Unlocking Collective Intelligence in Decentralized AI

by

Gauri Gupta

Submitted to the Program in Media Arts and Sciences, School of Architecture and Planning, in partial fulfillment of the requirements for the degree of

Master of Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2024

AUTHOR
Gauri Gupta
Program in Media Arts and Sciences
May 17, 2024

CERTIFIED BY
Ramesh Raskar
Thesis Supervisor
Associate Professor of Media Arts and Sciences

ACCEPTED BY
Joseph A. Paradiso
Academic Head
Program in Media Arts and Sciences

Unlocking Collective Intelligence in Decentralized AI

by

Gauri Gupta

Submitted to the Program in Media Arts and Sciences,
on May 17, 2024, in partial fulfillment of the requirements for the degree of
Master of Science

Abstract

In the current evolving digital landscape, vast repositories of data and knowledge often remain siloed and untapped due to privacy concerns and centralized control. Thus, despite the transformative potential of artificial intelligence, its utilization in societal sectors lags behind other industries. For example in healthcare, data privacy and lack of incentives and trust in the system prevent collaboration on a large scale. This necessitates the development of efficient methods for decentralized learning while preserving privacy to generate wisdom whose quality is on par with the case of data centralization. It involves first identifying and creating essential building blocks that encourage collaboration while preserving the decentralized nature of these critical digital paradigms. A key challenge here is to facilitate collaboration among distrustful, disconnected, and disincentivized entities possessing distinct assets such as data, models, and computation resources. Harnessing the collective wisdom latent within decentralized networks will unlock new avenues for innovation and human collaboration. Therefore, the primary aim of this thesis is to expedite AI adoption in decentralized systems by introducing novel algorithms and systems capable of extracting collective intelligence while preserving privacy.

This thesis addresses the following research questions: First, it delves into methods for training machine learning models collaboratively while simultaneously protecting the privacy of raw data and the proprietary nature of individual models. Second, it explores the coordination mechanisms among system nodes in the absence of a central authority or trusted server to ensure orderly collaboration. Specifically, it answers questions like who should a node talk to. When does random collaboration selection work? Finally, it investigates strategies for conducting crowd-sourced decision-making to obtain population-level predictive results, scaling efficiently to encompass millions of agents.

Thesis Supervisor: Ramesh Raskar
Title: Professor of Media Arts and Sciences

Unlocking Collective Intelligence in Decentralized AI

by

Gauri Gupta

This thesis has been reviewed and approved by the following committee members:

THESIS SUPERVISOR
Ramesh Raskar
Associate Professor of Media Arts and Sciences
Massachusetts Institute of Technology

THESIS READER
Phillip Isola
Associate Professor of Electrical Engineering and Computer Science
Massachusetts Institute of Technology

THESIS READER
Gauri Joshi
Associate Professor of Electrical and Computer Engineering
Carnegie Mellon University

ACKNOWLEDGMENTS

First, I am extremely grateful to my advisor Prof. Ramesh Raskar whose constant guidance and support have been instrumental in shaping my personal and professional growth over the past two years. Under his mentorship, I have learned to tackle impactful challenges fearlessly, with a "think big, do big" attitude towards life. I am also thankful to my thesis readers: Professor Phillip Isola, for the profound impact he had on me through the Deep Learning course, which I enjoyed the most at MIT; and Professor Gauri Joshi, for their insightful conversations and dedicated time as thesis readers.

I am grateful to each Camera Culture lab member for creating a stimulating and friendly environment. Collaborating with such intelligent and talented individuals, including my close collaborators Abhishek Singh, Ayush Chopra, and Charlie Lu, has been an enriching experience. I have thoroughly enjoyed our discussions, brainstorming sessions, paper-deadline dinners, and conference trips, and I hope to continue our collaborations. Special thanks to my collaborators and UROPs: Ritvik Kapila, Jonas Blanc, Alex Dang, Yichuan Shi, and Joyce for their invaluable contributions and meaningful discussions. In addition, I am grateful for all the advice from Praneeth Vepakomma, Vivek Sharma, and Nikhil Naik. Lastly, I have enjoyed the camaraderie and light-hearted conversations with my Camera Culture folks, also including Kush Tiwary, Sid Somasundaram, Hank Lin, Tzofi Klinghoffer, Nikhil Behari, Aaron Young and Akshat Dave. The distinct culture of the MIT Media Lab has been instrumental in shaping my personal and professional growth. Through engaging in fall and spring meetings, presenting demos, organizing conferences, and delivering talks, I've gained valuable insights that have contributed significantly to my development. I am profoundly grateful to MIT for providing me with the platform and resources to pursue my intellectual curiosity.

Finally, this thesis would not have been possible without the unwavering support of my family — Vipan Gupta, Shweta Gupta, and my brother Kartik Gupta — who have fostered an environment of learning, resilience, and excellence. Frequent calls and visits with my close friends Ritvik Kapila, and Shreya Sharma provided me with much-needed stability and grounding during the challenging phases of research. I am also grateful to friends Dhwani Trivedi, Arijit Dasgupta, Pragati Modi, Gauri Agarwal, and many others for our fond memories made in the last two years in the greater Boston area.

PUBLICATIONS

The thesis consists of the culmination of the following works:

(1) “CoDream: Exchanging dreams instead of models for federated aggregation with heterogeneous models" by authors **Gauri Gupta***, Abhishek Singh*, Ritvik Kapila, Yichuan Shi, Alex Dang, Sheshank Shankar, Mohammed Ehab, and Ramesh Raskar [148]. I spearheaded the conceptualization and conducted an extensive experimental analysis of our proposed approach. I express deep appreciation to all my co-authors for their invaluable contributions and dedication.

