Addressing Privacy in Recommender Systems with Federated Learning

Discussion Paper

Vito Walter Anelli[1], Tommaso Di Noia[1], Eugenio Di Sciascio[1], Antonio Ferrara[1,*] and Alberto Carlo Maria Mancino[1,*]

[1]Politecnico di Bari, Bari, Italy

Abstract

In recent years, recommender systems have successfully assisted user decision-making in various user-centered applications. In such scenarios, the modern approaches are based on collecting user-sensitive preferences. However, data collection is crucial since users now worry about the related privacy risks when sharing their data. This work presents a recommendation approach based on the Federated Learning paradigm, a distributed privacy-preserving approach to the recommendation. Here, users collaborate on the training while still controlling the amount of the shared sensitive data. This paper presents FPL: a pair-wise learning-to-rank approach based on Federated Learning. We show that it puts users in control of their data and reveals recommendation performance competing with centralized state-of-the-art approaches. The public implementation is available at https://split.to/sisinflab-fpl.

Keywords

Federated Learning, Collaborative Filtering, Pair-wise Learning, Matrix Factorization

1. Introduction

Recommender Systems (RSs) have emerged as a solution for mitigating the information-overload problem by assisting users with personalized recommendations. Generally, these models are trained in a *centralized fashion*, where massive proprietary sensitive data are hosted on a single server. In the last two decades, the RS mainstream research line has focused on Collaborative Filtering (CF) [2, 3], Content-based [4, 5], and hybrid [6] approaches. However, these models need sufficient data to provide accurate recommendations by exploiting users' behavior similarities. Proposed by Google, Federated Learning (FL) emerged as a *privacy-by-design* solution for machine learning models [7, 8, 9, 10]. FL addresses the ML-privacy limitations by horizontally distributing the training while having the clients train the global model on their local devices without sharing their private data [10]. Recent works on Federated Learning-based RSs have exhibited benefits for the users' privacy [11]. Federated Pair-wise Learning (FPL) [8] shows how a federated recommender system can exploit the Learning-to-Rank competitive performance while still letting users control their data. Indeed, one of the most significant

IIR2022: 12th Italian Information Retrieval Workshop, June 29 - June 30th, 2022, Milan, Italy
Extended version [1] published in the 58th volume of Journal of Intelligent Information Systems

[*]Corresponding author.

antonio.ferrara@poliba.it (A. Ferrara); alberto.mancino@poliba.it (A. C. M. Mancino)

advantages of employing FPL is that the users participating in the federated learning process can independently decide how much they are inclined to disclose their private sensitive preferences.

In this paper, we formally show how in FPL, the users can control their data. We investigate the risks related to the transmission of the gradients and how we address this drawback by putting users in control of their data. Moreover, we show that integrating FPL could lead to competitive performance in accuracy and provide users with a trustworthy model [12].

2. Foundations of Federated Pair-wise Learning

FPL: Federated Pair-wise Learning for Recommendation. Let be $\mathcal{U}$ and $\mathcal{I}$ the set of users and items, respectively, and $\mathbf{X} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{I}|}$ be the user-item matrix containing for each x_{ui} an implicit feedback 1 or 0. Inspired by the state-of-the-art MF [13, 14, 15, 16], each user u and item i are represented by the embedding vectors $\mathbf{p}_u$ and $\mathbf{q}_i$, respectively. The dot product between $\mathbf{p}_u$ and $\mathbf{q}_i$ can explain any observed user-item interaction x_{ui}, so that any non-observed interaction can be estimated as $\hat{x}_{ui} = b_i + \mathbf{p}_u^T \mathbf{q}_i$, where b_i is a bias term. FPL builds a global model $\Theta_S = \langle \mathbf{Q}, \mathbf{b} \rangle$ on a server S, which is aware of the whole catalog $\mathcal{I}$, and a private local model $\Theta_u = \langle \mathbf{p}_u \rangle$ on each client of the federation. In our federated setting, each user u holds her own feedback dataset $\mathbf{x}_u \in \mathbb{R}^{\mathcal{I}}$, which — compared with a centralized recommender system — corresponds to the u-th row of matrix $\mathbf{X}$, so only user u knows her own set of consumed items.

In FPL, the model is trained by rounds of communications composed of a four-step **protocol** (**Distribution→Computation→Transmission→Aggregation**), described in the following.

