Is Non-IID Data a Threat in Federated Online Learning to Rank?

Shuyi Wang
The University of Queensland
Brisbane, QLD, Australia
shuyi.wang@uq.edu.au

Guido Zuccon
The University of Queensland
Brisbane, QLD, Australia
g.zuccon@uq.edu.au

ABSTRACT

In this perspective paper we study the effect of non independent and identically distributed (non-IID) data on federated online learning to rank (FOLTR) and chart directions for future work in this new and largely unexplored research area of Information Retrieval. In the FOLTR process, clients participate in a federation to jointly create an effective ranker from the implicit click signal originating in each client, without the need to share data (documents, queries, clicks). A well-known factor that affects the performance of federated learning systems, and that poses serious challenges to these approaches, is that there may be some type of bias in the way data is distributed across clients. While FOLTR systems are on their own rights a type of federated learning system, the presence and effect of non-IID data in FOLTR has not been studied. To this aim, we first enumerate possible data distribution settings that may showcase data bias across clients and thus give rise to the non-IID problem. Then, we study the impact of each setting on the performance of the current state-of-the-art FOLTR approach, the Federated Pairwise Differentiable Gradient Descent (FPDGD), and we highlight which data distributions may pose a problem for FOLTR methods. We also explore how common approaches proposed in the federated learning literature address non-IID issues in FOLTR. This allows us to unveil new research gaps that, we argue, future research in FOLTR should consider. This is an important contribution to the current state of FOLTR field because, for FOLTR systems to be deployed, the factors affecting their performance, including the impact of non-IID data, need to be thoroughly understood.

CCS CONCEPTS

• **Information systems** → **Learning to rank**; **Combination, fusion and federated search**; **Collaborative search**.

KEYWORDS

federated online learning to rank, data heterogeneity, non-IID data

ACM Reference Format:
Shuyi Wang and Guido Zuccon. 2022. Is Non-IID Data a Threat in Federated Online Learning to Rank?. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22), July 11–15, 2022, Madrid, Spain.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3477495.3531709

SIGIR '22, July 11–15, 2022, Madrid, Spain

ACM ISBN 978-1-4503-8732-3/22/07…$15.00
https://doi.org/10.1145/3477495.3531709

1 INTRODUCTION

Online learning to rank (OLTR) [18, 31, 32, 54] aims to learn effective rankers from users search interactions, i.e., queries and clicks on search engine result pages (SERPs), by iteratively training and updating a production ranker through online interventions. The use of clicks, rather than relevance labels, reduces the high cost and time required to collect labels from editorial teams; it also better aligns with the user's true preferences than labels provided by third-party judges. The execution of this training process online rather than offline (e.g., as in counterfactual LTR [19]) addresses issues associated with rapid changes in query intents [55].

Traditional OLTR solutions assume the ranker resides on a central server that controls the production of SERPs, including the online intervention made to explore the ranker's parameter space based on the index and that logs every user interaction (queries, clicks). This architecture, however, is inadequate for search contexts where the data is private or confidential and cannot be shared with the central search service or where users demand their interactions to be private, i.e. not to share clicks on SERPs with the server. Federated OLTR [21, 43] (FOLTR) has been canvassed as a solution to such situations. In FOLTR, private user data is kept on the user's device. The data is used locally within the user device to learn updates to a globally shared ranker. Local updates from all clients in the federated system are then shared to a central server[1] (thus without sharing of actual user data), which is responsible for the aggregation of the local updates, the consequent update of the global model and the sharing of the new global model with the clients (see Figure 2 for a concrete example of a FOLTR system). The object of FOLTR is to federatively create a ranker that is more effective than each of the individual rankers users could create on each of the users private data – and ideally this federated ranker should perform as well as a ranker that is created using all user data in a centralised manner.

Research on the effectiveness of the FOLTR paradigm and the factors that affect its performance is still limited to date, with only a couple of proposed and empirically investigated methods [21, 43, 44]. Importantly, research on FOLTR has fully ignored a key issue affecting the performance of federated learning (FL) systems: the presence of bias in how the training data is divided across the clients that join the federation. In other words, the fact that clients may hold non-independent and identically distributed (non-IID) data [53].

Non-IID data can pose severe threat to the effectiveness of a federated learning method. Models trained federatively in the presence of non-IID data across the clients that participate in the federation,

[1] We note that while the use of a single central server is common among federated learning methods (and certainly is the only setup investigated so far for FOLTR), alternative setups are possible and include peer-to-peer federated systems with no central servers [22, 37, 41], and federated systems with multiple central servers.

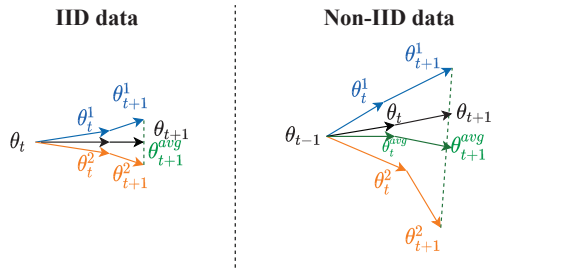


Figure 1: Illustration of the model divergence problem in FL, adapted from Zhu et al. [53]. θ_t is the ideal global model under centralised learning, and θ_t^{avg} is the average model created from the local models of Client 1 (θ_t^1) and Client 2 (θ_t^2) through FedAvg [29].

in fact, display significantly lower effectiveness, and at times experience difficulties for the model to converge [53]. Effectiveness degradation can mainly be attributed to the weight divergence between the local models resulting from the non-IID distribution of the data across the clients [53]. Local models with the same initial parameters will converge to different models because of the heterogeneity of the local data distributions. This divergence will increase as more communication rounds of the federated learning algorithm are performed. This slows down or even impedes model convergence, worsening the performance of the global model. An illustration of the phenomenon of model divergence for both IID and non-IID data in federated learning is given in Figure 1. The ideal global model (θ_t, under centralised learning) and actual global model (θ_t^{avg}, average model created through FedAvg [29]) coincide when data is IID, but diverge when data is non-IID, showing that this is a sizeable problem when the data is non-IID.

This perspective paper [2] provides a systematic understanding of when non-IID data may occur in the FOLTR setting and the impact of non-IID data in such cases. This crucial research sheds light on the factors that need to be considered when devising and deploying FOLTR methods. It also details the experimental conditions for simulating non-IID data in FOLTR, paving the way for the development and adaptation to OLTR of existing and new methods for dealing with non-IID data. With this regard, we also show how some of the methods proposed in the federated learning literature to deal with non-IID data can be cast in the FOLTR framework and the gaps that still exist in effectively addressing non-IID data in FOLTR.

2 RELATED WORK

2.1 Federated Learning with non-IID data

Zhu et al. have compiled a comprehensive survey on the impact of non-IID data on federated learning [53], also reviewing the current research on handling these challenges. Early work from Zhao et al. [52] shows a deterioration of the accuracy of federated learning if non-IID or heterogeneous data is present; they also provide a solution to this problem by creating a small subset of globally shared data between all clients (local devices). Li et al. [26] analyse the convergence of the federated learning algorithm `FedAvg` [29] (which is a component of the FOLTR method we rely upon for investigation [43]) on non-IID data and empirically show that data heterogeneity slows down the convergence. This raised attention to the presence of non-IID data in federated learning.

Generally speaking, existing approaches for handling non-IID issues in federated learning can be classified into three categories: data-based approaches, algorithm-based approaches, and system-based approaches [53]. Data sharing [52] and data augmentation [11] are two kinds of typical data-based approaches. While they achieve state-of-the-art performance, they fundamentally conflict with the objective of federated learning: that of not sharing data across clients. This is because, for example, methods such as data sharing require a subset of private data to be shared across all clients. While proposals have been made to use synthetic, rather than real, data for the data sharing mechanism [41] it is unclear (1) what the effectiveness loss of the sharing of synthetic data in place of real data is, and (2) whether the sharing of synthetic data could still jeopardise privacy as this synthetic data is typically generated from real data, and thus analysis of the synthetic data may reveal key aspects of and information contained in the real data. Algorithm-based approaches mainly focus on personalisation methods like local fine-tuning of a neural model [42] and Personalized FedAvg (Per-FedAvg) [12] – which are both limited mainly to neural models – or the casting of the federated learning process into a multi-task learning problem [39]. System based approaches adopt clustering [38] and tree-based structure [15] to deal with non-IID data. Limitations exist among all proposed approaches, and this is still a much unexplored line of research.

2.2 Federated Learning in IR

We provide an overview of the use of federated learning in OLTR in section 3. That section also introduces the FOLTR method used in the empirical experimentation in this paper: the Federated Pairwise Differentiable Gradient Descent (FPDGD) method [43], which is the current state-of-the-art in FOLTR.