(2) “When and how to make friends: Investigating strategies for collaborator selection in decentralized learning" by Jonas Blanc, Abhishek Singh, **Gauri Gupta**, Joyce Yuan, Martin Jaggi, and Ramesh Raskar [**?**]. My contributions include advising and brainstorming for the paper’s direction and actively participating in running and analyzing numerous experiments.

(3) “First 100 days of Pandemic; An Interplay of Pharmaceutical, Behavioral and Digital Interventions - A Study using Agent-Based Modeling" by **Gauri Gupta**, Ritvik Kapila, Ayush Chopra, and Ramesh Raskar. AAMAS 2024 **(Oral)** [57]. As the lead author, I conceived and executed the study performed, and thank my co-authors for their help in writing and meaningful discussions.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

A | INTRODUCTION

A.1 MOTIVATION

In today's interconnected world, data is increasingly siloed across diverse entities with varying resource constraints. The collaborative processing of such data holds promise for generating highly valuable insights. However, this collaborative potential faces substantial challenges, including strict privacy regulations, safeguarding trade secrets, computational constraints, communication bottlenecks, trust-related concerns, and competitive dynamics. Consequently, there is a need to develop methodologies for decentralized learning that can extract insights effectively and foster collaboration among distrustful, disconnected, and disincentivized entities. This thesis, adopting an interdisciplinary approach, systematically addresses these challenges, introducing methodologies to navigate the complexities inherent to decentralized learning.

Imagine a highly intelligent AI that can adeptly handle nuanced queries by even accessing data confined within silos. For instance, predicting one's COVID status based on daily activities like gym visits and interactions with specific friends requires tapping into the data that has been trapped in silos. This includes personal health records, mobility data stored in devices such as phones or smartwatches, or even the data at the nearest hospital. While this sounds fascinating, there are substantial challenges that need to be addressed.

Foremost among these challenges is data privacy, encompassing concerns related to sharing personal and sensitive information, adherence to data regulations, and safeguarding trade secrets. Secondly, trust-related queries emerge, questioning the concentration of data in a singular central entity and raising concerns about associated risks. Can we trust a single central entity with all our private data, risking the concentration of power in too few hands? Furthermore, the issue of ownership prompts reflection on who possesses access rights to these models.

The answer to all these questions lies in the decentralization of AI. Today, these models are built by centralizing all data in one place and then training models on top of it. However, we cannot centralize data for most real-world applications like healthcare, finance, location privacy, and many others. We need

a well-designed decentralized network model that can enable an ecosystem in which different data providers collaborate to train models. In this decentralized ecosystem, AI is owned by everyone and yet by no one, fostering democratic access to knowledge.

This thesis addresses the following research questions:

- How can we unlock data silos while collectively training machine learning models?
- If there is no central entity or truster server, how can the nodes in the system coordinate to collaborate without leading to chaos?
- Finally, how can we do crowdsourced prediction and get population-level prediction outcomes at the scale of millions of agents?

A.2 OUTLINE

A.2.1 Decentralization over Data using Collaborative Learning

The imperative for training machine learning models on decentralized data is driven by the need to unlock cross-silo and cross-device collaboration, as a significant portion of data held by individuals and organizations remains inaccessible due to prevailing privacy and regulatory constraints. The current collaborative learning paradigm, exemplified by Federated Learning (FL) [112], addresses this challenge by centrally aggregating clients' models rather than the data itself. However, to enhance the flexibility and utility of these systems, both algorithms, and infrastructures must adapt to accommodate the inherent heterogeneity in data, models, and computational resources.

Technical Challenge

1. Collaboration with heterogeneous: data and models: In decentralized learning, collaboration with heterogeneous resources poses a multifaceted challenge involving both heterogeneities in data and model architectures. Data with individuals may vary in terms of distribution, scale, and features, leading to challenges in aggregating information cohesively. Combining the collective knowledge from such diverse data to make accurate predictions is challenging. While plenty of FL techniques have been deployed in distributed learning, the intrinsic constraint of model homogeneity and handling non-iid data is

still a challenge. Recent studies [82, 101, 102, 80, 72, 86] have shown that FL encounters significant convergence issues when dealing with heterogeneous data (not independent and identically distributed, i.e. non-IID) resulting in slow training and suboptimal performance as compared to training on IID data. FL model homogeneity constraints have significantly curtailed the potential impact of Federated Learning in the current landscape by severely restricting participation from diverse entities. In a decentralized setup, each user is equipped with varying compute resources and bandwidth capabilities and thus should accommodate the diversity of resources rather than mandating uniform standards. This heterogeneity enhances inclusivity and adaptability across users but also creates interoperability roadblocks that complicate collaboration.

2. Confidential Models for Cross-Silo Collaboration: Conventional techniques require participants to share model weights. Consequently, in situations where organizations are hesitant to disclose their model weights due to confidentiality concerns, collaboration becomes impractical, even if they are willing to share valuable insights from their data. This deadlock highlights the urgent need for innovations that enable collaborative learning while safeguarding data and model confidentiality.

3. Scalability concerns: As the current era of model development is witnessing a rapid progression toward large-scale models with billions of parameters, the communication overhead incurred during the FL process becomes increasingly burdensome. The larger the model, the greater the volume of parameters that need to be transmitted during each communication round. This results in heightened network congestion, longer training times, and increased resource consumption. As communication scales linearly with model size, FL systems struggle to accommodate these large models, leading to performance degradation and system instability potentially leading to system collapse. Addressing these challenges requires innovative solutions that optimize communication protocols, reduce model complexity, and adapt FL algorithms to accommodate the constraints of large-scale models. Failure to mitigate these scalability concerns risks rendering FL impractical for deploying and training state-of-the-art models in real-world settings.

