM is a one-time offline cost proportional to the number of items. Training and inference cost is close to a SASRec-sized model plus a small gate, and each overlapping user adds one d -dimensional vector to store.

IV. COMPARATIVE ANALYSIS

PALS-CDR is the only method in Table 1 that combines order-aware modeling, text semantics, adaptive transfer and a formal privacy guarantee on the cross-domain channel. The comparison is at the level of design, drawn from how each method is specified in its paper; it reports no measured accuracy.

4.1 Comparison with existing methods

Method	Models interaction order	Uses text semantics	Adaptive transfer	What crosses domains	Formal privacy
CMF [2]	No	No	No	Shared latent factors	No
EMCDR [3]	No	No	No	Source user embeddings	No
CoNet [4]	No	No	No	Joint training on both domains	No
DDTCDR [5]	No	No	No	User embeddings, dual mapping	No

Table. 1. Design-level comparison. An entry of No means the property is not part of the method's published design, not that it cannot be added.

4.2 Comparison of privacy mechanisms for the transfer channel

Approach	What leaves the domain	What is protected	Main limitation
Pooling raw logs	Interaction records	Nothing	Exposes all behavior; often not permitted
Federated learning [11]	Model updates	Raw data stays local	Updates can still leak information unless noise is added
DP-SGD training [13]	A trained model	Training records,	Noise in every step can
Approach	What leaves the domain	What is protected	Main limitation
		formally	cost accuracy; protects the model, not a per-user channel
Private set intersection	The identifier overlap only	Which users are shared	Does not protect the content of representations
Clip and noise on user vectors (PALS-CDR)	One noised vector per overlapping user	Each released vector, formally	Budget accumulates over repeated releases; covers the vector, not the encoder, unless combined with DP-SGD

Table.2. Privacy mechanisms for cross-domain transfer. These are complementary: PALS-CDR can be combined with DP-SGD and with private set intersection.

4.3 Comparison of PALS-CDR design choices

Design choice	Common alternative	Why PALS-CDR chooses it
Text-derived item vectors plus optional ID embedding	ID embeddings only	Scores items with no interactions, and links domains with no item overlap
Per-user, per-step gated fusion	One shared mapping or plain concatenation	Weights source evidence per user and can lower reliance when noise makes it unreliable
One clipped, noised vector as the shared unit	Raw sequences or gradients	Bounded sensitivity gives a formal guarantee, and no gradients flow back
Frozen LLM on item text only	Prompts that contain user histories	No user data reaches the LLM
InfoNCE alignment on released vectors	Joint training across domains	Uses only released vectors, so it adds no privacy cost

Table.3. Design choices in Section 4 and the alternatives they replace.

4.4 Reading the comparison

Methods that model order well, such as π -Net and C²DSR, pool sequences and offer no privacy guarantee. Text-based

methods such as UniSRec and P5 handle new items but do not give per-user, privacy-aware transfer. Privacy-oriented methods protect data but do not model sequences or semantics. PALS-CDR is designed to close all three gaps at once; whether it does so in accuracy terms is an empirical question that this paper leaves to future work.

V. CHALLENGES AND FUTURE WORK

5.1 Challenges

- **Utility–privacy tension.** Clipping and noise reduce the signal a vector carries. Small ϵ may make source evidence close to useless, and the right operating point depends on the application.
- **Limited and sensitive overlap.** The framework assumes some users appear in both domains and can be linked. Linking identifiers is itself a privacy-relevant step, and in many settings the overlap is small or unavailable.
- **Scope of the guarantee.** The guarantee covers the released vector. It does not cover leakage through the trained encoder unless DP-SGD is used, and it does not cover side channels such as timing or model-inversion attacks on the target model.
- **Domain shift and negative transfer.** Some domain pairs share little taste signal. The gate reduces the risk but does not remove it.
- **LLM cost and bias.** Text embeddings depend on description quality and language, and LLM-derived representations may carry popularity or demographic bias into rankings. Embedding very large catalogs is also a cost.
- **Evaluation limits.** The framework is specified but not yet validated empirically; offline next-item metrics will be only a proxy for business outcomes, and review data differs from purchase or click logs.

5.2 Future work

- Extend to many source domains with a learned attention over domains, and to non-overlapping users with prototype-based transfer.
- Replace the pseudonymous identifier with private set intersection, and compare against secure aggregation and local differential privacy in the federated setting [11].
- Use per-user adaptive clipping and privacy budgets, so users who want more protection can choose it.
- Fine-tune a small LLM on item text with parameter-efficient methods [16, 20], and study whether richer text improves transfer at the same ϵ .
- Add fairness and exposure analyses, and evaluate online with A/B tests.
- Audit the channel empirically with membership-inference and attribute-inference attacks, to complement the formal bound.