Aside from its usage in OLTR, recent works have applied federated learning in other IR contexts. Zong et al. [56] provide a solution for cross-modal retrieval in a distributed data storage scenario, which uses federated learning to reduce the potential privacy risks and the high maintenance costs encountered when dealing with a large amount of training data. Wang et al. [45] study learning to rank (but not OLTR) in a cross-silo federated learning setting; this work is aimed at helping companies that have access to limited labelled data to collaboratively build a document retrieval system efficiently. Hartmann et al. [16] use federated learning to improve the ranking of suggestions in the Firefox URL bar, so that the training of the ranker on user interactions is performed in a privacy-preserving way; they show that this federated approach improves on the suggestions produced by the previously employed heuristics in Firefox. Yang et al. [48] describe the use of federated learning for search query suggestions in the Google Virtual Keyboard (GBoard) product. Here, a baseline model identifies relevant query suggestions given a user query; candidate suggestions are then filtered using a triggering model learnt using federated learning. Closest to FOLTR

[2] In this paper, if not specified otherwise, we only consider horizontal FL [47] and we believe our framework can be applied to both cross-device and cross-silo federated learning [20].

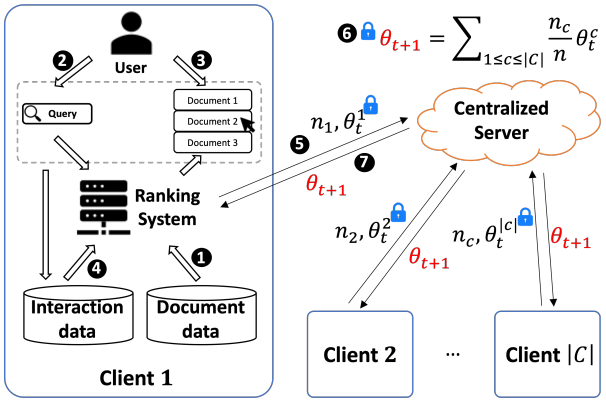


Figure 2: Schematic representation of the FOLTR setting.

is the work of Li and Ouyang [23], who devise an offline federated learning method for counterfactual learning to rank from historic click logs.

Aside from the previous examples, federated learning has also seen adoption in the area of personalised search [14], which aims to return search results that cater to the specific user's interests. While feature-based [4, 5, 17] and deep learning-based [13, 40, 50] methods are widely used in this area, user data privacy has been often overlooked – this is particularly the case when considering the user's query logs which are collected by the central server to create the personalised ranker. To tackle this issue, Yao et al. [49] recently proposed a privacy protection enhanced personalised search framework which adapts federated learning to the state-of-the-art personalised search model. While not directly related to the OLTR context we consider here, these related lines of research could benefit from the investigations and considerations reported in this paper, as the problem of non-IID data in these previous contexts has also been ignored.

3 FOLTR FRAMEWORK AND FPDGD

We next briefly describe the FOLTR framework, including the Federated Pairwise Differentiable Gradient Descent (FPDGD) method [43], which represents the current state-of-the-art in FOLTR and that we use as a representative method in our experiments to investigate the effect of non-IID data on FOLTR.

The federated online learning to rank setting is pictured in Figure 2. Searchable data is stored by each client (❶) and not shared with the centralised server or other clients. Different clients may hold all, a portion of, none of the same searchable data. Queries and user's clicks occur at a client side (❷ and ❸) and are not communicated to the centralised server or other clients: search is indeed entirely performed on the user device (❷). Each client exploits search interactions to perform local model updates to the ranker; for FPDGD, the routine executed by the client is shown in Algorithm 1, and the PDGD update is shown in Algorithm 2. Each client considers B interactions before updating the local ranker using the PDGD gradients. These local updates are then shared with the central server (❺), which in turn combines the ranker updates from the clients to produce a revised ranker (❻); for FPDGD, this is achieved according to the server routine in Algorithm 1. The new global model is then distributed to the user's device (❼).

Algorithm 1 FederatedAveraging PDGD.

- set of clients participating training: C, each client is indexed by c;
- number of local interactions for client c: n_c ($\sum_{c=1}^{|C|} n_c = n$)
- local interaction set: B, model weights: θ.

Server executes:

initialize θ_0; scoring function: f; learning rate: η
for each round $t = 1, \dots \infty$ **do**
  for each client $c \in C$ **in parallel do**
    $\theta^c_{t+1}, n_c \leftarrow \text{ClientUpdate}(c, \theta_t)$
  $\theta_{t+1} \leftarrow \sum_{c=1}^{|C|} \frac{n_c}{n} \theta^c_{t+1}$

ClientUpdate(c, θ_t): *// Run on client c*
for each local update i from 1 to B **do**
  $\theta^c_{t+1} \leftarrow \theta_t + \eta \nabla f^c_{\theta_t}$ *//PDGD update shown in Algorithm. 2*
return (θ^c_{t+1}, n_c) to server

Algorithm 2 Pairwise Differentiable Gradient Descent(PDGD) [31]

Input: initial weights: θ_1; scoring function: f; learning rate η.
for $t \leftarrow 1, \dots B$ **do**
  $q_t \leftarrow receive\_query(t)$ *// obtain a query from a user*
  $D_t \leftarrow preselect\_documents(q_t)$ *// preselect documents for query*
  $\mathbf{R}_t \leftarrow sample\_list(f_{\theta_t}, D_t)$ *// sample list*
  $\mathbf{c}_t \leftarrow receive\_clicks(\mathbf{R}_t)$ *// show result list to the user*
  $\nabla f_{\theta_t} \leftarrow \mathbf{0}$ *// initialize gradient*
  for $d_k >_\mathbf{c} d_l \in \mathbf{c}_t$ **do**
    $w \leftarrow \rho(d_k, d_l, R, D)$ *// initialize pair weight*
    $w \leftarrow w \frac{e^{f_{\theta_t}(\mathbf{d}_k)} e^{f_{\theta_t}(\mathbf{d}_l)}}{(e^{f_{\theta_t}(\mathbf{d}_k)} + e^{f_{\theta_t}(\mathbf{d}_l)})^2}$ *// pair gradient*
    $\nabla f_{\theta_t} \leftarrow \nabla f_{\theta_t} + w(f'_{\theta_t}(\mathbf{d}_k) - f'_{\theta_t}(\mathbf{d}_l))$ *// model gradient*
  $\theta_{t+1} \leftarrow \theta_t + \eta \nabla f_{\theta_t}$ *// update the ranking model*

4 TYPES OF NON-IID DATA IN FOLTR

We consider training a ranker for the OLTR system as a supervised learning task in an FL setup, with each client holding a subset of the data. Each data sample is denoted as (x, y), where x is the feature representation of the data and y is the label. The local distribution of the dataset in client i is denoted as $P_i(x, y)$. The presence of non-IID data can be represented as the difference between local data distributions: that is, for different clients i and j, $P_i(x, y) \neq P_j(x, y)$.

In federated learning, data across clients may not be IID due to different reasons: Kairouz et al. [20] and Zhu et al. [53] assert this can be due to how features x and labels y are distributed. However, the translation of these categories to FOLTR is not straightforward. In the following sections, we put forward several situations in which data specific to FOLTR could be distributed in a non-IID manner across clients. Specifically, we consider data in the FOLTR process may not be IID because of biases across clients due to:

- **Type 1:** document preferences (Section 5)
- **Type 2:** document label distribution skewness (Section 6)
- **Type 3:** click preferences (Section 7)
- **Type 4:** data quantity (Section 7)

The last data type, Type 4, i.e., the situation in which different clients hold different quantities of data (and in particular interaction data such as queries and clicks), does not necessarily imply that the data is non-IID. However, we note this case is often studied in the FL literature alongside non-IID data [24, 53], and thus we include this situation in our considerations of the non-IID problem. Each data type is defined and investigated in the next sections; in addition we provide a summary overview of the data types in Table 1.

We also note that commonly in federated learning, non-IID data occurs because the data is distributed across clients according to its features. In other words, the marginal distribution of the features belonging to the data held by each client may vary, i.e. for different clients i and j, $P_i(x) \neq P_j(x)$. This situation may occur in horizontal federated learning settings (also called homogeneous FL) [47], where each client holds different and overlapping datasets. In this case, the non-IID divergence is usually caused by inconsistent data distributions, e.g., feature imbalance of the training data local to each client. However, this case does not seem applicable to FOLTR (thus is not further studied in this paper). In FOLTR, each data item is represented by the feature vector of a query-document pair and its relevance label. The features often consist of variations of query-dependent features such as TF-IDF scores, BM25 scores, query length, as well as query-independent features such as PageRank, URL lengths, and so on [34]. In this case, bias in the feature distribution across clients would be rare as most features are dependent on the query-document pair.

Next, we describe the non-IID data types we put forward in this paper and analyse their impact on FOLTR. We empirically find that only Type 1 and partially also Type 2 data have a strong impact on the FOLTR. We thus predominantly focus our attention on these two data types while providing only a definition and a brief account of the remaining two data types in the paper due to space: we do, however, report all experiments results, thorough analysis and considerations in an online appendix available at https://github.com/ielab/2022-SIGIR-noniid-foltr.

5 TYPE 1: DOCUMENT PREFERENCES

Document preference skewness (Type 1) considers the situation when the conditional distribution $P_i(y|x)$ varies across the clients though $P_i(x)$ remains the same. This happens when different clients have different preferred candidate documents, although they are searching for the same query. As OLTR requires the user's implicit feedback as an optimization objective, which might be highly related to individual preferences, this setting appears to be of very likely occurrence.

