

Federated Recommender Systems with Learning to Rank

Vito Walter Anelli¹, Yashar Deldjoo¹, Tommaso Di Noia¹, Antonio Ferrara¹ and Fedelucio Narducci¹

¹Politecnico di Bari, Via E. Orabona, 4, 70126 Bari, Italy

Abstract

Recommendation services are extensively adopted in several user-centered applications as a tool to alleviate the information overload problem and help users in orienteering in a vast space of possible choices. In such scenarios, data ownership is a crucial concern since users may not be willing to share their *sensitive* preferences (e.g., visited locations, read books, bought items) with a central server. Unfortunately, data harvesting and collection is at the basis of modern, state-of-the-art approaches to recommendation. To address this issue, we extend Federated Pair-wise Learning (FPL), an architecture in which users collaborate in training a central factorization model while controlling the amount of sensitive data leaving their devices. The proposed approach implements pair-wise learning-to-rank optimization by following the *Federated Learning* principles, conceived originally to mitigate the privacy risks of traditional machine learning.

Keywords

Federated Learning, Recommender Systems, Information Retrieval, Learning to Rank

1. Introduction

Collaborative filtering (CF) models have been mainstream research in the recommender system (RS) community over the last two decades thanks to their performance accuracy [1, 2]. Among them, a prominent class uses the matrix factorization (MF) approach as the inference model. The MF model's main aim is to uncover user and item latent representations whose linear interaction explains observed feedback. To date, the majority of existing MF models are trained in a *centralized* fashion causing several concerns about the privacy of user data.


The consequent data scarcity dilemma can thereby jeopardize the training of MF models. Training high-quality MF models strongly relies on sufficient in-domain interaction data to ensure that enough co-occurrence information exists to shape similar behavioral/preference patterns in a user community. Although cross-domain recommendation approaches allow combating the issue of data scarcity, their applicability largely depends upon the availability of data providers that can collect/supply cross-domain in their platform (e.g., Amazon). However,

SEBD 2021: The 29th Italian Symposium on Advanced Database Systems, September 5-9, 2021, Pizzo Calabro (VV), Italy

✉ vitowalter.anelli@poliba.it (V. W. Anelli); yashar.deldjoo@poliba.it (Y. Deldjoo); tommaso.dinoia@poliba.it (T. D. Noia); antonio.ferrara@poliba.it (A. Ferrara); fedelucio.narducci@poliba.it (F. Narducci)



© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

these approaches remain out of focus in this work. In recent years, federated learning (FL) was proposed by Google as a mean to offer a *privacy-by-design* solution [3, 4, 5] for machine-learned models. Federated learning aims to meet ML privacy shortcomings by horizontally distributing the model’s training over user devices; thus, clients exploit private data without sharing them [5]. Despite its original formulation, the FL concept is extended to a more comprehensive idea of privacy-preserving, decentralized, collaborative ML techniques [6] where different data partitions share the same feature space (horizontal federation) or not (vertical federation). Weiss *et al.* [7] state that privacy can be preserved by limiting data collection, which is one of the main privacy concerns [8]. Indeed, the accuracy of RS based on the CF paradigm is strictly dependent on the amount of user preferences available.

Our idea is to put users in control of their sensitive data by allowing them to choose the amount of information to share with the server. Hence, if data collection from the server side is reduced, other threats related to retention, sales, and unauthorized data browsing are limited. The proposed system extends FPL [9, 10] (short for Federated Pair-wise Learning), is a *federated* factorization model for collaborative recommendation¹. It extends state-of-the-art factorization approaches to build a RS that puts users in control of their sensitive data. Users participating in the federation process can decide *if* and *to which extent* they are willing to disclose their *sensitive* private data (i.e., what they liked/consumed). FPL mainly leverages not-sensitive information (e.g., places the user has not visited) – which can be large and non-sensitive – to reach a competitive accuracy and, at the same time, respect a satisfactory balance between accuracy and privacy. We have carried out extensive experiments on real-world datasets [11] in the Point of Interest (PoI) domain by considering the accuracy of recommendation and diversity metrics. The experimental evaluation shows that FPL can provide high-quality recommendations, putting the user in control of the amount of sensitive data to share.

2. Approach

In this section, after a brief introduction of background technologies, we extend FPL [9, 10] (depicted in Fig. 1). To the best of our knowledge, FPL is the first attempt to put pair-wise optimization in federated recommender systems and give the users the possibility to select the trade-off between data disclosure and the recommendation utility.

