

Federated One-Shot Learning with Data Privacy and Objective-Hiding

Maximilian Egger*, *Student Member, IEEE*, Rüdiger Urbanke†, *Senior Member, IEEE*,
and Rawad Bitar*, *Member, IEEE*

*Technical University of Munich, Germany {maximilian.egger, rawad.bitar}@tum.de

†École Polytechnique Fédérale de Lausanne, Switzerland {rudiger.urbanke}@epfl.ch

Abstract—Privacy in federated learning is crucial, encompassing two key aspects: safeguarding the privacy of clients’ data and maintaining the privacy of the federator’s objective from the clients. While the first aspect has been extensively studied, the second has received much less attention.

We present a novel approach that addresses both concerns simultaneously, drawing inspiration from techniques in knowledge distillation and private information retrieval to provide strong information-theoretic privacy guarantees.

Traditional private function computation methods could be used here; however, they are typically limited to linear or polynomial functions. To overcome these constraints, our approach unfolds in three stages. In stage 0, clients perform the necessary computations locally. In stage 1, these results are shared among the clients, and in stage 2, the federator retrieves its desired objective without compromising the privacy of the clients’ data. The crux of the method is a carefully designed protocol that combines secret-sharing-based multi-party computation and a graph-based private information retrieval scheme. We show that our method outperforms existing tools from the literature when properly adapted to this setting.

Index Terms—Federated Learning, Objective-Hiding, Information-Theoretic Privacy, Private Information Retrieval, Secure Aggregation.

I. INTRODUCTION

We consider federated learning (FL), a framework where a federator and a set of clients with private data collaborate to train a neural network. Due to privacy constraints, the clients’ data cannot be directly shared with the federator or among the clients. This privacy concern has been extensively studied in the literature [2]–[6]. There exists a second, often overlooked, privacy concern: ensuring the privacy of the federator’s objective used to train the neural network. This aspect has not been explored in the literature to the same extent.¹

We present a novel approach that ensures the privacy of the clients’ data and simultaneously hides the objective of the federator through a careful combination of a secure aggregation method and a tailored private information retrieval (PIR) scheme. The key challenge of the overall problem arises from the complexity of the computations required for training the

neural network and the inherent heterogeneity of the clients’ data. E.g., training a neural network in a distributed manner typically requires each client to compute a gradient of a loss function with respect to the current neural network using their private data. This is a highly non-linear computation.

We pose a very general research question: *How can a federator use the clients’ private data to accomplish a task, while hiding their objective and maintaining the privacy of the clients’ data?*

An instantiation of this problem is fine-tuning a large-language model for one target objective out of many target objectives known to the clients. We do not impose any assumptions on the clients’ data; hence, the computed function might be different for each client. If the task were linear, the standard technique of *secure aggregation* [2] could be employed effectively. However, for non-linear tasks, such as training a neural network, averaging the final models reached by the clients fails to produce a meaningful model. Even when the clients’ data is similar, their resulting models may differ significantly, and the averaged model may lack meaningful utility.

The challenge of combining the knowledge of multiple non-linear models is commonly referred to as (federated) knowledge distillation, particularly in the context of multiple teacher models and a single student model [8]. In this framework, pre-trained teacher models—representing the clients’ models, or function outputs—transfer their knowledge to the student model, which corresponds to the federator’s model. We draw inspiration from these concepts to enable arbitrary function computations within model training processes. Additionally, the recently introduced concept of auditing for private prediction [9] highlights the importance of exploring privacy-preserving techniques in this domain.

Our work can be viewed as a generalization of private function computation, a well-established framework for outsourcing complex computations while preserving the privacy of the function being evaluated (e.g., [10]–[12]). This computation can be performed on datasets stored either centrally or distributively [13]. A stronger notion than function privacy is the additional protection of the underlying data used for computation, as discussed in [11]. This enhanced privacy is achieved using techniques such as secret sharing. While privacy guarantees can be categorized into information-theoretic and computational approaches, this work focuses exclusively on the former, ensuring privacy even against adversaries with unlimited computational power.

This project is funded by DFG (German Research Foundation) projects under Grant Agreement Nos. BI 2492/1-1 and WA 3907/7-1.

Part of the work was done when RB and ME visited RU at EPFL supported in parts by EuroTech Visiting Researcher Programme grants.

A preliminary version of this paper is accepted for presentation at the IEEE International Symposium on Information Theory 2025 [1].

¹The notion of intention-hiding has only appeared recently in a different setting in vertical FL [7], where the intention of model training is implemented using suitable data preprocessing in the form of a private set intersection.

The foundational concept underpinning private function computation is private information retrieval (PIR) [14], [15]. In PIR, a dataset is distributed across one or more servers, and a client retrieves a specific file or subset of data without revealing which file is of interest. Numerous studies have explored PIR from different angles, including single-server PIR, PIR with replicated data [16], PIR with MDS-coded data [15], PIR with secretly shared data [17], and PIR using graph-based replication, also referred to as non-replicated data storage [18]–[20]. This concept has been further extended to symmetric privacy, which ensures that the client learns nothing about the dataset beyond the specific file requested from the databases [21]. For a comprehensive overview of recent advancements and open challenges in PIR, we refer readers to the surveys [22], [23].

Beyond private function computation [10]–[12], [24]–[27], concepts from PIR have been extended to private function retrieval [14], [15], [28]–[33], private inner product retrieval [34], and private linear transformation [35], [36]. While these frameworks address private computation for linear functions, polynomial functions, or compositions of linear functions, they do not provide solutions for computing arbitrary functions. This limitation introduces new challenges, which we address in this work. Our approach can be seen as a generalized framework for the computation of arbitrary predictors, extending beyond the previously studied settings.

We propose an end-to-end solution for federated one-shot learning that ensures both data and objective privacy. Under the assumption of a limited number of colluding clients, our approach prevents the leakage of private data to other clients or the federator. Specifically targeting federated learning applications, we introduce a novel method that integrates concepts from graph-based (symmetric) private information retrieval, secret sharing, multi-party computation, coded storage, knowledge distillation, and ensemble learning.

Our solution requires a public unlabeled dataset accessible to all clients and the federator, a common and non-restrictive assumption in semi-supervised machine learning problems [37]. In Stage 0, each client is assigned a subset of objectives chosen from a pool of candidate objectives. For each assigned objective, the client trains a local model and uses it to label the shared public dataset. In Stage 1, clients use a carefully tailored secret sharing scheme to share the labels privately among each other and aggregate the received shares. Since secret sharing schemes are linear, the aggregate of the secret shares consists of shares of the aggregated labels. In Stage 2, the federator uses a symmetric PIR scheme to receive only the aggregated labels corresponding to their objective of interest. Thus, enabling the reconstruction of the federator’s model while leveraging data contributions from all clients and maintaining client privacy by observing only aggregated labels. Additionally, the federator’s objective of interest remains hidden from the clients through the use of a PIR scheme.

We focus on one-shot federated learning for several key reasons: (1) iterative schemes could compromise the privacy of the objective, (2) iterative methods incur significant communication overhead due to privacy mechanisms, and (3) collaborative iterative training using a shared public dataset introduces additional challenges. Further details are provided

in Remark 2.

Our contributions are summarized as follows:

- We formulate the general problem and propose a three-stage solution comprising the task assignment stage, the sharing stage and the query stage, with jointly designed codes to minimize the overall communication cost.
- Building on [19], we leverage tools from the duals of Reed-Solomon codes to design a flexible task assignment scheme. Then, we develop a graph-based PIR scheme tailored for the designed private coded storage used in the query stage. This approach generalizes the storage pattern to ramp Secret Sharing (e.g., McEliece-Sarwate Secret Sharing [38]), enhancing efficiency in generating and storing shares of the clients’ labels.
- We extend the framework to incorporate data privacy against the federator, i.e., ensuring no additional information beyond the desired function is leaked to the federator.
- We evaluate the rate of our scheme, demonstrating significant gains over existing PIR methods for graph-based coded data when computational resources are constrained. Additionally, we propose an optimized scheme utilizing star-product PIR for scenarios where computation is inexpensive.

Remark 1. *In scenarios where client privacy is not a primary concern, a simple approach is to independently download all labels from each client and aggregate the results at the federator. For cryptographic guarantees, symmetric privacy—protecting the clients’ results beyond the desired objective—can be achieved through oblivious transfer protocols for individual queries. Our proposed solution adheres to a stronger notion of information-theoretic privacy, safeguarding both individual client data, by only revealing the aggregate of the labels of the objective of interest, as well as the privacy of the objective itself.*

II. RELATED WORK

We review the following fundamental ingredients upon which our scheme is based on: graph-based PIR, secure aggregation in FL, private function computation, symmetric PIR and knowledge distillation.

a) *PIR*: While there has been an abundance of works, we specifically mention those closest to our interest: PIR for MDS coded data was studied, e.g., in [17]. Joint message encoding for PIR was studied in [39], and [40] studied the trade-off storage and download cost in PIR. Although the latter two are conceptually different, the ideas are loosely related. We focus in the following on graph-based PIR, whose methodology is most related to parts of our contribution.

b) *Graph-Based PIR*: The problem of PIR on graph-based data storage was first introduced in [18], in which the replication of files is modeled by (hyper-)graphs, where vertices correspond to storage nodes and (hyper-)edges correspond to files and connect nodes storing the same file. The proposed scheme is proved to be uncritical in regard to collusions as long as the graph exhibits non-cyclic structures. However, privacy of the stored data is not considered. A similar concept has arisen concurrently and termed PIR for non-replicated databases [20], later extended to optimal message sizes [41].

Many follow-up works have considered PIR on different graph structures, e.g., [42] where bounds on the capacity of PIR were derived for specific graphs, in particular for the star-graph (with one universal node storing all the messages), and the complete graph. A linear programming-based bound was given for general graphs. Follow-up works studies the capacity for a $K = 4$ star graph [43]. Privacy of the data through secret sharing was studied in [44] by means of Shamir Secret Sharing for X -secure and T -private PIR in a non-graph setting. Cross-space interference alignment is used to improve the rate of the PIR scheme by efficiently returning multiple information symbols of interest per query. This has later been extended to graph-based PIR for messages encoded by a Shamir Secret Sharing [19], which can be seen as the extension of [11] to X -security. The dual code of a Generalized Reed Solomon is used together with cross-subspace alignment for inference cancellation. Private function computation in the graph-based setting has recently been studied in [31] for X -security and T -privacy. A generalization of cross-subspace alignment using algebraic geometry codes [45] with secret sharing [46] was recently proposed. Those works are concerned with Shamir Secret Sharing, and hence not suitable for this problem. Graph-based secret sharing was independently studied in [47].

Semantic PIR in which the length of the messages may be different was considered in [48]. In [10], the authors studied PIR for replicated data and general functions such that the query space is a vector-space. Theorem 2 in [12] and a result in [49] appear to consider a general set of functions from non-colluding replicated databases. In [13], the authors consider a setting where the user decides how to store the messages on the non-colluding databases. There is no privacy of the data from the servers.

c) Secure Aggregation in FL: Secure aggregation for FL was first introduced in [2]. Follow-up works are concerned with the communication overhead of such methods, e.g., [50]–[54]. Alternative models are considered in, e.g., [55]–[57].

d) Private Function Computation and Distributed Computing: Private function computation was extensively studied in the literature for linear and polynomial functions, cf. [10]–[12], [24]–[27]. Tools from PIR were further applied to distributed computing. In [22], a review and survey on this topic is provided. For instance, polynomial computation from distributed data was studied in [58], and distributed matrix multiplication from MDS coded storage was studied in [59]. Tools from PIR have further been used in FL for private submodel learning termed private read update write [60].

e) Symmetric PIR: The capacity of symmetric PIR (SPIR) where the messages and the randomness are encoded with codes with different parameters are considered in [61]. Symmetric PIR from MDS coded data with potentially colluding databases was considered in [21]. SPIR with user-side common randomness was considered in [62], [63]. Random SPIR was introduced in [63], where the user is interested in a random message rather than a specific one. Closer to our work, symmetric private polynomial computation was studied in [24], where the authors also consider a finite set of candidate polynomial functions. Related to symmetric privacy with multiple servers, the combinations of multiple oblivious transfer protocols was recently studied in [64].

f) Knowledge Distillation in FL: Throughout this work, we make use of ensemble learning methods, which have been extensively studied, e.g., in [65]–[67], and for heterogeneous classifiers in [68]. The concept is also related to the student-teacher model in the setting of multiple teachers [8]. We refer interested readers to the survey in [69] for an extensive review of knowledge distillation in FL.

III. PRELIMINARIES AND SYSTEM MODEL

Notation. We denote finite fields of cardinality q by \mathbb{F}_q . For a natural number a we define the following set notation $[a] \triangleq \{1, \dots, a\}$. For a random variable X , we refer to the entropy as $H(X)$, and for two random variables X and Y , we denote the mutual information as $I(X; Y)$. Similarly, we denote the conditional entropy and the conditional mutual information conditioned on a random variable Z as $H(X | Z)$ and $I(X; Y | Z)$, respectively.

Private Function Computation. Each client $i \in [n]$ holds a private dataset \mathcal{D}_i . In some cases, these datasets consist of distinct samples drawn from the same underlying distribution, referred to as the homogeneous case. In other cases, the datasets are drawn from different distributions, referred to as the heterogeneous case.