5.1 Simulating Type 1 non-IID Data

The mechanism we use to simulate non-IID data of Type 1 and IID data to baseline the FOLTR effectiveness relies on a recent work that empirically studied and demonstrated how OLTR methods adapt when user's search intents change overtime [55]. In particular, Zhuang and Zuccon [55] created a collection for OLTR with several explicit intent types by adapting an existing TREC collection, as no dataset is available for studying this OLTR problem. Derived from ClueWeb09 and the TREC Web Track 2009 to 2012 [8], this intent change collection consists of 200 queries with 4 intents each and, on average, 512 candidate documents per query. Furthermore, query-document pairs' relevance judgements are provided per intent. We believe this is an appropriate collection to adapt to study the effect of Type 1 non-IID data on FOLTR. We can regard each intent as a type of user preference. As the average number of relevant documents per intent varies largely across all intent types, the learning difficulty of optimizing a ranker among different intents also varies. To avoid this bias, we follow Zhuang and Zuccon [55] and we re-label the original intent number for each query through random shuffling: this is possible because all intent types are independent across queries. In our experiments, we repeat this process of re-balancing 5 times, thus giving rise to results averaged across 5 FOLTR experiments. We refer to Zhuang and Zuccon [55] for further details on the dataset creation, and we further highlight that we have made available an implementation of the dataset creation procedure along with the actual dataset at https://github.com/ielab/2022-SIGIR-noniid-foltr.

To simulate non-IID data, after randomly shuffling all intents across 4 types, we let each intent represent one client preference. The client preferences differ from each other for the same query-document pair so as the corresponding relevance judgements. The federated setup involves 4 clients (represented by 4 types of intent) and the local updating time $B = 5$ with fixed global communication times $T = 10,000$. These settings are similar to those used in previous work on FOLTR [21, 43, 44] – in particular we refer the interested reader to the work of Wang et al. [44] to understand the relationships between number of clients, number of local updates B, and FOLTR effectiveness. For the implicit feedback in FOLTR, we simulate user clicks based on the popular *Simplified Dynamic Bayesian Network* (SDBN) click mode [7], following settings in previous work on OLTR [31, 33, 43, 54]. We limit SERP to 10 documents and use *nDCG*@10 for offline evaluation, cumulative discounted *nDCG*@10 [31] for online evaluation. We train a linear ranker and a neural ranker on the intent-change dataset. As in Zhuang and Zuccon [55], given that no held-out test set is available, we evaluate both online and offline performance on the original training set across all 4 intent types and average all results. For the IID setting, we merge all intents and mark a document as relevant as long as it is judged relevant for at least one of the intent types. Each client randomly picks a query from the training set and clicks documents based on the same preferences during the federated training with IID data. Other settings remain the same as the non-IID experiments.

5.2 Impact of Type 1 non-IID Data

The offline performance related to Type 1 data is shown in Figure 3; the corresponding online performance is shown in Table 2. From the offline performance, it is clear that the presence of non-IID data negatively impacts the performance of the learnt ranker, compared to those obtained when data is IID. In terms of online performance, rankers obtained in the presence of non-IID data are also worse than when trained with IID data. This can be explained as follows. Since each client has its preference (intent), the relevant documents are judged in different ways; this leads to the divergence of each client's local ranker update, as exemplified in Figure 1.

Table 1: Summary of non-IID data types in FOLTR.

Data type	Key characteristic	When it happens in FOLTR
Type 1	Document Preferences	Different clients have different preferred candidate documents, although they are searching for the same query.
Type 2	Document Label Distribution	Different clients hold candidate documents with different label distribution while the conditional feature distirbution is the shared.
Type 3	Click Preferences	Different clients have various preferred click behaviours when searching for the same query.
Type 4	Data Quantity	Different clients have different frequency on issuing queries and interacting with the searching system.

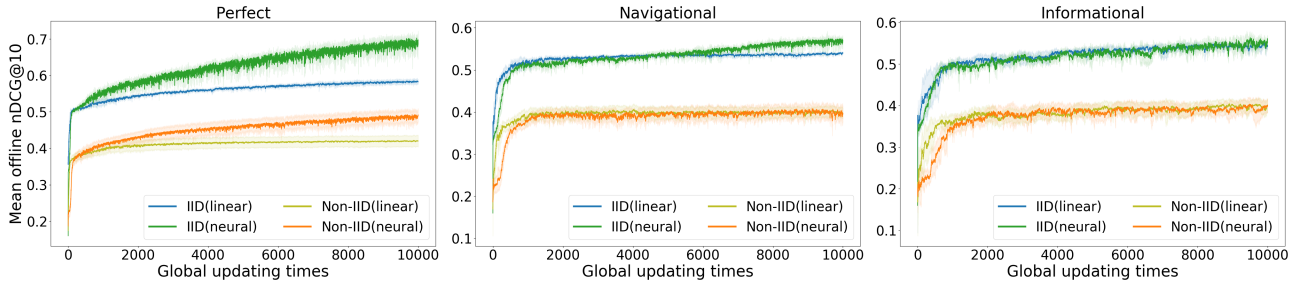


Figure 3: Offline performance (nDCG@10) on Type 1 data; results averaged across dataset splits and experimental runs.

Table 2: Online performance on Type 1 data, averaged across dataset splits and experimental runs. Significant differences between IID and non-IID are indicated by ▲ ($p < 0.05$)

ranker	data types	*perfect*	*navigational*	*informational*
linear	IID	1002.36 ▲	872.12 ▲	894.95 ▲
	non-IID	648.71	546.25	566.23
neural	IID	1061.57 ▲	834.08 ▲	842.87 ▲
	non-IID	668.38	505.64	490.29

In summary, we find that if data is distributed in a non-IID manner across clients according to Type 1, the effectiveness of FOLTR (and specifically of FPDGD) is seriously affected.

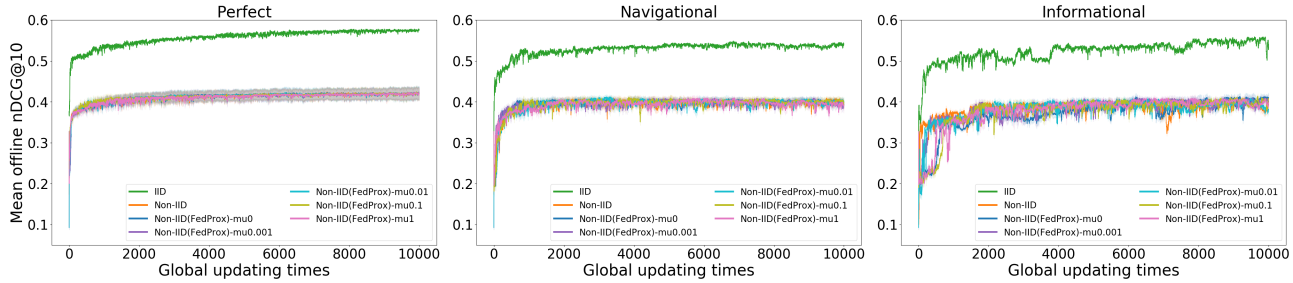
5.3 Dealing with Type 1 non-IID Data

The employed state-of-the-art FPDGD method is based on the FedAvg algorithm. The fact that FPDGD is affected by non-IID data may be due to the underlying federation algorithm, i.e. FedAvg itself. In federated learning literature, variations of this federation algorithm have been proposed to tackle the non-IID data problem directly. We select two of such methods, FedProx [25] and FedPer [1], and adapt them to the FPDGD method.

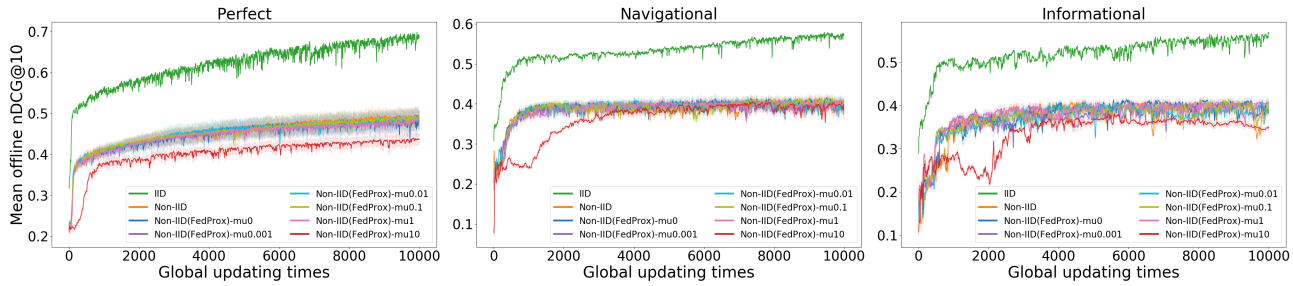
FedProx [25] improves the local objective of FedAvg. Specifically, it introduces an additional L_2 regularisation term (weighted according to a hyper-parameter μ) in the local objective function to limit the distance between the local model and the global model. We provide details of our adaptation of FedProx to FPDGD in the online appendix; the use of FedProx adds little computational overhead. However, the main drawback is that the hyper-parameter μ needs to be carefully tuned: a large μ may slow the convergence by forcing the updates to get close to the initial point, while a small μ may not make much difference compared to the use of FedAvg.