2.1. Background Technologies

Federated Learning. Federated learning (FL) is a paradigm initially envisioned by Google [3, 12, 5] to train a machine-learning model from data distributed among a loose federation of users’ devices (e.g., personal mobile phones). The rationale is to face the increasing issues of ownership and locality of data to mitigate the privacy risks (and leaks) resulting from centralized machine learning [13, 14]. In particular, given Θ denoting the parameters of a machine learning model, we consider a learning scenario where the objective is to minimize a generic loss function $G(\Theta)$. FL is a learning paradigm in which the users $u \in \mathcal{U}$ of a federation collaborate to solve the learning problem under the coordination of a central server S without sharing or exchanging

¹A public implementation of FPL is available at <https://github.com/sisinflab/FedBPR/>.

their raw data with S . From an algorithmic point of view, we start with S sharing Θ with the federation of devices. Then, specific methods solve a local optimization problem on the single device, i.e., using its data, and exploiting Θ . Afterwards, the client shares the parameters of its local model with S . The parameters provided by the clients are used to update Θ , which is sent back to the devices in a new iteration step.

Factorization Models and Pair-Wise Recommendation A recommendation problem over a set of users \mathcal{U} and a set of items \mathcal{I} is defined as the activity of finding for each user $u \in \mathcal{U}$ an item $i \in \mathcal{I}$ that maximizes a utility function $g : \mathcal{U} \times \mathcal{I} \rightarrow \mathbb{R}$. In this context, $\mathbf{X} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{I}|}$ is the user-item matrix containing for each x_{ui} an explicit or implicit feedback (e.g., rating or check-in, respectively) of user $u \in \mathcal{U}$ for item $i \in \mathcal{I}$. In the work at hand, an implicit feedback scenario is considered – i.e., feedback is, e.g., purchases, visits, clicks, views, check-ins –, with \mathbf{X} containing binary values. Therefore, $x_{ui} = 1$ and $x_{ui} = 0$ denote either user u has consumed or not item i , respectively.

In FPL, the underlying data model is a Factorization model, inspired by MF [15], a recommendation model that became popular in the last decade thanks to its state-of-the-art recommendation accuracy [16]. This technique aims to build a model Θ in which each user u and each item i is represented by the embedding vectors \mathbf{p}_u and \mathbf{q}_i , respectively, in the shared latent space \mathbb{R}^F . The algorithm relies on the assumption that \mathbf{X} can be factorized such that the dot product between \mathbf{p}_u and \mathbf{q}_i can explain any observed user-item interaction x_{ui} , and that any non-observed interaction can be estimated as $\hat{x}_{ui}(\Theta) = b_i(\Theta) + \mathbf{p}_u^T(\Theta) \cdot \mathbf{q}_i(\Theta)$ where b_i is a term denoting the bias of the item i . Among pair-wise approaches for learning-to-rank the items of a catalog, Bayesian Personalized Ranking (BPR) [17] is one of the most broadly adopted, thanks to its capabilities to correctly rank with *acceptable* computational complexity. In detail, given a training set defined by $\mathcal{K} = \{(u, i, j) \mid x_{ui} = 1 \wedge x_{uj} = 0\}$, BPR solves the optimization problem via the criterion $\max_{\Theta} \sum_{(u,i,j) \in \mathcal{K}} \ln \sigma(\hat{x}_{uij}(\Theta)) - \lambda \|\Theta\|^2$, where $\hat{x}_{uij}(\Theta) = \hat{x}_{ui}(\Theta) - \hat{x}_{uj}(\Theta)$ is a real value modeling the relation between user u , item i and item j , $\sigma(\cdot)$ is the sigmoid function, and λ is a model-specific regularization parameter to prevent overfitting.

Pair-wise optimization can be applied to a wide range of recommendation models, included factorization. Hereafter, we denote the model $\Theta = \langle \mathbf{P}, \mathbf{Q}, \mathbf{b} \rangle$, where $\mathbf{P} \in \mathbb{R}^{|\mathcal{U}| \times F}$ is a matrix whose u -th row corresponds to the vector \mathbf{p}_u , and $\mathbf{Q} \in \mathbb{R}^{|\mathcal{I}| \times F}$ is a matrix in which the i -th row corresponds to the vector \mathbf{q}_i . Finally, $\mathbf{b} \in \mathbb{R}^{|\mathcal{I}|}$ is a vector whose i -th element corresponds to the value b_i .