1. **Distribution.** S delivers the model Θ_S to a subset of selected users $\mathcal{U}^- \subseteq \mathcal{U}$.

2. **Computation.** Each user u generates T triples (u, i, j) from her local dataset and for each of them performs BPR stochastic optimization to compute the updates for the local $\mathbf{p}_u$ vector of Θ_u, and for $\mathbf{p}_i$, b_i, $\mathbf{p}_j$, and b_j of the received Θ_S, following:

$$\Delta\theta = \frac{e^{-\hat{x}_{uij}}}{1 + e^{-\hat{x}_{uij}}} \cdot \frac{\partial}{\partial\theta}\hat{x}_{uij} - \lambda\theta, \qquad \text{with} \quad \frac{\partial}{\partial\theta}\hat{x}_{uij} = \begin{cases} (\mathbf{q}_i - \mathbf{q}_j) & \text{if } \theta = \mathbf{p}_u, \\ \mathbf{p}_u & \text{if } \theta = \mathbf{q}_i, \\ -\mathbf{p}_u & \text{if } \theta = \mathbf{q}_j, \\ 1 & \text{if } \theta = b_i, \\ -1 & \text{if } \theta = b_j. \end{cases} \tag{1}$$

3. **Transmission.** Each client $u \in \mathcal{U}^-$ send back the updates for the computed item embedding and item bias to the server S. We should focus on how BPR computes the training output. Since for a triple (u, i, j), the server could be able to distinguish the consumed item i from the non-consumed one j (for instance, by analyzing the positive and the negative sign of Δb_i and Δb_j), we argue that sending all the updates computed by u may raise a privacy issue. FPL proposes a solution to overcome this vulnerability by sending the sole update $(\Delta\mathbf{q}_j, \Delta b_j)$ of each training triples (u, i, j). In this way, the user u shares only indistinguishably negative or missing values, which are assumed to be *non-sensitive*. Furthermore, FPL allows users to establish the number of consumed

items to share with the central server S, by introducing the parameter π. It is related to the probability of a user sending a specific update relative to a positive item $(\Delta\mathbf{q}_i, \Delta b_i)$ in addition to $(\Delta\mathbf{q}_j, \Delta b_j)$.

4. **Global aggregation.** All the received updates are aggregate by S in $\mathbf{Q}$ and $\mathbf{b}$ to build the new model $\Theta_S \leftarrow \Theta_S + \alpha \sum_{u\in\mathcal{U}^-} \Delta\Theta_u$, with α being the learning rate.

Privacy Analysis of FPL. FPL has not been conceived as a privacy-preserving framework but as a tool to control the trade-off between potentially exposed sensitive data and the recommendation quality. While federated learning hides, by design, users' raw data to the server, some *malicious* actors might still try to learn sensitive information. For this reason, federated learning alone does not consider providing privacy guarantees to users. In the context of FPL, the aim is to protect the existence of user-item transactions. While attempts of active reconstruction of the user profile are not considered here and are out of the scope of this work, we focus on the presence of an honest-but-curious server. Regarding Eq. 1, suppose a pair of positive and negative items i and j and the gradients received at the t-th round of communication. The notation of $\Delta\mathbf{q}_i^t$ and $\Delta\mathbf{q}_j^t$ could be extended by focusing on a single latent factor f:

$$\Delta\mathbf{q}_{i,f}^t = \mathbf{p}_{u,f}^{t-1}\sigma(\mathbf{p}_{u,f}^{t-1}(\mathbf{q}_{i,f}^{t-1} - \mathbf{q}_{j,f}^{t-1})), \tag{2}$$

$$\Delta\mathbf{q}_{j,f}^t = -\mathbf{p}_{u,f}^{t-1}\sigma(\mathbf{p}_{u,f}^{t-1}(\mathbf{q}_{i,f}^{t-1} - \mathbf{q}_{j,f}^{t-1})), \tag{3}$$