FedPer [1] tackles the presence of non-IID exclusively for deep neural networks by separating them into base layers and personalisation layers. The base layers are trained collaboratively through FedAvg, where all clients share the same base layers. Instead, the personalisation layers are trained locally using the clients' local data with stochastic gradient descent (SGD). This procedure works as follows: after initialisation, each client merges and updates its base and personalised layers locally using an SGD style algorithm. Each client only sends its base layers to the global server. The server updates the globally-shared base layers using FedAvg and sends back again the updated ones to each client. Intuitively, the base layers are updated globally to learn common high-level representations. In contrast, the distinct personalisation layers never leave the local device and capture the personalisation aspects required by the clients. Except for the training and the maintenance of the local personalisation layers, FedPer is quite similar to FedAvg. FedPer, however, reduces the communication costs as only part of the whole model is transferred and has shown enhanced learning performance under highly skewed non-IID data [1].

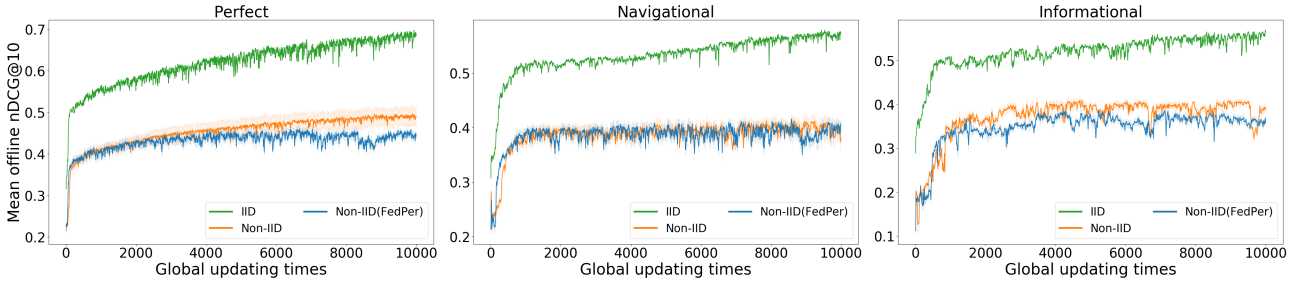
Our experimental results on FedProx and FedPer are shown in Figure 4; for FedProx we explored $\mu \in \{0.001, 0.01, 0.1, 1, 10\}$. The results clearly show that these federated learning methods, which successfully deal with non-IID data in general machine learning tasks, are not effective in the FOLTR context. In fact, not only do these methods not overcome the gap in effectiveness between IID and non-IID setups, but they even only provide limited improvements, if any, compared to FPDGD with FedAvg. This is an important finding because: (1) it shows a realistic case in which non-IID data largely affects FOLTR effectiveness, and (2) it shows that current methods developed in general FL for non-IID data do not



(a) **intent-change (linear ranker) - FedProx**



(b) **intent-change (neural ranker) - FedProx**



(c) **intent-change (neural ranker) - FedPer**

Figure 4: Offline performance on Type 1 data for FedProx and FedPer; results averaged across dataset splits and experimental runs.

work in FOLTR. Thus, a strong need for new methods specialised in the FOLTR settings emerges from these findings.

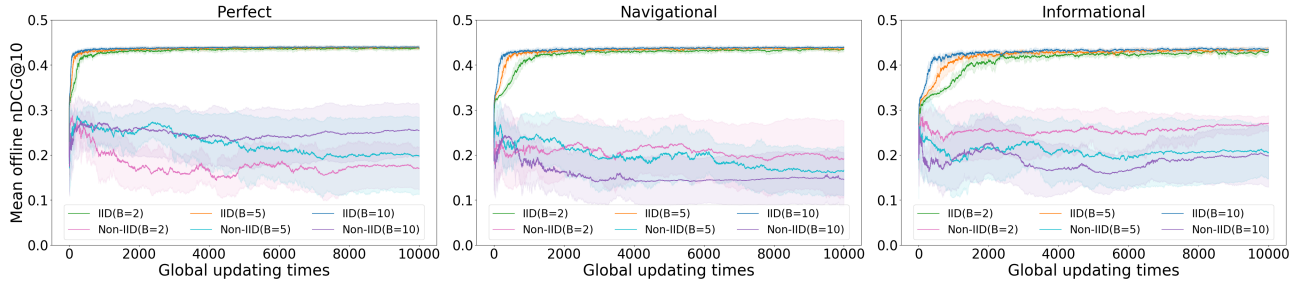
6 TYPE 2: DOCUMENT LABEL DISTRIBUTION SKEWNESS

Document label distribution skewness (Type 2) is a widely recognised type of non-IID data type in federated learning. In this setting, the label distributions $P_i(y)$ in each client are different while the conditional feature distribution $P_i(x|y)$ is shared across the clients. In terms of FOLTR, this is equivalent to the following situation. Assume a document is evaluated across the r-level relevance grades, from *not relevant* (0) to *perfectly relevant* $(r-1)$; then the label distribution on each client is such that, for client i, the probability of holding documents with relevance label k is $P_i(R=k)=p_k$, where $\sum_{k=0}^{r-1} p_k = 1, \forall k, p_k \in [0,1]$

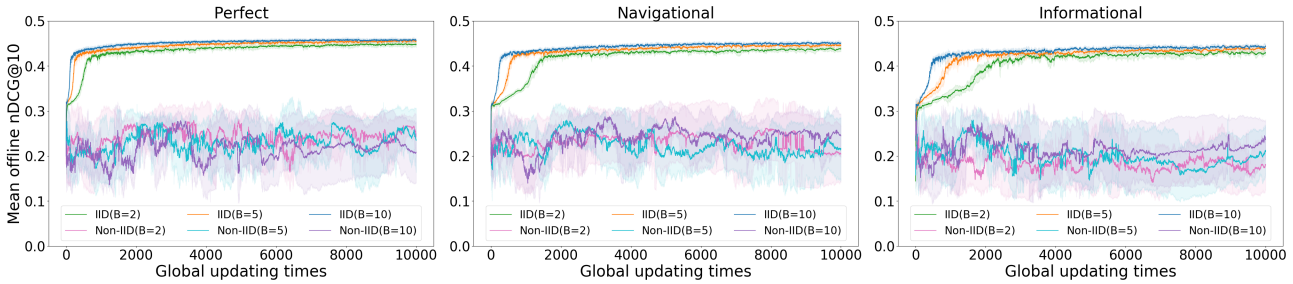
In practice, this may be represented by a situation like the following. Several hospitals are collaboratively creating a FOLTR ranker for clinical-decision-support [35, 36]. Certain hospitals hold a significantly larger portion of highly relevant health records for a certain disease, while some only a small fraction. In this circumstance, the document label distribution is skewed. Under the context of email search [30], different clients might have unique strategies for managing personal emails [46]. Some clients frequently clean up their inboxes and use folders to organise emails. In contrast, some hardly use folders or delete irrelevant messages, resulting in different label distribution when following a learning-to-rank approach.

6.1 Simulating Type 2 non-IID Data

In this section, we discuss how we synthetically simulate Type 2 non-IID data and IID data to baseline in the FOLTR effectiveness. For these experiments, we use the popular datasets MSLR-WEB10k [34] (10,000 queries), Yahoo [6] (29,900 queries) and Istella-S [27] (33,018 queries). We report the results for MSLR-WEB10k in the paper; results on the other datasets are similar and are provided in the online appendix. We simulate $|C|$ clients with each client performing

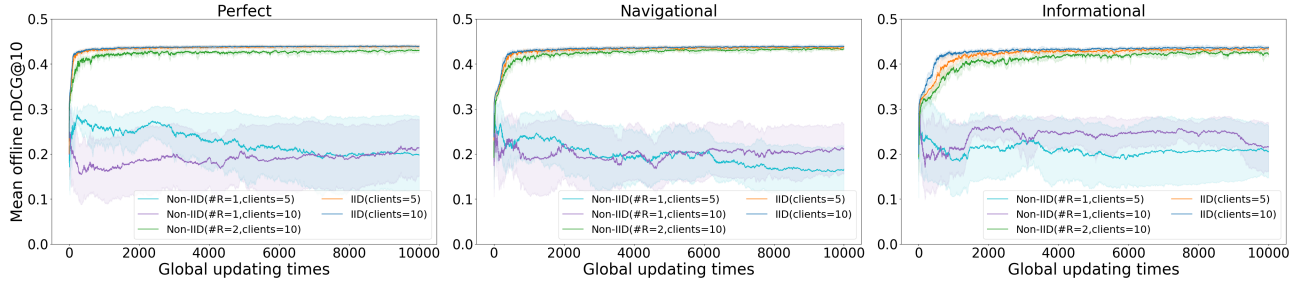


(a) **MSLR-WEB10k (linear ranker)**

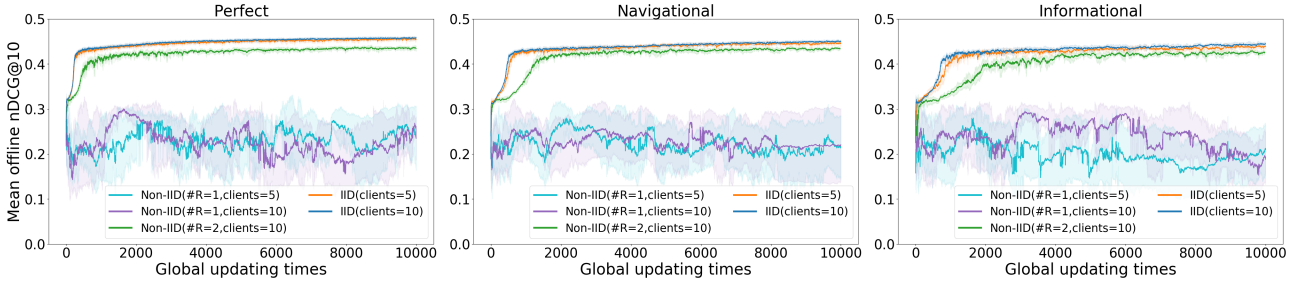


(b) **MSLR-WEB10k (neural ranker)**

Figure 5: Offline performance (nDCG@10) on MSLR-WEB10k for Type 2 ($\#R = 1$), under three instantiations of SDBN click model and three local updates setting ($B \in \{2, 5, 10\}$); results averaged across all dataset splits and experimental runs.