2.2. FPL: Federated Pair-wise Learning for Recommendation

Following the aforementioned federated learning principles, let us assume that users in \mathcal{U} consume items from a catalog \mathcal{I} and give feedback about them. S is aware of the whole catalog \mathcal{I} , while only user u knows her own set of consumed items. Given these conditions, the classic BPR-MF learning procedure [17] for model training can not be applied to the federated learning scheme [5]. Instead, we propose a novel learning paradigm (depicted in Figure 1) that is executed for a number E of epochs and works by rounds of communication that envisages **Distribution** \rightarrow **Computation** \rightarrow **Transmission** \rightarrow **Aggregation** sequences between the server and the clients.

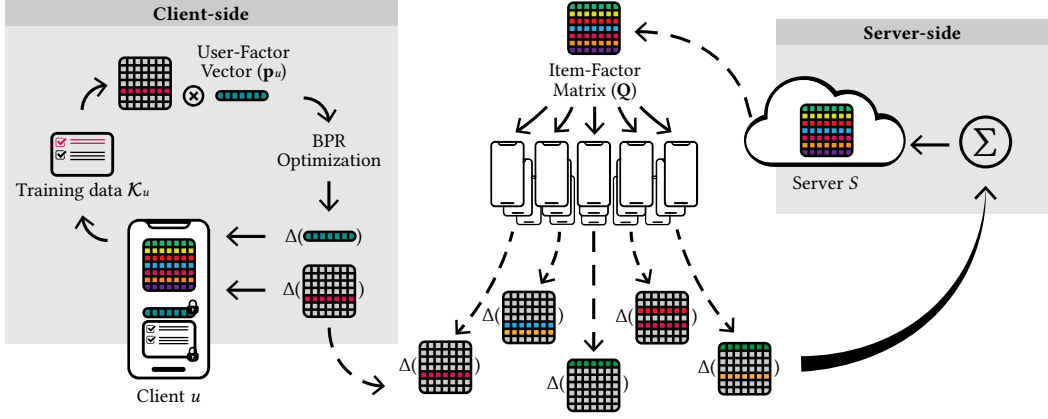


Figure 1: Item-Factor Matrix (center) is sent by the server to the federation of devices (left side) which perform the local training phase. Local outputs are sent to the server which aggregates them (right side).

In the FPL setting, a global model Θ_S is built on S such that $\Theta_S = \langle \mathbf{Q}, \mathbf{b} \rangle$, where $\mathbf{Q} \in \mathbb{R}^{|\mathcal{I}| \times F}$ and $\mathbf{b} \in \mathbb{R}^{|\mathcal{I}|}$ are the item-factor matrix and the bias vector (introduced in Section 2.1). On the other hand, on each device in the federation, FPL builds a model $\Theta_u = \langle \mathbf{p}_u \rangle$, which corresponds to the representation of user u in a latent space of dimensionality F . It is noteworthy that, in FPL, only user u holds the embedding vector \mathbf{p}_u ; therefore, each user u autonomously computes her personalized item ranking, by combining the global model Θ_S , sent by S to the devices in the federation, with her local model Θ_u . In such a setting, each user u holds her own private 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} . Each FPL client u hosts a user-specific training set $\mathcal{K}_u : \mathcal{U} \times \mathcal{I} \times \mathcal{I}$ defined by $\mathcal{K}_u = \{(u, i, j) \mid x_{ui} = 1 \wedge x_{uj} = 0\}$, where x_{ui} represents the i -th element of \mathbf{x}_u . Please note that, in the following, we refer to $X^+ = \sum_{u \in \mathcal{U}} |\{x_{ui} \mid x_{ui} = 1\}|$ as the number of positive interactions.

The number of rounds of communication performed in each learning epoch is a parameter denoted by the symbol *rpe* (*round-per-epoch*). Each round of communication is envisioned as a four-step **protocol**, described in the following.

1. **Distribution.** S randomly selects a subset of users $\mathcal{U}^- \subseteq \mathcal{U}$ and delivers them the model Θ_S .
2. **Computation.** Each user u generates T triples (u, i, j) from her dataset \mathcal{K}_u 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:

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

It is worth noticing that Rendle [17] suggests, in a centralized scenario, to adopt a uniform distribution (over \mathcal{K}) to choose the training triples randomly. The purpose is to avoid data is traversed item-wise or user-wise, since this may lead to slow convergence. Conversely, in a federated approach, we required to train the model user-wise since the training of each round of communication is performed separately on each client u knowing only data in \mathcal{K}_u . This is the reason why, in FPL, the designer can control of the number of triples T used for training, to tune the degree of local computation – i.e., how much the sampling is user-wise traversing.

