Communication-Efficient Loss Minimization over Heterogeneous Data with Federated Hierarchical Ensemble Aggregation via Distillation

Sayantan Chowdhury SAYANTAN.CHOWDHURY@MAIL.UTORONTO.CA
Ben Liang LIANG@ECE.UTORONTO.CA
University of Toronto, Canada

Ali Tizghadam ALI.TIZGHADAM@TELUS.COM
Ilijc Albanese ILIJC.ALBANESE@TELUS.COM
TELUS, Canada

Abstract

Distributed optimization through federated learning (FL) suffers from data heterogeneity particularly when the client datasets are highly imbalanced. Model aggregation via ensemble distillation is an effective solution to address this issue. However, there is no previous work on ensemble distillation in FL that considers hierarchical model aggregation, which is important for reducing communication overhead over a large network. In this work, we propose new methods to enable ensemble distillation for a hierarchical FL system. We develop a Federated Hierarchical Ensemble Aggregation via Distillation (FedHEAD) algorithm that performs ensemble distillation by reusing the clients' local data within each network sector of the hierarchy. We also extend it to FedHEAD+ so as to take advantage of reference data when it is available at the server. We provide theoretical analysis on FedHEAD and FedHEAD+, showing that under a wide range of conditions, our proposed schemes achieve faster convergence than existing non-hierarchical alternatives. Furthermore, extensive experiments over computer vision, natural language processing, and network traffic classification datasets show that the proposed schemes are robust towards hierarchical model aggregation in the network.

1. Introduction

Ensemble distillation has shown great potential to address the challenge of data heterogeneity in federated learning (FL) [3, 10, 14, 30]. In particular, FedDF [14] significantly improves upon the standard FedAvg [16] by adding a server distillation phase where the global model is aligned to the ensemble of all local models through distillation. However, the original ensemble distillation schemes require a publicly available reference dataset for knowledge transfer, which must have probability distribution similar to the actual aggregate data distribution of the clients. Such reference dataset is often unavailable in FL, where data are often privacy sensitive and it is difficult to gather a large amount of data samples is difficult to gather by any node in the network. Data-free distillation schemes for FL [3, 30] mitigate these issues, but they suffer from poor performance compared with those schemes using a reference dataset.

Furthermore, the existing ensemble distillation schemes naturally utilize only the server in FL to build the ensemble and require direct communication between the server and the clients. Therefore, they are not directly applicable in the emerging hierarchical FL systems [1, 5, 13, 15, 23, 29]. In hierarchical FL, typified by a client-edge-cloud system, the task of model aggregation is first performed by multiple edge servers, and then a final aggregation is performed at the cloud server. The communication with edge servers instead of the distant cloud server can greatly reduce latency and communication cost over a large network. However, since the models are aggregated in multiple locations, there is no single node where all the client models are present to build an ensemble.

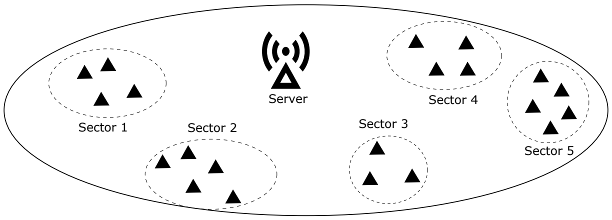


Figure 1: Federated learning over network sectors.

This renders ensemble distillation in a hierarchical FL a particularly challenging problem. We provide a brief survey of the relevant works on FL using knowledge distillation in Appendix A in the supplementary material.

In this paper, we introduce a novel federated hierarchical ensemble aggregation via distillation (FedHEAD) scheme that facilitates hierarchical FL with ensemble distillation. It does not require any reference dataset. Instead, we reuse the client data for distillation utilizing a decentralized strategy in FL. Furthermore, in case a reference dataset becomes available at the server, we extend our proposed scheme to FedHEAD+, which performs server distillation on top of FedHEAD.

Our main contributions are summarized as follows:

- We propose a novel FedHEAD framework that facilitates hierarchical FL using distillation over a network partitioned into sectors. It requires no reference dataset and reuses the client's local datasets for aligning the global model to the ensemble predictions. Since FedHEAD randomly chooses a client leader at each communication round, the existing convergence analysis techniques do not apply. Hence, we provide new analysis for strongly convex and smooth loss functions with bounded gradients, which indicates that under a wide range of conditions, FedHEAD converges faster than FedAvg. This is the first study theoretically showing that a reference data-free ensemble distillation scheme can converge faster than FedAvg.

- In the case that a reference dataset is available at the FL server, we further extend our scheme to FedHEAD+ in order to make use of the additional data. FedHEAD+ involves a server distillation phase on top of FedHEAD and thus can boost the learning performance further. We also provide convergence analysis for FedHEAD+ and find the conditions under which FedHEAD+ converges faster than FedDF.

- We conduct extensive experiments on computer vision, natural language processing, and network traffic classification tasks using a variety of datasets including SVHN, CIFAR10, CIFAR100, AG News, SST-2, Unicauca75, and Unicauca141. In all cases with varying degrees of data heterogeneity, we observe that FedHEAD and FedHEAD+ outperform a wide range of alternatives. More interestingly, our experiments demonstrate that FedHEAD, without using a reference dataset, often achieves a test accuracy at par with FedDF, which uses a reference dataset.

2. Preliminaries

We consider the distributed optimization of a global machine learning model through FL, over a network partitioned into M sectors as shown in Fig. 1. Each sector may belong to an organization (e.g., hospitals, service providers, etc., owned by a parent institution). Under such circumstances, it is well-known that hierarchical FL can substantially reduce the communication overhead while facilitating collaboration among all clients in the network.

For notational convenience only, we assume there is an equal number of clients per sector denoted by K, i.e., there are totally $N = KM$ clients in the network. Each client has a local dataset with true labels. The local dataset of the k-th client in the m-th sector is denoted by $\mathcal{D}_{km}$. Then the local loss for the k-th client in the m-th sector is given by

$$\mathcal{L}_{km}(\mathbf{w}) = \tfrac{1}{|\mathcal{D}_{km}|} \textstyle\sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}_{km}} l(f(\mathbf{x};\mathbf{w}),\mathbf{y}), \tag{1}$$

where $|\cdot|$ denotes the size of a set, $f(\cdot;\mathbf{w})$ is a classifier with model parameters $\mathbf{w}$, and the loss $l(f(\mathbf{x};\mathbf{w}),\mathbf{y})$ is computed between the classifier's output for a single datapoint $\mathbf{x}$ and its true label $\mathbf{y}$. We define the sector loss for the m-th sector as

$$\mathcal{L}_m(\mathbf{w}) = \textstyle\sum_{k=1}^{K} \tfrac{p_{km}}{p_m} \mathcal{L}_{km}(\mathbf{w}). \tag{2}$$

Finally, the global loss is formulated as:

$$\mathcal{L}(\mathbf{w}) = \textstyle\sum_{m=1}^{M} p_m \mathcal{L}_m(\mathbf{w}) = \sum_{m=1}^{M}\sum_{k=1}^{K} p_{km}\mathcal{L}_{km}(\mathbf{w}), \tag{3}$$

where $p_{km} = \frac{|\mathcal{D}_{km}|}{\sum_{m=1}^{M}\sum_{k=1}^{K}|\mathcal{D}_{km}|}$ and $p_m = \sum_{k=1}^{K} p_{km}$. The goal of our FL system is to learn a model $\mathbf{w}$ by minimizing the global loss function in a distributed manner without sharing local data.

In standard non-hierarchical FL, each client trains a local model for a few rounds on its local dataset. The local model parameters for the k-th client in the m-th sector are denoted by $\mathbf{w}_{km}$. After local training, all the clients send their models to the server for aggregating the model parameters. The server aggregates the local models into a global model and broadcasts the global parameters to all clients. This constitutes one *communication round*. In Appendix B, we discuss how model averaging can be leveraged for hierarchical FL and elaborate upon the motivation behind designing our proposed algorithms.

3. Proposed Federated Hierarchical Ensemble Aggregation via Distillation

We present a detailed description of the proposed FedHEAD algorithm and its extension FedHEAD+ in this section. We also provide theoretical analysis for both schemes under a general non-iid setting.

3.1. Description of FedHEAD Algorithm

In the FedHEAD framework, the server selects a client leader randomly from each sector in every communication round. This client leader will work as a mediator between the clients in that sector and the distant server. We reuse the local data of the client leader for ensemble distillation. The client leader is randomly chosen owing to the fact that the local data distribution of each client is skewed when the data are not iid among clients. By distilling over the dataset of different clients in different communication rounds, we mitigate the skewness of the distillation dataset.

The steps involved in FedHEAD are described in Algorithm 1 provided in Appendix C in the supplementary material. The server initializes a global model for all clients. At the beginning of every communication round, the server randomly designates a client in each sector as client leader and notifies all the clients in a sector who their leader is. The probability of choosing the k-th client in the m-th sector as leader is $\frac{p_{km}}{p_m}$. The server also provides each leader the client weights in their sector, $\{p_{km}\}_{k=1}^{K}, \ \forall m$, which will be later used by the client leader for model aggregation and ensemble formation.

Then, FedHEAD proceeds with two distinct phases: namely, the local training phase and the sector distillation phase.

Local training phase. Each client trains their model on their local data for n_l rounds and sends the trained model to their corresponding client leader. Let $\mathbf{w}_{km}^t$ denote the model of the k-th client in the m-th sector at the t-th communication round. Upon receiving the models from the clients, the client leaders perform pre-distillation sector aggregation as follows: $\mathbf{w}_m^t = \sum_{k=1}^K \frac{p_{km}}{p_m} \mathbf{w}_{km}^t, \ \forall m$. Next, the client leaders forward the aggregated models $\{\mathbf{w}_m^t\}_{m=1}^M$ to the server. The server performs pre-distillation server aggregation: $\mathbf{z}^t = \sum_{m=1}^M p_m \mathbf{w}_m^t$ and sends back the aggregated model $\mathbf{z}^t$ to the client leaders. Thus, only the client leader communicates with the distant server, and hierarchical aggregation is accommodated in FedHEAD. Note that pre-distillation aggregation performs global synchronization for distillation, which prevents client drift [7].

Sector distillation phase. Once each client leader receives the aggregated model $\mathbf{z}^t$ from the server, the distillation begins. Each client leader forms an ensemble of the local models in its sector. For a sample point $\mathbf{x}$, let us denote the output from the local model of the k-th client in the m-th sector as $f(\mathbf{x}; \mathbf{w}_{km}^t)$. For distillation, the ensemble model in each sector, $\bar{f}_m^t(\mathbf{x}) := \sum_{k=1}^K \frac{p_{km}}{p_m} f(\mathbf{x}; \mathbf{w}_{km}^t), \forall m$, is used as a teacher, and the aggregated model $\mathbf{z}^t$ serves as a student. Each client leader performs ensemble distillation on its local dataset for n_s rounds. Note that since the distillation is performed by reusing local datasets, FedHEAD does not require any reference data unlike FedDF. However, as the local datasets are non-iid, it is recommended to use early stopping by monitoring validation loss similar to FedDF. In particular, the client leaders keep track of validation loss during sector distillation and once the loss is plateaued for some consecutive number of iterations, the distillation stops. After distillation, the client leaders again send the distilled models $\{\mathbf{z}_m^t\}_{m=1}^M$ to the server for post-distillation aggregation as follows: $\mathbf{w}^t = \sum_{m=1}^M p_m \mathbf{z}_m^t$. Finally, the server broadcasts $\mathbf{w}^t$ to the client leaders and the client leaders distribute $\mathbf{w}^t$ as the updated global model within each sector.