where $\sigma(\cdot)$ returns values in the range $(0, 1)$. These equations show that the modules of $\Delta\mathbf{q}_{i,f}^t$ and $\Delta\mathbf{q}_{j,f}^t$ (which must be sent to the server) are identical, while their signs are opposite. Moreover, the sign of the update depends on both the existence/absence of a transaction for k and on $sgn(\mathbf{p}_{u,f}^{t-1})$. Therefore, the sign of a gradient does not directly reveal the presence or absence of an item in the user's training set. However, the pairs of positive and negative gradients disclose user preference patterns. In a round of communication, all the updates for the consumed items share the same sign, as well as all the updates for the non-consumed items have the same positive or negative sign, depending on $sgn(\mathbf{p}_{u,f}^{t-1})$. If the server S is honest-but-curious (i.e., it may try to inspect the updates to obtain some user information), as soon as it obtains enough information adequate to identify one or more consumed/non-consumed items, the entire user dataset will be exposed. To avoid this problem, FPL puts users in control of their data. If the users adopt the *privacy-oriented* masking procedure during the Transmission phase, they can decide the fraction of updates for positive items to send. In the case of exposure of the user transactions, only a fraction is given up. As a consequence, FPL has to work in a data scarcity scenario, where the fraction of used data is defined by the parameter π (this could be fixed by the system designer or actively decided by the users). In the experimental section, we empirically show how FPL is resilient to missing data in the federated scenario.

3. Experimental Results

In the following, we report the accuracy performance of FPL. It has been evaluated on the *Foursquare* dataset [17] in the Point-of-Interest domain since it contains data usually perceived as sensible. To mimic a federation of devices in a single country, we have extracted check-ins

Table 1
Results of accuracy metrics for baselines and FPL on the three datasets. The metrics refers to the top 10 recommendation list.

		Brazil		Canada		Italy	
		P@10	**R@10**	**P@10**	**R@10**	**P@10**	**R@10**
Centralized	**Random**	0.00013	0.00015	0.00030	0.00035	0.00030	0.00029
	Top-Pop	0.01909	0.02375	0.04239	0.04679	0.04634	0.05506
	User-kNN	0.10600	0.13480	0.07639	0.07533	0.06881	0.07833
	Item-kNN	0.07716	0.09607	0.04006	0.03881	0.04663	0.05356
	VAE	0.10320	0.13153	0.06060	0.06317	0.10421	0.21324
	BPR-MF	0.07702	0.09494	0.03694	0.03650	0.04560	0.05458
Federated	**FCF**	0.03089	0.03749	0.03724	0.03836	0.03126	0.03708
	***s*FPL** *	0.07757	0.09581	0.04515	0.04550	0.04701	0.05600
	***s*FPL+** *	0.08682	0.11004	0.05701	0.05665	**0.05595**	0.06229
	***p*FPL** *	0.07771	0.09582	0.04582	0.04637	0.04642	0.05465
	***p*FPL+** *	**0.08733**	**0.11085**	**0.05761**	**0.05755**	0.05565	**0.06291**

* Best π obtained for each the proposed FPL variations across three countries (Brazil, Canada, and Italy) are: *s*FPL = (0.5, 0.1, 0.4), *s*FPL+ = (0.9, 0.4, 0.2), *p*FPL = (0.8, 0.1, 1), *p*FPL+ = (0.8, 0.3, 0.1)

for three countries, namely Brazil, Canada, and Italy. Moreover, we have split each dataset by adopting a realistic temporal hold-out 80-20 splitting on a per-user basis [18, 19]. FPL has been evaluated with different values of π in [0.0, 1.0] with step 0.1, in order to assess the impact of user sharing more (high π) or less (low π) positive feedbacks. Hence, four configurations have been considered regarding variations in computation ad communication. In ***s*FPL** and ***p*FPL**, the model is updated for each round of communication involving one client, or all the clients, respectively. In these configurations, the clients' local training involves only one triple (u, i, j) from their local dataset. In contrast, in the correspondent ***s*FPL+** and ***p*FPL+** configurations, the number of selected triples is proportional to the number of each user's positive feedback. FPL has been compared against six centralized models (**Random**, **Most Popular**, **BPR-MF** [20], **User-kNN** and **Item-kNN** [21], **VAE** [22]), and a federated recommendation approach based on MF (**FCF** [23]). The accuracy performances of the results are reported in table 1, comparing the four configurations of FPL and the state-of-the-art baselines. We notice that VAE and User-kNN outperform the other models, while Item-kNN and BPR-MF show similar results. Regarding FPL, when comparing *s*FPL and *s*FPL+ with their parallelized configurations (*p*FPL and *p*FPL+), we observe that the increased parallelism does not affect the performance significantly. On the other hand, increasing the local computation (*s*FPL+ and *p*FPL+) boots the performance up to 24%. The results show that FPL behaves better than BPR-MF in precision and recall. These performances are surprising considering that FPL exploits less feedback per round since they are reduced by π. It is also notable that FPL outperforms FCF and preserves privacy to a greater extent since sharing gradients of all rated items in FCF can result in a data leak [24].