Related Work

FL [112] techniques have adequately addressed the diversity in data sources, but not enough attention has been paid to the heterogeneity of models and computational resources. This is largely due to a foundational assumption within Federated Learning, wherein the data owner and computation owner

are presumed to be the same entity. Split Learning [60] where every client only performs partial computation locally, and the remaining computation required to train deep networks is offloaded to a server that typically has a lot more computation resources. Some recent knowledge-distillation (KD) [118] techniques present an alternate paradigm that allows clients to share knowledge while allowing for heterogeneous models. However, these KD algorithms depart from the model averaging paradigm, making them incompatible with secure aggregation.

Approach

The thesis chapter C presents a unified framework **CoDream** to address the above challenges of accommodating diverse models for cross-device collaboration while simultaneously addressing the privacy and scalability concerns posed by model weight sharing in FL. Unlike conventional FL approaches that share model weights, CoDream is a collaborative learning technique that shares and aggregates "wisdom" across participants to gain collective knowledge while keeping models decentralized. This "wisdom" is updated as models learn from each other and improve. We define this "wisdom" in the representation space of models using "dreams" [148]. Clients collaboratively optimize randomly initialized data using federated optimization in the input data space, similar to how randomly initialized model parameters are optimized in FL. The key insight is that jointly optimizing this data can effectively capture the properties of the global data distribution. This paradigm shift offers a promising solution to the challenges posed by heterogeneous and confidential models in decentralized settings. Sharing knowledge in data space offers numerous benefits.

- It allows model-agnostic collaborative learning as clients gain the flexibility to employ varied model architectures based on their computational capacity. The approach enables seamless collaboration among entities with disparate resources and model capabilities.
- By sharing representations of data rather than model weights, we mitigate the risk of compromising the confidentiality of models. This model-agnostic representation-sharing approach allows organizations to collaborate without explicitly exposing their confidential model parameters. Further, the proposed approach is compatible with secure aggregation, thus preserving the privacy benefits of federated learning

- The Communication is independent of the size of the model parameters and remains constant even if the model increases in depth and width, alleviating scalability concerns.

A.2.2 Decentralization over Orchestration using Self-orchestration

Centralized learning of machine learning models requires minimal coordination but limits collaboration and puts too much faith in a single entity. Furthermore, as the system scales, centralization faces limitations regarding the available resources (communication, computation, etc.) within a single location. This prompts the system to shift towards embracing the decentralization of data and resources. In a fully decentralized scenario with no central server, this introduces a new set of challenges, as clients now require self-coordination. More attention is needed to address some of the key challenges in coordination aspects that become apparent in the absence of a centralized coordinator. Who should each user collaborate with? Should they collaborate with others at random or only with similar clients? In the context of coordinating interactions among diverse participating entities, as encountered in the realm of collaborative learning, a pressing concern is the development of techniques that facilitate the identification of suitable cohorts for collaboration.

For instance, consider a scenario where a hospital is engaged in the training of a machine-learning model for chest X-ray image analysis. It may be negatively impacted by collaborating with another hospital that is simultaneously training its model using MRI datasets. The key challenge here thus is identifying the most pertinent collaborators whose data distributions align closely with the objectives of the user. This problem is of significant practical relevance, particularly within the decentralized framework where the absence of a central coordinating server necessitates the autonomous identification of suitable collaboration cohorts.

Technical Challenge

In a fully decentralized setting, where no central server exists, strategic collaboration is crucial. The indiscriminate collaboration with every available entity is a far-from-ideal approach, as it can lead to very high communication costs. Further engaging in collaboration with every other individual for knowledge exchange may not be desirable, as relevant knowledge for one's objective could be diluted among numerous available resources and can potentially introduce detrimental effects on a user's model updates. Therefore, clients must selec-

tively choose their collaborators. The empirical findings in Chapter D highlight the benefits of collaborating with peers having similar data distributions, particularly in the presence of domain shifts between groups of clients. However, we observe that selecting the most similar clients based on their similarity can lead to feedback loops within a small set of clients, resulting in the emergence of isolated cliques in the collaboration graph, thereby reducing diversity in collaboration. Thus identifying the key collaborators is a challenging task.

In a system of diverse local users, identifying and forming communities comprising users with *similar* data distributions is challenging. This is due to the challenge of identifying effective criteria for clients to signal their dataset distribution is crucial for collaborative learning. While using raw data is not feasible due to privacy constraints, sharing models is also not an optimal solution due to architectural diversity and model weight permutations. Moreover, meta-data has limited utility in capturing dataset statistics. Consequently, an alternative approach that effectively navigates these challenges is required to facilitate the formation of relevant and effective collaboration cohorts.

Related Work

Decentralized FL works for peer-to-peer (P2P) communication between clients [66, 141]. Current systems mostly rely on random communication [76]; however, this greatly suffers from the heterogeneity in the clients' data distributions. In the context of non-IID data, some approaches assign *trust* or collaboration weights to other clients, either by learning them [99], measuring similarity on an unlabelled public dataset [48], or by clustering clients [152]

Numerous peer-to-peer collaborative learning methodologies [104, 19, 99] have previously delved into the intricate problem of collaborator discovery, predominantly through the lens of federated learning. This approach typically involves the exchange of model weights among clients, who subsequently rank other clients by computing statistical metrics based on these weights. While these techniques demonstrate promise in the context of collaborator discovery, there are certain limitations, particularly when it comes to scalability in a more decentralized setup.