We discuss the communication cost of FedHEAD in Appendix D in the supplementary material.

3.2. Convergence Analysis of FedHEAD

Since a client leader is randomly selected in every communication round for sector distillation, to prove the convergence of FedHEAD is non-trivial. Here, we give a convergence analysis of FedHEAD in a general non-iid setting.

Let us consider the stochastic gradient descent (SGD) updates throughout different phases in FedHEAD. During the local training phase, each client updates their models computing the gradient of the loss against the true labels. Recall that we denote the loss of the k-th client in the m-th sector by $\mathcal{L}_{km}(\cdot)$. Next, in the sector distillation phase, each client leader updates their models computing the gradient of the loss against the sector ensemble predictions. We denote the distillation loss by $\tilde{\mathcal{L}}_{km}(\cdot)$ if the k-th client in the m-th sector is designated as a client leader. $\tilde{\mathcal{L}}_{km}(\cdot)$ is given by

$$\tilde{\mathcal{L}}_{km}(\mathbf{w}) = \tfrac{1}{|\mathcal{D}_{km}|} \textstyle\sum_{(\mathbf{x},\mathbf{y}) \in \mathcal{D}_{km}} l(f(\mathbf{x}; \mathbf{w}), \bar{f}_m^t(\mathbf{x})). \tag{4}$$

Similar to (2) and (3), we can also define the sector distillation loss for the m-th sector as $\tilde{\mathcal{L}}_m(\cdot) = \sum_{k=1}^K \frac{p_{km}}{p_m} \tilde{\mathcal{L}}_{km}(\cdot)$ and the global distillation loss as $\tilde{\mathcal{L}}(\cdot) = \sum_{m=1}^M p_m \tilde{\mathcal{L}}_m(\cdot) = \sum_{m=1}^M \sum_{k=1}^K p_{km} \tilde{\mathcal{L}}_{km}(\cdot)$.

We require Assumptions 1, 2, 3, 4, and 5 about $\mathcal{L}_{km}(\cdot)$ and $\tilde{\mathcal{L}}_{km}(\cdot)$, regarding the properties of smoothness, strong convexity, bounded variance of stochastic gradient, bounded second moment of stochastic gradient and ϵ-noisy distillation, respectively. They are commonly assumed properties in the literature on convergence analysis. The details of these assumptions are provided in Appendix E.

Furthermore, we define the degree of client data heterogeneity, $\Gamma_{\text{client}} = \mathcal{L}^* - \sum_{m=1}^{M} \sum_{k=1}^{K} p_{km}\mathcal{L}^*_{km}$ and sector data heterogeneity, $\Gamma_{\text{sec}} = \mathcal{L}^* - \sum_{m=1}^{M} p_m \mathcal{L}^*_m$ where $\mathcal{L}^*_{km}, \mathcal{L}^*_m, \mathcal{L}^*$ are the minimum of the local, sector and global losses defined in (1), (2) and (3). From the definition of the expected risk minimization (ERM), $\sum_{m=1}^{M} \sum_{k=1}^{K} p_{km}\mathcal{L}^*_{km} \leq \sum_{m=1}^{M} p_m \mathcal{L}^*_m \leq \mathcal{L}^*$. Hence, $\Gamma_{\text{sec}} \leq \Gamma_{\text{client}}$.

Let $\mathbf{w}^1$, $\mathbf{w}^{t+1}$ and $\mathbf{w}^*$ be the initial global model, the global model after t communication rounds, and the minimizer of the global loss $\mathcal{L}(\cdot)$, respectively. Let us denote $\Delta_1 = \mathbb{E}||\mathbf{w}^1 - \mathbf{w}^*||^2$. We further consider a single iteration index τ that runs through all the local training, sector distillation, and server distillation phases. In other words, we define a virtual global model θ^τ as follows:

$$\theta^\tau = \begin{cases} \sum_{m=1}^{M} \sum_{k=1}^{K} p_{km}\mathbf{w}^\tau_{km}, \ \tau \in \text{local training phase}, \\ \sum_{m=1}^{M} p_m \mathbf{z}^\tau_m, \ \tau \in \text{sector distillation phase}, \end{cases} \tag{5}$$

where $\mathbf{w}^\tau_{km}$ is the model at the k-th client in the m-th sector at the τ-th iteration, and $\mathbf{z}^\tau_m$ is the model at the m-th client leader at the τ-th iteration. Note that in Algorithm 1, we only access θ^τ during the pre-distillation aggregation and post-distillation aggregation, but it is a mathematical construct that will be used throughout our proof.

Our analysis further considers early stopping during the sector distillation phase according to following criterion.

Definition 1 (Early stopping criterion for sector distillation) *The SGD update during sector distillation stops at the τ-th iteration if $\tilde{\mathcal{L}}(\theta^\tau) < \tilde{\mathcal{L}}(\mathbf{w}^*)$, $\tau \in$ sector distillation phase and θ^τ is returned as the final model.*

Note that this criterion prevents overfitting on the ensemble predictions and drifting away from $\mathbf{w}^*$, but it is not feasible to implement since we do not have access to $\mathbf{w}^*$. However, in practice, as shown in [14], early stopping can be implemented by monitoring validation loss.

Now, we proceed to show the convergence of FedHEAD.

Theorem 2 *Suppose Assumptions 1, 2, 3, 4, and 5 hold, and $\bar{n}_s \leq n_s$ is the average number of sector distillation rounds per communication round before early stopping according to Definition 1. FedHEAD guarantees that after t communication rounds*

$$\mathbb{E}[\mathcal{L}(\mathbf{w}^{t+1})] - \mathcal{L}^* \leq \frac{\kappa}{(n_l + \bar{n}_s)t + \gamma}\left(\frac{2B_l + 2B_s}{\mu} + \frac{\mu\gamma}{2}\Delta_1\right) \tag{6}$$

where $\kappa = \frac{L}{\mu}$, $\gamma = \max\{n_l, n_s, 8\kappa\}$,

$$B_l = 6L\Gamma_{client} + \sigma^2 \sum_{m=1}^{M} \sum_{k=1}^{K} p^2_{km} + 8(n_l - 1)^2 G^2, \tag{7}$$

$$B_s = 6L\Gamma_{sec} + 6L\epsilon + (1-\delta)\sigma^2 \sum_{m=1}^{M} p^2_m + (1-\delta)G^2 \sum_{m=1}^{M} p^2_m + 8(1-\delta)(n_s - 1)^2 G^2, \tag{8}$$

and $L, \mu, \sigma, G, \delta$ and ϵ are defined in Assumptions 1-5.

Furthermore, we add the following observation comparing the convergence rate of FedHEAD and FedAvg.

Remark 3 *For T_{comm} communication rounds, the convergence rates of FedHEAD and FedAvg are given by $O\left(\frac{B_l+B_s+\gamma G^2}{\mu(n_l+\bar{n}_s)T_{comm}}\right)$ and $O\left(\frac{B_l+\gamma G^2}{\mu n_l T_{comm}}\right)$, respectively. FedHEAD converges faster than FedAvg if $\bar{n}_s \geq n_l \frac{B_s}{B_l}$. In particular, when $n_s = n_l$, if $\frac{\delta}{\epsilon} \geq \frac{3L}{4n_s^2 G^2}$, there exists a $\bar{n}_s$ such that FedHEAD converges faster than FedAvg.*

The proof of Theorem 2 and Remark 3 mainly depend on the observation that the local training phase of FedHEAD is similar to FedAvg among clients minimizing the loss against true labels, and the sector distillation phase of FedHEAD is based on distributed optimization among client leaders minimizing the loss against ensemble predictions. Extra care is given to further account for the random selection of client leaders. The details are provided in Appendix E in the supplementary material.

3.3. Extension to FedHEAD+

For FedHEAD, we do not need a reference dataset for distillation. However, if an unlabeled reference dataset is available at the server (same as in FedDF), we want to make use of it. Therefore, we propose FedHEAD+, which performs ensemble distillation at the server on top of FedHEAD. Thus, for FedHEAD+, in addition to the local training and sector distillation phases of FedHEAD, we also have a server distillation phase.

Server distillation phase. After post-distillation server aggregation in Algorithm 1, we have the aggregated model $\mathbf{w}^t$. Now the server builds an ensemble of the client leaders' models, $\bar{f}^t(\mathbf{x}) := \sum_{m=1}^{M} p_m f(\mathbf{x}; \mathbf{z}_m^t)$. Then, ensemble distillation is performed for n_g rounds using reference dataset $\mathcal{D}_{\text{ref}}$ with $\mathbf{w}^t$ as a student and the ensemble as a teacher. Similar to FedDF, it is recommended to use early stopping by monitoring validation loss. At the end of distillation, the server broadcasts distilled $\mathbf{w}^t$ to the client leaders, which is the same as in Algorithm 1.

Note that since we have already performed ensemble distillation at each sector, every client leader's model contains the knowledge of the corresponding sector. The distillation at the server then aligns the model with global consensus. Clearly, FedHEAD+ does not incur any communication cost beyond FedHEAD, which is indicated in Table 1 given in Appendix D. The convergence analysis of FedHEAD+ is discussed in Appendix F and further details are provided in Appendix G in the supplementary material.

We include our experimental results in Appendix H in the supplementary material.

4. Conclusion

In this paper, we propose a new FedHEAD scheme that performs ensemble distillation in hierarchical FL in the distributed optimization of a global learning model, to reduce both the impact of data heterogeneity and the cost of communication. A salient feature of this scheme is that local datasets of the clients are reused for the purpose of distillation. With theoretical analysis, we show that FedHEAD converges faster than FedAvg under mild conditions. Furthermore, we extend the proposed scheme to FedHEAD+ to make use of a reference dataset when it is available at the server. We derive conditions under which FedHEAD+ converges faster than FedDF. Our experimental results confirm that FedHEAD and FedHEAD+ outperform their respective counterparts over a variety of CV, NLP, and NTC tasks. Furthermore, unlike other data-free distillation schemes in FL, FedHEAD often performs at par with FedDF, even though it does not require a reference dataset and it supports hierarchical ensemble distillation for significantly reduced communication overhead. Further experiments on the impact of the number of sectors and random choice of client leader reveal that FedHEAD and FedHEAD+ are robust towards hierarchical aggregation over network sectors.

References

[1] M. Abad, E. Ozfatura, D. Gunduz, and O. Ercetin. Hierarchical federated learning across heterogeneous cellular networks. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

[2] I. Bistritz, A. J. Mann, and N. Bambos. Distributed distillation for on-device learning. In *Proc. Neural Information Processing Systems (NeurIPS)*, 2020.

[3] H. Chen, C. Wang, and H. Vikalo. The best of both worlds: Accurate global and personalized models through federated learning with data-free hyper-knowledge distillation. In *Proc. International Conference on Learning Representations*, 2023.

[4] Y. Deng, J. Ren, C. Tang, F. Lyu, Y. Liu, and Y. Zhang. A hierarchical knowledge transfer framework for heterogeneous federated learning. In *Proc. IEEE INFOCOM Conference on Computer Communications*, 2023.

[5] J. Feng, L. Liu, Q. Pei, and K. Li. Min-max cost optimization for efficient hierarchical federated learning in wireless edge networks. *IEEE Transactions on Parallel and Distributed Systems*, 33(11):2687–2700, 2022.

[6] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[7] S. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proc. International Conference on Machine Learning*, 2020.

[8] A. Krizhevsky. Learning multiple layers of features from tiny images. *Technical Report*, 2009.