(a) **MSLR-WEB10k (linear ranker)**



(b) **MSLR-WEB10k (neural ranker)**

Figure 6: Offline performance (nDCG@10) on MSLR-WEB10k for Type 2 ($\#R = 2$), under three instantiations of SDBN click model with local updates setting ($B = 5$); results averaged across all dataset splits and experimental runs.

B interactions (queries) locally to contribute to each global model update and restrict the global communication times $T = 10,000$. For simulating querying behaviour, for each client participating in the federated OLTR, we sample B queries randomly, in line with previous work on FOLTR [21, 43]. For each query, we use the local ranking model (i.e. that held by the client) to rank documents; we limit SERP to 10 documents. For the click behaviour, we rely on the

Table 3: Online performance on MSLR-WEB10k for Type 2 ($\#R = 1$), averaged across dataset splits and runs.

		linear ranker			*neural ranker*		
	click	$B = 2$	$B = 5$	$B = 10$	$B = 2$	$B = 5$	$B = 10$
IID	*per.*	742.10	778.56	798.05	716.55	781.18	815.8
	nav.	698.25	743.35	771.04	649.83	728.96	775.87
	inf.	672.23	722.23	757.10	612.76	693.35	748.64
non-IID	*per.*	1589.23	1589.23	1589.23	1589.23	1589.23	1589.23
	nav.	1589.23	1589.23	1589.23	1589.23	1589.23	1589.23
	inf.	1589.23	1589.23	1589.23	1589.23	1589.23	1589.23

same SDBN click models as Section 5. We train both a linear ranker and a neural ranker same as Wang et al. [43].

We specifically consider two types of non-IID data for Type 2: non-IID subtype 1 and non-IID subtype 2. The main difference between the two types is the number of different labels (i.e. the graded relevance assessments) in each client's local dataset. Following similar partitioning strategies by Li et al. [24], suppose each client only has data samples for k different labels. We first generate all possible k-combinations of the relevance set R and randomly assign to $\binom{R}{k}$ clients. Then, for the query-document pairs of each label, we randomly and equally divide them into the clients who own the label. In this way, the number of labels in each client is fixed, and there is no overlap between the samples of different clients.

In non-IID subtype 1, each client only holds query-document pairs from one specific value of relevance label. We use $\#R = 1$ to denote this partitioning strategy. This federated setup involves $|C| = 5$ clients. We also vary the local updating time $B \in \{2, 5, 10\}$ to investigate the impact of local updating with a fixed global communication time $T = 10,000$. For non-IID subtype 2, each client holds data samples from two relevance labels – we denote this as $\#R = 2$. We simulate $|C| = 10$ clients and for fair comparison between the two non-IID subtypes, we also simulate $|C| = 10$ clients for $\#R = 1$ (with each label distributed on two different clients).

The IID experimental setting is the same as the non-IID in terms of federation, ranker parameters and evaluation procedure, except that each client now randomly picks a query from the whole training set with all graded judgements during the training period.

6.2 Impact of Type 2 non-IID Data

The offline performance for $\#R = 1$ on MSLR-WEB10k is shown in Figure 5 and the corresponding online performance is shown in Table 3. From the offline results, it is clear that the rankers learned with non-IID data under-fit the generalized held-out test set under all three settings of local updating times (B). For the *perfect* click model, a larger number of B achieves better test performance. However, when it comes to noisier clicks (*navigational* or *informational*), the trend is reversed, although differences are minimal and the model performance fluctuates. For the online results, all non-IID settings appear to over-fit the maximum value ($online\_ndcg = 1589.23$) as each client's local data only contains data from one relevance label.

Offline results for $\#R = 2$ are shown in Figure 6. In this case, the effectiveness of the learnt rankers is much higher than for $\#R = 1$: a diversity in labels held by a client prevents major losses in FOLTR effectiveness. This result is also consistent with previous results in general federated learning with non-IID data: Li et al. [24] found that the most challenging setting is when each client only has data samples from a single class (label). We further note that another reason for the performance gap is the pairwise loss used in FPDGD [43]: when each client only has one relevance label, it is hard to infer preferences between document pairs (as they both have the same label). However, given labels from two levels of relevance ($\#R = 2$), pairwise differences can be effectively inferred. This suggests that the results obtained here for Type 2 data may not generalise to other FOLTR methods beyond FPDGD if they do not rely on the pairwise preference mechanism. We further note, however, that FPDGD is the current state-of-the-art method and that the only available alternative [21] displays highly variable and sensibly worse performance compared to FPDGD [44, 45]. Therefore, new methods of FOLTR must also be validated in the presence of Type 2 data.

In summary, we find that if data is distributed in a non-IID manner across clients according to Type 2, the effectiveness of FOLTR (and specifically of FPDGD) is seriously affected in the case of $\#R = 1$; however, if $\#R = 2$ then gaps in effectiveness compared to IID settings are minimal.

6.3 Dealing with Type 2 non-IID Data

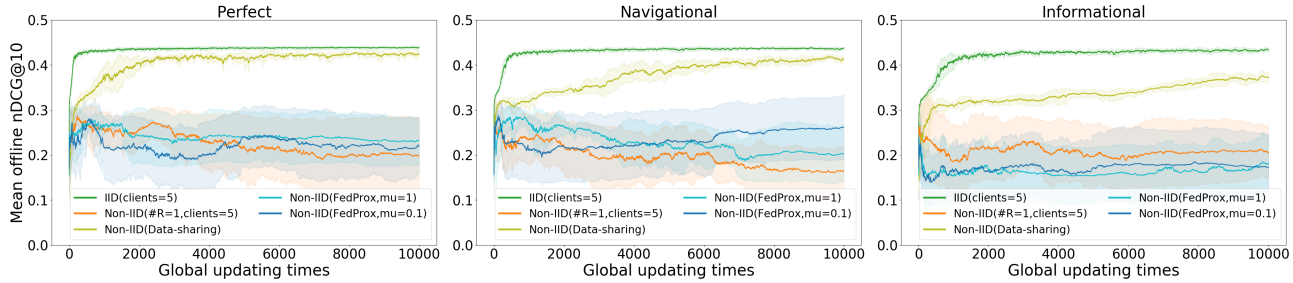
To mitigate the effect of Type 2 non-IID data, we investigate three existing methods from the federated learning literature: Data-sharing, FedProx and FedPer. FedProx and FedPer have been described in Section 5.3. Data-sharing was first proposed by Zhao et al. [52]. They attribute the performance reduction observed on non-IID data to the weight divergence, which is further affected by the divergence between the local data distribution and the overall distribution. They then introduce a straightforward idea to improve FedAvg: slightly reduce the divergence that causes the global model to underperform. This can be achieved as follows. A globally shared dataset G characterised by the overall data distribution is centralised on the server, and a warm-up global model is trained from G. Then, a random α proportion of G is sent to all clients to update the local model by both local training data and the shared data from G. Lastly, the server aggregates the local models from the clients and updates the global model with FedAvg. Experimental results on machine learning tasks show that data sharing can significantly enhance the global model performance in the presence of non-IID data. However, the shortcomings are also pronounced. It is challenging to collect uniformly distributed global datasets in real-world scenarios because either the global server needs some prior knowledge about the local data distributions or each client needs to share parts of the local data (violating the privacy requirement underlying FL).

Figure 7 reports the results for Data-sharing, FedProx and FedPer on MSLR-10k dataset under label distribution skewness $\#R = 1$. We randomly select 10% of the entire dataset as the globally shared data and simulate $|C| = 5$ clients with $B = 5$ local updates before each global update. Results show that the global performance can be significantly enhanced with data-sharing for both linear and neural rankers. On the other hand, neither FedProx nor FedPer provides statistically significant gains over the basic FPDGD on Type 2 non-IID data (with $\#R = 1$).