Approach

This thesis in Chapter D delves into the intricacies of random collaboration, exploring when and why it suffices. While collaborating with similar users is beneficial, we discover a paradox in selecting users based on similarity. The

findings reveal a feedback loop between collaboration choices and outcomes, leading to "collaborator collapse." This phenomenon traps users in isolated cliques, resulting in suboptimal performances. We identify two types of cliques: "diversity collapse" (users from the same community) and "relevance collapse" (users from different communities). To address this challenge, we propose consensus-based collaborator selection strategies, harnessing users' opinions and evolving network dynamics. Extensive experiments demonstrate that consensus-based methods effectively prevent collaborator collapse and outperform random selection strategies. Changes in the *opinions* lead to the changes in *neighbors*, resulting in an adaptive network.

Furthermore, we tackle the challenge of what to share and advertise by leveraging "knowledge" within the data space, which serves as a potent and innovative approach for the identification of relevant collaborators. This notion of sharing "knowledge" within the data space distinguishes this approach from conventional Federated Learning (FL) approaches, where the exchange of models typically serves as a surrogate for knowledge about a client's private dataset. However, this thesis will explore how sharing "knowledge" within the data space, using dreams (Chapter C), represents a more fitting and effective approach for reasoning about data distribution and task-related considerations. This approach accommodates the inherent diversity of model architectures and remains unaffected by the permutations of model weights, making it a robust and adaptable solution. This thesis will develop methodologies that facilitate the generation of data-space samples without compromising the privacy of the underlying raw data.

A.2.3 Decentralization over Prediction using Large Population Models

Many of the grand societal challenges, such as pandemics, housing crises, or immunosenescence, may be characterized as emergent phenomena resulting from complex interactions within a large population of autonomous agents (eg: humans, cells). The emergent effects are sensitive to the scale of the input population, simulation parameters, and modeling assumptions.

Pandemics, notably the recent COVID-19 outbreak, have impacted both public health and the global economy. We need a profound understanding of disease progression and efficient response strategies to prepare for potential future outbreaks. For instance, we are interested in answering questions like are fast RTPCR tests better than slow but accurate POC tests? The answer is complex and challenging. What might be optimal for the individual citizen

may not be the same for the aggregate population. Population outcomes can be non-obvious due to complex individual interactions. For instance, it turns out that prioritizing speed over accuracy can result in a better population outcome.

As we move forward, there is a need to go from simply accounting for contacts to deriving actionable intelligence from contact data. By crowdsourcing such "contact intelligence" responsibly, we can enable predictive capabilities to inform critical policy decisions. Further, interfacing high-resolution autonomous agents with decentralized real agents can help aggregate individual insights and make reliable decisions for the collective. The potential of such large population models to accelerate scientific discovery and facilitate critical decision-making is huge.

Technical Challenge and Related Work

Using ABMs for practical decision-making requires recreating populations with great detail, calibrating to heterogeneous data sources, and assimilating granular real-world feedback. Computational and data access bottlenecks constrain this utility. Conventional ABMs slow [15, 22], are difficult to scale to large populations [22], tough to calibrate with real-world data [131], and prone to misspecification due to hand-crafted rules.

The utility of ABMs for practical decision-making depends upon several factors. These include their accuracy in replicating the population behavior [131, 55]. Furthermore, most prior work either studies the effect of only one intervention at a time or simulates very few agents [43]. Real-world deployment of intervention strategies intricately linked to each other should be scalable to large populations and need to be studied with a combined effect of each of these interventions [85, 38, 68]. However, decision-making in such scenarios is challenging due to the multifarious intricacies of complex societies characterized by heterogeneous populations, diverse behavioral patterns, and differential access to resources [9, 12, 33]. The interplay of various interventions, their mutual impacts, and the factors influencing their effectiveness adds further layers of complexity. Further, conventional ABMs rely on synthetic populations generated using sparse summary statistics from real-world observations. Privacy, not data sparsity, is the cause of limited granularity as data is siloed with individuals.

Approach

The thesis chapter E, introduces a general pipeline using ABMs that simulates a real-world synergy of interventions at scale, encompassing pharmaceutical, behavioral, and digital strategies [57]. We adopt a tensorized approach [38] which enables a fast, parallelized simulation, allowing analysis of emergent behavior on a large-scale population for millions of agents in a few seconds. This framework offers extensive detail to capture the complexities observed in the real-world adoption of these interventions. The model provides a comprehensive system that simulates interventions with real-world challenges of deployment or adoption, representing them through quantifiable parameters. Further, ethically crowdsourcing the data can help guide urgent decisions, as demonstrated by contact tracing applications during the pandemic. For example, our MIT-SafePaths protocol provided digital contact tracing to over a million people across 5 US states and territories [1].

B PRELIMINARIES

B.1 COLLABORATIVE LEARNING

The problem of collaborative data synthesis has been previously explored using federated learning, generative modeling, and knowledge distillation techniques. These approaches are illustrated in Fig. 1. Further, existing works can be categorized based on two characteristics: how knowledge is extracted from the teachers and how the students aggregate and acquire the teachers' knowledge. `CoDream` introduces the idea of collaboration in the representation space(data space precisely).