3. **Transmission.** The clients in \mathcal{U}^- send back to S a portion of the updates for the computed item factor vector and item bias. More in detail, since the training output of a triple (u, i, j) in BPR lets the server distinguish the consumed item i from the non-consumed one j (for example just by analyzing the positive and the negative sign of Δb_i and Δb_j), while they show the same absolute value, we argue that sending all the updates computed by u may allow S to reconstruct \mathcal{K}_u thus raising a privacy issue. Since our primary goal is to put users in control of their data, 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) , user u would share with S indistinguishably negative or missing values, which are assumed to be *non-sensitive* data. Furthermore, in FPL we introduce the parameter π , which allows users to control of the number of consumed items to share with the central server S . The parameter π works as a probability that clients send also a specific positive item update $(\Delta\mathbf{q}_i, \Delta b_i)$ in addition to $(\Delta\mathbf{q}_j, \Delta b_j)$.
4. **Global aggregation.** S aggregates all the received updates 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 (each row of the matrix \mathbf{Q} and each element of \mathbf{b} is updated by summing up the contribution of all clients in \mathcal{U}^- for the corresponding item).

3. Experimental Setup

In this section, we introduce the experimental setting designed to analyze the performance of FPL. To this extent, we introduce the choice of the datasets with a brief analysis of their characteristics. Then, we describe the state-of-the-art algorithms we have involved. For the sake of reproducibility, for each method, we report the explored hyper-parameters in a specific section. Lastly, we present the evaluation protocol, and the metrics considered in the study.

Table 1

Characteristics of the evaluation datasets used in the offline experiment: $|\mathcal{U}|$ is the number of users, $|\mathcal{I}|$ the number of items, X^+ the number of records.

Dataset	$ \mathcal{U} $	$ \mathcal{I} $	X^+	$\frac{X^+}{ \mathcal{U} }$	$\frac{X^+}{ \mathcal{I} }$	$\frac{X^+}{ \mathcal{I} \cdot \mathcal{U} }$ %
Brazil	17,473	47,270	599,958	34.34	12.69	0.00073%
Canada	1,340	29,518	63,514	47.40	2.15	0.00161%
Italy	1,353	25,522	54,088	39.98	2.20	0.00157%

3.1. Datasets

The evaluation of FPL needs to meet some particular constraints: the availability of transaction data to obtain a reliable experimental setting and a domain that guarantees the presence of data the user may prefer to protect. In our view, the optimal domain would be that of the Point-of-Interest (PoI), which concerns data that users usually perceive as sensitive. Among the many available datasets, we think that a very good candidate is the *Foursquare* dataset [11]. Indeed, it is often considered as a reference for evaluating PoI recommendation models.

To mimic a federation of devices in a single country, we have extracted check-ins for three countries, namely Brazil, Canada, and Italy. Since our only constraint was to obtain datasets with different size/sparsity characteristics, we took the liberty of choosing three countries of recent RecSys conference venues. To fairly evaluate FPL against the baselines, we have kept users with more than 20 interactions². Moreover, we have split the datasets by adopting a realistic temporal hold-out 80-20 splitting on a per-user basis [18, 19]. Table 1 shows the characteristics of the resulting training sets.

3.2. Baselines

To evaluate the efficacy of FPL, we have conducted the experiments by considering non-personalized methods (random and most popular recommendation), and different recommendation approaches, including the centralized **BPR-MF** implementation [17], **VAE** [20], and **FCF** [21], which is, to date, the only federated recommendation approach based on MF (since no source code is available, we reimplemented and considered it in the reader’s interest). To evaluate the impact of feedback deprivation on recommendation accuracy, we have evaluated different values of π in the range [0.0, 1.0]. We remember that $\pi = 0.0$ means u is not sharing any update $(\Delta \mathbf{q}_i, \Delta b_i)$ with S regarding her positive items feedback, while $\pi = 1.0$ means u is sharing the updates on all positive items. Hence, we have considered four different configurations regarding computation and communication:

- **sFPL**: it aims to reproduce the stochastic learning approach of centralized factorization model with pair-wise learning, where the central model is updated sequentially; therefore, we set $|\mathcal{U}^-| = 1$ to involve just one random client per round, and it extracts solely one triple (u, i, j) from its dataset ($T = 1$) for the training phase;

²The limitations of the Collaborative Filtering in a cold-start user setting are well-known in the literature.