Let $\mathcal{F} = \{h_t\}_{t=1}^T$ denote the pool of T candidate functions (or objectives), public, and hence, known to all parties. Among these, the federator is interested in a specific function h_j , where the index j is unknown to the clients. Specifically, the federator aims to compute $h_j \left(\bigcup_{i \in [n]} \mathcal{D}_i \right)$, leveraging the combined data from all clients, $\bigcup_{i \in [n]} \mathcal{D}_i$. Our solution is designed for scenarios where h_j is additively separable, i.e., $h_j \left(\bigcup_{i \in [n]} \mathcal{D}_i \right) = \sum_{i \in [n]} h_j(\mathcal{D}_i)$.

This property is naturally satisfied by linear models, and as we will demonstrate in the next section, it can also be extended to certain non-linear models, including neural networks.

Non-linear Training Processes. We now expand on the previous paragraph and discuss how standard tools from knowledge distillation can ensure additive separability even in cases where highly non-linear training processes are involved. The core idea is that the clients train models to label a public unlabeled dataset for each objective. The function h_t is then the composition of the local model training and the local model applied to the public data.

More precisely, let $w_{i,t}$ be the local model trained at client i for objective t . Let \mathcal{D}_{pub} denote the public dataset consisting of s samples $\{\mathbf{x}_\ell\}_{\ell \in [s]}$. Define $h_t(\mathcal{D}_i) = w_{i,t}(\mathcal{D}_{\text{pub}})$, where $w_{i,t}(\mathcal{D}_{\text{pub}}) \triangleq \{w_{i,t}(\mathbf{x}_\ell)\}_{\ell \in [s]} = \{y_{i,\ell}^{(t)}\}_{\ell \in [s]}$. Here, $y_{i,\ell}^{(t)}$, $\ell \in [s]$, refers to the label of client i and objective t for data sample ℓ in \mathcal{D}_{pub} . Note that $y_{i,\ell}^{(t)}$ is of dimension c , and each entry is quantized to at most $\lfloor q/n+1 \rfloor$ levels. In summary, in this case, h_t is a function that takes as input a training dataset, trains a model, and outputs the labels for a fixed public dataset, i.e., $h_t : \mathcal{D} \mapsto \{y_\ell^{(t)}\}_{\ell \in [s]}$.

Task Assignment. Although the best performance can be reached if all clients compute a function (or train a model) for each objective, training a model for all T objectives might be expensive. Hence, for each t , we assign the task of computing h_t to only a subset of the clients. We model this assignment

of tasks by a hypergraph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with clients $[n]$ represented by vertices, and objectives represented by hyperedges \mathcal{E} . The binary incident matrix $\mathbf{I} \in [0, 1]^{n \times T}$ has its (i, t) entry equal to 1 if client i computes the model for objective t , and 0 otherwise. We assume a symmetric setup, where the column weight of \mathbf{I} is constant and equal to ρ , i.e., exactly ρ clients compute a model for each objective t . We denote by $\mathcal{I}(e_t)$ the set of clients incident with hyperedge e_t . Hence, all clients $i \in \mathcal{I}(e_t)$ compute a model for the objective t . Further, let $\mathcal{I}(i)$ denote the set of all edges incident with client i .

Privacy Guarantees. During the execution of the protocol, clients share messages amongst each other and with the federator. Let \mathcal{M}_i be the set of all messages received by client i , and for a set $\mathcal{T} \subseteq [n]$, let $\mathcal{M}_{\mathcal{T}}$ be the set of all messages received by all clients in \mathcal{T} , i.e., $\mathcal{M}_{\mathcal{T}} \triangleq \{\mathcal{M}_i\}_{i \in \mathcal{T}}$. Let $\mathcal{Q}_{\mathcal{T}}^{(t)}$ be the set of all queries received from the federator by clients $i \in \mathcal{T}$ for objective t . After the sharing stage is complete, let \mathcal{S}_i be the data stored by client i , and $\mathcal{S}_{\mathcal{T}}$ be the data stored by all clients $i \in \mathcal{T}$ and let $\mathbf{Y}_{i,\ell}^{(t)}$ be the random variable representing the prediction of the public sample \mathbf{x}_{ℓ} using objective t . We consider information-theoretic privacy notions, defined formally in the sequel, assuming at most z_s colluding clients that target compromising other clients' individual data privacy and at most z_q clients trying to infer the federator's objective. The multifold privacy guarantees for the clients' data and the federator's objective are formally stated in Definitions 1 to 3.

Definition 1 (Data Privacy from Clients). *No client's data is leaked to any other set of at most z_s colluding clients, i.e., for all $i \in [n]$ and any client collusion set $\mathcal{T}_s \subset [n] \setminus \{i\}$, $|\mathcal{T}_s| \leq z_s$,*

$$\mathbf{I}(\{h_t(\mathcal{D}_i)\}_{t \in \mathcal{I}(i), i \in [n]}; \mathcal{M}_{\mathcal{T}_s} \mid \{h_t(\mathcal{D}_i)\}_{t \in \mathcal{I}(i), i \in \mathcal{T}_s}) = 0.$$

where $\{h_t(\mathcal{D}_i)\}_{t \in \mathcal{I}(i)} = \{\mathbf{Y}_{i,\ell}^{(t)}\}_{t \in \mathcal{I}(i), \ell \in [s]}$ for the special case of one-shot FL.

Since the clients' training data might be correlated, the conditioning ensures that nothing further is leaked beyond what is known to the colluding clients from their own computations.

Definition 2 (Objective-Hiding). *The identity j of the function of interest to the federator is private from any z_q colluding clients, i.e., for each $\mathcal{T}_q \subset [n]$, $|\mathcal{T}_q| \leq z_q$, it holds that*

$$\mathbf{I}(\{\mathcal{Q}_{\mathcal{T}_s}^{(t)}\}_{t \in [T]}, \mathcal{S}_{\mathcal{T}_q}, \mathcal{M}_{\mathcal{T}_q}; j) = 0.$$

Going beyond the above privacy measures, we further require that no information beyond the aggregate clients' function results is revealed to the federator. With \mathcal{A}^i being the answer received by the federator from client i , and $\mathcal{A}^{[n]}$ the set of answers from all clients $i \in [n]$, we have the following definition of data privacy against the federator.

Definition 3 (Data Privacy from Federator). *The federator's knowledge about the clients data is limited to the quantity $\sum_{i=1}^n h_j(\mathcal{D}_i)$ of interest, i.e.,*

$$\mathbf{I}\left(\mathcal{A}^{[n]}, \{\mathcal{Q}_{[n]}^{(t)}\}_{t \in [T]}; \{h_t(\mathcal{D}_i)\}_{t \in \mathcal{I}(i), i \in [n]} \mid \sum_{i \in \mathcal{I}(e_j)} h_j(\mathcal{D}_i)\right) = 0,$$

where for FL we have $h_t(\mathcal{D}_i) = \{\mathbf{Y}_{i,\ell}^{(t)}\}_{\ell \in [s]}$, and $\sum_{i \in \mathcal{I}(e_j)} h_j(\mathcal{D}_i) = \{\sum_{i \in \mathcal{I}(e_j)} \mathbf{Y}_{i,\ell}^{(j)}\}_{\ell \in [s]}$.

In PIR, Definition 2 corresponds to the user privacy, i.e., hiding the identity of the queried file, and Definition 3 is referred to as the symmetric privacy guarantee. The communication cost is formally determined as the size of all transmitted messages, i.e., $H(\mathcal{M}_{[n]}, \mathcal{A}^{[n]})$. The task assignment stage incurs no communication overhead. Therefore, we will analyze R_{share} and R_{PIR} , the rates of the sharing and query (PIR) stages of our scheme, which are formally defined next.

$$R_{\text{share}} = \frac{H(\sum_{i \in \mathcal{I}(e_j)} h_j(\mathcal{D}_i))}{\sum_{i=1}^n H(\mathcal{M}_i)}, R_{\text{PIR}} = \frac{H(\sum_{i \in \mathcal{I}(e_j)} h_j(\mathcal{D}_i))}{\sum_{i=1}^n H(\mathcal{A}^i)}$$

IV. PROBLEM ILLUSTRATION THROUGH THE LENS OF FINE-TUNING LARGE-LANGUAGE MODELS

To illustrate the problem and the principle idea of our solution, we take the example of fine-tuning large language models (LLMs), which recently gained significant attention through the progress and capabilities of generative models such as GPT-4, Llama 3 and Mistral 7B. Imagine a pre-trained and generic LLM suitable for a variety of tasks. The federator is interested in fine-tuning the model according to a specific objective j , for instance, sentiment analysis [70], article classification [71] or question classification [72]. Note that both functions (models) and datasets can differ across objectives. While the clients have suitable labeled data at hand that can be used for supervised learning, their data should be kept private. Knowing the different objectives (or functions), each client i can individually fine-tune the LLM with respect to every possible objective $t \in [T]$, including the objective j of interest. Thereby, each client obtains a model $w_{i,t}$ for each objective of interest. Since the average of clients' models for the same objective trained on their individual data is not meaningful, we make use of a public and unlabelled dataset \mathcal{D}_{pub} , consisting of s samples \mathbf{x}_{ℓ} , $\ell \in [s]$, to transfer the knowledge from clients to the federator. Each client i labels the public samples \mathbf{x}_{ℓ} for each objective t , i.e., each trained LLM $w_{i,t}$, thereby obtaining s labels $y_{i,\ell}^{(t)}$ for each $t \in [T]$.

Since the individual labels contain sensitive information about the clients' data [73], their privacy is as important as that of the fine-tuned LLM models. For the federator, it is beneficial to receive the average of the predictions in order to reconstruct a suitable model [74]. In fact, such averaged predictions for all samples of the public dataset might improve the performance due to the diversity of the clients' data, leading to better generalization results [74]. To make the federator obtain the sum of all predictions without revealing any information about individual labels to the federator or to other clients, we borrow ideas from multi-party computation [75]. Here, each client stores a message, and they collaboratively want to compute the sum of the messages without leaking individual ones to any subset of z_s clients trying to compromise privacy. In FL, such concepts are well-known as the secure aggregation of local models (or gradients) at each iteration [2], [54]. Such secure aggregation techniques lead to high computational overheads, especially when using basic versions of secret sharing such as Shamir Secret Sharing [76].

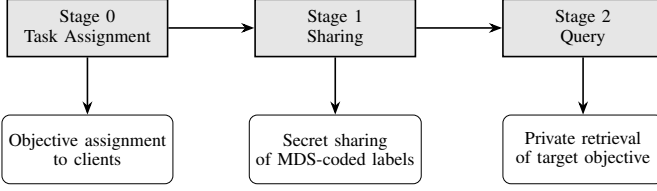


Fig. 1. High-level description of a three-stage protocol that first establishes in two stages (assignment and sharing) an MDS-coded data storage pattern based on secret sharing that encodes the aggregated clients computation results for all objectives, and then queries the result for the objective of interest.

Example 1. To illustrate the challenges, we explain and contrast two approaches to solve this motivating example. Assume for simplicity unbounded compute power of the clients, i.e., each client computes the function result for all objectives, and let $\rho = n = 5$ and $z_s = z_q = z = 1$. The two approaches are: (i) designing a scheme for our framework using known techniques from knowledge distillation, secure aggregation, and methods from the PIR literature [19]; and (ii) rethinking the co-design of secure aggregation and PIR in this setting, through a new coding technique we introduce to lower the communication costs.

We will see that the first approach, while solving the problem, incurs a high communication cost in the sharing phase. The second approach reduces this cost, yet requires the assumption of unbounded computation, i.e., $\rho = n$. Since the method of Approach 2 does not directly generalize to the case where $\rho < n$, we construct a scheme in Section V that build on the concepts from Approach 2 and design a method for arbitrary incident matrices \mathbf{I} with $\rho \leq T$ that alleviate the interferences resulting from an arbitrary choice of \mathbf{I} by leveraging dual properties of Reed-Solomon codes and a careful choice of the evaluation points. Fig. 1 shows the three-stage concept on a high level, and Fig. 2 summarizes the functionality of the sharing and the query stage.

Approach 1 (Simplified Solution for Shamir Secret Sharing). The task assignment stage is trivial as all clients compute the output of all objectives. In the sharing stage, each client i constructs for each sample ℓ and each objective t a secret sharing

$$f_{i,\ell}^{(t)}(x) = y_{i,\ell}^{(t)} + x r_{i,\ell}^{(t)},$$

encoding the private label $y_{i,\ell}^{(t)} \in \mathbb{F}_q^c$ (encoded by a one-hot encoding into a vector of length c , the number of classes) using $z = 1$ term $r_{i,\ell}^{(t)}$ of the size of the label, chosen independently and uniformly at random from \mathbb{F}_q^c . Each client i_1 then sends the evaluation (share) $f_{i_1,\ell}^{(t)}(\alpha_{i_2})$ to client i_2 and receives the share $f_{i_2,\ell}^{(t)}(\alpha_{i_1})$ for all $i_2 \in [n]$ for each label ℓ and objective t . Aggregating the received shares, client i_1 obtains a share $F_\ell^{(t)}(\alpha_{i_1})$ of the sum-secret sharing $F_\ell^{(t)}(x) = \sum_{i=1}^n f_{i,\ell}^{(t)}(x)$, which can be viewed as a codeword of a Generalized Reed Solomon (GRS) code \mathcal{C} with dimension $k_C = 2$ and length n .