VI. CONCLUSION

This paper presented PALS-CDR, a cross-domain recommendation framework that models customers' ordered interactions with a Transformer, gives items a shared semantic space through a frozen LLM, and moves information between domains only as a clipped, noised vector with a stated (ϵ, δ) guarantee. An adaptive gate decides per user and step how much of that vector to use, which both personalizes transfer and gives a route to robustness against noise. We specified the architecture, objective and privacy accounting, and compared the framework with representative methods on design-level criteria.

The central claim to be established by future experiments is that transfer remains useful at a meaningful privacy budget. Future work will extend the framework to many domains, stronger identity protection and empirical privacy audits.

REFERENCES

1. Zhu, F., Wang, Y., Chen, C., Zhou, J., Li, L., Liu, G. Cross-domain recommendation: challenges, progress, and prospects. In IJCAI, 2021.
2. Singh, A. P., Gordon, G. J. Relational learning via collective matrix factorization. In KDD, 2008.
3. Man, T., Shen, H., Jin, X., Cheng, X. Cross-domain recommendation: an embedding and mapping approach. In IJCAI, 2017.
4. Hu, G., Zhang, Y., Yang, Q. CoNet: collaborative cross networks for cross-domain recommendation.

In CIKM, 2018.

5. Li, P., Tuzhilin, A. DDTCDR: deep dual transfer cross domain recommendation. In WSDM, 2020.
6. Hidasi, B., Karatzoglou, A., Baltrunas, L., Tikk, D. Session-based recommendations with recurrent neural networks. In ICLR, 2016.
7. Kang, W.-C., McAuley, J. Self-attentive sequential recommendation. In ICDM, 2018.
8. Sun, F., et al. BERT4Rec: sequential recommendation with bidirectional encoder representations from Transformer. In CIKM, 2019.
9. Ma, M., Ren, P., Lin, Y., Chen, Z., Ma, J., de Rijke, M. π -Net: a parallel information-sharing network for shared-account cross-domain sequential recommendations. In SIGIR, 2019.
10. Cao, J., Cong, X., Sheng, J., Liu, T., Wang, B. Contrastive cross-domain sequential recommendation. In CIKM, 2022.
11. McMahan, H. B., Moore, E., Ramage, D., Hampson, S., Aguera y Arcas, B. Communication-efficient learning of deep networks from decentralized data. In AISTATS, 2017.
12. Dwork, C., Roth, A. The algorithmic foundations of differential privacy. Foundations and Trends in Theoretical Computer Science, 9(3–4), 2014.
13. Abadi, M., et al. Deep learning with differential privacy. In ACM CCS, 2016.
14. Hou, Y., Mu, S., Zhao, W. X., Li, Y., Ding, B., Wen, J.-R. Towards universal sequence representation learning for recommender systems. In KDD, 2022.
15. Geng, S., Liu, S., Fu, Z., Ge, Y., Zhang, Y. Recommendation as language processing (RLP): a unified pretrain, personalized prompt & predict paradigm (P5). In RecSys, 2022.
16. Bao, K., et al. TALLRec: an effective and efficient tuning framework to align large language model with recommendation. In RecSys, 2023.
17. Wu, L., et al. A survey on large language models for recommendation. World Wide Web, 27, 2024.
18. Vaswani, A., et al. Attention is all you need. In NeurIPS, 2017.
19. Reimers, N., Gurevych, I. Sentence-BERT: sentence embeddings using Siamese BERT-networks. In EMNLP-IJCNLP, 2019.
20. Hu, E. J., et al. LoRA: low-rank adaptation of large language models. In ICLR, 2022.
21. van den Oord, A., Li, Y., Vinyals, O. Representation learning with contrastive predictive coding. arXiv:1807.03748, 2018.
22. Mironov, I. Rényi differential privacy. In IEEE Computer Security Foundations Symposium, 2017.
23. Ni, J., Li, J., McAuley, J. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In EMNLP-IJCNLP, 2019.
24. Krichene, W., Rendle, S. On sampled metrics for item recommendation. In KDD, 2020.
25. Kingma, D. P., Ba, J. Adam: a method for stochastic optimization. In ICLR, 2015.
26. Ammad-ud-din, M., et al. Federated collaborative filtering for privacy-preserving personalized recommendation system. arXiv:1901.09888, 2019.
27. Chai, D., Wang, L., Chen, K., Yang, Q. Secure federated matrix factorization. IEEE Intelligent Systems, 2021.
28. Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L. BPR: Bayesian personalized ranking from implicit feedback. In UAI, 2009.
29. Järvelin, K., Kekäläinen, J. Cumulated gain-based evaluation of IR techniques. ACM Transactions on Information Systems, 20(4), 2002.
30. Zhu, Y., et al. Personalized transfer of user preferences for cross-domain recommendation. In WSDM, 2022.
31. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. In NAACL-HLT, 2019.
32. Zhao, W. X., et al. A survey of large language models. arXiv:2303.18223, 2023.



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details