[9] Y. Le and X. Yang. Tiny imagenet visual recognition challenge. *Stanford course CS 231N*, 2015.

[10] D. Li and J. Wang. Fedmd: Heterogenous federated learning via model distillation. *ArXiv preprint*, arXiv:1910.03581, 2019.

[11] Q. Li, B. He, and D. Song. Model-contrastive federated learning. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[12] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang. On the convergence of fedavg on non-iid data. In *Proc. International Conference on Learning Representations, (ICLR)*, 2020.

[13] W. Lim, J. Ng, Z. Xiong, D. Niyato, C. Miao, and D. Kim. Dynamic edge association and resource allocation in self-organizing hierarchical federated learning networks. *IEEE Journal on Selected Areas in Communications*, 39(12):3640–3653, 2021.

[14] T. Lin, L. Kong, S. U. Stich, and M. Jaggi. Ensemble distillation for robust model fusion in federated learning. In *Proc. Advances in Neural Information Processing Systems*, 2020.

[15] L. Liu, J. Zhang, S. Song, and K. Letaief. Client-edge-cloud hierarchical federated learning. In *Proc. IEEE International Conference on Communications (ICC)*, 2020.

[16] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proc. International Conference on Artificial Intelligence and Statistics*, 2017.

[17] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, A. Ng, and Z. Zhang. Reading digits in natural images with unsupervised feature learning. In *Proc. NeurIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[18] S. Park, K. Hong, and G. Hwang. Towards understanding ensemble distillation in federated learning. In *Proc. International Conference on Machine Learning*, 2023.

[19] J. Rojas. Universidad del Cauca traffic dataset, 2017. URL https://www.kaggle.com/datasets/jsrojas/ip-network-traffic-flows-labeled-with-87-apps.

[20] J. Rojas. Universidad del Cauca traffic dataset, 2019. URL https://www.kaggle.com/datasets/jsrojas/labeled-network-traffic-flows-114-applications.

[21] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. Manning, A. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proc. Conference on Empirical Methods in Natural Language Processing*, 2013.

[22] Y. Tan, G. Long, L. Liu, T. Zhou, Q. Lu, J. Jiang, and C. Zhang. Fedproto: Federated prototype learning over heterogeneous devices. *ArXiv preprint*, arXiv:2105.00243, 2021.

[23] Z. Wang, H. Xu, J. Liu, H. Huang, C. Qiao, and Y. Zhao. Resource-efficient federated learning with hierarchical aggregation in edge computing. In *Proc. IEEE INFOCOM Conference on Computer Communications*, 2021.

[24] Y. Xu, Y. Xu, Q. Qian, H. Li, and R. Jin. Towards understanding label smoothing. *ArXiv preprint*, arXiv:2006.11653, 2020.

[25] L. Yuan, F. Tay, G. Li, T. Wang, and J. Feng. Revisiting knowledge distillation via label smoothing regularization. In *Proc. IEEE CVPR*, 2020.

[26] M. Yurochkin, M. Agarwal, S. Ghosh, K. Greenewald, N. Hoang, and Y. Khazaeni. Bayesian nonparametric federated learning of neural networks. In *Proc. International Conference on Machine Learning*, 2019.

[27] J. Zhang, S. Guo, X. Ma, H. Wang, W. Xu, and F. Wu. Parameterized knowledge transfer for personalized federated learning. In *Proc. NeurIPS*, 2021.

[28] X. Zhang, J. Zhao, and Y. LeCun. Character-level convolutional networks for text classification. In *Proc. NeurIPS*, 2015.

[29] X. Zhou, X. Ye, K. Wang, W. Liang, N. Nair, S. Shimizu, Z. Yan, and Q. Jin. Hierarchical federated learning with social context clustering-based participant selection for internet of medical things applications. *IEEE Transactions on Computational Social Systems*, 10(4): 1742–1751, 2023.

[30] Z. Zhu, J. Hong, and J. Zhou. Data-free knowledge distillation for heterogeneous federated learning. In *Proc. International Conference on Machine Learning*, 2021.

Supplementary Material

Appendix A. Related Works

Knowledge distillation in non-hierarchical FL. FedMD was proposed in [10] where each client has access to a *labeled* reference dataset. Before the original FL begins, the client models are pretrained on the reference dataset. Once FL commences, the clients share their predictions on the reference dataset and learn from the global consensus. Subsequently, the authors of [14] introduced FedDF to alleviate the need for true labels in the reference dataset. In every communication round, using an *unlabeled* reference dataset, the server distills knowledge from the ensemble of the client models to the global model. Another knowledge distillation-based personalized FL scheme was proposed in [27] where the clients update their models on both private and reference datasets and share a knowledge coefficient matrix that is aggregated at the server.

Collecting a large reference dataset is often infeasible in a privacy-sensitive FL setting. Hence, in recent years, data-free knowledge distillation schemes for FL have been proposed. In [30], the authors introduced FedGen, where a generator is used at the server instead of a reference dataset. The authors in [3] introduced FedHKD, a hyper-knowledge distillation framework, where in addition to sharing model parameters, the clients also share means of latent representations as well as the logit outputs for each class. However, when compared with FedMD, FedGen and FedHKD give degraded performance as data-free distillation results in less effective knowledge transfer.

Knowledge distillation in hierarchical FL. As far as we are aware, there does not exist any prior work on ensemble distillation over hierarchical FL. In a recent work termed FedHKT [4], the clients associated with each edge first collaborate to train an edge server model. Then, the knowledge from the ensemble of edge server models is transferred to a larger cloud server model using a reference dataset. Finally, the cloud server model is distilled back to the edge server models using the reference dataset again. However, FedHKT does not really perform hierarchical FL since the model aggregation only occurs in the edge servers, and not in the cloud server. Indeed, FedDF can be directly modified to give identical performance as FedHKT by first distilling from the ensemble to a larger server model and then from that larger server model to a smaller server model that is identical in size to the client models.

Convergence analysis of FL with distillation. The convergence of ensemble distillation in distributed setting has been investigated in [18] and [2]. However, the analysis in [18] is limited to kernel ridge regression (KRR). The analysis in [2] assumes mean squared loss. The authors in [14], provide a worst-case generalization bound for FL using ensemble distillation. To the best of our knowledge, there exists no prior work that provides a general mathematical analysis of ensemble distillation even for non-hierarchical FL.

In contrast, our convergence analysis for FedHEAD and FedHEAD+ is for ensemble distillation in hierarchical FL, with or without a reference dataset, and it is more general with standard assumptions such as smoothness and strong convexity of the loss function.

Appendix B. Naive Aggregation via Hierarchical Averaging

The advantage of model aggregation via averaging is that it is flexible enough to accommodate hierarchical FL. For a network partitioned into multiple sectors as shown in Fig. 1, a communication-efficient strategy is to perform model aggregation first at each sector level and then at the server level. This eliminates the need for all clients to directly send their models to the server. In each communication round, the aggregated model at the m-th sector can be written as

$$\mathbf{w}_m = \textstyle\sum_{k=1}^{K} \frac{|\mathcal{D}_{km}|}{\sum_{k=1}^{K} |\mathcal{D}_{km}|} \mathbf{w}_{km} = \sum_{k=1}^{K} \frac{p_{km}}{p_m} \mathbf{w}_{km}. \tag{9}$$

Next, the server-level aggregation is performed to obtain the global model as follows:

$$\mathbf{w} = \textstyle\sum_{m=1}^{M} p_m \mathbf{w}_m = \sum_{m=1}^{M} \sum_{k=1}^{K} p_{km} \mathbf{w}_{km}. \tag{10}$$

Note that this hierarchical aggregation gives us the same global model as in the standard FedAvg.

However, the downside of aggregation via averaging is that it severely suffers from data heterogeneity. As explained previously, ensemble distillation can address this issue by aligning the global model with the consensus of the local models [14].

Appendix C. FedHEAD Algorithm

Algorithm 1 FedHEAD Algorithm

```
Initialize: Broadcast initial global model, $\mathbf{w}^1$ to all clients.
for $t = 2, ..., T_{\text{comm}}$ do
    Server executes:
    Select a client leader randomly out of $K$ clients in each sector with probability of choosing the $k$-th client $\frac{p_{km}}{p_m}$.
    Notify the clients in each sector of the index of their client leader.
    Send the $m$-th client leader the client weights, $\{p_{km}\}_{k=1}^{K}, \forall m$.
    Clients execute in parallel:
    Train global model using local datasets for $n_l$ rounds.
    Send models $\{\mathbf{w}_{km}^t\}_{k=1,m=1}^{K,M}$ to corresponding client leaders.
    Client leaders execute in parallel:
    Perform pre-distillation sector aggregation: $\mathbf{w}_m^t = \sum_{k=1}^{K} \frac{p_{km}}{p_m} \mathbf{w}_{km}^t, \forall m$.
    Send aggregated models $\{\mathbf{w}_m^t\}_{m=1}^{M}$ to server.
    Server executes:
    Perform pre-distillation server aggregation: $\mathbf{z}^t = \sum_{m=1}^{M} p_m \mathbf{w}_m^t$.
    Broadcast $\mathbf{z}^t$ to client leaders.
    Client leaders execute in parallel:
    Build ensemble of models in each sector, $\sum_{k=1}^{K} \frac{p_{km}}{p_m} f(\mathbf{x}; \mathbf{w}_{km}^t), \forall m$.
    Perform ensemble distillation for $n_s$ rounds using each client leader's local dataset with $\mathbf{z}^t$ as a student and the ensemble as a teacher.
    Send distilled models $\{\mathbf{z}_m^t\}_{m=1}^{M}$ to server.
    Server executes:
    Perform post-distillation aggregation: $\mathbf{w}^t = \sum_{m=1}^{M} p_m \mathbf{z}_m^t$.
    Broadcast $\mathbf{w}^t$ to client leaders.
    Client leaders execute in parallel:
    Broadcast $\mathbf{w}^t$ to clients in each sector as the updated global model.
end for
```

Appendix D. Communication Cost of FedHEAD

Since the communication within each sector is localized, the total communication cost is dominated by the communication between the sectors and the distant server. It is shown in Table 1 for different algorithms after T_{comm} communication rounds. Note that in FedHEAD, each client leader talks with the server twice per communication round. For FedDF, the communication cost scales linearly with the number of clients, N, because all clients in the network need to send their models to the server in every communication round. However, thanks to hierarchical aggregation for FedAvg and FedHEAD, the communication cost between the sectors and the server is independent of N.

Table 1: Total communication cost for different FL algorithms (assuming unit cost per round trip).

Algorithm	Hierarchical	Total comm. cost betn. the sectors and the server
FedAvg (Hier. Agg.)	✓	MT_{comm}
FedDF	✗	NT_{comm}
FedHEAD	✓	$2MT_{\text{comm}}$
FedHEAD+	✓	$2MT_{\text{comm}}$

Appendix E. Proof of Theorem 2 and Remark 3

E.1. Assumptions on $\mathcal{L}_{km}(\cdot)$ and $\tilde{\mathcal{L}}_{km}(\cdot)$

We require the following set of assumptions about $\mathcal{L}_{km}(\cdot)$ and $\tilde{\mathcal{L}}_{km}(\cdot)$, which are common in the literature.