- **sFPL+**: we increase client local computation by raising to $\frac{X^+}{|\mathcal{U}|}$ the number of triples T extracted from \mathcal{K}_u by each client involved in the round of communication;
- **pFPL**: we enable parallelism by involving all clients in each round of communication ($\mathcal{U}^- = \mathcal{U}$); we keep $T = 1$;
- **pFPL+**: we extend pFPL by letting each client sample $T = \frac{X^+}{|\mathcal{U}|}$ triples from \mathcal{K}_u ; the rationale is that the overall training samples are exactly X^+ , as in centralized BPR-MF.

In Rendle *et al.* [17], authors suggest to set the number of triples in one epoch of BPR to X^+ , which corresponds to the number of optimizations steps. A particular choice is to randomly sampling $T = \frac{C^+}{|\mathcal{U}|}$ triples per user. To make a federated training epoch of FPL comparable to BPR and among different configurations, we set *rpe* to obtain always the same number of interactions ρ between clients and server in one epoch. This value is equal to the overall number of optimization steps in one epoch of the centralized pair-wise learning. In detail, we set $\rho = X^+$ and then $rpe = \rho \cdot |\mathcal{U}^-|$ which results in $rpe = X^+$ for sFPL, and $rpe = \frac{X^+}{|\mathcal{U}^-|}$ for pFPL.

3.3. Reproducibility

For the splitting strategy, we have adopted a **temporal hold-out** 80/20 to separate our datasets in training and test set. Moreover, to find the most promising learning rate α , we have further split the training set, adopting a temporal hold-out 80/20 strategy on a user basis to extract her validation set. **VAE** has been trained by considering three autoencoder topologies, with the following number of neurons per layer: 200-100-200, 300-100-300, 600-200-600. We have chosen candidate models by considering the best models after training for 50, 100, and 200 epochs, respectively. For the **factorization models**, we have performed a grid search in BPR-MF for $\alpha \in \{0.005, 0.05, 0.5\}$ varying the number of latent factors in $\{10, 20, 50\}$. Then, to ensure a fair comparison, we have exploited the same learning rate and number of latent factors to train FPL and **FCF**, and we explored the models in the range of $\{10, \dots, 50\}$ iterations. We have set *user-* and *positive item-*regularization parameter to $\frac{1}{20}$ of the learning rate. The *negative item-*regularization parameter is $\frac{1}{200}$ of the learning rate, as suggested in *mymedialite*³ implementation as well as by Anelli *et al.* [22].

3.4. Evaluation Metrics

We have evaluated the performance of FPL under the accuracy and diversity perspective. The accuracy of the models is measured by exploiting Precision ($P@N$) and Recall ($R@N$). They respectively represent, for each user, the proportion of relevant recommended items in the recommendation list, and the fraction of relevant items that have been altogether suggested. We have assessed the statistical significance of results by adopting Student’s paired T-test considering p-values < 0.05 ⁴. The results are in general statistically significant but the differences among BPR-MF, sFPL, and pFPL, which is a very important result. To measure the diversity of recommendations, we have measured the Item Coverage ($IC@N$), and the Gini

³<http://www.mymedialite.net/>

⁴The complete results are available at <https://github.com/sisinflab/fpl-results/>.

Table 2

Results of accuracy and beyond-accuracy metrics for baselines and FPL on the three datasets. For each configuration of FPL and for each dataset, the experiment with the best π is shown (see the bottom part for details). For all metrics, the greater the better.