In the query stage, to privately retrieve the labels of interest (i.e., the $y_{i,\ell}^{(t)}$'s for objective $t = j$), we design the queries by the following polynomial:

$$q_\ell^{(t,j)}(x) = \delta_\ell^{(t,j)} + x k_\ell^{(t)},$$

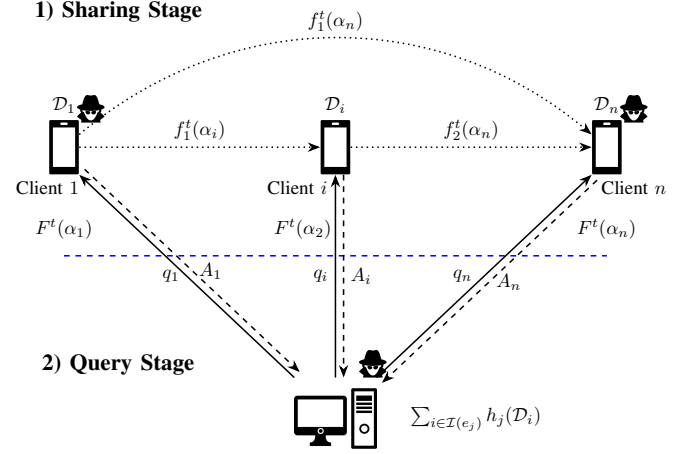


Fig. 2. Illustration of the sharing and query phase of our protocol. The objective of interest is hidden from the curious clients. The clients do not learn about the other clients' results. The federator only learns the aggregate clients' results for the objective of interest.

where $\delta_\ell^{(t,j)} = \begin{cases} 1 & \text{if } t = j \\ 0 & \text{otherwise} \end{cases}$, and $k_\ell^{(t)} \in \mathbb{F}_q^c$ are chosen independently and uniformly at random. Each client i receives the query polynomial evaluated at α_i , and returns $\mathcal{A}_\ell^i = \sum_{t=1}^T F_\ell^{(t)}(\alpha_i) q_\ell^{(t,j)}(\alpha_i)$, which is an evaluation of the degree-2 polynomial

$$F_\ell^{(t)}(x) q_\ell^{(t,j)}(x) = \sum_{i=1}^n y_{i,\ell}^{(j)} + x \left(\sum_{i=1}^n r_{i,\ell}^{(j)} + \sum_{t=1}^T k_\ell^{(t)} \sum_{i=1}^n y_{i,\ell}^{(t)} \right) + x^2 \left(\sum_{t=1}^T k_\ell^{(t)} \sum_{i=1}^n r_{i,\ell}^{(t)} \right),$$

where vector-products are element-wise. Interpolating this polynomial from any subset of the answers $\{\mathcal{A}_\ell^i\}_{i=1}^n$ of size 3 reveals the desired aggregate of the labels $\sum_{i=1}^n y_{i,\ell}^{(j)}$. The communication cost of this scheme is $Tsn(n-1) + 3s$ symbols in \mathbb{F}_q^c since the queries and answers are sent for each of the s labels. Note that we used the same collusion parameter z for storage and query codes, which need not hold true in general.

While the rate of the PIR scheme in Approach 1 can be made optimal by using tools from cross-subspace alignment using multiple Shamir secret sharing schemes, the communication cost in the sharing stage is considerably large for the scheme when $n - z \gg 2$ and amplified by the number of objectives T , which gets clear from the above example. Next, we illustrate how we further reduce communication costs by using ramp secret sharing and an adapted PIR scheme.

Approach 2 (Simplified Solution for McEliece Sarwate Secret Sharing). Each client i constructs for each pair of two labels² $y_{i,\ell}^{(t)}, y_{i,\ell+1}^{(t)} \in \mathbb{F}_q^c, \ell \in \{1, 3, 5, \dots, s-1\}$ and for each objective t a secret sharing

$$f_{i,\ell}^{(t)}(x) = y_{i,\ell}^{(t)} + x y_{i,\ell+1}^{(t)} + x^2 r_{i,\ell}^{(t)}.$$

Each client i_1 then sends the evaluation $f_{i_1,\ell}^{(t)}(\alpha_{i_2})$ to client i_2 and receives a share $f_{i_2,\ell}^{(t)}(\alpha_{i_1})$ from each client $i_2 \in [n]$ for

²Assuming for simplicity an even number of labels s . This idea can be generalized to jointly encoding an arbitrary number of labels.

each label and each objective t . Each client obtains a share $F_\ell^{(t)}(\alpha_{i_1})$ of the sum-secret sharing $F_\ell^{(t)}(x) = \sum_{i=1}^n f_{i,\ell}^{(t)}(x)$. To privately retrieve the labels of interest, we design the following query polynomial:

$$q_\ell^{(t,j)}(x) = \delta_\ell^{(t,j)} + x^2 k_\ell^{(t)},$$

where $\delta_\ell^{(t,j)} = \begin{cases} 1 & \text{if } t = j \\ 0 & \text{otherwise} \end{cases}$, and $k_\ell^{(t)} \in \mathbb{F}_q^c$ are chosen independently and uniformly at random. Each client i receives the query polynomial evaluated at α_i , and returns $A_\ell^i = \sum_{t=1}^T F_\ell^{(t)}(\alpha_i) q_\ell^{(t,j)}(\alpha_i)$, which is an evaluation of the degree-4 polynomial

$$\begin{aligned} F_\ell^{(t)}(x) q_\ell^{(t,j)}(x) &= \sum_{i=1}^n y_{i,\ell}^{(j)} + x \sum_{i=1}^n y_{i,\ell+1}^{(j)} \\ &+ x^2 \left(\sum_{i=1}^n r_{i,\ell}^{(j)} + \sum_{t=1}^T k_\ell^{(t)} \sum_{i=1}^n y_{i,\ell}^{(t)} \right) \\ &+ x^3 \sum_{t=1}^T k_\ell^{(t)} \sum_{i=1}^n y_{i,\ell+1}^{(t)} + x^4 \sum_{t=1}^T k_\ell^{(t)} \sum_{i=1}^n r_{i,\ell}^{(t)}. \end{aligned}$$

Interpolating this polynomial from the set of any 5 answers in $\{A_\ell^i\}_{i=1}^n$ reveals $\sum_{i=1}^n y_{i,\ell}^{(j)}$ and $\sum_{i=1}^n y_{i,\ell+1}^{(j)}$. Hence, the communication cost is $T \frac{s}{2} n(n-1) + \frac{5s}{2}$ symbols in \mathbb{F}_q^c .

Approach 2 does not suffer from the drawback of high communication costs in the sharing stage. However, it cannot be directly applied to the graph-based setting. A careful design of a suitable PIR scheme for arbitrary incident matrices \mathbf{I} will be needed. Requiring each client to compute the label for all samples and all objective functions exhibits a good utility of the resulting model at the federator, but incurs large computation costs. Hence, a trade-off between the computation complexity and the utility arises in this setting. To leverage this trade-off, we seek solutions that allow reduced computation and simultaneously balance the communication costs in the sharing and the query phase, requiring the design of a specifically tailored graph-based PIR scheme to generalize the scheme in Approach 2.

Remark 2. We resort to ideas from federated knowledge distillation due to the difficulty in hiding the objective in classical FL settings. Classical FL [77] is an iterative process where at each iteration, the clients compute a gradient based on their individual data and the current global model. They report the result to the federator, who aggregates the received gradients and updates the global model accordingly. After synchronizing the clients with the latest model, the process repeats. While gradient computation can be seen as an arbitrary non-linear function computation, hiding the objective of the federator in an iterative scheme is difficult. Even if the objective is completely hidden from the clients at the time of computing the gradients, the clients are potentially able to infer the federator's objective by simply observing consecutive global model updates. The idea is similar to that of model inversion attacks, which allows the federator to obtain information about the clients' private data after seeing their model updates (gradients). Further, incorporating our framework into an iterative process would incur extensive communication cost.

V. GENERAL SCHEME

We present the end-to-end one-shot FL scheme that preserves the privacy of the federator's objective of interest and the clients' data, made of three stages: the task allocation, the sharing and the query stage. We present the scheme for the case that the function results are the labels of a public dataset. Nonetheless, our solution applies for any function that is additively separable.

a) *Task Assignment Stage:* The task assignment stage requires each client i to compute a set of objectives denoted by $\mathcal{I}(i)$, i.e., the client computes a function $h_t(\mathcal{D}_i)$, or trains a model $w_{t,i}$, for each objective $t \in \mathcal{I}(i)$. For each model, the client creates (predicts) a label out of c possible classes for each sample of the public unlabeled dataset, obtaining s labels $y_{i,\ell}^{(t)}$, $\ell \in [s]$ for each objective $t \in \mathcal{I}(i)$. The labels of the public dataset can contain either hard or soft information, i.e., a one-hot encoded vector of the class labels or a vector encoding the class probabilities with finite precision. The precision will affect the finite field size q required to store all information necessary without creating finite field overflows in the aggregation process of the clients, i.e., with γ quantization steps, we require $q \geq (\gamma - 1)n$. Having quantized the labels $y_{i,\ell}^{(t)}$ into vectors from \mathbb{F}_q^c enables information-theoretic privacy guarantees through secret sharing.

b) *Sharing Stage:* We dedicate an evaluation point α_i for each client $i \in [n]$, where $\alpha_i = \alpha^i$, $i \in [n]$ for a generator element α of the field \mathbb{F}_q , $q \geq \max\{\rho + k_C - z_s, (\gamma - 1)n\}$, where $k_C = \frac{\rho - z_q + z_s + 1}{2}$ is the dimension of the storage code. The storage code is constructed by Multi-Party Computation between the clients. For each objective $t \in [T]$, each client $i \in \mathcal{I}(e_t)$ splits the set of all s labels $y_{i,\ell}^{(t)} \in \mathbb{F}_q^c$ into $P = \frac{s}{k_C - z_s}$ partitions³ $\mathcal{P}_1, \dots, \mathcal{P}_P$ each of size $k_C - z_s$, where the labels of partition p are referred to by $y_{i,p,u}^{(t)}$ for $u \in [k_C - z_s]$. For each objective t , each client $i \in \mathcal{I}(e_t)$ encodes each partition $p \in [P]$ into a secret sharing as

$$f_{i,p}^{(t)}(x) = \sum_{u=1}^{k_C - z_s} x^{u-1} y_{i,p,u}^{(t)} + \sum_{\tau=1}^{z_s} x^{k_C - z_s + \tau - 1} r_{i,p,\tau}^{(t)}, \quad (1)$$

where $\forall \tau \in [k_C - z_s]$ and $p \in [P]$ the vectors $r_{i,p,\tau}^{(t)} \in \mathbb{F}_q^c$ are uniformly chosen from \mathbb{F}_q^c . Each client i sends the evaluation $f_{i,p}^{(t)}(\alpha_{i_1})$ to client i_1 . Each client i_1 aggregates all received contributions

$$\begin{aligned} F_p^{(t)}(\alpha_{i_1}) &\triangleq \sum_{i \in \mathcal{I}(e_t)} f_{i,p}^{(t)}(\alpha_{i_1}) \\ &= \sum_{u=1}^{k_C - z_s} x^{u-1} \sum_{i \in \mathcal{I}(e_t)} y_{i,p,u}^{(t)} + \sum_{\tau=1}^{z_s} x^{k_C - z_s + \tau - 1} r_{p,\tau}^{(t)}, \end{aligned}$$

where $r_{p,\tau}^{(t)} \triangleq \sum_{i=1}^n r_{i,p,\tau}^{(t)}$, thus obtains a codeword from an (n, k_C) -GRS. This procedure incurs a total communication cost of $\frac{Tsc \cdot \rho(\rho-1)}{k_C - z_s}$ symbols in \mathbb{F}_q given that⁴ $(k_C - z_s) | sc$. The rate for each secret sharing is $\frac{k_C - z_s}{\rho(\rho-1)}$. Since the federator

³Assuming for simplicity that $(k_C - z_s) | s$.

⁴This is more general than the description above, where we would require that $(k_C - z_s) | s$ instead $(k_C - z_s) | sc$, but is achievable by splitting into fractions of the labels.

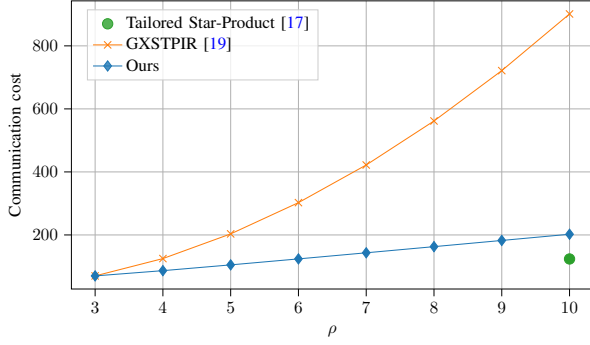


Fig. 3. Communication Cost in sc symbols in \mathbb{F}_q compared to our three-stage protocol paired with GXSTPIR [19] and the Star-Product scheme with optimized storage code dimension k_C^* according to Lemma 1. The latter is limited to $\rho = n$, i.e., to non-graph-based settings. The parameters are chosen as $n = 10$, $T = 10$, and $z_s = z_q = 1$.

is interested in only one objective $j \in [T]$, the overall sharing rate is $\frac{k_C - z_s}{T\rho(\rho-1)}$.

c) *Query Stage*: Having in place a (ρ, k_C) -GRS storage code (in fact, we have TP of such codes where each information symbol is of size $\frac{sc}{P(k_C - z_s)}$ symbols in \mathbb{F}_q , we design an (S)PIR scheme that allows the federator to retrieve the labels corresponding to the objective t of interest without revealing its identity.