Assumption 1 (Smoothness) *$\mathcal{L}_{km},\ \forall k, m$, are L-smooth, i.e., for all $\mathbf{u}$ and $\mathbf{v}$, $\mathcal{L}_{km}(\mathbf{u}) \leq \mathcal{L}_{km}(\mathbf{v}) + (\mathbf{u}-\mathbf{v})^T \nabla \mathcal{L}_{km}(\mathbf{v}) + \frac{L}{2}||\mathbf{u}-\mathbf{v}||^2,\ \forall k, m$. The same assumption holds for $\tilde{\mathcal{L}}_{km}(\cdot),\ \forall k, m$.*

Assumption 2 (Strong convexity) *$\mathcal{L}_{km},\ \forall k, m$, are μ-strongly convex, i.e., for all $\mathbf{u}$ and $\mathbf{v}$, $\mathcal{L}_{km}(\mathbf{u}) \geq \mathcal{L}_{km}(\mathbf{v}) + (\mathbf{u}-\mathbf{v})^T \nabla \mathcal{L}_{km}(\mathbf{v}) + \frac{\mu}{2}||\mathbf{u}-\mathbf{v}||^2,\ \forall k, m$. The same assumption holds for $\tilde{\mathcal{L}}_{km}(\cdot),\ \forall k, m$.*

Assumption 3 (Bounded variance of stochastic gradient) *Let ξ be a stochastic sample. The variance of the stochastic gradient is bounded, i.e., for all $\mathbf{w}$, $\mathbb{E}\,||\nabla \mathcal{L}_{km}(\mathbf{w}, \xi) - \nabla \mathcal{L}_{km}(\mathbf{w})||^2 \leq \sigma^2,\ \forall k, m$. Furthermore, for $\tilde{\mathcal{L}}_{km}(\cdot),\ \forall k, m$, we assume the variance of the stochastic gradient is more tightly bounded, i.e., for all $\mathbf{w}$, $\mathbb{E}\left|\left|\nabla \tilde{\mathcal{L}}_{km}(\mathbf{w}, \xi) - \nabla \tilde{\mathcal{L}}_{km}(\mathbf{w})\right|\right|^2 \leq (1-\delta)\sigma^2,\ \forall k, m$, where $0 \leq \delta \leq 1$.*

Assumption 4 (Bounded second moment of stochastic gradient) *The second moment of the stochastic gradient is uniformly bounded, i.e., for all $\mathbf{w}$, $\mathbb{E}\,||\nabla \mathcal{L}_{km}(\mathbf{w}, \xi)||^2 \leq G^2,\ \forall k, m$. Furthermore, for $\tilde{\mathcal{L}}_{km}(\cdot),\ \forall k, m$, we assume the second moment of the stochastic gradient is more tightly bounded i.e., for all $\mathbb{E}\left|\left|\nabla \tilde{\mathcal{L}}_{km}(\mathbf{w}, \xi)\right|\right|^2 \leq (1-\delta)G^2,\ \forall k, m$, where $0 \leq \delta \leq 1$.*

Assumption 5 (ϵ-noisy distillation) *Distillation losses are ϵ-noisy, i.e., for all $\mathbf{w}$, $|\tilde{\mathcal{L}}_{km}(\mathbf{w}) - \mathcal{L}_{km}(\mathbf{w})| \leq \frac{\epsilon}{2},\ \forall k, m$.*

Assumptions 3 and 4 about $\tilde{\mathcal{L}}_{km}(\cdot),\ \forall k, m$ are due to the fact that knowledge distillation is a sophisticated label smoothing technique [24, 25] and δ could be viewed as the benefit of distillation. However, since the ensemble teacher's predictions are noisy, the effectiveness of distillation also depends on how accurate the teacher is, which is indicated by ϵ in Assumption 5. Whether distilling from an ensemble teacher improves performance depends upon the balance between δ and ϵ.

E.2. Proof of Theorem 2

To track the performance of learning over time, we define $\Delta_\tau = \mathbb{E}||\theta^\tau - \mathbf{w}^*||^2$. The expectation is taken over all sources of randomness in the corresponding FL algorithm. For FedHEAD, the randomness is due to stochastic gradients as well as the choice of the client leader in each communication round. Note that this definition is consistent with the definition of Δ_1 in the statement of Theorem 2 since $\theta^1 = \mathbf{w}^1$.

Local training phase. In the local training phase of FedHEAD, the clients run FedAvg computing the gradient of loss against the true labels. Therefore, we can borrow part of the results on the convergence of FedAvg from [12]. Let us denote the stochastic gradient at the k-th client in m-th sector as $\mathbf{g}_{km}^\tau(\xi) = \nabla\mathcal{L}_{km}(\mathbf{w}_{km}^\tau, \xi)$ and the full gradient, $\mathbf{g}_{km}^\tau = \mathbb{E}\mathbf{g}_{km}^\tau(\xi)$. The learning rate at τ-th iteration is given by $\eta_\tau = \frac{\beta}{\tau+\gamma}$, where $\beta > \frac{1}{\mu}$ and $\gamma > 0$ such that $\eta_1 \le \frac{1}{4L}$ and $\eta_\tau \le 2\min\{\eta_{\tau+n_l}, \eta_{\tau+n_s}\},\ \forall\tau$. The following three lemmas is a restating of Lemma 1, 2, and 3 from [12] and thus are given without proof.

Lemma 4 *Suppose Assumption 1 and 2 hold. If $\eta_\tau \le \frac{1}{4L}$, then in the local training phase of FedHEAD, we have*

$$\begin{aligned}\mathbb{E}\left\|\theta^{\tau+1} - \mathbf{w}^*\right\|^2 \le (1-\mu\eta_\tau)\mathbb{E}\left\|\theta^\tau - \mathbf{w}^*\right\|^2 + 6L\eta_\tau^2\Gamma_{client} \\ + \eta_\tau^2\mathbb{E}\left\|\sum_{m=1}^{M}\sum_{k=1}^{K} p_{km}\mathbf{g}_{km}^\tau(\xi) - \sum_{m=1}^{M}\sum_{k=1}^{K} p_{km}\mathbf{g}_{km}^\tau\right\|^2 \\ + 2\sum_{m=1}^{M}\sum_{k=1}^{K} p_{km}\mathbb{E}\left\|\theta^\tau - \mathbf{w}_{km}^\tau\right\|^2.\end{aligned} \tag{11}$$

Lemma 5 *Suppose Assumption 3 holds. We have*

$$\mathbb{E}\left\|\sum_{m=1}^{M}\sum_{k=1}^{K} p_{km}\mathbf{g}_{km}^\tau(\xi) - \sum_{m=1}^{M}\sum_{k=1}^{K} p_{km}\mathbf{g}_{km}^\tau\right\|^2 \le \sigma^2\sum_{m=1}^{M}\sum_{k=1}^{K} p_{km}^2. \tag{12}$$

Lemma 6 *Suppose Assumption 4 holds, and η_τ is non-increasing and $\eta_\tau \le 2\eta_{\tau+n_l}$ for all $\tau \ge 0$. We have*

$$\sum_{m=1}^{M}\sum_{k=1}^{K} p_{km}\mathbb{E}\left\|\theta^\tau - \mathbf{w}_{km}^\tau\right\|^2 \le 4\eta_\tau^2(n_l-1)^2G^2. \tag{13}$$

Plugging (12) and (13) into (11), we obtain a recurrence relation for a single-step SGD update during the local training phase of FedHEAD as follows:

$$\Delta_{\tau+1} \le (1-\mu\eta_\tau)\Delta_\tau + \eta_\tau^2 B_l,\ \tau \in \text{local training phase}, \tag{14}$$

where $B_l = 6L\Gamma_{\text{client}} + \sigma^2\sum_{m=1}^{M}\sum_{k=1}^{K} p_{km}^2 + 8(n_l-1)^2G^2$.

Sector distillation phase. Next, in the sector distillation phase, the client leaders run FedAvg computing gradient against sector ensemble predictions. Hence, the model is updated using the stochastic gradient of distillation loss on the client leaders until early stopping according to Definition 1. However, the main complication in sector distillation phase is that, the m-th client leader itself is randomly selected out of all the clients in the m-th sector.

Let us denote the id of the client that is selected as client leader in the m-th sector as k'. Then, we write the stochastic gradient of the distillation loss at the client leader in m-th sector as $\tilde{\mathbf{g}}^{\tau}_{k'm}(\xi) = \nabla\tilde{\mathcal{L}}_{k'm}(\mathbf{z}^{\tau}_m, \xi)$ and the full gradient of the m-th sector as $\tilde{\mathbf{g}}^{\tau}_m = \mathbb{E}\tilde{\mathbf{g}}^{\tau}_{k'm}(\xi)$ where the expectation is taken over two sources of randomness: the choice of stochastic sample, ξ, and the choice of client leader, k'. We observe $\tilde{\mathbf{g}}^{\tau}_m = \mathbb{E}\tilde{\mathbf{g}}^{\tau}_{k'm}(\xi) = \sum_{k=1}^{K}\mathbb{E}[\mathbb{1}(k'=k)\tilde{\mathbf{g}}^{\tau}_{km}(\xi)] = \sum_{k=1}^{K}\mathbb{E}[\mathbb{1}(k'=k)]\tilde{\mathbf{g}}^{\tau}_{km} = \sum_{k=1}^{K}\frac{p_{km}}{p_m}\tilde{\mathbf{g}}^{\tau}_{km}$ where we use $\mathbb{E}[\mathbb{1}(k'=k)] = \Pr(k'=k) = \frac{p_{km}}{p_m}$, which is the probability of choosing the k-th client as leader in the m-th sector.

To proceed, we require three additional lemmas as follows.

Lemma 7 *Suppose Assumption 1, 2 and 5 hold. If $\eta_\tau \le \frac{1}{4L}$, then in the sector distillation phase of FedHEAD, we have*

$$\mathbb{E}\left\|\theta^{\tau+1}-\mathbf{w}^*\right\|^2 \le (1-\mu\eta_\tau)\mathbb{E}\left\|\theta^{\tau}-\mathbf{w}^*\right\|^2 + 6L\eta_\tau^2\Gamma_{sec} + 6L\eta_\tau^2\epsilon + \eta_\tau^2\mathbb{E}\left\|\sum_{m=1}^{M}p_m\tilde{\mathbf{g}}^{\tau}_{k'm}(\xi) - \sum_{m=1}^{M}p_m\tilde{\mathbf{g}}^{\tau}_m\right\|^2 + 2\sum_{m=1}^{M}p_m\mathbb{E}\left\|\theta^{\tau}-\mathbf{z}^{\tau}_m\right\|^2. \tag{15}$$