	Brazil				Canada				Italy			
	PR	RE	IC	Gini	PR	RE	IC	Gini	PR	RE	IC	Gini
Random	0.00013	0.00015	46120	0.709455	0.00030	0.00035	10815	0.26809	0.00030	0.00029	10478	0.28914
Top-Pop	0.01909	0.02375	19	0.000203	0.04239	0.04679	18	0.00030	0.04634	0.05506	19	0.00035
VAE *	0.10320	0.13153	5503	0.02117	0.06060	0.06317	1044	0.00652	0.10421	0.21324	165	0.02336
BPR-MF	0.07702	0.09494	2552	0.00756	0.03694	0.03650	1216	0.00998	0.04560	0.05458	19	0.00036
FCF	0.03089	0.03749	911	0.00095	0.03724	0.03836	504	0.00174	0.03126	0.03708	403	0.00158
sFPL	0.07757	0.09581	1581	0.00561	0.04515	0.04550	451	0.00243	0.04701	0.05600	18	0.00036
sFPL+	0.08682	0.11004	5200	0.01449	0.05701	0.05665	1510	0.01259	0.05595	0.06229	932	0.00789
pFPL	0.07771	0.09582	2114	0.00638	0.04582	0.04637	425	0.00213	0.04642	0.05465	96	0.00056
pFPL+	0.08733	0.11085	3820	0.01106	0.05761	0.05755	1214	0.00981	0.05565	0.06291	936	0.00725

Best π obtained for each of the proposed FPL variations for Brazil, Canada, and Italy are: sFPL = (0.5, 0.1, 0.4), pFPL = (0.8, 0.1, 1)

* VAE does not always produce recommendations for all the users. For Italy, the reported results cover the 14% of the users. For this reason, it is not marked with **bold** in the table.

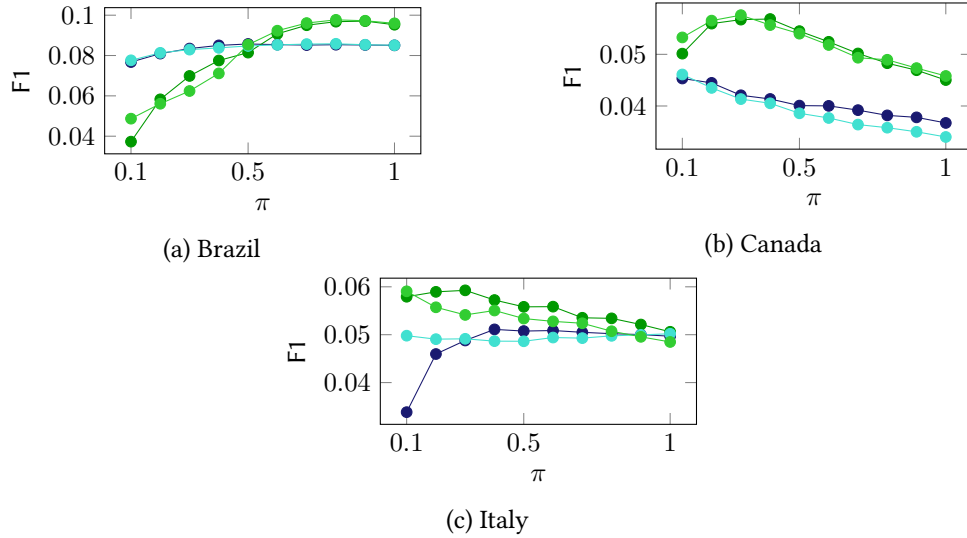


Figure 2: F1 performance at different values of π in the range $[0.1, 1]$. Dark blue is sFPL, dark green is sFPL+, light blue is pFPL, light green is pFPL+.

Index ($Gini@N$). IC provides the number of diverse items recommended to users. It also conveys the sense of the degree of personalization [23]. $Gini$ measures distributional inequality, i.e., how unequally different items a RS provides users with [24]. A higher value of $Gini$ [18] corresponds to higher personalization.

4. Discussion

The main goal of the experiments is assessing whether it is possible to obtain a recommendation performance comparable to a centralized pair-wise learning approach while allowing the users to control their data. In this respect, Table 2 shows the accuracy and diversity results of the comparison between the state-of-the-art baselines and the four configurations of FPL presented in Section 3. By focusing on accuracy metrics, we may notice that User-kNN outperforms the other approaches in the three datasets, while the performance of Item-kNN and BPR-MF approximately settle in the same range of values. This is possibly due to the user-item ratio [25], that favors the user-based schemes (see Table 1). On the other hand, it is important to investigate the differences of FPL with respect to BPR-MF, which is a pair-wise centralized approach, being FPL the first federated pair-wise recommender based on a factorization model. The performance of BPR-MF against FPL, in the configuration sFPL, shows how precision and recall in sFPL are slightly outperforming BPR-MF for the three datasets. This result is surprising since the two methods share the sequential training, but sFPL exploits a π reduced to 0.5, 0.1, and 0.4, respectively, for Brazil, Canada, and Italy. This behavior is more evident in Figure 2, where the harmonic mean between Precision and Recall (F1) is plotted for different values of π . If we look at the dark blue line, we may observe how the best result does not correspond to $\pi = 1$. In the last three rows of Table 2, we explore an increasing of the local computation (sFPL+), or an increased parallelism (pFPL), or a combination of both (pFPL+). In detail, we observe that sFPL+ takes advantage of the increased local computation, and FPL significantly outperforms BPR-MF for the three datasets; for instance, for Canada, we observe an interesting increase in precision. Instead, when comparing pFPL with sFPL, we observe that the increased parallelism does not affect the performance significantly. Even then, the increased local computation boosts the Precision and Recall performance, up to 24% for precision in the Italy dataset. The results confirm that *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 recommendation. 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.*