Consider for all $t \in [T], p \in [P]$ the following query polynomial

$$q_p^{(t,j)}(x) = \delta_p^{(t,j)} + \sum_{\tau=1}^{z_q} x^{k_C - z_s + \tau - 1} k_{p,\tau}^{(t)}, \quad (2)$$

where $\delta_p^{(t,j)} = \begin{cases} 1 & \text{if } t = j \\ 0 & \text{otherwise} \end{cases}$, and $k_{p,\tau}^{(t)} \in \mathbb{F}_q^c$ are chosen independently and uniformly at random. For each $t \in T$, client $i \in \mathcal{I}(e_t)$ receives an evaluation $q_p^{(t,j)}(\alpha_i)$ of this polynomial, which it multiplies (element-wise) by all $F_p^{(t)}(\alpha_i)$ for $p \in [P]$, and additionally by

$$\nu_{t,i} \triangleq \left(\prod_{i_1 \in \mathcal{I}(e_t) \setminus \{i\}} (\alpha_i - \alpha_{i_1}) \right)^{-1}.$$

The choice of $\nu_{t,i}$ is justified by the duals of GRS codes and will allow for arbitrary task assignment patterns. The federator receives the answers $\mathcal{A}^i = \{A_1^i(\alpha_i), \dots, A_P^i(\alpha_i)\}$, where $A_p^i(\alpha_i)$ are evaluations of the following answer polynomial

$$\begin{aligned} A_p^i(x) &= \sum_{t \in \mathcal{I}(i)} \nu_{t,i} F_p^{(t)}(x) q_p^{(t,j)}(x) \\ &= \nu_{j,i} \sum_{u=1}^{k_C - z_s} x^{u-1} \sum_{i \in \mathcal{I}(e_j)} y_{i,p,u}^{(j)} + \sum_{t \in \mathcal{I}(i)} \nu_{t,i} \sum_{\tau=1}^{k_C + z_q - 1} x^{k_C - z_s + \tau - 1} a_{p,\tau,i}^{(t)}, \end{aligned}$$

and $\forall \tau \in [k_C + z_q - 1]$, $a_{p,\tau,i}^{(t)}$ are interference terms being potentially different for each client. The multiplication $F_p^{(t)}(x) q_p^{(t,j)}(x)$ is done element-wise.

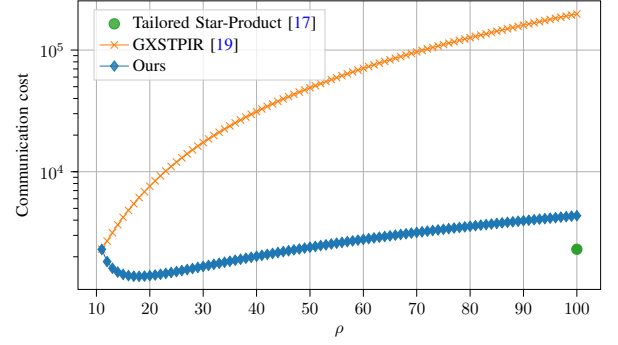


Fig. 4. Communication Cost in sc symbols in \mathbb{F}_q compared to our three-stage protocol paired with GXSTPIR [19] and the Star-Product scheme with optimized storage code dimension k_C^* according to Lemma 1. The latter is limited to $\rho = n$, i.e., to non-graph-based settings. The parameters are chosen as $n = 100$, $T = 20$, and $z_s = z_q = 5$.

d) *Reconstruction Stage*: For $\vartheta \in [k_C - z_s]$, the federator sums over all clients' answers to obtain

$$\begin{aligned} A^{(\vartheta)} &\triangleq \sum_{i=1}^n \alpha_i^{-\vartheta} A_p^i(\alpha_i) \\ &= \sum_{u=1}^{k_C - z_s} \left(\sum_{i \in \mathcal{I}(e_j)} y_{i,p,u}^{(j)} \right) \sum_{i \in \mathcal{I}(e_j)} \nu_{j,i} \alpha_i^{u - \vartheta - 1}. \end{aligned}$$

Let $\bar{y}_{p,u} \triangleq \sum_{i \in \mathcal{I}(e_j)} y_{i,p,u}^{(j)}$. By computing $A^{(\vartheta)}$ for all $\vartheta \in [k_C - z_s]$, the federator can obtain the desired information as

$$(\bar{y}_{p,1}, \bar{y}_{p,2}, \dots, \bar{y}_{p,k_C - z_s})^T = \mathbf{P}^{-1} (A^{(1)}, A^{(2)}, \dots, A^{(k_C - z_s)})^T,$$

where \mathbf{P} is the following invertible matrix

$$\mathbf{P} = \sum_{i \in \mathcal{I}(e_j)} \begin{pmatrix} \nu_{j,i} \alpha_i^{-1} & 0 & \dots & 0 & 0 \\ \nu_{j,i} \alpha_i^{-2} & \nu_{j,i} \alpha_i^{-1} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \nu_{j,i} \alpha_i^{-k_C + z_s} & \dots & \dots & \dots & \nu_{j,i} \alpha_i^{-1} \end{pmatrix}.$$

Having obtained all aggregated labels for the objective j of interest, the federator retrain a suitable model leveraging the public dataset and the heterogeneity of the clients' data.

e) *Properties of the Scheme*: We state in the following the most important properties of our proposed scheme, which is the sharing rate, the PIR rate, and the privacy guarantee.

Proposition 1 (Sharing Rate). *The rate of the proposed sharing scheme is*

$$R_{\text{share}} = \frac{\rho - z_s - z_q + 1}{2T\rho(\rho-1)}.$$

Proof. We assume in the worst case that all clients' function results are independent and uniformly distributed. By the above choice of k_C , for each objective the number of labels contained in each secret sharing according to (1) is $\frac{\rho - z_s - z_q + 1}{2}$, and the number of shares (of the same size) transmitted is given by $\rho(\rho-1)$. Since the federator is only interested in one out of all T objectives, the rate deteriorates by T . \square

Theorem 1 (PIR Rate). *The rate of the proposed PIR scheme is*

$$R_{\text{PIR}} = \frac{\rho - z_q - z_s + 1}{2n} \stackrel{(*)}{=} \frac{\rho - k_C - z_q + 1}{n}$$

TABLE I
COMPARISON OF SECRET SHARING AND PIR RATES, AND COMMUNICATION COSTS. THE PARAMETER k_C^* IS GIVEN BY LEMMA 1.

Method	Sharing Rate	PIR Rate	Total Communication Cost in \mathbb{F}_q
Ours	$\frac{\rho - z_s - z_q + 1}{2T\rho(\rho - 1)}$	$\frac{\rho - z_q - z_s + 1}{2n}$	$\frac{2Tsc \cdot \rho(\rho - 1)}{\rho - z_s - z_q + 1} + \frac{2sc \cdot n}{\rho - z_q - z_s + 1}$
GXSPIR [19]	$\frac{1}{T\rho(\rho - 1)}$	$\frac{\rho - z_q - z_s}{n}$	$Tsc \cdot \rho(\rho - 1) + \frac{sc \cdot n}{\rho - z_s - z_q}$
$\rho = n$ (modified [17])	$\frac{k_C^* - z_s}{Tn(n - 1)}$	$\frac{(k_C^* - z)(n - k_C^* - z_q + 1)}{nk_C^*}$	$\frac{Tsc \cdot n(n - 1)}{k_C^* - z_s} + \frac{sc}{k_C^* - z_s} \frac{k_C^* n}{n - k_C^* - z_q + 1}$

where (\star) holds for $\rho \leq 2k_C + z_q - 1$ and $z_s = 2k_C - \rho + z_q - 1$.

Proof. The rate of the PIR scheme for each $e_t \in \mathcal{E}$ is $\frac{k_C - z_s}{2k_C + z_q - z_s - 1}$. Setting $\rho = 2k_C + z_q - z_s - 1$, we obtain for $\rho \leq 2k_C + z_q - 1$ a per-objective rate of $\frac{\rho - k_C - z_q + 1}{\rho}$, where $z_s = 2k_C - \rho + z_q - 1$. Since we also need to query clients $i \in [n] \setminus \mathcal{I}(e_j)$ for reasons of privacy, the overall rate of the PIR scheme is $\frac{\rho - k_C - z_q + 1}{n}$. \square

The communication cost is, hence, $\frac{n}{\rho - k_C - z_q + 1}$. Let the set of all messages received by client i_1 be $\mathcal{M}_i \triangleq \{f_{i,p}^{(t)}(\alpha_{i_1})\}_{i \in \mathcal{I}(e_t) \setminus \{i_1\}, t \in \mathcal{I}(i_1)}$. Further, let the set of all messages received by all clients in $\mathcal{T} \subset [n]$ be $\mathcal{M}_{\mathcal{T}} \triangleq \{\mathcal{M}_i\}_{i \in \mathcal{T}}$.

Theorem 2 (Privacy from Clients and Objective-Hiding). *The clients' computation results are private against any set of z_s clients (cf. Definition 1). The objective j is private against any set of z_q clients (cf. Definition 2), i.e., for any two sets of clients $\mathcal{T}_s, \mathcal{T}_q \subset [n] \setminus \{i\}$, $|\mathcal{T}_s| \leq z_s$, $|\mathcal{T}_q| \leq z_q$, we have*

$$\mathbb{I}\left(\{Y_{i,\ell}^{(t)}\}_{t \in \mathcal{I}(i), \ell \in [s]}; \mathcal{M}_{\mathcal{T}_s} \mid \{Y_{i,\ell}^{(t)}\}_{t \in \mathcal{I}(i), i \in \mathcal{T}_s, \ell \in [s]}\right) = 0, \forall i \in [n],$$

$$\mathbb{I}\left(\{Q_{p,\mathcal{T}_q}^{(t)}\}, \mathcal{S}_{\mathcal{T}_q}; j\right) = 0.$$

Theorem 3 (Correctness). *The sum of labels of interest $\sum_{i \in \mathcal{I}(e_j)} y_{i,\ell}^{(j)}$, $\ell \in [s]$, is decodable from the answers, i.e.,*

$$\mathbb{H}\left(\left\{\sum_{i \in \mathcal{I}(e_j)} Y_{i,p,u}^{(j)}\right\}_{p \in [P], u \in [k_C - z_s]} \mid \{A_p^i(\alpha_i)\}_{p \in [P], i \in \mathcal{I}(e_j)}\right) = 0.$$

We compare the communication cost of our scheme in Figs. 3 and 4 to the application of two PIR schemes from the literature for $n = 10$ and $n = 100$, respectively, and provide the corresponding rates in Table I. The details of the comparison are deferred to Section VII.

VI. EXTENSION TO PRIVACY FROM THE FEDERATOR

While the above scheme complies with the privacy notion according to Definition 1, privacy from the federator as in Definition 3 is not ensured. Therefore, recall the answer polynomial $A_p^i(x)$ above. The crucial aspect is that the interference terms $a_{p,\tau,i}^{(t)}$ contain sensitive information about $y_{i,p,u}^{(j)}$ for $u \in [k_C - z_s]$ and $t \neq j$, i.e., they depend on the function results of the clients, thereby leaking potential information about clients' results beyond the objective j of interest. To ensure user-side privacy, where the federator does not learn anything about the clients models beyond the objective of interest, we assume the existence of shared randomness among

all clients unknown to the federator. Leveraging this shared randomness, the clients construct a randomized polynomial

$$R_p(x) = \sum_{\tau=1}^{k_C + z_q - 1} x^{k_C - z_s + \tau - 1} s_{p,\tau},$$

where $\{s_{p,\tau}\}_{\tau \in [k_C + z_q - 1]}$ is known to all clients, but unknown to the federator. For each partition $p \in [P]$, each client i replies to the queries $q_p^{(t,j)}(x)$, $t \in \mathcal{I}(i)$ with the answer

$$A'_p(\alpha_i) = \sum_{t=1}^T \nu_{t,i} F_p^{(t)}(\alpha_i) q_p^{(t,j)}(\alpha_i) + R_p(\alpha_i),$$

which is an evaluation of the *re-randomized* polynomial

$$A'_p(x) = \nu_{j,i} \sum_{u=1}^{k_C - z_s} x^{u-1} \sum_{i \in \mathcal{I}(e_j)} y_{i,p,u}^{(j)} + \sum_{\tau=1}^{k_C + z_q - 1} x^{k_C - z_s + \tau - 1} \left(s_{p,\tau} + \sum_{t \in \mathcal{I}(i)} \nu_{t,i} a_{p,\tau,i}^{(t)} \right),$$

which is, by the one-time pad, guarantees the privacy of the $a_{p,\tau,i}^{(t)}$ that contain potentially sensitive information about the clients' computation beyond the objective of interest. The recovery process as in Section V remains unchanged. Let the answer \mathcal{A}^i received from client i be $\mathcal{A}^i \triangleq \{A'_p(\alpha_i)\}_{p \in [P]}$, then we have the following privacy statement.