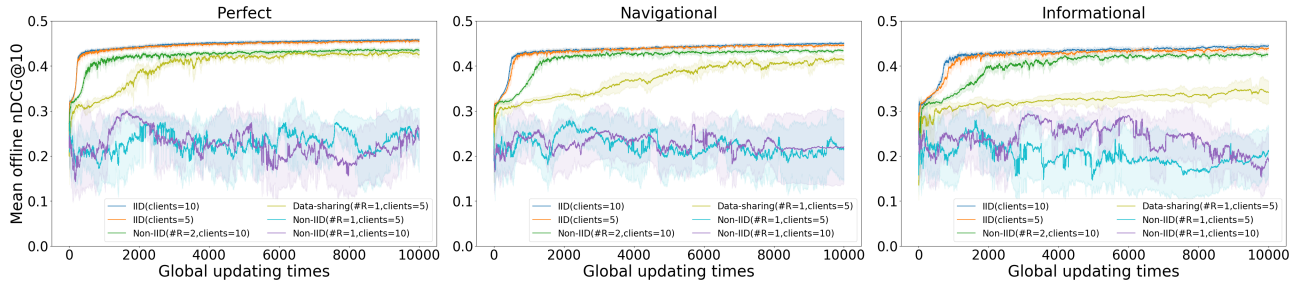
7 OTHER DATA TYPES

7.1 Type 3: Click Preferences

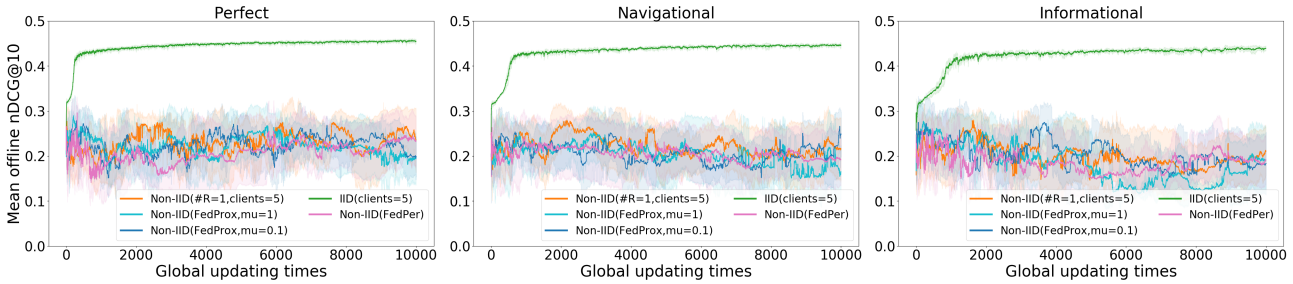
Next, we consider as a source of non-IID data the noise and biases caused by the different click preferences arising from different



(a) **MSLR-WEB10k (linear ranker)**



(b) **MSLR-WEB10k (neural ranker) - data sharing**



(c) **MSLR-WEB10k (neural ranker) - FedProx & FedPer**

Figure 7: Offline performance on MSLR-10k when using Data-sharing, FedProx and FedPer on Type 2 non-IID data (with $\#R = 1$); results averaged across dataset splits and experimental runs.

clients that participate in the FOLTR training; we term this type of non-IID data as *click preference skewness* (Type 3).

The mechanism to emulate non-IID data of Type 3 and IID data baseline in our FOLTR experiments is as follows. We study two widely used click models: the *Simplified Dynamic Bayesian Network* (SDBN) click model [7] and the *Position-Based Model* (PBM) [10]. For non-IID settings in SDBN, each client chooses one of three widely-used instantiations of SDBN, namely *perfect, navigational, informational*. For the non-IID settings with PBM, we generate 5 instantiations based on varying the $\eta \in \{0, 0.5, 1, 1.5, 2\}$ parameter: each client is represented by one click type. Thus the federated setup involves 3 clients for SDBN clicks and 5 for PBM. We set the local updating time $B = 5$ with fixed global communication times $T = 10,000$. In the IID setting, at every time, each client is simulated based on a click model randomly picked from all click models instantiations detailed above and used in the non-IID setting; this provides a fair comparison between the IID and non-IID settings. We experiment on MSLR-WEB10k, Yahoo and Istella-S.

For both online and offline performance, and all datasets, our experimental results show that the difference between non-IID and IID data for Type 3 is not significant; for further details we refer the reader to the Appendix in this paper and the online appendix.

7.2 Type 4: Data Quantity

Finally, we consider the case of *data quantity skewness* (Type 4); this occurs when the number of training data varies across different clients. It is a common scenario in real-world applications. For example, in FOLTR, some clients tend to issue more queries and interact more with the searching system than others. Thus, they have more data for training than others. The situation represented by Type 4 may occur in combination with the other data types. In our empirical experiments, we have studied Type 4 data both on its own and combined with the document preferences skew (Type 1) and the document label distribution skew with $\#R = 1$ (Type 2).

Type 4 data is simulated by assigning different numbers of queries (Q) to each client during the same local updating period, thus leading to different local updating times for each client. The

number of queries varies in {1, 3, 5, 7, 9} and we simulate $|C| = 5$ clients in total with fixed global communication times $T = 10,000$. Experiments are carried out on MSLR-WEB10k, Yahoo and Istella-S.

When mixing other non-IID types with Type 4, we follow the same experimental settings of previous non-IID types, and we also assign different numbers of queries to each client (from {1, 3, 5, 7, 9}) during the same local updating period. Instead, in the IID simulation, each client has 5 iterations of searching for different queries. For both IID and non-IID, we use SDBN click models for click simulation and train a linear ranker using FPDGD on MSLR-WEB10k for Type 1 with $\#R = 1$, and the dataset from Zhuang and Zuccon [55] for Type 2.

Empirical results[3] show that if data is distributed in a non-IID manner across clients according to Type 4, the effectiveness of FPDGD is not impacted. We stress that this result may be specific to FPDGD because it uses the FedAvg paradigm and does not generalise to other FOLTR methods.

8 OUTLOOK AND DISCUSSION

In this paper, we provide a new perspective on the problem of data distribution across clients for federated online learning to rank. Next, we summarise the our key findings and draw directions for future research.

Impact of non-IID data. We found that the presence of non-IID characteristics in the distribution of document preferences (Type 1) and specific cases of document labels (Type 2) have severe effects on the effectiveness of FPDGD. Conversely, if data is distributed across clients in a non-IID manner concerning click preferences (Type 3) or data quantity (Type 4), no significant effects on the quality of FPDGD are observed. These findings contribute an understanding of under which data distributions it is safe to use FOLTR and when it is not. We believe this paper will encourage researchers to include non-IID data settings when evaluating new FOLTR methods.

Calling for FOLTR methods to address non-IID issues. Our paper charts directions to direct future work on non-IID data in FOLTR concerning the creation of techniques that provide remedies to Type 1 and 2, while deeming solutions for Type 3 and 4 data less critical. Importantly, we show that existing solutions employed in general federated learning to mitigate the non-IID data problem do not apply to the FOLTR setting, despite some of these non-IID cases (and especially Type 1) being likely to occur across many FOLTR systems. Thus, researching how to address non-IID data in FOLTR is a worthwhile area of investigation.

Privacy should be a high priority when dealing with non-IID data. Our analysis found that only the data-sharing technique could address to significant extents Type 2 non-IID data. However, this and similar methods, although performing well, require the prior knowledge about the users' local data distributions – and thus require users to share private data, largely defeating the purpose of federated learning. We note that recent work has considered the sharing of synthetic, rather than real, data [41]. In such a setting, real data would be used by each client to generate synthetic data, and the synthetic data only would be shared in the federation. However, we could not find evidence of the loss in effectiveness associated with the use of synthetic rather than real data in the data-sharing scheme. Furthermore, it is unclear what the privacy guarantees are in such a synthetic data sharing scheme. Specifically, we wonder whether the use of synthetic data could jeopardise privacy as this synthetic data is generated from the real data: thus analysis of the synthetic data may reveal key aspects of and information contained in the real data. Thus, how to guarantee user's privacy needs when designing effective FOLTR algorithms on non-IID data is still an open question.

Real-world datasets and benchmarks for FOLTR with non-IID data are need. The experiments put forward in this perspective paper to substantiate our views on the non-IID data problem in FOLTR are based on simulations. While simulations are prevalent in information retrieval and especially in its evaluation [2, 3, 9, 28, 51], a key aspect we had to simulate was the nature of the non-IID data, including their distributions. On one hand, this allows us to carefully control the experiments; on the other it limits the generalisability of the findings to real non-IID data that may occur in FOLTR settings. We therefore want to conclude with a call for action for information retrieval practitioners in this area: there is the pressing need for FOLTR benchmark datasets that provide standard simulations on real-world non-IID scenarios as well as standard hyper-parameter settings so that future FOLTR algorithms can be fairly studied.

9 CONCLUSION

The goal of FOLTR is to learn an effective ranker in a federated (without the need for searchable and interaction data to reside on a central server) and online (by exploiting users clicks on SERPs as they occur) manner. In such a FOLTR setup, user data and interactions reside with the user's client, and not in a central server. Clients then do not need to share such data. Instead, they only share ranker updates with a central server whose responsibility is to collect such updates from the clients and aggregate them into a global model. The global model is then pushed back to the clients in an iterative manner as search interactions occur.