Proof Given a fixed set of the sectors' client leaders in each communication round, the sector distillation phase resembles FedAvg among M client leaders. However, the loss function is no longer against true labels but the distillation loss against ensemble predictions. Therefore, we rewrite Lemma 4 for M participants substituting the loss function against true labels with the distillation loss against ensemble predictions. Thus, firstly, the stochastic gradients needs to be replaced by $\tilde{\mathbf{g}}^{\tau}_{k'm}(\xi)\ \forall m$. Secondly, before early stopping according to Definition 1, we have $\tilde{\mathcal{L}}(\theta^{\tau}) \ge \tilde{\mathcal{L}}(\mathbf{w}^*)$. Thirdly, Γ_{client} will be changed to $\sum_{m=1}^{M}p_m\tilde{\mathcal{L}}_m(\mathbf{w}^*) - \sum_{m=1}^{M}p_m\tilde{\mathcal{L}}^*_m$ where $\tilde{\mathcal{L}}^*_m$ is the minimum of the distillation loss $\tilde{\mathcal{L}}_m(\cdot)$. We have

$$\begin{aligned}
&\sum_{m=1}^{M}p_m\tilde{\mathcal{L}}_m(\mathbf{w}^*) - \sum_{m=1}^{M}p_m\tilde{\mathcal{L}}^*_m\\
&= \sum_{m=1}^{M}p_m\tilde{\mathcal{L}}_m(\mathbf{w}^*) - \sum_{m=1}^{M}p_m\mathcal{L}_m(\mathbf{w}^*) + \sum_{m=1}^{M}p_m\mathcal{L}_m(\mathbf{w}^*) - \sum_{m=1}^{M}p_m\mathcal{L}^*_m + \sum_{m=1}^{M}p_m\mathcal{L}^*_m - \sum_{m=1}^{M}p_m\tilde{\mathcal{L}}^*_m\\
&\le \sum_{m=1}^{M}p_m|\tilde{\mathcal{L}}_m(\mathbf{w}^*) - \mathcal{L}_m(\mathbf{w}^*)| + \mathcal{L}^* - \sum_{m=1}^{M}p_m\mathcal{L}^*_m + \sum_{m=1}^{M}p_m|\mathcal{L}^*_m - \tilde{\mathcal{L}}^*_m|\\
&\overset{(a)}{\le} \sum_{m=1}^{M}p_m|\tilde{\mathcal{L}}_m(\mathbf{w}^*) - \mathcal{L}_m(\mathbf{w}^*)| + \mathcal{L}^* - \sum_{m=1}^{M}p_m\mathcal{L}^*_m + \sum_{m=1}^{M}p_m\max_{\mathbf{w}}|\mathcal{L}_m(\mathbf{w}) - \tilde{\mathcal{L}}_m(\mathbf{w})|\\
&\overset{(b)}{\le} \frac{\epsilon}{2} + \mathcal{L}^* - \sum_{m=1}^{M}p_m\mathcal{L}^*_m + \frac{\epsilon}{2}\\
&= \Gamma_{\text{sec}} + \epsilon
\end{aligned} \tag{16}$$

where (a) is due to the fact $|\min_x f(x) - \min_x g(x)| \leq \max_x |f(x) - g(x)|$, and (b) is obtained from Assumption 5. Making the three modifications mentioned above, we deduce Lemma 7 from Lemma 4. ■

Lemma 8 *Suppose Assumption 3 and 4 hold. We have*

$$\mathbb{E}\left\|\sum_{m=1}^{M} p_m \tilde{\mathbf{g}}_{k'm}^{\tau}(\xi) - \sum_{m=1}^{M} p_m \tilde{\mathbf{g}}_m^{\tau}\right\|^2 \leq (1-\delta)\sigma^2 \sum_{m=1}^{M} p_m^2 + (1-\delta)G^2 \sum_{m=1}^{M} p_m^2. \tag{17}$$

Proof

$$\begin{aligned}
&\mathbb{E}\left|\left|\sum_{m=1}^{M} p_m \tilde{\mathbf{g}}_{k'm}^{\tau}(\xi) - \sum_{m=1}^{M} p_m \tilde{\mathbf{g}}_m^{\tau}\right|\right|^2 \\
&= \mathbb{E}\left|\left|\sum_{m=1}^{M}\sum_{k=1}^{K} p_m \mathbb{1}(k'=k)\tilde{\mathbf{g}}_{km}^{\tau}(\xi) - \sum_{m=1}^{M}\sum_{k=1}^{K} p_{km}\tilde{\mathbf{g}}_{km}^{\tau}\right|\right|^2 \\
&= \sum_{m=1}^{M} p_m^2 \mathbb{E}\left|\left|\sum_{k=1}^{K}\mathbb{1}(k'=k)\left(\tilde{\mathbf{g}}_{km}^{\tau}(\xi) - \sum_{k=1}^{K}\frac{p_{km}}{p_m}\tilde{\mathbf{g}}_{km}^{\tau}\right)\right|\right|^2 \\
&= \sum_{m=1}^{M}\sum_{k=1}^{K} p_m^2 \Pr(k'=k)\mathbb{E}\left|\left|\tilde{\mathbf{g}}_{km}^{\tau}(\xi) - \sum_{k=1}^{K}\frac{p_{km}}{p_m}\tilde{\mathbf{g}}_{km}^{\tau}\right|\right|^2 \\
&= \sum_{m=1}^{M} p_m^2 \sum_{k=1}^{K}\frac{p_{km}}{p_m}\mathbb{E}\left|\left|\tilde{\mathbf{g}}_{km}^{\tau}(\xi) - \sum_{k=1}^{K}\frac{p_{km}}{p_m}\tilde{\mathbf{g}}_{km}^{\tau}\right|\right|^2 \\
&= \sum_{m=1}^{M} p_m^2 \sum_{k=1}^{K}\frac{p_{km}}{p_m}\mathbb{E}\left|\left|\tilde{\mathbf{g}}_{km}^{\tau}(\xi) - \tilde{\mathbf{g}}_{km}^{\tau}\right|\right|^2 + \sum_{m=1}^{M} p_m^2 \sum_{k=1}^{K}\frac{p_{km}}{p_m}\left|\left|\tilde{\mathbf{g}}_{km}^{\tau} - \sum_{k=1}^{K}\frac{p_{km}}{p_m}\tilde{\mathbf{g}}_{km}^{\tau}\right|\right|^2 \\
&\overset{(a)}{\leq} \sum_{m=1}^{M} p_m^2 \sum_{k=1}^{K}\frac{p_{km}}{p_m}\mathbb{E}\left|\left|\tilde{\mathbf{g}}_{km}^{\tau}(\xi) - \tilde{\mathbf{g}}_{km}^{\tau}\right|\right|^2 + \sum_{m=1}^{M} p_m^2 \sum_{k=1}^{K}\frac{p_{km}}{p_m}\left|\left|\tilde{\mathbf{g}}_{km}^{\tau}\right|\right|^2 \\
&\overset{(b)}{\leq} (1-\delta)\sigma^2 \sum_{m=1}^{M} p_m^2 + (1-\delta)G^2 \sum_{m=1}^{M} p_m^2
\end{aligned} \tag{18}$$

where (a) is due to the fact that the variance is smaller than the second moment, and (b) is obtained from Assumption 3 and 4. ■

Lemma 9 *Suppose Assumption 4 holds, η_τ is non-increasing, and $\eta_\tau \leq 2\eta_{\tau+n_s}$ for all $\tau \geq 0$. We have*

$$\sum_{m=1}^{M} p_m \mathbb{E}\left\|\theta^{\tau} - \mathbf{z}_m^{\tau}\right\|^2 \leq 4\eta_\tau^2 (n_s - 1)^2 G^2. \tag{19}$$

Proof Since we perform global synchronization through pre-distillation aggregation (see lines 10-13 in Algorithm 1), for any $\tau \in$ sector distillation phase, there exists a $\tau_0 \leq \tau$, such that $\tau - \tau_0 \leq n_s - 1$ and $\mathbf{z}_m^{\tau_0} = \theta^{\tau_0}$, $\forall m$. Also, $\theta^\tau = \sum_{m=1}^M p_m \mathbf{z}_m^\tau$. Following the proof of Lemma 6 in [12], we have $\sum_{m=1}^M p_m \mathbb{E}\left\|\theta^\tau - \mathbf{z}_m^\tau\right\|^2 \leq 4\eta_\tau^2 (n_s-1)^2 G^2$. ■

We now continue the proof of Theorem 2. Plugging (17) and (19) into (15), we obtain a recurrence relation for a single-step SGD update during the sector distillation phase of FedHEAD as follows:

$$\Delta_{\tau+1} \leq (1-\mu\eta_\tau)\Delta_\tau + \eta_\tau^2 B_s,\ \tau \in \text{sector distillation phase}, \tag{20}$$

where $B_s = 6L\Gamma_{\text{sec}} + 6L\epsilon + (1-\delta)\sigma^2 \sum_{m=1}^M p_m^2 + (1-\delta)G^2 \sum_{m=1}^M p_m^2 + 8(1-\delta)(n_s-1)^2 G^2$.

Overall bound on Δ_τ. Let $v = \max\left\{\frac{\beta^2 B_l}{\beta\mu - 1}, \frac{\beta^2 B_s}{\beta\mu-1}, (\gamma+1)\Delta_1\right\}$. We prove that $\Delta_\tau \leq \frac{v}{\tau+\gamma}$, $\forall\tau$, by induction. Firstly, note that the inequality holds for Δ_1. Let it be true for some $\Delta_\tau, \tau \in$ local training phase. Then, we have

$$\begin{aligned}
\Delta_{\tau+1} &\leq (1-\mu\eta_\tau)\Delta_\tau + \eta_\tau^2 B_l \\
&\leq (1 - \frac{\beta\mu}{\tau+\gamma})\frac{v}{\tau+\gamma} + \frac{\beta^2 B_l}{(\tau+\gamma)^2} \\
&= \frac{\tau+\gamma-1}{(\tau+\gamma)^2}v + \left[\frac{\beta^2 B_l}{(\tau+\gamma)^2} - \frac{\beta\mu-1}{(\tau+\gamma)^2}v\right] \\
&\overset{(a)}{\leq} \frac{\tau+\gamma-1}{(\tau+\gamma)^2}v \\
&\leq \frac{\tau+\gamma-1}{(\tau+\gamma)^2-1}v \\
&= \frac{v}{\tau+\gamma+1}
\end{aligned} \tag{21}$$

where (a) is obtained by the definition of v. Note that the result also holds for $\tau \in$ sector distillation phase due to the definition of v. Therefore, from Assumption 1, we get

$$\mathbb{E}[\mathcal{L}(\theta^{\tau+1})] - \mathcal{L}^* \leq \frac{L}{2}\Delta_{\tau+1} \leq \frac{L}{2}\frac{v}{\tau+\gamma+1} \leq \frac{L}{2(\tau+\gamma+1)}\left(\frac{\beta^2 B_l}{\beta\mu-1} + \frac{\beta^2 B_s}{\beta\mu-1} + (\gamma+1)\Delta_1\right). \tag{22}$$

We set $\beta = \frac{2}{\mu}$ and $\gamma = \max\{n_l, n_s, 8\kappa\} - 1$. One can check $\eta_1 \leq \frac{1}{4L}$ and $\eta_\tau \leq 2\min\{\eta_{\tau+n_l}, \eta_{\tau+n_s}\}$, $\forall\tau$. After t communication rounds, the total number of iterations $\tau = (n_l + \bar{n}_s)t$ where $\bar{n}_s \leq n_s$ is the average number of sector distillation rounds per communication round before early stopping according to Definition 1, and $\theta^{\tau+1} \equiv \mathbf{w}^{t+1}$. Thus, we have

$$\mathbb{E}[\mathcal{L}(\mathbf{w}^{t+1})] - \mathcal{L}^* \leq \frac{\kappa}{(n_l+\bar{n}_s)t + \gamma + 1}\left(\frac{2B_l}{\mu} + \frac{2B_s}{\mu} + \frac{\mu(\gamma+1)}{2}\Delta_1\right). \tag{23}$$

Rewriting $\gamma := \gamma - 1$, we obtain the statement of Theorem 2.

E.3. Details of Remark 3

To compare against FedAvg, we reproduce Theorem 1 of [12] here.

Theorem 10 *Suppose Assumptions 1, 2, 3, and 4 hold. FedAvg guarantees that after t communication rounds*

$$\mathbb{E}[\mathcal{L}(\mathbf{w}^{t+1})] - \mathcal{L}^* \leq \frac{\kappa}{n_l t + \gamma} \left(\frac{2B_l}{\mu} + \frac{\mu\gamma}{2}\Delta_1 \right) \tag{24}$$

where $\kappa = \frac{L}{\mu}$, $\gamma = \max\{n_l, 8\kappa\}$, B_l is defined in (7), and $L, \mu, \sigma, G, \delta$ and ϵ are defined in Assumptions 1-5.