We have seen how the proposed system can generate recommendations with a quality that is comparable with the centralized pair-wise learning approach. Moreover, the increased local computation causes a considerable improvement in the accuracy of recommendations. On the other side, the training parallelism does not significantly affects results. Finally, when the ***local computation*** *is combined with* ***parallelism****, the results show a further improvement.*

References

[1] V. W. Anelli, Y. Deldjoo, T. D. Noia, A. Ferrara, F. Narducci, User-controlled federated matrix factorization for recommender systems, J. Intell. Inf. Syst. 58 (2022) 287–309.

[2] B. McFee, L. Barrington, G. R. G. Lanckriet, Learning content similarity for music recommendation, IEEE Trans. Audio, Speech & Language Processing 20 (2012) 2207–2218.

[3] J. Yuan, W. Shalaby, M. Korayem, D. Lin, K. AlJadda, J. Luo, Solving cold-start problem in large-scale recommendation engines: A deep learning approach, in: 2016 IEEE Int. Conf. on Big Data, BigData 2016, Washington DC, USA, December 5-8, 2016, IEEE Computer Society, 2016, pp. 1901–1910.

[4] V. Bellini, G. M. Biancofiore, T. D. Noia, E. D. Sciascio, F. Narducci, C. Pomo, Guapp: A conversational agent for job recommendation for the italian public administration, in: 2020 IEEE Conference on Evolving and Adaptive Intelligent Systems, EAIS 2020, Bari, Italy, May 27-29, 2020, IEEE, 2020, pp. 1–7. URL: https://doi.org/10.1109/EAIS48028.2020.9122756. doi:10.1109/EAIS48028.2020.9122756.

[5] V. W. Anelli, A. Bellogín, A. Ferrara, D. Malitesta, F. A. Merra, C. Pomo, F. M. Donini, T. D. Noia, V-elliot: Design, evaluate and tune visual recommender systems, in: H. J. C. Pampín, M. A. Larson, M. C. Willemsen, J. A. Konstan, J. J. McAuley, J. Garcia-Gathright, B. Huurnink, E. Oldridge (Eds.), RecSys '21: Fifteenth ACM Conference on Recommender Systems, Amsterdam, The Netherlands, 27 September 2021 - 1 October 2021, ACM, 2021, pp. 768–771. URL: https://doi.org/10.1145/3460231.3478881. doi:10.1145/3460231.3478881.

[6] V. W. Anelli, T. D. Noia, E. D. Sciascio, A. Ragone, J. Trotta, How to make latent factors interpretable by feeding factorization machines with knowledge graphs, in: ISWC (1), volume 11778 of *Lecture Notes in Computer Science*, Springer, 2019, pp. 38–56.

[7] J. Konecný, B. McMahan, D. Ramage, Federated optimization: Distributed optimization beyond the datacenter, CoRR abs/1511.03575 (2015). arXiv:1511.03575.

[8] V. W. Anelli, Y. Deldjoo, T. D. Noia, A. Ferrara, F. Narducci, How to put users in control of their data in federated top-n recommendation with learning to rank, in: SAC '21: The 36th ACM/SIGAPP Symposium on Applied Computing, Virtual Event, Republic of Korea, March 22-26, 2021, ACM, 2021, pp. 1359–1362. URL: https://doi.org/10.1145/3412841.3442010. doi:10.1145/3412841.3442010.

[9] J. Konecný, H. B. McMahan, D. Ramage, P. Richtárik, Federated optimization: Distributed machine learning for on-device intelligence, CoRR abs/1610.02527 (2016). arXiv:1610.02527.

[10] B. McMahan, E. Moore, D. Ramage, S. Hampson, B. A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Proc. of 20th Int. Conf. on Artificial Intelligence and Stat., 2017, pp. 1273–1282. URL: http://proceedings.mlr.press/v54/mcmahan17a.html.