Afterwards, we varied π in the range $\{0.1, \dots, 1.0\}$ to assess how removal of the updates for consumed items affects the final recommendation accuracy, and we plotted the accuracy performance by considering F1 in Figure 2. As previously observed, the best performance rarely corresponds to $\pi = 1$. On the contrary, a general trend can be observed: the training reaches a peak for a certain value of π – depending on the dataset –, and then the system performance decays in accuracy when increasing the amount of shared positive updates. In rare cases, e.g., sFPL, and pFPL for Brazil dataset, the decay is absent, but results that are very close for different values of π . The general behavior suggests that the system learning exploits the updates of positive items to absorb information about popularity. This consideration is coherent with the mathematical formulation of the learning procedure, and it is also supported by the observation that for Canada and Italy FPL reaches the peak before with respect to Brazil. Indeed, Canada and Italy datasets are less sparse than Brazil, and the increase of information about positive items may lead to push up too much the popular items (this is a characteristic of pair-wise learning), while the same behavior in Brazil can be observed for values of π very close to 1.

The same mathematical background, for sFPL+ and pFPL+ with Brazil dataset, which is very sparse, explains the higher value of π needed to reach good performance. Here, the lack of positive information with a vast catalog of items, confuses the training that cannot exploit item popularity. Now, we can positively assert that *user can receive high-quality recommendations also when decides to disclose a small amount of her sensitive data. However, it should be noted that the more the dataset is sparse, the more the amount of sensitive data should be large.*

5. Conclusion and Future Work

Inspired by the potential ubiquity of the federated learning paradigm, we extend FPL, a novel federated learning framework that exploits pair-wise learning for factorization models. To this purpose, we have designed a model to leave the user-specific information of the original factorization model in the clients' devices. With FPL, a user may be completely in control of her sensitive data and could share no positive feedback with the centralized server. The framework can be envisioned as a general factorization model in which clients can tune the amount of information shared among devices. We have conducted an exploratory, but extensive, experimental evaluation to analyze the degree of accuracy, the diversity of the recommendation results, the trade-off between accuracy, and amount of shared transactions. We have assessed that the proposed model shows performance comparable with several state-of-the-art baselines and the classic centralized factorization model with pair-wise learning. The evaluation shows that clients may share a small portion of their data with the server and still receive high-performance recommendations. We believe that the proposed privacy-oriented paradigm may open the doors to a new class of ubiquitous recommendation engines.

References

- [1] B. McFee, L. Barrington, G. R. G. Lanckriet, Learning content similarity for music recommendation, *IEEE Trans. Audio, Speech & Language Processing* 20 (2012) 2207–2218.
- [2] 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.
- [3] J. Konečný, B. McMahan, D. Ramage, Federated optimization: Distributed optimization beyond the datacenter, *CoRR abs/1511.03575* (2015). [arXiv:1511.03575](https://arxiv.org/abs/1511.03575).
- [4] V. W. Anelli, Y. Deldjoo, T. Di Noia, A. Ferrara, Towards effective device-aware federated learning, in: *International Conference of the Italian Association for Artificial Intelligence*, Springer, 2019, pp. 477–491.
- [5] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, et al., Communication-efficient learning of deep networks from decentralized data, *arXiv preprint arXiv:1602.05629* (2016).
- [6] Q. Yang, Y. Liu, T. Chen, Y. Tong, Federated machine learning: Concept and applications, *ACM TIST* 10 (2019) 12:1–12:19.
- [7] S. Weiss, The need for a paradigm shift in addressing privacy risks in social networking