Theorem 4 (Symmetric Privacy). *In addition to the privacy satisfied according to Theorem 2, the federator learns nothing beyond the aggregation of ρ clients' predictions for the objective j of interest. Formally,*

$$\mathbb{I}\left(\mathcal{A}^{[n]}, \{Q_{p,[n]}^{(t)}\}; \{Y_{i,\ell}^{(t)}\} \mid \left\{\sum_{i \in \mathcal{I}(e_j)} Y_{i,\ell}^{(j)}\right\}_{\ell \in [s]}\right) = 0.$$

where for clarity we define $\{Q_{p,[n]}^{(t)}\} \triangleq \{Q_{p,[n]}^{(t)}\}_{t \in [T], p \in [P]}$ and $\{Y_{i,\ell}^{(t)}\} \triangleq \{Y_{i,\ell}^{(t)}\}_{t \in \mathcal{I}(i), i \in [n], \ell \in [s]}$.

Proof. By the one-time pad, all interference terms $\sum_{t \in \mathcal{I}(i)} \nu_{t,i} a_{p,\tau,i}^{(t)}$ containing sensitive information are perfectly hidden from the federator, and hence, all labels $\{Y_{i,\ell}^{(t)}\}_{i \in [n], t \in \mathcal{I}(i), \ell \in [s]}$ with the exception of $\{\sum_{i \in \mathcal{I}(e_j)} Y_{i,\ell}^{(j)}\}_{\ell \in [s]}$ are statistically independent from the answers $\mathcal{A}^{[n]}$. Further, the queries $\{Q_{p,[n]}^{(t)}\}_{t \in [T], p \in [P]}$ are independent of the labels. This holds even if all clients' polynomials could be exactly reconstructed by the federator. \square

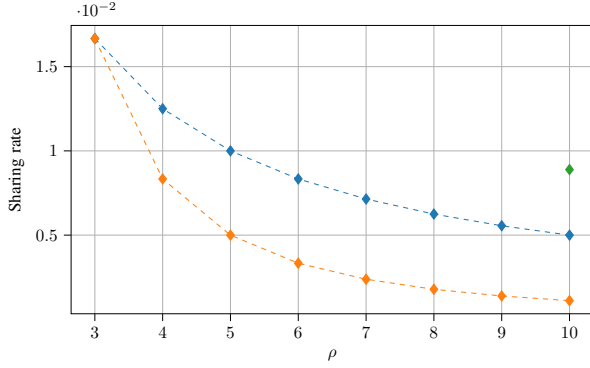


Fig. 5. Secret Sharing rates compared to GXSTPIR [19] and the Star-Product scheme with optimized storage code dimension k_C^* according to Lemma 1. The latter is limited to $\rho = n$, i.e., to non-graph-based settings. The parameters are chosen as $n = 10$, $T = 10$, and $z_s = z_q = 1$.

VII. COMPARISON TO CROSS-SUBSPACE-ALIGNMENT AND STAR-PRODUCT CODES

Our scheme encapsulates a secret sharing stage and a new PIR scheme for graph-based MDS coded storage patterns specifically tailored to generalized secret sharing schemes. When restricting to the suboptimal Shamir Secret Sharing in the sharing stage, the method of [19] can be used instead of our PIR scheme. In a non-graph-based setting (when $\rho = n$), known methods from PIR over MDS coded data apply for generalized secret sharing schemes due to their MDS structure. However, for optimal overall communication costs, we formulate and solve an optimization problem that trades the sharing rate against the PIR rate and finds the optimal operating point. We elaborate on the two extreme cases in the following, and provide a comparison to our scheme.

a) *Star-Product PIR [17]*: Considering non-graph-based settings (i.e., $\rho = n$), since secret sharing is a Reed-Solomon code, known results from private information retrieval over MDS-coded data such as those in [17] can be applied in our framework. Applying such results yields an interesting trade-off between the design of the storage code and the query code. This trade-off is not present in our scheme due to our construction being tailored to this setting. Star-product-based PIR schemes consist of a storage code \mathcal{C} and a query code \mathcal{D} . On a high level, each client (or server) returns codewords from the star-product code $\mathcal{C} \star \mathcal{D}$, and the messages of interest are encoded as erasures in the code. The rate of this PIR scheme is $(d_{\mathcal{C} \star \mathcal{D}} - 1)/n$, while being private against $z_q = d_{\mathcal{D}^\perp} - 1$ colluders, where $d_{\mathcal{D}^\perp}$ is the minimum distance of the dual code of \mathcal{D} . Given a desired z_q , the dual of \mathcal{D} must satisfy $d_{\mathcal{D}^\perp} = z_q + 1$ and have dimension $k_{\mathcal{D}^\perp} = n - z_q$. Hence, \mathcal{D} is an (n, z_q) -GRS code. If we choose the same code locators for both codes, then the star product $\mathcal{C} \star \mathcal{D}$ is an $(n, \min\{k_C + z_q - 1, n\})$ -GRS code with minimum distance $d_{\mathcal{C} \star \mathcal{D}} = n - k_C - z_q + 2$ given that $k_C + z_q - 1 \leq n$. The PIR scheme achieves a rate of $\frac{d_{\mathcal{C} \star \mathcal{D}} - 1}{n} = \frac{n - k_C - z_q + 1}{n}$ [17]. Fixing z_q , it is desirable to choose a storage code with a small parameter k_C . However, in this case the randomness of the storage code is recovered as a by-product, and hence, the rate comes with an additional factor of $\frac{k_C - z_s}{k_C}$. The rate results as $\frac{n - k_C - z_q + 1}{n} \frac{k_C - z_s}{k_C}$. The download cost of the PIR scheme is given by $\frac{sc}{k_C - z_s} \frac{k_C n}{n - k_C - z_q + 1}$ symbols in \mathbb{F}_q . However, each of

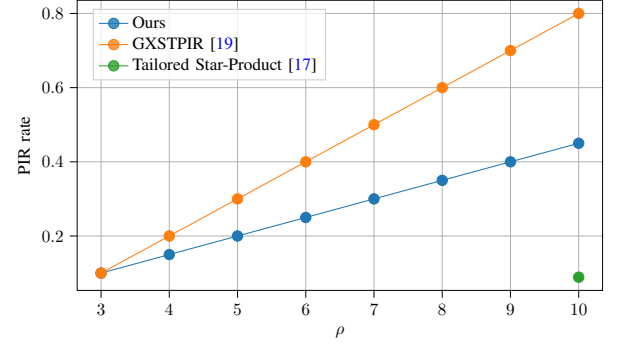


Fig. 6. Private information retrieval rates compared to GXSTPIR [19] and the Star-Product scheme with optimized storage code dimension k_C^* according to Lemma 1. The latter is limited to $\rho = n$, i.e., to non-graph-based settings. The parameters are chosen as $n = 10$, $T = 10$, and $z_s = z_q = 1$.

the clients is required to train T models for each potential function.

Lemma 1. *The lowest communication cost is given by*

$$\frac{Tsc \cdot n(n-1)}{k_C^* - z_s} + \frac{sc}{k_C^* - z_s} \frac{k_C^* n}{n - k_C^* - z_q + 1}$$

symbols in \mathbb{F}_q , where k_C^* is chosen from $\{\lfloor k'_C \rfloor, \lceil k'_C \rceil\}$ as given below to minimize the above cost. Defining $c_1 \triangleq (n - z_q + 1)$ and $c_2 \triangleq Tn(n-1)$, we have

$$k'_C = \frac{1}{c_2 - n} \left(c_1 c_2 - \sqrt{c_1^2 c_2^2 - (c_2 - n)(c_1^2 c_2 + n c_1 z_s)} \right).$$

Approach 3. *Consider the setting of Example 1. We have $n = 5$ clients $i \in [5]$ and s labels, split into $P = s/2$ partitions of size two. We assume no collusions between clients, i.e., $z_s = z_q = z = 1$. Let for some choice of T the optimal code dimension be $k_C^* = 3$. For each $p \in [P]$, each client i creates shares of the form*

$$f_{i,p}^{(t)}(x) = y_{i,p,1}^{(t)} + x \cdot y_{i,p,2}^{(t)} + x^2 \cdot r_{i,p,1}^{(t)},$$

and sends to each client $i_1 \neq i$ a share $f_{i,p}^{(t)}(\alpha_{i_1})$, keeping the share $f_{i,p}^{(t)}(\alpha_i)$ to itself. Summing the shares of all incoming clients, each client i obtains $\forall p \in [P]$ a share $F_p^{(t)}(\alpha_i)$ of the polynomial $F_p^{(t)}(x)$, corresponding to a codeword of an $(n, 3)$ -GRS $(\alpha, \mathbf{1}_n)$ code \mathcal{C} , where $\alpha = (\alpha_1, \dots, \alpha_n)$ are the evaluators and $\mathbf{1}_n$ the column multipliers. The communication cost in the sharing stage is $T \frac{s}{2} n(n-1)$ symbols in \mathbb{F}_q^c .

For privacy against $z_q = 1$, we choose a query code \mathcal{D} whose dual has dimension $k_{\mathcal{D}^\perp} = n - z = 4$, hence an $(n, n - k_{\mathcal{D}^\perp})$ -GRS $(\alpha, \mathbf{1}_n)$. The star product code $\mathcal{C} \star \mathcal{D}$ then is an $(n, 3)$ -GRS $(\alpha, \mathbf{1}_n)$ code, with minimum distance $d_{\mathcal{C} \star \mathcal{D}} = 3$. The overall rate of the PIR scheme is $\frac{d_{\mathcal{C} \star \mathcal{D}} - 1}{n} = \frac{2}{5}$. The overall communication cost of the scheme is $T \frac{s}{2} n(n-1) + \frac{5}{2s}$ symbols in \mathbb{F}_q^c , which is the same as for Approach 2. However, this method does not apply to graph-based settings with arbitrary task assignments due to the interplay of storage and query code, but can yield better results for $\rho = n$ as shown in Fig. 3.

b) *Graph-based PIR with Cross-Subspace Alignment:*

When restricting to Shamir Secret Sharing schemes, the clients would construct a secret sharing for each prediction

individually. If all secret sharing instances follow the same construction, cross-subspace alignment can improve the rate of the PIR scheme. In fact, the rate of GXSTPIR was shown to be $\frac{\rho - z_s - z_q}{n}$ [19]. For XSTPIR (non-graph based), let $\rho = n$. Then we have a rate of $\frac{\rho - z_s - z_q}{\rho}$ [44], where Shamir Secret Sharing schemes are shown to be capacity-achieving through cross-subspace alignment. However, being restricted to Shamir Secret Sharing is undesirable in our case since the rate of the sharing stage is $\frac{k_C - z_s}{\rho(\rho - 1)}$ for $\rho \leq k_C$, which is the worst for $k_C - z_s = 1$, where the rate is $\frac{1}{\rho(\rho - 1)}$. The optimal rate results when ρ is maximal, which yields $\frac{\rho - z_s}{\rho(\rho - 1)}$. In comparison, with our proposed PIR scheme for arbitrary $k_C - z_s \leq \rho - z_s$, we obtain a PIR rate of $\frac{k_C - z_s}{2k_C + z_q - z_s - 1}$ for $\rho \geq 2k_C + z_q - z_s - 1$, which shows that for non-fixed ρ , large $k_C - z_s$ are desirable. Equivalently, we can write $\frac{\rho - k_C - z_q + 1}{\rho} = \frac{\rho - z_q - z_s + 1}{2\rho}$, hence, asymptotically in ρ , the rate goes to $\frac{1}{2}$.

Remark 3 (Cross-Subspace Alignment in our Scheme). *The authors of [19] use cross-subspace alignment by a careful choice of evaluation points for different Shamir Secret Sharing instances to construct capacity-achieving PIR schemes. On the contrary, we jointly design the storage and query code to reduce the overall communication cost dominated by the sharing stage. This leads to occupying all dimensions of the answers with only one secret sharing. If the communication cost in the sharing stage was of lower importance, ideas from cross-subspace alignment could be incorporated into our framework by a deliberate choice of the evaluation points for different McEliece-Sarwate secret sharing instances.*

We compare in Fig. 3 the overall communication cost of our proposed three-stage protocol for the proposed tailored PIR scheme to the usage of prior work from the PIR literature as a substitute for our PIR scheme, as a function of $\rho \leq n$ for the case when $n = 10$, $T = 10$, and $z_s = z_q = 1$. In Table I, we provide a summary of the sharing and PIR rates and the communication costs for the three schemes as a function of the system parameters. We plot the separate sharing and PIR rates for the same parameters above in Figs. 5 and 6, which exhibit contrasting behavior as a function of ρ . Hence, the computation cost and, consequently, the utility determined by ρ incur a trade-off between the sharing and the PIR rates. One can further find that our proposed solution in combination with the newly designed PIR scheme outperforms the usage of existing graph-based PIR schemes in the query stage of our method. When $\rho = n$, we show that the usage of a star-product-based PIR scheme in the query stage with parameters specifically optimized for MDS coded data in form of secret sharing gives the smallest communication cost. We depict the communication cost for $n = 100$ clients, $T = 20$, and $z_s = z_q = 5$ in Fig. 4, and find that our scheme uniformly outperforms the application of existing graph-based PIR schemes. For $\rho = n = 100$, using the tailored star-product-based PIR scheme is beneficial. According to Definitions 1 to 3, no privacy leakage is incurred by our scheme.