Despite federated learning receiving substantial attention, research in FOLTR is still in its early stages, with only two methods available at the time of writing [21, 43]. Importantly, studies that have proposed FOLTR methods have ignored an important issue that has been shown to affect the performance of federated learning systems [53]: that of the data not being distributed across the federated clients in an identical and independent manner (non-IID data). This paper provides the first analysis of the impact of non-IID data on FOLTR and it charts directions for future research. Our findings and observations may be valid also in other contexts that consider to create a ranker from interaction data in a federated manner, e.g., in federated counterfactual leaning to rank [23].

We make code, experimental details and results available at https://github.com/ielab/2022-SIGIR-noniid-foltr.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their insightful feedback in further shaping the paper. We would also like to thank Dr Bevan Koopman and Dr Harrisen Scells for their thoughtful comments on earlier drafts of this paper.

[3] In the Appendix in this paper and in the online appendix.

REFERENCES

[1] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. 2019. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818* (2019).

[2] Leif Azzopardi. 2016. Simulation of interaction: A tutorial on modelling and simulating user interaction and search behaviour. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 1227–1230.

[3] Krisztian Balog, David Maxwell, Paul Thomas, Shuo Zhang, and Bloomberg London. 2021. Report on the 1st Simulation for Information Retrieval Workshop (Sim4IR 2021) at SIGIR 2021. (2021).

[4] Paul N Bennett, Ryen W White, Wei Chu, Susan T Dumais, Peter Bailey, Fedor Borisyuk, and Xiaoyuan Cui. 2012. Modeling the impact of short-and long-term behavior on search personalization. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*. 185–194.

[5] Mark J Carman, Fabio Crestani, Morgan Harvey, and Mark Baillie. 2010. Towards query log based personalization using topic models. In *International Conference on Information and Knowledge Management*. 1849–1852.

[6] Olivier Chapelle and Yi Chang. 2011. Yahoo! learning to rank challenge overview. In *Proceedings of the Yahoo! Learning to Rank Challenge*. PMLR, 1–24.

[7] Olivier Chapelle and Ya Zhang. 2009. A dynamic bayesian network click model for web search ranking. In *International Conference on World Wide Web*. 1–10.

[8] Charles L Clarke, Nick Craswell, and Ellen M Voorhees. 2012. *Overview of the TREC 2012 web track*. Technical Report.

[9] Michael D Cooper. 1973. A simulation model of an information retrieval system. *Information Storage and Retrieval* 9, 1 (1973), 13–32.

[10] Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. 2008. An experimental comparison of click position-bias models. In *International Conference on Web Search and Data Mining*. 87–94.

[11] Moming Duan, Duo Liu, Xianzhang Chen, Yujuan Tan, Jinting Ren, Lei Qiao, and Liang Liang. 2019. Astraea: Self-balancing federated learning for improving classification accuracy of mobile deep learning applications. In *IEEE International Conference on Computer Design*. IEEE, 246–254.

[12] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. 2020. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. *Advances in Neural Information Processing Systems* 33 (2020).

[13] Songwei Ge, Zhicheng Dou, Zhengbao Jiang, Jian-Yun Nie, and Ji-Rong Wen. 2018. Personalizing search results using hierarchical RNN with query-aware attention. In *International Conference on Information and Knowledge Management*. 347–356.

[14] M Rami Ghorab, Dong Zhou, Alexander O'connor, and Vincent Wade. 2013. Personalised information retrieval: survey and classification. *User Modeling and User-Adapted Interaction* 23, 4 (2013), 381–443.

[15] Avishek Ghosh, Jichan Chung, Dong Yin, and Kannan Ramchandran. 2020. An efficient framework for clustered federated learning. *arXiv preprint arXiv:2006.04088* (2020).

[16] Florian Hartmann, Sunah Suh, Arkadiusz Komarzewski, Tim D Smith, and Ilana Segall. 2019. Federated learning for ranking browser history suggestions. *arXiv preprint arXiv:1911.11807* (2019).

[17] Morgan Harvey, Fabio Crestani, and Mark J Carman. 2013. Building user profiles from topic models for personalised search. In *International Conference on Information and Knowledge Management*. 2309–2314.

[18] Katja Hofmann. 2013. Fast and reliable online learning to rank for information retrieval. In *ACM SIGIR Forum*, Vol. 47. ACM New York, NY, USA, 140–140.

[19] Rolf Jagerman, Harrie Oosterhuis, and Maarten de Rijke. 2019. To model or to intervene: A comparison of counterfactual and online learning to rank from user interactions. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*. 15–24.

[20] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. 2021. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning* 14, 1–2 (2021), 1–210.

[21] Eugene Kharitonov. 2019. Federated online learning to rank with evolution strategies. In *International Conference on Web Search and Data Mining*. 249–257.

[22] Anusha Lalitha, Osman Cihan Kilinc, Tara Javidi, and Farinaz Koushanfar. 2019. Peer-to-peer federated learning on graphs. *arXiv preprint arXiv:1901.11173* (2019).

[23] Chang Li and Hua Ouyang. 2021. Federated Unbiased Learning to Rank. *arXiv preprint arXiv:2105.04761* (2021).

[24] Qinbin Li, Yiqun Diao, Quan Chen, and Bingsheng He. 2022. Federated Learning on Non-IID Data Silos: An Experimental Study. In *IEEE International Conference on Data Engineering*.

[25] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems* 2 (2020), 429–450.

[26] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. 2020. On the Convergence of FedAvg on Non-IID Data. In *8th International Conference on Learning Representations*.

[27] Claudio Lucchese, Franco Maria Nardini, Salvatore Orlando, Raffaele Perego, Fabrizio Silvestri, and Salvatore Trani. 2016. Post-learning optimization of tree ensembles for efficient ranking. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*. 949–952.

[28] David Maxwell and Leif Azzopardi. 2016. Simulating interactive information retrieval: Simiir: A framework for the simulation of interaction. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 1141–1144.

[29] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1273–1282.

[30] Kanika Narang, Susan T Dumais, Nick Craswell, Dan Liebling, and Qingyao Ai. 2017. Large-scale analysis of email search and organizational strategies. In *ACM SIGIR Conference on Human Information Interaction and Retrieval*. 215–223.

[31] Harrie Oosterhuis and Maarten de Rijke. 2018. Differentiable unbiased online learning to rank. In *International Conference on Information and Knowledge Management*. 1293–1302.

[32] Harrie Oosterhuis and Maarten de Rijke. 2021. Unifying Online and Counterfactual Learning to Rank: A Novel Counterfactual Estimator that Effectively Utilizes Online Interventions. In *International Conference on Web Search and Data Mining*. 463–471.

[33] Harrie Oosterhuis, Anne Schuth, and Maarten de Rijke. 2016. Probabilistic multileave gradient descent. In *European Conference on Information Retrieval*. Springer, 661–668.

[34] Tao Qin and Tie-Yan Liu. 2013. Introducing LETOR 4.0 datasets. *arXiv preprint arXiv:1306.2597* (2013).

[35] Kirk Roberts, Matthew Simpson, Dina Demner-Fushman, Ellen Voorhees, and William Hersh. 2016. State-of-the-art in biomedical literature retrieval for clinical cases: a survey of the TREC 2014 CDS track. *Information Retrieval Journal* 19, 1 (2016), 113–148.

[36] Kirk Roberts, Matthew S Simpson, Ellen M Voorhees, and William R Hersh. 2015. Overview of the TREC 2015 Clinical Decision Support Track. In *Text REtrieval Conference*.

[37] Abhijit Guha Roy, Shayan Siddiqui, Sebastian Pölsterl, Nassir Navab, and Christian Wachinger. 2019. Braintorrent: A peer-to-peer environment for decentralized federated learning. *arXiv preprint arXiv:1905.06731* (2019).

[38] Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. 2020. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE transactions on neural networks and learning systems* (2020).

[39] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet Talwalkar. 2017. Federated multi-task learning. *arXiv preprint arXiv:1705.10467* (2017).

[40] Yang Song, Hongning Wang, and Xiaodong He. 2014. Adapting deep ranknet for personalized search. In *International Conference on Web Search and Data Mining*. 83–92.

[41] Han Wang, Luis Muñoz-González, David Eklund, and Shahid Raza. 2021. Non-IID data re-balancing at IoT edge with peer-to-peer federated learning for anomaly detection. In *Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. 153–163.

[42] Kangkang Wang, Rajiv Mathews, Chloé Kiddon, Hubert Eichner, Françoise Beaufays, and Daniel Ramage. 2019. Federated evaluation of on-device personalization. *arXiv preprint arXiv:1910.10252* (2019).

[43] Shuyi Wang, Bing Liu, Shengyao Zhuang, and Guido Zuccon. 2021. Effective and Privacy-preserving Federated Online Learning to Rank. In *ICTIR '21: The 2021 ACM SIGIR International Conference on the Theory of Information Retrieval*. 3–12.

[44] Shuyi Wang, Shengyao Zhuang, and Guido Zuccon. 2021. Federated Online Learning to Rank with Evolution Strategies: A Reproducibility Study. In *European Conference on Information Retrieval*. Springer, 134–149.

[45] Yansheng Wang, Yongxin Tong, Dingyuan Shi, and Ke Xu. 2021. An Efficient Approach for Cross-Silo Federated Learning to Rank. In *International Conference on Data Engineering*. IEEE, 1128–1139.