B.1.1 Federated Learning (FL)

FL aims to minimize the expected risk $\min_\theta \mathbb{E}_{\mathcal{D}\sim p(\mathcal{D})}\ell(\mathcal{D},\theta)$ where θ is the model parameters, $\mathcal{D}$ is a tuple of samples $(X \in \mathcal{X}, Y \in \mathcal{Y})$ of labeled data in supervised learning in the data space $\mathcal{X} \subset \mathbb{R}^d$ and $\mathcal{Y} \subset \mathbb{R}$, and ℓ is some risk function such as mean square error or cross-entropy [87, 114]. In the absence of access to the true distribution, FL aims to optimize the empirical risk instead given by:

$$\min_\theta \sum_{k\in K} \frac{1}{|\mathcal{D}_k|}\ell(\mathcal{D}_k,\theta), \tag{1}$$

$\mathcal{D}$ is assumed to be partitioned across K clients, where each client k owns each $\mathcal{D}_k$ and $\mathcal{D} = \cup_{k\in K}\mathcal{D}_k$. The optimization proceeds with the server broadcasting θ^r to each user k that locally optimizes $\theta_k^{r+1} = \arg\min_{\theta^r} \ell(\mathcal{D}_k,\theta^r)$ for M rounds and sends local updates either in the form of θ_k^{r+1} or $\theta_k^{r+1} - \theta_k^r$ (*pseudo-gradient*) to the server to aggregate local updates and send the aggregated weights back to the clients.

B.1.2 Collaborative Data Synthesis

Generative modeling techniques for proxy data synthesis either pool locally generated data on the server [150, 54] or use `FedAvg` with generative models [137,

164]. Like FL, `FedAvg` over generative models is also not model agnostic. Data Free techniques employ a generative model to generate synthetic samples as substitutes for the original data DENSE [170], Fedgen [177], and FedFTG [171] learn a generative model of data on the server. While not focused on knowledge distillation, dataset distillation techniques such as Fed-D3 [150] and DOS-FL [175] perform dataset distillation locally and share the distilled datasets with the server, where a global model is trained with one-shot communication. [77] and [20] aggregate the last layer's output on the private data across different samples for every class.

B.1.3 Knowledge Distillation (KD) for collaboration

Knowledge Distillation in FL is an alternative to `FedAvg` that aims to facilitate knowledge sharing among clients that cannot acquire this knowledge individually [32, 105, 4, 34].

Knowledge extraction: One of the key components of KD is to match the output distribution between the teacher and the student conditioned on the same input. Therefore, the same input is needed between the teacher and the student. However, FL techniques typically assume that the student and the teacher have different data siloes. This constraint has led to two general approaches to KD-based FL:

- **Proxy data-based** techniques assume publicly available datasets to obtain the output distribution among different users. Cronus [32], Federated model fusion [105], and FedBE [34]. Instead of relying on a publicly available dataset, [4] utilize the student's dataset to obtain the teacher's predictions by exchanging the teacher's model. A general drawback of these approaches is that the proxy dataset has to be a sufficiently rich representation of the original data distribution to enable knowledge sharing, which can be an overly constraining assumption for FL.

- **Data Free** techniques as discussed before employ a generative model to generate synthetic samples on the server.

Knowledge Acquisition: The other key component of KD is the alignment between the teacher's and the student's output distribution $\mathsf{KL}(p(f_{\theta_t}(x))||p(f_{\theta_s}(x)))$ is computed and minimized by optimizing the parameters of the student. Existing approaches can be broadly categorized into where this distribution alignment is performed.

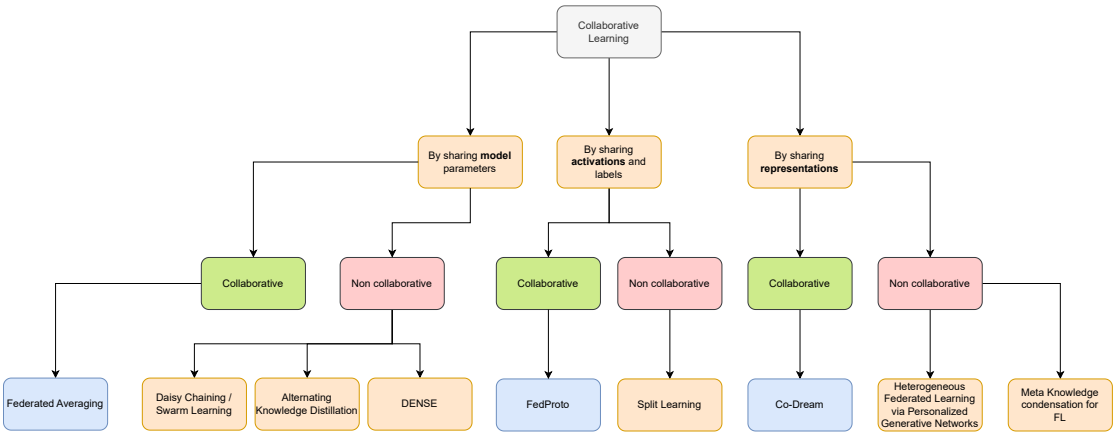


Figure 1: Different approaches to collaborative learning. By collaborative, we mean techniques that perform aggregation of *knowledge* before sharing it with the server and non-collaborative refers to techniques that directly share some information without aggregataion

- **Client model regularization** - The clients regularize their local model training by treating other clients as teachers. [94] regularizes on all labels with respect to the aggregated teacher model except the original label to avoid catastrophic forgetting. [4] perform distillation locally by using the teacher model on the client's data. [177] distribute the synthetic data generator learned by the server to the clients for regularization during training.

- **Ensemble distillation** techniques regularize the global server model by utilizing proxy or synthetic data. FedBE [34], Ensemble Distillation [105], FedAux [142], and FedFTG [171] aggregate soft labels from different clients.

However, all these existing approaches lack active client collaboration in the knowledge synthesis process. Clients share their local models or locally generated data with the server without contributing to knowledge synthesis. We believe that collaborative synthesis is crucial for secure aggregation and bridging the gap between KD and FL. Our approach `CoDream` enables clients to synthesize dreams collaboratively while remaining compatible with secure aggregation techniques and being model agnostic.