VIII. CONCLUSION

In this work, we introduced a new notion of objective-hiding for federated one-shot learning complemented by data

privacy for the clients' data. We use tools from multi-party computation, knowledge distillation and (S)PIR, and propose a new three-stage protocol that achieves information-theoretic privacy of the federator's objective and the clients' data under a limited collusion assumption. To minimize the joint communication cost in the sharing and query stages of our framework, we proposed a novel graph-based PIR scheme for flexible task assignments specifically tailored to the setting at hand and leveraging the properties of dual GRS codes.

Going further, the problems of mitigating the effect of stragglers and dropouts among clients, considering the presence of malicious clients deliberately trying to corrupt the process, and considering different privacy notions such as differential privacy [78] and subset privacy [79] remain open in this setting.

APPENDIX

A. Proof of Theorem 2

We first prove the clients' data privacy from any other set \mathcal{T}_s of at most z_s colluding clients. Note that each client $i_1 \in \mathcal{T}_s$ receives the following set of messages from all other clients: $\mathcal{M}_{i_1} \triangleq \{f_{i,p}^{(t)}(\alpha_{i_1})\}_{i \in \mathcal{I}(e_t), t \in \mathcal{I}(i_1) \setminus \{i_1\}, p \in [P]}$, i.e., for each objective $t \in \mathcal{I}(i_1)$ one share from each client that was assigned the same objective t , per partition $p \in [P]$. Hence, for each objective $t \in [T]$ and each partition $p \in [P]$ and set of clients in \mathcal{T}_s receive at most z_s shares. Since, by design, at most z_s secret shares of any two objectives $t, t' \in [T]$ and two partitions p, p' are pair-wise independent of each other, it suffices to prove that for all clients $i \in [n]$ any for any pair of t, p , we have $\mathbb{I}\left(\{Y_{i,p,u}^{(t)}\}_{u \in [k_C - z_s]}; \mathcal{M}_{\mathcal{T}_s} \mid \{Y_{i,\ell}^{(t)}\}_{t \in \mathcal{I}(i), i \in \mathcal{T}_s, \ell \in [s]}\right) = 0$, where $Y_{i,p,u}^{(t)}$ is the random variable corresponding to $y_{i,p,u}^{(t)}$ as defined in Section V. The set of critical messages is then given by $\{f_{i,p}^{(t)}(\alpha_{i_1})\}_{i_1 \in \mathcal{T}_s}$. Hence, we need to prove that

$$\begin{aligned} & \mathbb{I}\left(\{Y_{i,p,u}^{(t)}\}_{u \in [k_C - z_s]}; \{f_{i,p}^{(t)}(\alpha_{i_1})\}_{i_1 \in \mathcal{T}_s} \mid \{Y_{i,\ell}^{(t)}\}_{t \in \mathcal{I}(i), i \in \mathcal{T}_s, \ell \in [s]}\right) \\ & \mathbb{H}\left(\{Y_{i,p,u}^{(t)}\}_{u \in [k_C - z_s]} \mid \{Y_{i,\ell}^{(t)}\}_{t \in \mathcal{I}(i), i \in \mathcal{T}_s, \ell \in [s]}\right) \\ & - \mathbb{H}\left(\{Y_{i,p,u}^{(t)}\}_{u \in [k_C - z_s]} \mid \{Y_{i,\ell}^{(t)}\}_{t \in \mathcal{I}(i), i \in \mathcal{T}_s, \ell \in [s]}, \{f_{i,p}^{(t)}(\alpha_{i_1})\}_{i_1 \in \mathcal{T}_s}\right) \\ & \mathbb{H}\left(\{Y_{i,p,u}^{(t)}\}_{u \in [k_C - z_s]} \mid \{Y_{i,\ell}^{(t)}\}_{t \in \mathcal{I}(i), i \in \mathcal{T}_s, \ell \in [s]}\right) \\ & - \mathbb{H}\left(\{Y_{i,p,u}^{(t)}\}_{u \in [k_C - z_s]} \mid \{Y_{i,\ell}^{(t)}\}_{t \in \mathcal{I}(i), i \in \mathcal{T}_s, \ell \in [s]}\right) = 0, \end{aligned}$$

which holds since any z_s shares do not reveal anything about the privacy labels $\{Y_{i,p,u}^{(t)}\}_{u \in [k_C - z_s]}$ beyond the correlation with the colluders' information.

We now prove the privacy of the objective j of interest in the following, in particular, the queries or shares observed and held by any set of z_q clients does not leak any information about j . When clear from the context, we omit the subscripts

from the set notation for readability.

$$\begin{aligned}
& \mathcal{I}\left(\{Q_{p,\mathcal{T}_q}^{(t)}\}, \mathcal{M}_{\mathcal{T}_q}, \mathcal{S}_{\mathcal{T}_q}; j\right) \\
& \stackrel{(a)}{=} \mathcal{I}\left(\{Q_{p,\mathcal{T}_q}^{(t)}\}, \mathcal{M}_{\mathcal{T}_q}, \{Y_{i,\ell}^{(t)}\}_{i \in \mathcal{T}_q}; j\right) \\
& \stackrel{(b)}{=} \mathcal{I}\left(\{Q_{p,\mathcal{T}_q}^{(t)}\}; j\right) + \mathcal{I}\left(\mathcal{M}_{\mathcal{T}_q}, \{Y_{i,\ell}^{(t)}\}_{i \in \mathcal{T}_q}; j \mid \{Q_{p,\mathcal{T}_q}^{(t)}\}\right) \\
& = \mathcal{I}\left(\{Q_{p,\mathcal{T}_q}^{(t)}\}; j\right) + \mathcal{H}\left(\mathcal{M}_{\mathcal{T}_q}, \{Y_{i,\ell}^{(t)}\}_{i \in \mathcal{T}_q} \mid \{Q_{p,\mathcal{T}_q}^{(t)}\}\right) \\
& \quad - \mathcal{H}\left(\mathcal{M}_{\mathcal{T}_q}, \{Y_{i,\ell}^{(t)}\}_{i \in \mathcal{T}_q} \mid \{Q_{p,\mathcal{T}_q}^{(t)}\}, j\right) \\
& \stackrel{(c)}{=} \mathcal{I}\left(\{Q_{p,\mathcal{T}_q}^{(t)}\}; j\right) + \mathcal{H}\left(\mathcal{M}_{\mathcal{T}_q}, \{Y_{i,\ell}^{(t)}\}_{i \in \mathcal{T}_q}\right) - \mathcal{H}\left(\mathcal{M}_{\mathcal{T}_q}, \{Y_{i,\ell}^{(t)}\}_{i \in \mathcal{T}_q}\right)
\end{aligned}$$

where (a) is by the definition of $\mathcal{S}_{\mathcal{T}_q}$, (b) follows from the chain rule of mutual information, (c) from the definition of conditional mutual information and by independence of $\{Q_{p,\mathcal{T}_q}^{(t)}\}, j$ of $\mathcal{M}_{\mathcal{T}_q}, \{Y_{i,\ell}^{(t)}\}_{i \in \mathcal{T}_q}$.

Assuming w.l.o.g. that $j = 1$, then

$$\begin{aligned}
\mathcal{I}\left(\{Q_{p,\mathcal{T}_q}^{(t)}\}, \mathcal{S}_{\mathcal{T}_q}; j\right) &= \sum_{t=1}^T \mathcal{I}\left(\{Q_{p,\mathcal{T}_q}^{(t)}\}; j \mid \{Q_{p,\mathcal{T}_q}^{(t')}\}_{t' < t}\right) \\
&\stackrel{(d)}{=} \sum_{t=1}^T \mathcal{I}\left(\{Q_{p,\mathcal{T}_q}^{(t)}\}; j\right) \stackrel{(e)}{\leq} \sum_{t=1}^T \mathcal{I}\left(\{Q_{p,\mathcal{T}_q}^{(t)}\}; \delta_p^{(t,j)}\right) = 0,
\end{aligned}$$

where (d) is because any set of at most z_q shares $Q_{p,\mathcal{T}_q}^{(t)}$ encoding the query for objective t is independent of another set of secret shares $Q_{p,\mathcal{T}_q}^{(t')}$ encoding the query for objective t' by the properties of secret sharing. (e) holds by the data processing inequality, and the last equality holds since (2) is a secret sharing according to McEliece-Sarwate [38] where any set of at most z_q shares are statistically independent of the private message $\delta_p^{(t,j)}$. This concludes the proof.

B. Proof of Theorem 3

Proof. The federator receives the answers $\mathcal{A}^i = \{A_1^i(\alpha_i), \dots, A_p^i(\alpha_i)\}$, where $A_p^i(\alpha_i)$ are evaluations of the following answer polynomial

$$\begin{aligned}
A_p^i(x) &= \sum_{t \in \mathcal{I}(i)} \nu_{t,i} F_p^{(t)}(x) q_p^{(t,j)}(x) \\
&= \sum_{t \in \mathcal{I}(i)} \nu_{t,i} \left(\sum_{u=1}^{k_C - z_s} x^{u-1} \sum_{i \in \mathcal{I}(e_t)} y_{i,p,u}^{(t)} + \sum_{\tau=1}^{z_s} x^{k_C - z_s + \tau - 1} r_{p,\tau}'^{(t)} \right) \\
&\quad \left(\delta_p^{(t,j)} + \sum_{\tau=1}^{z_q} x^{k_C - z_s + \tau - 1} k_{p,\tau}^{(t)} \right) \\
&= \sum_{t \in \mathcal{I}(i)} \nu_{t,i} \left(\sum_{u=1}^{k_C - z_s} x^{u-1} \sum_{i \in \mathcal{I}(e_t)} y_{i,p,u}^{(t)} \right) \delta_p^{(t,j)} \\
&+ \sum_{t \in \mathcal{I}(i)} \nu_{t,i} \left(\sum_{u=1}^{k_C - z_s} x^{u-1} \sum_{i \in \mathcal{I}(e_t)} y_{i,p,u}^{(t)} \right) \left(\sum_{\tau=1}^{z_q} x^{k_C - z_s + \tau - 1} k_{p,\tau}^{(t)} \right) \\
&+ \sum_{t \in \mathcal{I}(i)} \nu_{t,i} \left(\sum_{\tau=1}^{z_s} x^{k_C - z_s + \tau - 1} r_{p,\tau}'^{(t)} \right) \delta_p^{(t,j)} \\
&+ \sum_{t \in \mathcal{I}(i)} \nu_{t,i} \left(\sum_{\tau'=1}^{z_s} x^{k_C - z_s + \tau' - 1} r_{p,\tau'}'^{(t)} \right) \left(\sum_{\tau=1}^{z_q} x^{k_C - z_s + \tau - 1} k_{p,\tau}^{(t)} \right)
\end{aligned}$$

$$\begin{aligned}
&= \nu_{j,i} \sum_{u=1}^{k_C - z_s} x^{u-1} \sum_{i \in \mathcal{I}(e_j)} y_{i,p,u}^{(j)} + \nu_{j,i} \sum_{\tau=1}^{z_s} x^{k_C - z_s + \tau - 1} r_{p,\tau}'^{(j)} \\
&+ \sum_{u=1}^{k_C - z_s} \sum_{\tau=1}^{z_q} x^{k_C - z_s + u + \tau - 2} \sum_{t \in \mathcal{I}(i)} \nu_{t,i} \left(k_{p,\tau}^{(t)} \sum_{i \in \mathcal{I}(e_t)} y_{i,p,u}^{(t)} \right) \\
&+ \sum_{\tau'=1}^{z_s} \sum_{\tau=1}^{z_q} x^{2k_C - 2z_s + \tau' + \tau - 2} \sum_{t \in \mathcal{I}(i)} \nu_{t,i} r_{p,\tau'}'^{(t)} k_{p,\tau}^{(t)},
\end{aligned}$$