[46] Steve Whittaker and Candace Sidner. 1996. Email overload: exploring personal information management of email. In *Conference on Human Factors in Computing Systems: Common Ground*. 276–283.

[47] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)* 10, 2 (2019), 1–19.

[48] Timothy Yang, Galen Andrew, Hubert Eichner, Haicheng Sun, Wei Li, Nicholas Kong, Daniel Ramage, and Françoise Beaufays. 2018. Applied federated learning: Improving google keyboard query suggestions. *arXiv preprint arXiv:1812.02903* (2018).

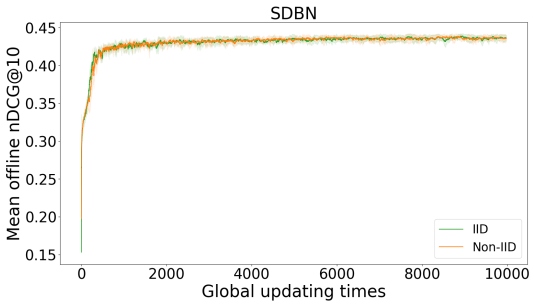
[49] Jing Yao, Zhicheng Dou, and Ji-Rong Wen. 2021. FedPS: A Privacy Protection Enhanced Personalized Search Framework. In *The Web Conference 2021*. 3757–3766.

[50] Jing Yao, Zhicheng Dou, Jun Xu, and Ji-Rong Wen. 2020. RLPer: A Reinforcement Learning Model for Personalized Search. In *The Web Conference 2020*. 2298–2308.

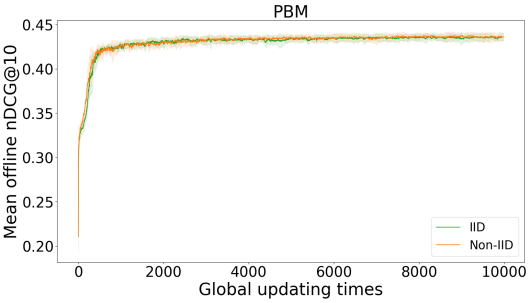
[51] Yinan Zhang, Xueqing Liu, and ChengXiang Zhai. 2017. Information retrieval evaluation as search simulation: A general formal framework for ir evaluation. In

Table 4: Instantiations of SDBN click model for simulating user behaviour in experiments. $rel(d)$ denotes the relevance label for document d. Note that in the intent-change dataset, only two-levels of relevance are used. We demonstrate the values for intent-change in bracket.

	$P(click = 1 \mid rel(d))$				
rel(d)	0	1	2	3	4
perfect	0.0 (0.0)	0.2 (1.0)	0.4 (-)	0.8 (-)	1.0 (-)
navigational	0.05 (0.05)	0.3 (0.95)	0.5 (-)	0.7 (-)	0.95 (-)
informational	0.4 (0.3)	0.6 (0.7)	0.7 (-)	0.8 (-)	0.9 (-)
	$P(stop = 1 \mid click = 1, rel(d))$				
rel(d)	0	1	2	3	4
perfect	0.0 (0.0)	0.0 (0.0)	0.0 (-)	0.0 (-)	0.0 (-)
navigational	0.2 (0.2)	0.3 (0.9)	0.5 (-)	0.7 (-)	0.9 (-)
informational	0.1 (0.1)	0.2 (0.5)	0.3 (-)	0.4 (-)	0.5 (-)



(a) **MSLR-WEB10k (linear ranker) under SDBN clicks**



(b) **MSLR-WEB10k (linear ranker) under PBM clicks**

Figure 8: Offline performance (nDCG@10) on MSLR-WEB10k for Type 3, separately under SDBN and PBM click model; results averaged across all dataset splits and experimental runs.

Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval. 193–200.

[52] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. 2018. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582* (2018).

[53] Hangyu Zhu, Jinjin Xu, Shiqing Liu, and Yaochu Jin. 2021. Federated learning on non-IID data: A survey. *Neurocomputing* 465 (2021), 371–390.

[54] Shengyao Zhuang and Guido Zuccon. 2020. Counterfactual Online Learning to Rank. In *European Conference on Information Retrieval*. Springer, 415–430.

[55] Shengyao Zhuang and Guido Zuccon. 2021. How do Online Learning to Rank Methods Adapt to Changes of Intent?. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[56] Linlin Zong, Qiujie Xie, Jiahui Zhou, Peiran Wu, Xianchao Zhang, and Bo Xu. 2021. FedCMR: Federated Cross-Modal Retrieval. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1672–1676.

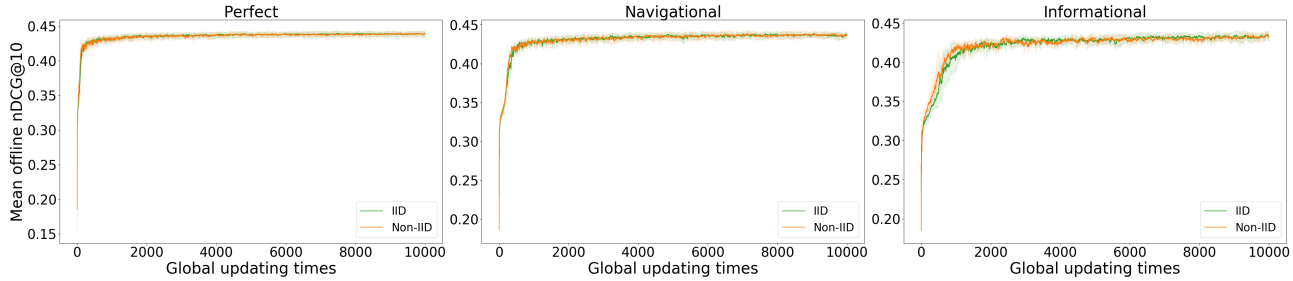
APPENDIX

Table 4 reports the values of the parameters of the *SDBN* click models we used in the experiments.

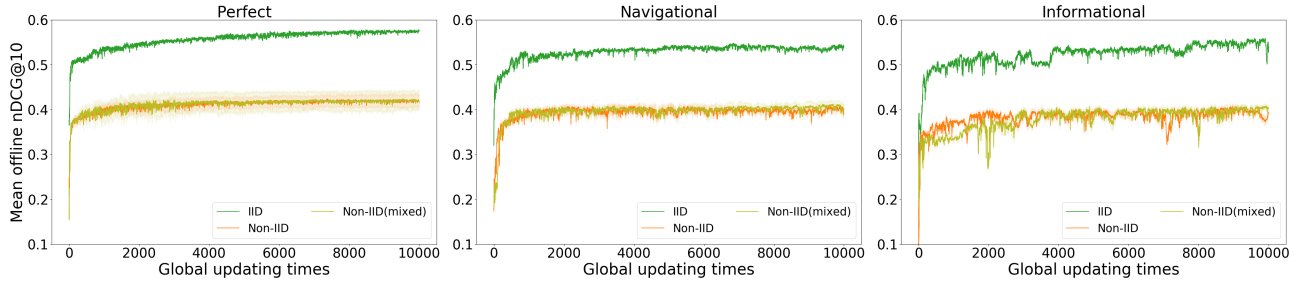
Figure 8 reports the results of our experiments for non-IID data type 3: click preferences.

Figure 9 reports the results of our experiments for non-IID data type 4: data quantity.

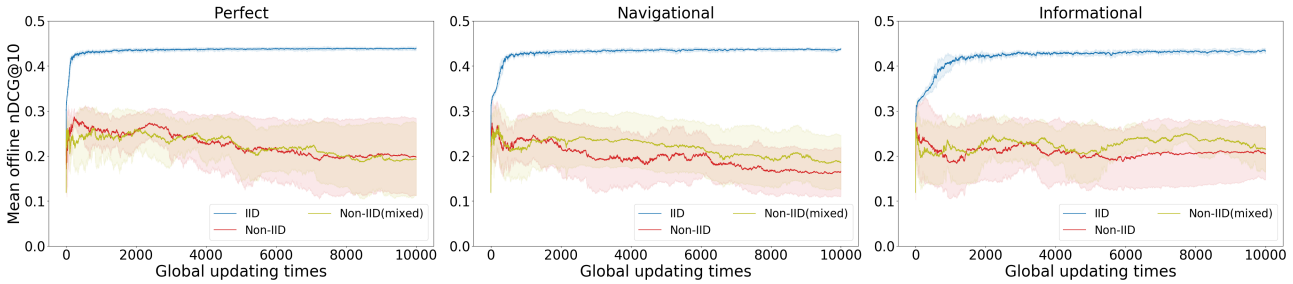
For both type 3 and type 4 data experiments, as well as for other data types, the interested reader can find additional analysis and figures in the online appendix.



(a) **MSLR-WEB10k (linear ranker)**



(b) **intent-change (linear ranker) mixed with type 1**



(c) **MSLR-WEB10k (linear ranker) mixed with type 2 ($\#R = 1$)**

Figure 9: Offline performance (nDCG@10) on MSLR-WEB10k and intent-change for Type 4, under three instantiations of SDBN click model; results averaged across all dataset splits and experimental runs.