Independently of knowledge distillation, a few recent works have accommodated model heterogeneity by sub-model extraction [30, 42, 71, 6] and factorizing model weights with low-rank approximations [116]. Most of these techniques only support heterogeneity for a specific class of models, for example - the

ResNet family. In contrast, KD-based approaches are more flexible and only require the input and output dimensionality to be the same.

B.2 PEER-TO-PEER LEARNING (P2P)

B.2.1 Personalized FL

Personalixed FL [47, 78] assumes different clients represent different tasks and trains a global model in a way that can be efficiently fine-tuned on each client's local data. Another approach to personalize FL is by clustering clients with similar objectives using hierarchical clustering [143, 27], K-means, [74], or K-means++ [46]. In IFCA [53], clients select the model that gives the best performance on their local data among K global models. In [108], clustering is part of the objective and is optimized through Expectation Maximization. Except for hierarchical clustering, these techniques rely on a fixed number of clusters, which is impractical to assume in a decentralized setting.

B.2.2 P2P

P2P learning refers to a collaborative in a serverless scenario where individuals form cohorts or come together to share knowledge, skills, and experiences. In this model, participants take on both the roles of learners and teachers, creating a dynamic exchange of information within a group or community. P2P selection techniques relax the assumption of a centralized orchestrator but typically do not perform any selection. Collaboration happens either between every users [161, 172, 129], neighbors dictated by a fixed topology [91, 135, 81] or (pseudo)-random users [96, 141]. While gossip-based approaches [73, 154, 41] optimize collaboration for knowledge diffusion, they aim to train a single global model.

B.3 AGENT-BASED MODELLING

Agent-based models (ABMs) are discrete simulators that allow entities (agents) with designated characteristics to interact within a given computational environment, replicating complex systems [24, 123, 139, 174, 44, 67]. Agent-based models allow us to take a bottom-up view of a system through its components. Based on a decentralized source of individual mobility patterns, behavioral

patterns, and local clinical data, they allow the simulation of heterogeneous populations through complex time-varying actions and interactions and make crowdsourced policy decisions i.e. by harnessing the wisdom of the crowd. Recently, ABMs have been widely employed in epidemiology to understand disease progression and the efficacy of interventions by providing relevant information to investigate and predict the behavior of the pandemic [110, 140, 2, 8, 85]. Several studies have utilized ABMs to evaluate the effectiveness of different interventions, such as social distancing, quarantine, lockdown, and vaccination [68, 38, 140]. ABMs have also been used in prior works for addressing policy-related queries like evaluating the importance of test turnaround time versus its sensitivity [92], and the benefits of postponing the vaccine's second dose to focus on the distribution of the first dose [140].

C | COLLABORATIVE LEARNING OVER DECENTRALIZED DATA

C.1 INTRODUCTION

Overview		Resources			Flexibility / Utility		Security	
Approach	What is shared?	Comm.	Comp.	Memory	Heterogeneous models	Heterogeneous tasks	Compatible with Secure Agg.	Levels of Privacy
Fed. Learning	Predictive Model[1]	Baseline	Baseline	Baseline	No	Yes	Yes	1
Fed. Gen. Modeling	Generative Model[1]	High	High	High	No	Yes	Yes	1
Syn. Data Sharing	Synthetic Data[2]	Low	High	High	Yes	Yes	No	2
Data-Free KD	Predictive Model[1]	High	High	High	Yes	No	No	1
CoDream	Dreams[2]	Same	High	Same	Yes	Yes	Yes	2

Figure 2: Landscape of FL techniques. Here we use Fed.-Federated, Gen.-Generative, Syn.-Synthetic, Pred.-Predictive, Comm.-Communication, Comp.-Computation, Het.-Heterogeneous, Agg.-Aggregation. By levels of privacy, we mean how distant the shared updates are from raw data. Sharing synthetic data[2] and *dreams*[2] are two levels of indirection away from the raw data than sharing models[1].

In many application areas, such as healthcare and finance, data is distributed among silos owned by different organizations, making it difficult to train machine learning (ML) models on large datasets collaboratively. Centralizing data is not always feasible due to regulatory and privacy concerns. Federated Learning (FL) [114] addresses this problem by centrally aggregating clients' models instead of their data. Informally, FL circumvents privacy concerns in two steps: 1) sharing client's models instead of data offers confidentiality as data does not leave the trusted local device, and 2) the aggregation step in FL is a linear operation (weighted average), which makes it compatible with secure aggregation techniques. The efficacy of the first layer of privacy becomes pronounced when the number of samples per client is substantial, while the significance of the second layer becomes apparent in ecosystems with numerous clients.

FL assumes that all clients agree on the same model architecture and are willing to share their local models. Due to resource constraints, however, this can potentially reduce the number of clients in the ecosystem, eliminating the benefit of the second layer. Some recent knowledge-distillation (KD) [118] techniques present an alternate paradigm that allows clients to share knowledge

while allowing heterogeneous models. However, these KD algorithms depart from the model averaging paradigm, making them incompatible with secure aggregation. Hence, they also can not derive privacy benefits from the second layer.

Alternatively, if we could generate samples representing data distribution characteristics while maintaining privacy benefits at both layers, we would eliminate the need to aggregate the client models. Sharing samples offers much higher flexibility for training models and supports arbitrary model architectures and tasks. However, the problem is challenging because collaboratively learning a generative model leads to the same problems as FL for predictive models. Instead of learning a generative model, we solve this dilemma by optimizing data collaboratively instead of parameters.

We design a novel framework for collaboratively synthesizing a proxy of siloed data distributions without centralizing data or client models. These collaboratively synthesized representations of data, which we call *dreams*, can be used to train ML models. We show that dreams capture the knowledge embedded within local models and also facilitate the aggregation of local knowledge without ever sharing the raw data or models. Our key idea is to begin with randomly initialized samples and apply federated optimization on these samples to extract knowledge from the client's local models trained on their original dataset. Unlike synthetic data, the goal of optimizing *dreams* is to enable KD, rather than generate realistic data (maximize likelihood of data).