For $\vartheta \in [k_C - z_s]$, summing over all clients, we have

$$\begin{aligned}
\sum_{i=1}^n \alpha_i^{-\vartheta} A_p^i(\alpha_i) &= \sum_{i=1}^n \sum_{t \in \mathcal{I}(i)} \alpha_i^{-\vartheta} \nu_{t,i} F_p^{(t)}(\alpha_i) q_p^{(t,j)}(\alpha_i) \\
&= \sum_{t=1}^T \sum_{i \in \mathcal{I}(e_t)} \alpha_i^{-\vartheta} \nu_{t,i} F_p^{(t)}(\alpha_i) q_p^{(t,j)}(\alpha_i) \\
&= \sum_{u=1}^{k_C - z_s} \sum_{i \in \mathcal{I}(e_j)} \left(\nu_{j,i} \alpha_i^{u - \vartheta - 1} \sum_{i \in \mathcal{I}(e_j)} y_{i,p,u}^{(j)} \right) \\
&+ \sum_{\tau=1}^{z_s} r_{p,\tau}'^{(j)} \sum_{i \in \mathcal{I}(e_j)} \nu_{j,i} \alpha_i^{k_C - z_s + \tau - \vartheta - 1} \\
&+ \sum_{t=1}^T \sum_{u=1}^{k_C - z_s} \sum_{\tau=1}^{z_q} \left(k_{p,\tau}^{(t)} \sum_{i \in \mathcal{I}(e_t)} y_{i,p,u}^{(t)} \right) \sum_{i \in \mathcal{I}(e_t)} \nu_{t,i} \alpha_i^{k_C - z_s + u + \tau - \vartheta - 2} \\
&+ \sum_{t=1}^T \sum_{\tau'=1}^{z_s} \sum_{\tau=1}^{z_q} r_{p,\tau'}'^{(t)} k_{p,\tau}^{(t)} \sum_{i \in \mathcal{I}(e_t)} \nu_{t,i} \alpha_i^{2k_C - 2z_s + \tau' + \tau - \vartheta - 2} \\
&= \sum_{u=1}^{k_C - z_s} \left(\sum_{i \in \mathcal{I}(e_j)} y_{i,p,u}^{(j)} \right) \sum_{i \in \mathcal{I}(e_j)} \nu_{j,i} \alpha_i^{u - \vartheta - 1}
\end{aligned}$$

where the latter step holds because $\sum_{i \in \mathcal{I}(e_t)} \nu_{t,i} \alpha_i^\zeta = 0$ for all $0 \leq \zeta \leq \rho - 2$. It is ensured that $k_C - z_s \leq \frac{\rho - z_s - z_q + 1}{2}$. Consequently, we have 1) that $0 \leq \tau - 1 \leq k_C - z_s + \tau - \vartheta - 1 \leq k_C - z_s + \tau - 2 \leq k_C - z_s - 3$, 2) that $2 \leq u + \tau \leq k_C - z_s + u + \tau - \vartheta - 2 \leq k_C - z_s + u + \tau - 3 \leq 2(k_C - z_s) + z_q - 3 \leq \rho - z_s - 2 \leq \rho - 2$, and 3) that $k_C - z_s \leq k_C - z_s + \tau' + \tau - 2 \leq 2k_C - 2z_s + \tau' + \tau - \vartheta - 2 \leq 2k_C - 2z_s + \tau' + \tau - 3 \leq 2k_C - 2z_s + z_s + z_q - 3 \leq \rho - 2$. Hence, in all cases, ζ is between 0 and $\rho - 2$, which is why all terms except the first vanish.

Similarly, all terms corresponding to $\alpha_i^{u - \vartheta - 1}$ vanish when $\vartheta < u$, and hence we obtain from $\alpha_i \in [k_C - z_s]$ the following linear system of equations. For notational convenience, we let $\bar{y}_{p,u} \triangleq \sum_{i \in \mathcal{I}(e_j)} y_{i,p,u}^{(j)}$. Further, let $A^{(\vartheta)} \triangleq \sum_{i=1}^n \alpha_i^{-\vartheta} A_p^i(\alpha_i)$. Without loss of generality, let clients $i \in [\rho]$ store the logits for objective j . We have for

$$\mathbf{P} = \sum_{i \in \mathcal{I}(e_j)} \begin{pmatrix} \nu_{j,i} \alpha_i^{-1} & 0 & \cdots & 0 & 0 \\ \nu_{j,i} \alpha_i^{-2} & \nu_{j,i} \alpha_i^{-1} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \nu_{j,i} \alpha_i^{-k_C + z_s} & \cdots & \cdots & \cdots & \nu_{j,i} \alpha_i^{-1} \end{pmatrix}$$

that

$$\begin{pmatrix} A^{(1)} \\ A^{(2)} \\ \dots \\ A^{(k_C - z_s)} \end{pmatrix} = \mathbf{P} \begin{pmatrix} \bar{y}_{p,1} \\ \bar{y}_{p,2} \\ \dots \\ \bar{y}_{p,k_C - z_s} \end{pmatrix}$$

from which the desired values $\bar{y}_{p,1}, \bar{y}_{p,2}, \dots, \bar{y}_{p,k_C - z_s}$ can be obtained. Using the triangular shape of the matrix and applying Lemma 2 proves that \mathbf{P} is invertible when $q > \rho + k_C - z_s - 1$, and $\alpha_i, i \in [n]$, are chosen as $\alpha_i = \alpha^i, i \in [n]$ for a generator α of the field \mathbb{F}_q . We state and prove the lemma in the following. This also concludes the proof of the theorem.

Lemma 2. For $q > \rho + k_C - z_s - 1$, and $\alpha_i = \alpha^i, i \in [n]$, for any generator element α of the field \mathbb{F}_q , we have $\sum_{i \in \mathcal{I}(e_j)} \nu_{j,i} \alpha_i^{-\vartheta} \neq 0$ for all $\vartheta \in [k_C - z_s]$.

Proof. Assuming without loss of generality that $\mathcal{I}(e_j) = [\rho]$, we write $\sum_{i \in \mathcal{I}(e_j)} \nu_{j,i} \alpha_i^{-\vartheta}$ as the inner product of two code-words:

$$\sum_{i \in \mathcal{I}(e_j)} \nu_{j,i} \alpha_i^{-\vartheta} = (\nu_{j,1}, \dots, \nu_{j,\rho}) (\alpha_1^{-\vartheta}, \dots, \alpha_\rho^{-\vartheta})^T$$

Let $(\nu_{j,1}, \dots, \nu_{j,\rho}) \in \mathcal{C}$, where \mathcal{C} is a generalized Reed-Solomon code with dimension 1 and length ρ . If and only if $(\alpha_1^{-\vartheta}, \dots, \alpha_\rho^{-\vartheta}) \in \mathcal{C}^\perp$, it holds that $(\nu_{j,1}, \dots, \nu_{j,\rho}) (\alpha_1^{-\vartheta}, \dots, \alpha_\rho^{-\vartheta})^T = 0$. We need to show that $(\alpha_1^{-\vartheta}, \dots, \alpha_\rho^{-\vartheta}) \notin \mathcal{C}^\perp$. By the definition of generalized Reed-Solomon codes and their dual, the generator matrix $\mathbf{G}_{\mathcal{C}^\perp}$ of \mathcal{C}^\perp is given by

$$\mathbf{G}_{\mathcal{C}^\perp} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ \alpha_1 & \alpha_2 & \dots & \alpha_\rho \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{\rho-2} & \alpha_2^{\rho-2} & \dots & \alpha_\rho^{\rho-2} \end{pmatrix} \quad (3)$$

Hence, $(\nu_{j,1}, \dots, \nu_{j,\rho}) (\alpha_1^{-\vartheta}, \dots, \alpha_\rho^{-\vartheta})^T = 0$ for all $2 - \rho \leq \vartheta \leq 0$ (which we used above). Since we have $\alpha_i^{q-1} = 1$, we can write

$$\begin{aligned} \sum_{i \in \mathcal{I}(e_j)} \nu_{j,i} \alpha_i^{-\vartheta} &= (\nu_{j,1}, \dots, \nu_{j,\rho}) (\alpha_1^{-\vartheta}, \dots, \alpha_\rho^{-\vartheta})^T \\ &= (\nu_{j,1}, \dots, \nu_{j,\rho}) (\alpha_1^{q-\vartheta-1}, \dots, \alpha_\rho^{q-\vartheta-1})^T \end{aligned}$$

If $q - \vartheta - 1 \leq \rho - 2$, then from (3), it can be seen that the above inner product is 0. If $q - \vartheta - 1 > \rho - 2$, we must show that the matrix

$$\begin{pmatrix} 1 & 1 & \dots & 1 \\ \alpha_1 & \alpha_2 & \dots & \alpha_\rho \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{\rho-2} & \alpha_2^{\rho-2} & \dots & \alpha_\rho^{\rho-2} \\ \alpha_1^{q-\vartheta-1} & \alpha_2^{q-\vartheta-1} & \dots & \alpha_\rho^{q-\vartheta-1} \end{pmatrix}$$

is full rank for every $\vartheta \in [1, k_C - z_s]$. Note that only the first $\rho - 1$ rows correspond to the structure of a transposed Vandermonde matrix, hence the latter row is not necessarily linearly independent from the first $\rho - 1$ rows. By choosing

each evaluation point α_i to be the i -th power of a primitive element α , we can rewrite the matrix as

$$\begin{pmatrix} 1 & 1 & \dots & 1 \\ \alpha & \alpha^2 & \dots & \alpha^\rho \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{\rho-2} & \alpha^{2(\rho-2)} & \dots & \alpha^{\rho(\rho-2)} \\ \alpha^{q-\vartheta-1} & \alpha^{2(q-\vartheta-1)} & \dots & \alpha^{\rho(q-\vartheta-1)} \end{pmatrix},$$

which exhibits the structure of a Vandermonde matrix. To satisfy that all powers $0, \dots, q - \vartheta - 1$ of α are distinct, we require that $\text{ord}(\alpha) > q - 2 \geq q - \vartheta - 1$. Hence, the order of α must be $\text{ord}(\alpha) \geq q - 1$, which is satisfied if α is a generator of the field \mathbb{F}_q . In this case, the above matrix is Vandermonde, thus full rank with all rows and columns being linearly independent. Hence, $(\alpha^{q-\vartheta-1}, \alpha^{2(q-\vartheta-1)}, \dots, \alpha^{\rho(q-\vartheta-1)})$ is linearly independent of the rows in $\mathbf{G}_{\mathcal{C}^\perp}$, and hence $(\alpha_1^{q-\vartheta-1}, \alpha_2^{q-\vartheta-1}, \dots, \alpha_\rho^{q-\vartheta-1}) \notin \mathcal{C}^\perp$. For $\vartheta \in [1, k_C - z_s]$, we have $q - \vartheta - 1 \geq q - k_C + z_s - 1$. Hence, the statement holds for $q - k_C + z_s - 1 > \rho - 2$, and thus for $q > \rho + k_C - z_s - 1$. \square

This concludes the proof of Theorem 3. \square

C. Proof of Lemma 1

Proof. Considering the rate of the secret sharing stage and the PIR scheme following the construction in [17], the communication cost in \mathbb{F}_{c+1} is given by

$$\frac{Tsc \cdot n(n-1)}{k_C - z_s} + \frac{sc}{k_C - z_s} \frac{k_C n}{n - k_C - z_q + 1},$$

which is a convex optimization over the convex set $z_q + 1 \leq k_C \leq n - z_q$ since being a sum of convex functions over that set. For $c_1 = (n - z_q + 1)$ and $c_2 = Tn(n-1)$, we have

$$\begin{aligned} &\frac{\partial}{\partial k_C} \frac{Tsc \cdot n(n-1)}{k_C - z_s} + \frac{sc}{k_C - z_s} \frac{k_C n}{n - k_C - z_q + 1} \\ &= sc \left(-\frac{Tn(n-1)}{(k_C - z_s)^2} + \frac{n(k_C^2 - z_s(n - z_q + 1))}{(k_C - z_s)^2(n - k_C - z_q + 1)^2} \right) \\ &= sc \left(-\frac{c_2(c_1^2 + k_C^2 - 2c_1k_C) - n(k_C^2 - c_1z_s)}{(k_C - z_s)^2(n - k_C - z_q + 1)^2} \right) \\ &= scn \left(-\frac{k_C^2(c_2 - n) - 2k_Cc_1c_2 + c_1^2c_2 + nc_1z_s}{(k_C - z_s)^2(n - k_C - z_q + 1)^2} \right). \end{aligned}$$

Setting the derivative to zero, we obtain

$$k'_C = \frac{1}{2(c_2 - n)} \left(2c_1c_2 - \sqrt{4c_1^2c_2^2 - 4(c_2 - n)(c_1^2c_2 + nc_1z_s)} \right),$$

and hence the statement above. \square

REFERENCES

- [1] M. Egger, R. Urbanke, and R. Bitar, "Federated one-shot learning with data privacy and objective-hiding," accepted for presentation at *IEEE International Symposium on Information Theory (ISIT)*, 2025.
- [2] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *ACM Conference on Computer and Communications Security*, 2017, pp. 1175–1191.
- [3] P. Kairouz, H. B. McMahan, B. Aven, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *Foundations and trends® in machine learning*, vol. 14, no. 1–2, pp. 1–210, 2021.