Now, for strongly convex loss functions, we have $\Delta_1 \leq \frac{4G^2}{\mu^2}$. Substituting this in (6) and (24), and setting $t = T_{\text{comm}}$, we obtain the convergence rates for FedHEAD and FedAvg in terms of dominant terms $O\left(\frac{B_l+B_s+\gamma G^2}{\mu(n_l+\bar{n}_s)T_{\text{comm}}}\right)$ and $O\left(\frac{B_l+\gamma G^2}{\mu n_l T_{\text{comm}}}\right)$, respectively. The convergence rate for FedHEAD is higher than FedAvg if $\frac{B_l+B_s}{n_l+\bar{n}_s} \leq \frac{B_l}{n_l}$. By mediant inequality, this is true if $\frac{B_s}{\bar{n}_s} \leq \frac{B_l}{n_l} \implies \bar{n}_s \geq n_l \frac{B_s}{B_l}$.

Furthermore, when $n_s = n_l$, if $B_s \leq B_l$, there exists a $\bar{n}_s$ such that $\bar{n}_s \leq n_s$ and $\bar{n}_s \geq n_l \frac{B_s}{B_l}$. As the variance is smaller than the second moment, we must have $\sigma^2 < G^2$. Also, since $\sum_{m=1}^M p_m^2 \leq 1$ and $n_s \geq 1$, we have $\sigma^2 \sum_{m=1}^M p_m^2 + G^2 \sum_{m=1}^M p_m^2 + 8(n_s - 1)^2 G^2 \leq 8n_s^2 G^2$. Thus, we can approximate $B_l \approx 6L\Gamma_{\text{client}} + 8n_l^2 G^2$ and $B_s \approx 6L\Gamma_{\text{sec}} + 6L\epsilon + 8(1-\delta)n_s^2 G^2$. Since $\Gamma_{\text{sec}} \leq \Gamma_{\text{client}}$, we have $\frac{\delta}{\epsilon} \geq \frac{3L}{4n_s^2 G^2} \implies B_s \leq B_l$. Hence, we obtain Remark 3 comparing FedHEAD against FedAvg.

Appendix F. Convergence Analysis of FedHEAD+

To perform convergence analysis on FedHEAD+, let us denote the distillation loss using the reference dataset at the server by $\tilde{\mathcal{L}}_{\text{ref}}(\mathbf{w}) = \frac{1}{|\mathcal{D}_{\text{ref}}|} \sum_{(\mathbf{x},\mathbf{y}) \in \mathcal{D}_{\text{ref}}} l(f(\mathbf{x};\mathbf{w}), \bar{f}^t(\mathbf{x}))$. We make an additional Assumption 6 as detailed in Appendix G, which postulates that the set of assumptions regarding $\tilde{\mathcal{L}}_{km}(\cdot)\ \forall k, m$, also applies to $\tilde{\mathcal{L}}_{\text{ref}}(\cdot)$.

Furthermore, only for the sake of analysis, let us denote the loss computed over the reference dataset against true labels as $\mathcal{L}_{\text{ref}}(\mathbf{w}) = \frac{1}{|\mathcal{D}_{\text{ref}}|} \sum_{(\mathbf{x},\mathbf{y}) \in \mathcal{D}_{\text{ref}}} l(f(\mathbf{x};\mathbf{w}), \mathbf{y})$. Then, we write the degree of reference data heterogeneity as $\Gamma_{\text{ref}} = \mathcal{L}_{\text{ref}}(\mathbf{w}^*) - \mathcal{L}^*_{\text{ref}}$ where $\mathcal{L}^*_{\text{ref}}$ is the minimum of the loss $\mathcal{L}_{\text{ref}}(\cdot)$. Since the reference data distribution at the server is different from the client data distribution, the minimizer of the loss over reference dataset $\mathcal{L}_{\text{ref}}(\cdot)$ is different from $\mathbf{w}^*$.

Similarly to the analysis on FedHEAD above, we define the virtual global model θ^τ below:

$$\theta^\tau = \begin{cases} \sum_{m=1}^M \sum_{k=1}^K p_{km}\mathbf{w}_{km}^\tau, \ \tau \in \text{local training phase}, \\ \sum_{m=1}^M p_m \mathbf{z}_m^\tau, \ \tau \in \text{sector distillation phase}, \\ \mathbf{w}^\tau, \ \tau \in \text{server distillation phase}, \end{cases} \tag{25}$$

where $\mathbf{w}^\tau$ is the student model in server distillation phase, which is updated, through computing the gradients against server ensemble predictions. In addition to early stopping in the sector distillation phase according to Definition 1, we also consider early stopping in the server distillation phase as follows.

Definition 11 (Early stopping criterion in server distillation phase) *The SGD update during server distillation stops at the τ-th iteration if $\tilde{\mathcal{L}}_{ref}(\theta^\tau) < \tilde{\mathcal{L}}_{ref}(\mathbf{w}^*)$, $\tau \in$ server distillation phase and θ^τ is returned as the final model.*

We have the following theorem about the convergence of FedHEAD+.

Theorem 12 *Suppose Assumptions 1, 2, 3, 4, 5, and 6 hold, and $\bar{n}_s \leq n_s$ and $\bar{n}_g \leq n_g$ are the average number of sector and server distillation rounds per communication round before early stopping according to Definitions 1 and 11. FedHEAD+ guarantees that after t communication rounds*

$$\mathbb{E}[\mathcal{L}(\mathbf{w}^{t+1})] - \mathcal{L}^* \leq \frac{\kappa}{(n_l + \bar{n}_s + \bar{n}_g)t + \gamma}\left(\frac{2B_l + 2B_s + 2B_g}{\mu} + \frac{\mu\gamma}{2}\Delta_1\right) \tag{26}$$

where $\kappa = \frac{L}{\mu}$, $\gamma = \max\{n_l, n_s, 8\kappa\}$, B_l and B_s are defined in (7) and (8), respectively,

$$B_g = 2L\Gamma_{ref} + 2L\epsilon + (1-\delta)\sigma^2, \tag{27}$$

and $L, \mu, \sigma, G, \delta$ and ϵ are defined in Assumptions 1-5.

Furthermore, we add the following observation comparing the convergence rate of FedHEAD+ and FedDF.

Remark 13 *For T_{comm} communication rounds, the convergence rate of FedHEAD+ and FedDF are given by $O\left(\frac{B_l+B_s+B_g+\gamma G^2}{\mu(n_l+\bar{n}_s+\bar{n}_g)T_{comm}}\right)$ and $O\left(\frac{B_l+B_g+\gamma G^2}{\mu(n_l+\bar{n}_g)T_{comm}}\right)$, respectively. FedHEAD+ converges faster than FedDF if $\bar{n}_s \geq n_l \frac{B_s\left(1+\frac{\bar{n}_g}{n_l}\right)}{B_l+B_g}$ In particular, when $n_g = n_s = n_l$, if $\Gamma_{ref} \geq 3\Gamma_{client}$ and $\delta \geq \frac{1}{2} + \frac{5L\epsilon}{8n_s^2G^2}$, there exists a $\bar{n}_s$ such that FedHEAD+ converges faster than FedDF.*

The proof of Theorem 12 and Remark 13 is similar to that of FedHEAD except for the additional server distillation phase. The details are provided in Appendix G in the supplementary material.

Interestingly, from our analysis of FedDF to arrive at Remark 13, we can also conclude that FedDF has faster convergence than FedAvg (see Theorem 15 in Appendix G). Note that the same conclusion was not proven but only inferred through numerical study in (Lin et al. 2020), but our analysis here provides a definitive proof.

We further note that the performance of FedDF and FedHEAD+ depends on the quality of the reference dataset. If the reference dataset collected by the server is not similar to the client dataset, the server ensemble distillation on the reference dataset will be ineffective for both FedDF and FedHEAD+. In that case, FedHEAD will converge faster.

Appendix G. Proof of Theorem 12 and Remark 13

G.1. Assumption on $\tilde{\mathcal{L}}_{\text{ref}}(\cdot)$

We require an additional assumption about $\tilde{\mathcal{L}}_{\text{ref}}(\cdot)$.

Assumption 6 (Distillation using reference dataset) *Assumptions 1, 2, 3, 4, and 5 regarding $\tilde{\mathcal{L}}_{km}(\cdot)$, $\forall k, m$, also hold for $\tilde{\mathcal{L}}_{ref}(\cdot)$.*

G.2. Proof of Theorem 12

For FedHEAD+, in addition to the local training and sector distillation phases, we have a server distillation phase where the model $\mathbf{w}^\tau$ is updated, through computing the gradients against server ensemble predictions until early stopping according to Definition 11. We define $\Delta_\tau = \mathbb{E}||\theta^\tau - \mathbf{w}^*||^2$. The expectation is taken over stochastic gradients as well as the choice of the client leader in each communication round for FedHEAD+.

Server distillation phase. Let us denote the stochastic gradient computed using the reference dataset at the τ-th iteration as $\tilde{\mathbf{g}}^{\tau}_{\text{ref}}(\xi) = \nabla\tilde{\mathcal{L}}_{\text{ref}}(\mathbf{w}^{\tau}, \xi)$ and the full gradient $\tilde{\mathbf{g}}^{\tau}_{\text{ref}} = \mathbb{E}\tilde{\mathbf{g}}^{\tau}_{\text{ref}}(\xi)$.

We first show a recurrence relationship concerning Δ_τ in the following lemma.

Lemma 14 *Suppose Assumption 6 holds. If $\eta_\tau \leq \frac{1}{4L}$, then in the server distillation phase of FedHEAD+, we have*

$$\mathbb{E}\left\|\theta^{\tau+1} - \mathbf{w}^*\right\|^2 \leq (1-\mu\eta_\tau)\mathbb{E}\left\|\theta^{\tau} - \mathbf{w}^*\right\|^2 + 2\eta_\tau^2 L\Gamma_{ref} + 2\eta_\tau^2 L\epsilon + \eta_\tau^2(1-\delta)\sigma^2. \tag{28}$$