We design our framework into three stages: knowledge extraction C.4.1, knowledge aggregation C.4.2, and knowledge acquisition C.4.3. We perform extensive investigation to test `CoDream` by (1) establishing the feasibility of `CoDream` as a way for clients to synthesize samples collaboratively, (2) showing the utility of synthesized samples by learning predictive models (3) validating `CoDream` as an alternative to FL and (4) performing empirical validation of our framework by benchmarking with existing algorithms and ablation studies across various design choices.

The key factors of our approach are: (1) **Flexibility**: Our proposed technique, `CoDream`, collaboratively optimizes *dreams* to aggregate knowledge from the client's local models. By sharing *dreams* in the data space rather than model parameters, our method is model-agnostic. (2) **Scalability**: Furthermore, communication does not depend on the model parameter size, alleviating scalability concerns. (3) **Privacy**: Just like `FedAvg` [113], `CoDream` also exhibits two-fold privacy: Firstly, clients share *dreams'* updates instead of raw data. Secondly, the linearity of the aggregation algorithm allows clients to securely aggregate their

dreams without revealing their individual updates to the server. In summary, our contributions are as follows:

- A novel framework `CoDream`, for collaborative data synthesis through federated optimization in input space, serving as a proxy for the global data distribution.
- Our approach introduces a novel perspective to FL by aggregating "knowledge" instead of local model parameters. This unique aggregation framework leads to model-agnostic learning and addresses scalability concerns while preserving privacy through compatibility with secure aggregation.
- Extensive empirical validation of `CoDream`, including benchmarking against existing algorithms and ablation studies across various design choices, further emphasizes its potential for collaborative optimization and adaptability for personalized learning.

C.2 BACKGROUND

Knowledge Distillation facilitates the transfer of knowledge from a teacher model ($f(\theta_T)$) to a student model ($f(\theta_S)$) by incorporating an additional regularization term into the student's training objective [29, 69]. This regularization term (usually computed with Kullback-Leibler (KL) divergence $\mathsf{KL}(f(\theta_T, \mathcal{D})||f(\theta_S, \mathcal{D}))$) encourages the student's output distribution to match the teacher's outputs.

DeepDream for Knowledge Extraction [119] first showed that features learned in deep learning models could be *extracted* using gradient-based optimization in the feature space. Randomly initialized features are optimized to identify patterns that maximize a given activation layer. Regularization such as TV-norm and ℓ_1-norm has been shown to improve the quality of the resulting images. Starting with a randomly initialized input $\hat{x} \sim \mathcal{N}(0,\ I)$, label y, and pre-trained model f_θ, the optimization objective is

$$\min_{\hat{x}} \mathsf{CE}\left(f_\theta(\hat{x}),\ y\right)\ +\ \mathcal{R}(\hat{x}), \tag{2}$$

where CE is cross-entropy and $\mathcal{R}$ is some regularization. DeepInversion [168] showed that the knowledge distillation could be further improved by matching batch normalization statistics with the training data at every layer.

C.3 RELATED WORK

The problem of collaborative data synthesis has been previously explored using generative modeling and federated learning techniques. Figure 2 compares existing decentralization solutions regarding shared resources, utility, and privacy.

Generative modeling techniques either pool locally generated data on the server [150, 54] or use `FedAvg` with generative models [137, 164]. Like FL, `FedAvg` over generative models is also not model agnostic. While we share the idea of generative data modeling, we do not expose individual clients' updates or models directly to the server.

Knowledge Distillation in FL is an alternative to `FedAvg` that aims to facilitate knowledge sharing among clients that cannot acquire this knowledge individually [32, 105, 4, 34]. However, applying KD in FL is challenging because the student and teacher models need to access the same data, which is difficult in FL settings.

Data-free Knowledge Distillation algorithms address this challenge by employing a generative model to generate synthetic samples as substitutes for the original data [170, 171, 177]. These data-free KD approaches are not amenable to secure aggregation and must use the same architecture for the generative model.

However, all these existing approaches lack active client collaboration in the knowledge synthesis process. Clients share their local models or locally generated data with the server without contributing to knowledge synthesis. We believe that collaborative synthesis is crucial for secure aggregation and bridging the gap between KD and FL. Our approach `CoDream` enables clients to synthesize dreams collaboratively while remaining compatible with secure aggregation techniques and being model agnostic.

C.4 CODREAM

Our approach `CoDream` consists of three key stages: **knowledge extraction**, **knowledge aggregation** and **knowledge acquisition**. In the knowledge extraction stage, each client extracts useful data representations, referred to as "dreams", from their locally trained models (teachers). Starting with random

[1] Aggregation of local updates occurs in model parameter space
[2] Aggregation of local updates occurs in the data space