- applications, in: IFIP International Summer School on the Future of Identity in the Information Society, Springer, 2007, pp. 161–171.
- [8] A. J. P. Jeckmans, M. Beye, Z. Erkin, P. H. Hartel, R. L. Lagendijk, Q. Tang, Privacy in recommender systems, in: N. Ramzan, R. van Zwol, J. Lee, K. Clüver, X. Hua (Eds.), *Social Media Retrieval, Computer Communications and Networks*, Springer, 2013, pp. 263–281.
- [9] V. W. Anelli, Y. Deldjoo, T. Di Noia, A. Ferrara, F. Narducci, How to put users in control of their data in federated top-n recommendation with learning to rank, in: *Proceedings of the 36th Annual ACM Symposium on Applied Computing, SAC '21*, Association for Computing Machinery, New York, NY, USA, 2021, p. 1359–1362. URL: <https://doi.org/10.1145/3412841.3442010>. doi:10.1145/3412841.3442010.
- [10] V. W. Anelli, Y. Deldjoo, T. D. Noia, A. Ferrara, F. Narducci, Federank: User controlled feedback with federated recommender systems, in: D. Hiemstra, M. Moens, J. Mothe, R. Perego, M. Potthast, F. Sebastiani (Eds.), *Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 - April 1, 2021, Proceedings, Part I*, volume 12656 of *Lecture Notes in Computer Science*, Springer, 2021, pp. 32–47. URL: https://doi.org/10.1007/978-3-030-72113-8_3. doi:10.1007/978-3-030-72113-8_3.
- [11] 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.
- [12] J. Konečný, H. B. McMahan, D. Ramage, P. Richtárik, Federated optimization: Distributed machine learning for on-device intelligence, *CoRR* abs/1610.02527 (2016). [arXiv:1610.02527](https://arxiv.org/abs/1610.02527).
- [13] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečný, S. Mazzocchi, H. B. McMahan, T. V. Overveldt, D. Petrou, D. Ramage, J. Roselander, Towards federated learning at scale: System design, *CoRR* abs/1902.01046 (2019). [arXiv:1902.01046](https://arxiv.org/abs/1902.01046).
- [14] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al., Advances and open problems in federated learning, *arXiv preprint arXiv:1912.04977* (2019).
- [15] Y. Koren, R. M. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, *IEEE Computer* 42 (2009) 30–37.
- [16] D. kumar Bokde, S. Girase, D. Mukhopadhyay, Role of matrix factorization model in collaborative filtering algorithm: A survey, *CoRR* abs/1503.07475 (2015). [arXiv:1503.07475](https://arxiv.org/abs/1503.07475).
- [17] S. Rendle, C. Freudenthaler, Z. Gantner, L. Schmidt-Thieme, BPR: bayesian personalized ranking from implicit feedback, in: J. A. Bilmes, A. Y. Ng (Eds.), *UAI 2009, Proc. of the Twenty-Fifth Conf. on Uncertainty in Artificial Intelligence*, Montreal, QC, Canada, June 18-21, 2009, AUAI Press, 2009, pp. 452–461.
- [18] A. Gunawardana, G. Shani, Evaluating recommender systems, in: F. Ricci, L. Rokach, B. Shapira (Eds.), *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] D. Liang, R. G. Krishnan, M. D. Hoffman, T. Jebara, Variational autoencoders for collaborative filtering, in: *Proceedings of the 2018 World Wide Web Conference*, 2018, pp. 689–698.

- [21] 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](https://arxiv.org/abs/1901.09888).
- [22] V. W. Anelli, T. D. Noia, E. D. Sciascio, C. Pomo, A. Ragone, On the discriminative power of hyper-parameters in cross-validation and how to choose them, in: Proc. of the 13th ACM Conf. on Recommender Systems, ACM, 2019, pp. 447–451.
- [23] G. Adomavicius, Y. Kwon, Improving aggregate recommendation diversity using ranking-based techniques, *IEEE Trans. Knowl. Data Eng.* 24 (2012) 896–911.
- [24] P. Castells, N. J. Hurley, S. Vargas, Novelty and diversity in recommender systems, in: F. Ricci, L. Rokach, B. Shapira (Eds.), *Recommender Systems Handbook*, Springer, 2015, pp. 881–918.
- [25] G. Adomavicius, J. Zhang, Impact of data characteristics on recommender systems performance, *ACM Trans. Management Inf. Syst.* 3 (2012) 3:1–3:17. URL: <https://doi.org/10.1145/2151163.2151166>. doi:10.1145/2151163.2151166.