Proof

$$\begin{aligned}
&\mathbb{E}\left\|\theta^{\tau+1} - \mathbf{w}^*\right\|^2 \\
&\leq \mathbb{E}\left\|\theta^{\tau} - \eta_\tau\tilde{\mathbf{g}}^{\tau}_{\text{ref}}(\xi) - \mathbf{w}^* - \eta_\tau\tilde{\mathbf{g}}^{\tau}_{\text{ref}} + \eta_\tau\tilde{\mathbf{g}}^{\tau}_{\text{ref}}\right\|^2 \\
&= \mathbb{E}\left\|\theta^{\tau} - \mathbf{w}^* - \eta_\tau\tilde{\mathbf{g}}^{\tau}_{\text{ref}}\right\|^2 - 2\eta_\tau\mathbb{E}\langle\theta^{\tau} - \mathbf{w}^* - \eta_\tau\tilde{\mathbf{g}}^{\tau}_{\text{ref}}, \tilde{\mathbf{g}}^{\tau}_{\text{ref}}(\xi) - \tilde{\mathbf{g}}^{\tau}_{\text{ref}}\rangle + \eta_\tau^2\mathbb{E}\left\|\tilde{\mathbf{g}}^{\tau}_{\text{ref}}(\xi) - \tilde{\mathbf{g}}^{\tau}_{\text{ref}}\right\|^2 \\
&\overset{(a)}{\leq} \mathbb{E}\left\|\theta^{\tau} - \mathbf{w}^* - \eta_\tau\tilde{\mathbf{g}}^{\tau}_{\text{ref}}\right\|^2 + \eta_\tau^2(1-\delta)\sigma^2 \\
&= \mathbb{E}\left\|\theta^{\tau} - \mathbf{w}^*\right\|^2 - 2\eta_\tau\mathbb{E}\langle\theta^{\tau} - \mathbf{w}^*, \tilde{\mathbf{g}}^{\tau}_{\text{ref}}\rangle + \eta_\tau^2\left\|\tilde{\mathbf{g}}^{\tau}_{\text{ref}}\right\|^2 + \eta_\tau^2(1-\delta)\sigma^2 \\
&\overset{(b)}{\leq} \mathbb{E}\left\|\theta^{\tau} - \mathbf{w}^*\right\|^2 - 2\eta_\tau\left(\tilde{\mathcal{L}}_{\text{ref}}(\theta^{\tau}) - \tilde{\mathcal{L}}_{\text{ref}}(\mathbf{w}^*) + \frac{\mu}{2}\mathbb{E}\left\|\theta^{\tau} - \mathbf{w}^*\right\|^2\right) + \eta_\tau^2\left\|\tilde{\mathbf{g}}^{\tau}_{\text{ref}}\right\|^2 + \eta_\tau^2(1-\delta)\sigma^2 \\
&\overset{(c)}{\leq} (1-\mu\eta_\tau)\mathbb{E}\left\|\theta^{\tau} - \mathbf{w}^*\right\|^2 - 2\eta_\tau\left(\tilde{\mathcal{L}}_{\text{ref}}(\theta^{\tau}) - \tilde{\mathcal{L}}_{\text{ref}}(\mathbf{w}^*)\right) + 2\eta_\tau^2 L\left(\tilde{\mathcal{L}}_{\text{ref}}(\theta^{\tau}) - \tilde{\mathcal{L}}^*_{\text{ref}}\right) + \eta_\tau^2(1-\delta)\sigma^2 \\
&= (1-\mu\eta_\tau)\mathbb{E}\left\|\theta^{\tau} - \mathbf{w}^*\right\|^2 - 2\eta_\tau(1-\eta_\tau L)\left(\tilde{\mathcal{L}}_{\text{ref}}(\theta^{\tau}) - \tilde{\mathcal{L}}_{\text{ref}}(\mathbf{w}^*)\right) + 2\eta_\tau^2 L\left(\tilde{\mathcal{L}}_{\text{ref}}(\mathbf{w}^*) - \tilde{\mathcal{L}}^*_{\text{ref}}\right) + \eta_\tau^2(1-\delta)\sigma^2 \\
&\overset{(d)}{\leq} (1-\mu\eta_\tau)\mathbb{E}\left\|\theta^{\tau} - \mathbf{w}^*\right\|^2 + 2\eta_\tau^2 L\left(\tilde{\mathcal{L}}_{\text{ref}}(\mathbf{w}^*) - \tilde{\mathcal{L}}^*_{\text{ref}}\right) + \eta_\tau^2(1-\delta)\sigma^2 \\
&\overset{(e)}{\leq} (1-\mu\eta_\tau)\mathbb{E}\left\|\theta^{\tau} - \mathbf{w}^*\right\|^2 + 2\eta_\tau^2 L\Gamma_{\text{ref}} + 2\eta_\tau^2 L\epsilon + \eta_\tau^2(1-\delta)\sigma^2
\end{aligned} \tag{29}$$

where $\tilde{\mathcal{L}}^*_{\text{ref}}$ is the minimum of the loss $\tilde{\mathcal{L}}_{\text{ref}}(\cdot)$, (a) is due to the bounded variance of stochastic gradients, (b) is due to strong convexity, (c) uses the Polyak-Lojasiewicz condition for L-smooth functions, (d) is due to early stopping criterion in Definition 11 and $\eta_\tau L \leq 1$, and (e) is obtained as follows:

$$\begin{aligned}
\tilde{\mathcal{L}_{\text{ref}}}(\mathbf{w}^*) - \tilde{\mathcal{L}}^*_{\text{ref}} &\leq \mathcal{L}_{\text{ref}}(\mathbf{w}^*) - \mathcal{L}^*_{\text{ref}} + |\tilde{\mathcal{L}}_{\text{ref}}(\mathbf{w}^*) - \mathcal{L}_{\text{ref}}(\mathbf{w}^*)| + |\mathcal{L}^*_{\text{ref}} - \tilde{\mathcal{L}}^*_{\text{ref}}| \\
&\overset{(a)}{\leq} \Gamma_{\text{ref}} + |\tilde{\mathcal{L}}_{\text{ref}}(\mathbf{w}^*) - \mathcal{L}_{\text{ref}}(\mathbf{w}^*)| + \max_{\mathbf{w}}|\mathcal{L}_{\text{ref}}(\mathbf{w}) - \tilde{\mathcal{L}}_{\text{ref}}(\mathbf{w})| \\
&\overset{(b)}{\leq} \Gamma_{\text{ref}} + \epsilon
\end{aligned} \tag{30}$$

where (a) is due to the fact $|\min_x f(x) - \min_x g(x)| \leq \max_x |f(x) - g(x)|$, and (b) is obtained from Assumption 5. ■

Overall bound on Δ_τ. Combining Lemma 14 with the recurrence relation for FedHEAD obtained in the proof of Theorem 2, we write the recurrence relation for FedHEAD+ as follows:

$$\Delta_{\tau+1} \leq \begin{cases} (1-\mu\eta_\tau)\Delta_\tau + \eta_\tau^2 B_l, \ \tau \in \text{local training phase}, \\ (1-\mu\eta_\tau)\Delta_\tau + \eta_\tau^2 B_s, \ \tau \in \text{sector distillation phase}, \\ (1-\mu\eta_\tau)\Delta_\tau + \eta_\tau^2 B_g, \ \tau \in \text{server distillation phase}, \end{cases} \tag{31}$$

where B_l, B_s, and B_g are defined in (7), (8) and (27), respectively. As done previously, letting $v = \max\left\{\frac{\beta^2 B_l}{\beta\mu-1}, \frac{\beta^2 B_s}{\beta\mu-1}, \frac{\beta^2 B_g}{\beta\mu-1}, (\gamma+1)\Delta_1\right\}$ and using induction, we obtain the statement of Theorem 12. Note that after t communication rounds, the total number of iterations is $(n_l + \bar{n}_s + \bar{n}_g)t$, where $\bar{n}_g \leq n_g$ is the average number of server distillation rounds per communication round before early stopping according to Definition 11.

G.3. Details of Remark 13

To compare against FedDF, we first note that FedDF is a degraded form of FedHEAD+ with the sector distillation phase removed. Therefore, we directly have the following theorem about the convergence of FedDF.

Theorem 15 *Suppose Assumptions 1, 2, 3, 4, 5, and 6 hold, and $\bar{n}_g \leq n_g$ is the average number of server distillation rounds per communication round before early stopping according to Definition 11. FedDF guarantees that after t communication rounds*

$$\mathbb{E}[\mathcal{L}(\mathbf{w}^{t+1})] - \mathcal{L}^* \leq \frac{\kappa}{(n_l + \bar{n}_g)t + \gamma}\left(\frac{2B_l + 2B_g}{\mu} + \frac{\mu\gamma}{2}\mathbb{E}||\mathbf{w}^1 - \mathbf{w}^*||^2\right) \tag{32}$$

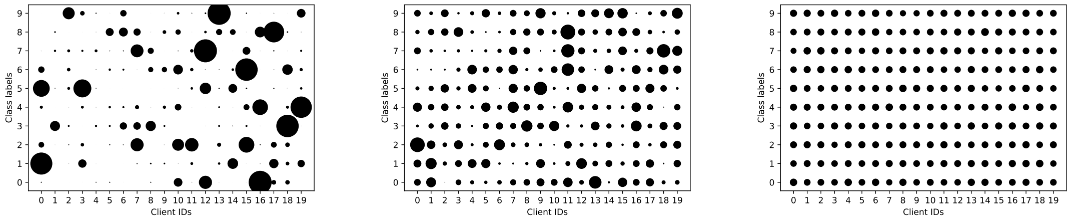
where $\kappa = \frac{L}{\mu}$, $\gamma = \max\{n_l, 8\kappa\}$, B_l and B_g are defined in (7) and (27), respectively, and $L, \mu, \sigma, G, \delta$ and ϵ are defined in Assumptions 1-5.

Now, for strongly convex loss functions, $\Delta_1 \leq \frac{4G^2}{\mu^2}$. Substituting this in (32) and (26) and setting $t = T_{\text{comm}}$, we obtain the convergence rate for FedDF and FedHEAD+ in terms of their dominant terms $O\left(\frac{B_l+B_s+B_g+\gamma G^2}{\mu(n_l+\bar{n}_s+\bar{n}_g)T_{\text{comm}}}\right)$ and $O\left(\frac{B_l+B_g+\gamma G^2}{\mu(n_l+\bar{n}_g)T_{\text{comm}}}\right)$, respectively. The convergence rate for FedHEAD+ is higher than FedDF if $\frac{B_l+B_s+B_g}{n_l+\bar{n}_s+\bar{n}_g} \leq \frac{B_l+B_g}{n_l+\bar{n}_g}$. By mediant inequality, this is true if $\frac{B_s}{\bar{n}_s} \leq \frac{B_l+B_g}{n_l+\bar{n}_g} \implies \bar{n}_s \geq n_l \frac{B_s\left(1+\frac{\bar{n}_g}{n_l}\right)}{B_l+B_g}$.

Furthermore, when $n_g = n_s = n_l$, if $B_s \leq \frac{B_l+B_g}{2}$, there exists a $\bar{n}_s$ such that $\bar{n}_s \leq n_s$ and $\bar{n}_s \geq n_l \frac{B_s\left(1+\frac{\bar{n}_g}{n_l}\right)}{B_l+B_g}$. As before, we can approximate $B_l \approx 6L\Gamma_{\text{client}} + 8n_l^2G^2$, $B_s \approx 6L\Gamma_{\text{sec}} + 6L\epsilon + 8(1-\delta)n_s^2G^2$, and $B_g \approx 2L\Gamma_{\text{ref}} + 2L\epsilon$. If $\Gamma_{\text{ref}} \geq 3\Gamma_{\text{client}}$ and $\delta \geq \frac{1}{2} + \frac{5L\epsilon}{8n_s^2G^2}$, we have $B_s \leq \frac{B_l+B_g}{2}$. Hence, we obtain Remark 13 comparing FedHEAD+ against FedDF.

A side benefit of obtaining Theorem 15 is that we can observe FedDF has a higher convergence rate in comparison with FedAvg in Theorem 10 if $\frac{B_l+B_g}{n_l+\bar{n}_g} \leq \frac{B_l}{n_l}$. By mediant inequality, this is true if $\frac{B_g}{\bar{n}_g} \leq \frac{B_l}{n_l} \implies \bar{n}_g \geq n_l\frac{B_g}{B_l}$.

Furthermore, when $n_g = n_l$, if $B_g \leq B_l$, there exists a $\bar{n}_g$ such that $\bar{n}_g \leq n_g$ and $\bar{n}_g \geq n_l\frac{B_g}{B_l}$. As before, we can approximate $B_l \approx 6L\Gamma_{\text{client}} + 8n_l^2G^2$, and $B_g \approx 2L\Gamma_{\text{ref}} + 2L\epsilon$. If $\Gamma_{\text{ref}} \leq 3\Gamma_{\text{client}}$ and $\frac{L\epsilon}{4n_l^2G^2} \leq 1$, we have $B_g \leq B_l$. Note that this conclusion was not proven but only inferred through numerical study in [14], since the generalization bound for FedDF derived in [14] has the same scaling as that of FedAvg. In contrast, our analysis here provides a definitive proof.