[11] V. W. Anelli, Y. Deldjoo, T. D. Noia, A. Ferrara, F. Narducci, Federank: User controlled feedback with federated recommender systems, in: ECIR (1), volume 12656 of *Lecture Notes in Computer Science*, Springer, 2021, pp. 32–47.

[12] C. Ardito, T. D. Noia, E. D. Sciascio, D. Lofú, G. Mallardi, C. Pomo, F. Vitulano, Towards a trustworthy patient home-care thanks to an edge-node infrastructure, in: R. Bernhaupt, C. Ardito, S. Sauer (Eds.), Human-Centered Software Engineering - 8th IFIP WG 13.2

International Working Conference, HCSE 2020, Eindhoven, The Netherlands, November 30 - December 2, 2020, Proceedings, volume 12481 of *Lecture Notes in Computer Science*, Springer, 2020, pp. 181–189. URL: https://doi.org/10.1007/978-3-030-64266-2_11. doi:10.1007/978-3-030-64266-2_11.

[13] Y. Koren, R. M. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, IEEE Computer 42 (2009) 30–37.

[14] V. W. Anelli, T. D. Noia, E. D. Sciascio, A. Ferrara, A. C. M. Mancino, Sparse feature factorization for recommender systems with knowledge graphs, in: H. J. C. Pampín, M. A. Larson, M. C. Willemsen, J. A. Konstan, J. J. McAuley, J. Garcia-Gathright, B. Huurnink, E. Oldridge (Eds.), RecSys '21: Fifteenth ACM Conference on Recommender Systems, Amsterdam, The Netherlands, 27 September 2021 - 1 October 2021, ACM, 2021, pp. 154–165. URL: https://doi.org/10.1145/3460231.3474243. doi:10.1145/3460231.3474243.

[15] V. W. Anelli, T. D. Noia, E. D. Sciascio, A. Ragone, J. Trotta, Semantic interpretation of top-n recommendations, IEEE Trans. Knowl. Data Eng. 34 (2022) 2416–2428. URL: https://doi.org/10.1109/TKDE.2020.3010215. doi:10.1109/TKDE.2020.3010215.

[16] V. W. Anelli, T. D. Noia, P. Lops, E. D. Sciascio, Feature factorization for top-n recommendation: From item rating to features relevance, in: RecSysKTL, volume 1887 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2017, pp. 16–21.

[17] D. Yang, D. Zhang, B. Qu, Participatory cultural mapping based on collective behavior data in location-based social networks, ACM TIST 7 (2016) 30:1–30:23.

[18] A. Gunawardana, G. Shani, Evaluating recommender systems, in: Recommender Systems Handbook, Springer, 2015, pp. 265–308.

[19] V. W. Anelli, T. D. Noia, E. D. Sciascio, A. Ragone, J. Trotta, Local popularity and time in top-n recommendation, in: European Conf. on Information Retrieval, volume 11437, Springer, 2019, pp. 861–868.

[20] S. Rendle, C. Freudenthaler, Z. Gantner, L. Schmidt-Thieme, BPR: bayesian personalized ranking from implicit feedback, in: Proc. of the 25th Conf. on Uncertainty in Artificial Intelligence, 2009, pp. 452–461.

[21] Y. Koren, Factor in the neighbors: Scalable and accurate collaborative filtering, ACM Transactions on Knowledge Discovery from Data (TKDD) 4 (2010) 1–24.

[22] D. Liang, R. G. Krishnan, M. D. Hoffman, T. Jebara, Variational autoencoders for collaborative filtering, in: Proc. of 2018 WWW Conf., 2018, pp. 689–698.

[23] M. Ammad-ud-din, E. Ivannikova, S. A. Khan, W. Oyomno, Q. Fu, K. E. Tan, A. Flanagan, Federated collaborative filtering for privacy-preserving personalized recommendation system, CoRR abs/1901.09888 (2019). arXiv:1901.09888.

[24] D. Chai, L. Wang, K. Chen, Q. Yang, Secure federated matrix factorization, CoRR abs/1906.05108 (2019). URL: http://arxiv.org/abs/1906.05108. arXiv:1906.05108.