- [4] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, Y. Li, X. Liu, and B. He, "A survey on federated learning systems: Vision, hype and reality for data privacy and protection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 4, pp. 3347–3366, 2021.
- [5] J. Wen, Z. Zhang, Y. Lan, Z. Cui, J. Cai, and W. Zhang, "A survey on federated learning: challenges and applications," *International Journal of Machine Learning and Cybernetics*, vol. 14, no. 2, pp. 513–535, 2023.
- [6] M. Ye, X. Fang, B. Du, P. C. Yuen, and D. Tao, "Heterogeneous federated learning: State-of-the-art and research challenges," *ACM Computing Surveys*, vol. 56, no. 3, pp. 1–44, 2023.
- [7] F. Tang, S. Liang, G. Ling, and J. Shan, "IHFVL: a privacy-enhanced intention-hiding vertical federated learning framework for medical data," *Cybersecurity*, vol. 6, no. 1, p. 37, 2023.
- [8] Y. Liu, W. Zhang, and J. Wang, "Adaptive multi-teacher multi-level knowledge distillation," *Neurocomputing*, vol. 415, pp. 106–113, 2020.
- [9] K. Chadha, M. Jagielski, N. Papernot, C. Choquette-Choo, and M. Nasr, "Auditing private prediction," *arXiv preprint arXiv:2402.09403*, 2024.
- [10] D. Karpuk, "Private computation of systematically encoded data with colluding servers," in *IEEE International Symposium on Information Theory (ISIT)*, 2018, pp. 2112–2116.
- [11] N. Raviv and D. A. Karpuk, "Private polynomial computation from lagrange encoding," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 553–563, 2019.
- [12] H. Sun and S. A. Jafar, "The capacity of private computation," *IEEE Transactions on Information Theory*, vol. 65, no. 6, pp. 3880–3897, 2018.
- [13] R. Tandon, M. Abdul-Wahid, F. Almoualem, and D. Kumar, "PIR from storage constrained databases-coded caching meets pir," in *IEEE International Conference on Communications (ICC)*, 2018, pp. 1–7.
- [14] M. Mirmohseni and M. A. Maddah-Ali, "Private function retrieval," in *Iran Workshop on Communication and Information Theory*, 2018, pp. 1–6.
- [15] S. A. Obead and J. Kliewer, "Achievable rate of private function retrieval from mds coded databases," in *IEEE International Symposium on Information Theory (ISIT)*, 2018, pp. 2117–2121.
- [16] B. Chor, E. Kushilevitz, O. Goldreich, and M. Sudan, "Private information retrieval," *Journal of the ACM*, vol. 45, no. 6, pp. 965–981, 1998.
- [17] R. Freij-Hollanti, O. W. Gnille, C. Hollanti, and D. A. Karpuk, "Private information retrieval from coded databases with colluding servers," *SIAM Journal on Applied Algebra and Geometry*, vol. 1, no. 1, pp. 647–664, 2017.
- [18] N. Raviv, I. Tamo, and E. Yaakobi, "Private information retrieval in graph-based replication systems," *IEEE Transactions on Information Theory*, vol. 66, no. 6, pp. 3590–3602, 2019.
- [19] Z. Jia and S. A. Jafar, "On the asymptotic capacity of x-secure t-private information retrieval with graph-based replicated storage," *IEEE Transactions on Information Theory*, vol. 66, no. 10, pp. 6280–6296, 2020.
- [20] K. Banawan and S. Ulukus, "Private information retrieval from non-replicated databases," in *IEEE International Symposium on Information Theory (ISIT)*, 2019, pp. 1272–1276.
- [21] Q. Wang and M. Skoglund, "Symmetric private information retrieval from mds coded distributed storage with non-colluding and colluding servers," *IEEE Transactions on Information Theory*, vol. 65, no. 8, pp. 5160–5175, 2019.
- [22] S. Ulukus, S. Avestimehr, M. Gastpar, S. A. Jafar, R. Tandon, and C. Tian, "Private retrieval, computing, and learning: Recent progress and future challenges," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 3, pp. 729–748, 2022.
- [23] R. G. D'Oliveira and S. El Rouayheb, "A guided walk through coded private information retrieval," *IEEE BITS the Information Theory Magazine*, 2024.
- [24] J. Zhu, Q. Yan, X. Tang, and S. Li, "Symmetric private polynomial computation from lagrange encoding," *IEEE Transactions on Information Theory*, vol. 68, no. 4, pp. 2704–2718, 2022.
- [25] S. A. Obead, H.-Y. Lin, E. Rosnes, and J. Kliewer, "Private linear computation for noncolluding coded databases," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 3, pp. 847–861, 2022.
- [26] B. Tahmasebi and M. A. Maddah-Ali, "Private function computation," in *IEEE International Symposium on Information Theory (ISIT)*, 2020, pp. 1118–1123.
- [27] —, "Private sequential function computation," in *IEEE International Symposium on Information Theory (ISIT)*, 2019, pp. 1667–1671.
- [28] S. A. Obead, H.-Y. Lin, E. Rosnes, and J. Kliewer, "Private polynomial function computation for noncolluding coded databases," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 1800–1813, 2022.
- [29] Y. Zhang, T. Etzion, and E. Yaakobi, "Bounds on the length of functional PIR and batch codes," *IEEE Transactions on Information Theory*, vol. 66, no. 8, pp. 4917–4934, 2020.
- [30] H. Jia and Z. Jia, "X-secure t-private linear computation with graph based replicated storage," in *IEEE International Symposium on Information Theory (ISIT)*, 2023, pp. 1586–1591.
- [31] —, "The asymptotic capacity of x-secure t-private linear computation with graph based replicated storage," *IEEE Transactions on Information Theory*, 2024.
- [32] N. Esmati, A. Heidarzadeh, and A. Sprintson, "Multi-server private linear computation with joint and individual privacy guarantees," in *International Symposium Problems of Redundancy in Information and Control Systems*, 2021, pp. 1–6.
- [33] A. Heidarzadeh and A. Sprintson, "Private computation with individual and joint privacy," in *IEEE International Symposium on Information Theory (ISIT)*, 2020, pp. 1112–1117.
- [34] M. H. Mousavi, M. A. Maddah-Ali, and M. Mirmohseni, "Private inner product retrieval for distributed machine learning," in *IEEE International Symposium on Information Theory (ISIT)*, 2019, pp. 355–359.
- [35] F. Kazemi and A. Sprintson, "Multi-server private linear transformation with joint privacy," in *International Symposium Problems of Redundancy in Information and Control Systems*, 2021, pp. 182–187.
- [36] A. Heidarzadeh, N. Esmati, and A. Sprintson, "Single-server private linear transformation: The joint privacy case," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 3, pp. 899–911, 2022.
- [37] S.-S. Learning, "Semi-supervised learning," *MIT Press*, vol. 5, p. 2, 2006.
- [38] R. J. McEliece and D. V. Sarwate, "On sharing secrets and reed-solomon codes," *Communications of the ACM*, vol. 24, no. 9, pp. 583–584, 1981.
- [39] H. Sun and C. Tian, "Breaking the mds-pir capacity barrier via joint storage coding," *Information*, vol. 10, no. 9, p. 265, 2019.
- [40] C. Tian, "On the storage cost of private information retrieval," *IEEE Transactions on Information Theory*, vol. 66, no. 12, pp. 7539–7549, 2020.
- [41] S. N. Keramaati and S. Salehkalibar, "Private information retrieval from non-replicated databases with optimal message size," in *Iran Workshop on Communication and Information Theory*, 2020, pp. 1–6.
- [42] B. Sadeh, Y. Gu, and I. Tamo, "Bounds on the capacity of pir over graphs," in *2021 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2021, pp. 1913–1918.
- [43] Y. Yao and S. A. Jafar, "The capacity of 4-star-graph pir," in *2023 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2023, pp. 1603–1608.
- [44] Z. Jia, H. Sun, and S. A. Jafar, "Cross subspace alignment and the asymptotic capacity of x-secure t-private information retrieval," *IEEE Transactions on Information Theory*, vol. 65, no. 9, pp. 5783–5798, 2019.
- [45] O. Makkonen, D. A. Karpuk, and C. Hollanti, "Algebraic geometry codes for cross-subspace alignment in private information retrieval," in *IEEE International Symposium on Information Theory (ISIT)*, 2024, pp. 2874–2879.
- [46] O. Makkonen, D. Karpuk, and C. Hollanti, "Secret sharing for secure and private information retrieval: A construction using algebraic geometry codes," *arXiv preprint arXiv:2408.00542*, 2024.
- [47] R. De Prisco, A. De Santis, and F. Palmieri, "Bounds and protocols for graph-based distributed secret sharing," *IEEE Transactions on Dependable and Secure Computing*, vol. 21, no. 1, pp. 434–448, 2023.
- [48] S. Vithana, K. Banawan, and S. Ulukus, "Semantic private information retrieval," *IEEE Transactions on Information Theory*, vol. 68, no. 4, pp. 2635–2652, 2021.
- [49] S. A. Obead, H.-Y. Lin, E. Rosnes, and J. Kliewer, "On the capacity of private nonlinear computation for replicated databases," in *IEEE Information Theory Workshop (ITW)*, 2019, pp. 1–5.
- [50] J. H. Bell, K. A. Bonawitz, A. Gascón, T. Lepoint, and M. Raykova, "Secure single-server aggregation with (poly)logarithmic overhead," in *ACM SIGSAC Conference on Computer and Communications Security*, 2020, p. 1253–1269.
- [51] J. So, B. Güler, and A. S. Avestimehr, "Turbo-aggregate: Breaking the quadratic aggregation barrier in secure federated learning," *IEEE Journal on Selected Areas in Information Theory*, vol. 2, no. 1, pp. 479–489, 2021.
- [52] S. Kadhe, N. Rajaraman, O. O. Koyluglu, and K. Ramchandran, "Fast-SecAgg: Scalable secure aggregation for privacy-preserving federated learning," *arXiv preprint arXiv:2009.11248*, 2020.
- [53] J. So, C. J. Nolet, C.-S. Yang, S. Li, Q. Yu, R. E. Ali, B. Güler, and S. Avestimehr, "LightSecAgg: a lightweight and versatile design for secure aggregation in federated learning," in *Machine Learning and Systems*, vol. 4, 2022, pp. 694–720.

- [54] T. Jahani-Nezhad, M. A. Maddah-Ali, S. Li, and G. Caire, "Swiftagg+: Achieving asymptotically optimal communication loads in secure aggregation for federated learning," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 4, pp. 977–989, 2023.
- [55] R. Schlegel, S. Kumar, E. Rosnes, and A. G. i. Amat, "Codedpaddedfl and codedsecagg: Straggler mitigation and secure aggregation in federated learning," *IEEE Transactions on Communications*, vol. 71, no. 4, pp. 2013–2027, 2023.
- [56] H. U. Sami and B. Güler, "Secure aggregation for clustered federated learning," in *IEEE International Symposium on Information Theory (ISIT)*, 2023, pp. 186–191.
- [57] M. Egger, C. Hofmeister, A. Wachter-Zeh, and R. Bitar, "Private aggregation in hierarchical wireless federated learning with partial and full collusion," *arXiv preprint arXiv:2306.14088*, 2024.
- [58] Z. Tan, D. Yuan, and Z. Huang, "Privacy-preserving polynomial computing over distributed data," *arXiv preprint arXiv:2309.09315*, 2023.
- [59] J. Zhu, S. Li, and J. Li, "Information-theoretically private matrix multiplication from mds-coded storage," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 1680–1695, 2023.
- [60] S. Vithana and S. Ulukus, "Private read update write (pruw) in federated submodel learning (fsl): Communication efficient schemes with and without sparsification," *IEEE Transactions on Information theory*, 2023.
- [61] Q. Wang, H. Sun, and M. Skoglund, "Symmetric private information retrieval with mismatched coded messages and randomness," in *2019 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2019, pp. 365–369.
- [62] Z. Wang and S. Ulukus, "Symmetric private information retrieval with user-side common randomness," in *IEEE International Symposium on Information Theory (ISIT)*, 2021, pp. 2119–2124.
- [63] —, "Digital blind box: Random symmetric private information retrieval," in *IEEE Information Theory Workshop*, 2022, pp. 95–100.
- [64] O. Farràs and J. Ribes-González, "One-out-of-q ot combiners," *IEEE Transactions on Information Theory*, 2023.
- [65] J. Kittler, M. Hatef, R. P. Duin, and J. Matas, "On combining classifiers," *IEEE transactions on pattern analysis and machine intelligence*, vol. 20, no. 3, pp. 226–239, 1998.
- [66] L. I. Kuncheva, *Combining pattern classifiers: methods and algorithms*. John Wiley & Sons, 2014.
- [67] R. Polikar, "Ensemble based systems in decision making," *IEEE Circuits and systems magazine*, vol. 6, no. 3, pp. 21–45, 2006.
- [68] J. Vongkulbhisal, P. Vinayavekhin, and M. Visentini-Scarzanella, "Unifying heterogeneous classifiers with distillation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3175–3184.
- [69] L. Li, J. Gou, B. Yu, L. Du, and Z. Y. D. Tao, "Federated distillation: A survey," *arXiv preprint arXiv:2404.08564*, 2024.
- [70] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1631–1642.
- [71] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," *Advances in neural information processing systems*, vol. 28, 2015.
- [72] X. Li and D. Roth, "Learning question classifiers," in *COLING 2002: The 19th International Conference on Computational Linguistics*, 2002.
- [73] H. Tang, Y. Fu, L. Sun, J. Xue, D. Liu, Y. Li, Z. Ma, M. Wu, J. Pan, G. Wan *et al.*, "Reducing the gap between streaming and non-streaming transducer-based asr by adaptive two-stage knowledge distillation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.
- [74] G. Hinton, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.
- [75] R. Cramer, I. B. Damgård *et al.*, *Secure multiparty computation*. Cambridge University Press, 2015.
- [76] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.
- [77] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.
- [78] C. Dwork, "Differential privacy," in *International colloquium on automata, languages, and programming*, 2006, pp. 1–12.
- [79] N. Raviv and Z. Goldfeld, "Perfect subset privacy for data sharing and learning," in *IEEE International Symposium on Information Theory (ISIT)*, 2022, pp. 1850–1855.