$(a)\ \alpha = 0.1$ $(b)\ \alpha = 1$ $(c)\ \alpha = 100$

Figure 2: Number of samples per class (proportional to dot size) distributed over the clients for different values of Dirichlet parameter, α.

Appendix H. Experimental Results

H.1. Datasets and Experimental Setup

We run experiments on SVHN [17], CIFAR10 [8] and CIFAR100 [8] datasets for computer vision (CV) tasks; AG News [28] and SST-2 [21] datasets for natural language processing (NLP) tasks; and Unicauca75 [19] and Unicauca141 [20] datasets for network traffic classification (NTC) tasks. Since the NTC datasets are huge, we draw a subset limiting the number of samples per class by 5000 and 2000 for Unicauca75 and Unicauca141, respectively.

As a default setting, we consider a network with a total of 20 clients randomly partitioned into two sectors. For simulating the data heterogeneity among clients, we follow the strategy of [11, 26] and create local datasets of clients using the Dirichlet distribution with parameter α. A smaller α indicates a more data heterogeneous setting whereas a larger α implies a lesser degree of data heterogeneity among the clients as illustrated in Fig. 2.

For computer vision tasks, we perform image classification using Resnet8 and Resnet32 [6]. For NLP tasks, we perform text classification using a bidirectional LSTM model consisting of an embedding layer followed by two bidirectional LSTM layers with 128 and 64 neurons, a global max pooling layer, five fully connected hidden layers with 1024, 512, 256, 128, 64 neurons, respectively, and finally, an output layer with softmax activation. For NTC tasks, we perform network traffic classification into application types using an MLP consisting of four fully connected hidden layers with 200 neurons each and an output layer with softmax activation. In all hidden layers, we apply RELU activation.

We compare FedHEAD and FedHEAD+ against four state-of-the-art FL baselines including FedAvg [16] and FedHKD [3], which do not require a reference dataset for distillation, and FedMD [10] and FedDF [14], which need a reference dataset. We do not compare against FedHKT [4] since as explained in Appendix A, FedHKT is not hierarchical and obtains unfair performance advantage over FedDF only by using a larger model at the server. Furthermore, we do not implement generator-based or prototype-based FL algorithms e.g., FedGen [30], FedProto [22], etc, as baselines because FedHKD has already been shown to outperform these methods [3].

For CIFAR10 and CIFAR100, we use 5000 images randomly sampled from CIFAR100 and Tiny Imagenet [9] respectively as reference datasets. For all other datasets, we use a subset of the training dataset with 5000 samples as the reference dataset. We report the test accuracy for all schemes to compare the performance of our proposed algorithms against the baselines.

Table 2: Test accuracy for CV tasks using Resnet8 with varying Dirichlet parameter, α.

Datasets	α	FedAvg	FedHKD	FedHEAD	FedMD	FedDF	FedHEAD+
SVHN	0.1	70.24±0.04%	66.01±0.04%	**79.93±0.04%**	69.84±0.04%	67.24±0.04%	78.31±0.04%
	1	86.33±0.03%	87.75±0.03%	**89.05±0.03%**	88.18±0.03%	84.79±0.03%	88.16±0.03%
CIFAR10	0.1	40.45±0.07%	40.01±0.07%	48.81±0.07%	45.32±0.07%	46.44±0.07%	**50.15±0.08%**
	1	55.78±0.08%	59.74±0.08%	60.45±0.06%	60.35±0.07%	60.58±0.07%	**61.03±0.07%**
CIFAR100	0.1	21.47±0.06%	22.72±0.06%	31.45±0.07%	23.74±0.05%	30.43±0.07%	**31.48±0.06%**
	1	28.47±0.06%	30.24±0.07%	**36.12±0.07%**	30.21±0.07%	34.46±0.07%	34.98±0.07%

Table 3: Test accuracy for NLP tasks with varying Dirichlet parameter, α.

Datasets	α	FedAvg	FedHKD	FedHEAD	FedMD	FedDF	FedHEAD+
AG News	0.1	86.38±0.04%	86.14±0.04%	86.83±0.05%	85.71±0.05%	85.43±0.05%	**87.02±0.04%**
	1	88.36±0.04%	88.53±0.04%	**88.89±0.05%**	87.59±0.05%	87.68±0.05%	88.66±0.05%
SST-2	0.1	87.71±0.05%	87.44±0.05%	**88.53±0.05%**	87.38±0.06%	87.45±0.06%	87.91±0.06%
	1	88.93±0.05%	88.72±0.05%	**89.60±0.05%**	87.29±0.06%	87.15±0.06%	88.84±0.06%

For local training, sector distillation, and server distillation, we use the Adam optimizer with a weight decay of 1×10^{-4} and a learning rate of 10^{-3}. Each client trains its model on the local dataset for 20 epochs. The number of distillation rounds at the sector and server is also set to 20. We use early stopping by monitoring validation loss and end distillation after the validation loss has plateaued for 5 consecutive epochs.

We build our models and run the experiments in Python 3.8 using Tesla P100 GPUs. The total number of communication rounds is set to 50. We use a batch size of 128 for both local training and distillation. In all experiments, we obtain a 95% confidence interval for inference by bootstrapping the test set.

H.2. Performance on CV Tasks

We present the performance of all algorithms on CV tasks in Table 2. For SVHN and $\alpha = 0.1$, we observe that FedAvg performs better than FedMD and FedDF. This implies that distillation using a reference dataset does not always improve performance. However, since the distillation in FedHEAD reuses the local datasets of the clients, FedHEAD outperforms FedAvg by achieving a test accuracy of 79.93%. Furthermore, FedHEAD+ falls short of FedHEAD, which confirms that distilling on the reference dataset has a negative impact in this case.

For CIFAR10, we see an opposite pattern. For $\alpha = 0.1$, FedAvg achieves an accuracy of 40.45% whereas the accuracy of FedMD and FedDF are 45.32% and 46.44%, respectively. FedHEAD outperforms all of them and achieves an accuracy of 48.81% without using a reference dataset. Furthermore, FedHEAD+ improves on top of FedHEAD achieving an accuracy of 50.15%. Finally, for CIFAR100, FedHEAD either achieves similar accuracy or outperforms FedHEAD+.

Table 4: Test accuracy for NTC tasks with varying Dirichlet parameter, α.

Datasets	α	FedAvg	FedHKD	FedHEAD	FedMD	FedDF	FedHEAD+
Unicauca75	0.1	63.51±0.04%	63.73±0.04%	64.01±0.03%	63.61±0.04%	63.92±0.03%	**64.79±0.04%**
	1	66.36±0.04%	66.42±0.04%	66.77±0.03%	66.51±0.04%	66.74±0.03%	**67.03±0.05%**
Unicauca141	0.1	77.42±0.04%	77.58±0.04%	77.86±0.04%	77.51±0.04%	77.61±0.03%	**78.30±0.04%**
	1	78.15±0.03%	78.34±0.03%	79.03±0.04%	78.16±0.03%	78.24±0.03%	**79.22±0.04%**

Table 5: Test accuracy for CIFAR10 using Resnet32 with varying number of clients, N, for $\alpha = 1$.

Schemes	$N = 20$	$N = 40$	$N = 60$
FedAvg	57.82±0.07%	54.55±0.07%	51.79±0.07%
FedHKD	60.91±0.07%	54.62±0.07%	52.23±0.07%
FedHEAD	62.30±0.06%	59.89±0.08%	58.10±0.07%
FedMD	62.19±0.07%	56.81±0.07%	53.98±0.07%
FedDF	62.43±0.07%	61.39±0.07%	58.81±0.07%
FedHEAD+	63.76±0.06%	62.53±0.07%	58.84±0.07%

H.3. Performance on NLP Tasks

Table 3 summarizes the performance of all algorithms on NLP tasks. For AG News and SST-2, FedHEAD and FedHEAD+ outperform all other methods for $\alpha = 0.1$ and 1. Take the performance on SST-2 with $\alpha = 1$. FedAvg and FedDF achieve an accuracy of 88.93% and 87.15%, respectively whereas FedHEAD and FedHEAD+ achieve an accuracy of 89.60% and 88.84%.

H.4. Performance on NTC Tasks

Table 4 shows the performance of all algorithms on NTC tasks. On Unicauca75 and Unicauca141 datasets, FedHEAD outperforms FedAvg and FedDF for both $\alpha = 0.1$ and 1. Furthermore, FedHEAD+ improves on top of FedHEAD. For example, in the case of Unicauca141 with $\alpha = 1$, FedAvg and FedDF achieve an accuracy of 78.15% and 78.24% where as FedHEAD achieves an accuracy of 79.03%. However, FedHEAD+ outperforms all of them by achieving an accuracy of 79.22%.

H.5. Impact of Number of Clients in the Network

In Table 5, we show the performance on CIFAR10 using Resnet32 varying the total number of clients, N, but keeping fixed the number of sectors, $M = 2$. We observe that generally the accuracy of all methods drops with an increasing number of clients in the network. Overall, FedHEAD outperforms FedAvg and FedHKD whereas FedHEAD+ outperforms FedMD and FedDF.

H.6. Impact of Number of Network Sectors

We investigate the impact of number of network sectors with varying number of clients in Table 6. For $\alpha = 0.1$ and $N = 20$, the client data distribution is very heterogeneous and the performance

Table 6: Test accuracy for CIFAR10 using Resnet8 with various numbers of clients, N, and number of sectors, M, for different values of α.

α	N	$M = 1$	$M = 2$	$M = 4$	$M = 8$
0.1	20	51.19±0.06%	48.81±0.07%	47.98±0.07%	45.12±0.07%
	40	54.72±0.07%	53.95±0.07%	50.04±0.07%	49.19±0.07%
	60	53.72±0.07%	52.94±0.07%	50.99±0.07%	49.79±0.07%
1	20	61.48±0.07%	60.45±0.06%	59.33±0.07%	59.29±0.07%
	40	58.53±0.07%	58.45±0.07%	58.61±0.07%	58.96±0.07%
	60	57.62±0.07%	57.82±0.08%	57.77±0.07%	57.83±0.07%

Table 7: Test accuracy for CV tasks using Resnet8 with fixed and randomized choices of client leaders for $\alpha = 0.1$.

Datasets	Fixed	Randomized
SVHN	77.53±0.04%	79.93±0.04%
CIFAR10	46.62 ±0.07%	48.81 ±0.07%
CIFAR100	29.76 ±0.07%	31.45 ±0.07%

deteriorates as the clients are segregated into more sectors. However, as the number of clients increases, the drop becomes less severe. For $\alpha = 1$, we observe that FedHEAD is robust towards the sectorization of the network and the performance does not drop much. Therefore, if the communication overhead is no issue, it is always better to partition the network into a fewer number of sectors, particularly in a highly non-iid setting. Furthermore, for $M = 1$, our system model reduces to the conventional non-hierarchical FL setup and FedHEAD achieves the best performance in this scenario. However, we remark that in this case the communication reduction benefit of hierarchical FL is lost.

H.7. Impact of Randomized Client Leader

Finally, we study the impact of choosing a random client as client leader at every communication round in FedHEAD as described in Algorithm 1. In Table 7, we show the performance on SVHN, CIFAR10, and CIFAR100 datasets by comparing the fixed and randomized choices of client leader. For the fixed choice, we designate the client with the largest dataset in each sector as the client leader throughout all communication rounds. We observe that our randomized choice of client leaders in FedHEAD results in better performance than the fixed one as it mitigates the skewness in client data distribution.