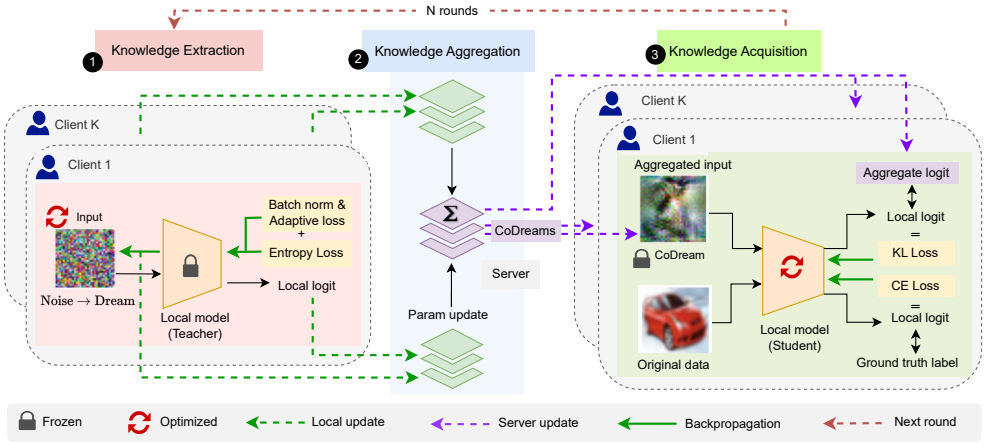


Figure 3: Overview of the `CoDream` pipeline comprising three stages: (1) Knowledge Extraction— each client generates *dreams*, representing the extracted knowledge from their local models (teacher). Starting with random noise images and frozen teacher models, clients optimize to reduce entropy on the output distribution while regularizing the batch norm and adaptive loss. The clients share their local updates of *dreams* and logits with the server. (2) Knowledge Aggregation—server aggregates *dreams* and soft labels from clients to construct a `CoDream` dataset. (3) Knowledge Acquisition—clients update their local models through two-stage training (i) on jointly optimized *co-dreams* with knowledge distillation (where clients act as students) and (ii) local dataset with cross-entropy loss.

noise images and fixed teacher models, clients optimize these images to facilitate knowledge sharing from their local models (Section C.4.1). Since this is a gradient-based optimization of the input *dreams*, we exploit the linearity of gradients to enable knowledge aggregation from all the clients. In the knowledge aggregation stage, the clients now jointly optimize these random noised images by aggregating the gradients from the local optimizations (Section C.4.2). Unlike traditional federated averaging (`FedAvg`), our aggregation occurs in the input data space over these *dreams*, making our approach compatible with heterogeneous client architectures. Finally, in the knowledge acquisition step, these collaboratively optimized images, or *dreams*, are then used for updating the server and clients without ever sharing the raw data or models. This is done by performing knowledge distillation on the global *dreams* where clients now act as students (Section C.4.3). Figure 18 gives an overview of the `CoDream` pipeline for each round. We further discuss these stages in more detail in the following subsections.

C.4.1 Local dreaming for extracting knowledge from models

In this stage, clients perform local *dreaming*, a model-inversion approach to extract useful information from the locally trained models. We use DeepDream [119] and DeepInversion [168] approaches that enable data-free knowledge extraction from the pre-trained models. However, these are not directly applicable to a federated setting because the client models are continuously evolving, as they learn from their own data as well as other clients. A given client should synthesize only those *dreams* over which they are highly confident. As the client models evolve, their confidence in model predictions also changes over time. A direct consequence of this non-stationarity is that it is unclear how the label y should be chosen in Eq 2. In DeepInversion, the teacher uniformly samples y from its own label distribution because the teacher has the full dataset. However, in the federated setting, data is distributed across multiple clients with heterogeneous data distributions.

To keep track of a given client's confidence, we take a simple approach of treating the entropy of the output distribution as a proxy for the teachers' confidence. We adjust Eq 2 so that the teacher synthesizes *dreams* without any classification loss by instead minimizing the entropy (denoted by $\mathcal{H}$) on the output distribution. Each client (teacher) starts with a batch of representations

sampled from a standard Gaussian ($\hat{x} = \mathcal{N}(0,1)$), and optimizes *dreams* using Eq 3. Formally, we optimize the following objective for synthesizing *dreams*:

$$\min_{\hat{x}} \left\{ \tilde{\ell}(\hat{x},\ \theta) \ =\ \mathcal{H}\left(f_\theta(\hat{x})\right) + \mathcal{R}_{bn}(\hat{x}) + \mathcal{R}_{adv}(\hat{x}) \right\} \tag{3}$$

where $\mathcal{H}$ is the entropy for the output predictions, $\mathcal{R}_{bn}$ is the feature regularization loss and $\mathcal{R}_{adv}$ is a student-teacher adversarial loss. $\mathcal{R}_{adv}$ helps extract knowledge from the clients that the clients know and the server does not know.

Batch-norm regularisation

To improve the *dreams* image quality, we enforce feature similarities at all levels by minimizing the distance between the feature map statistics for *dreams* and training distribution, which is stored in the batch normalization layers. Hence

$$\mathcal{R}_{bn}(\hat{x}) = \sum_l ||\mu^l_{feat} - \mu^l_{bn}|| + ||\sigma^l_{feat} - \sigma^l_{bn}|| \tag{4}$$

Adaptive teaching

In passive knowledge transfer, the student does not influence how the teacher extracts knowledge. However, by personalizing the teaching to what a student does not know and what the teacher does know, we can improve the student's performance and speed up learning. Due to gradient-based optimization, the teacher can synthesize examples that maximize the student's loss. The student, in turn, can optimize its model weights to minimize its loss. Additionally, by personalizing to the student, the teacher avoids extracting redundant knowledge. In our framework, we introduce adaptive teaching at two levels.

Intuitively, maximizing the differences between the students' and teachers' generation for a given teacher helps generate representations of what the teacher knows and what the student does not. We apply this adaptive teaching technique in `CoDream`. Under this setup, the clients operate as adaptive teachers for the server and minimize their loss while maximizing the loss with respect to the server. The server's knowledge can be viewed as the culmination of all the clients' knowledge compressed into a single model. The idea of flipping the gradients for learning what you don't know is interesting.

Thus to increase the diversity in generated *dreams*, we add an adversarial loss to encourage the synthesized images to cause student-teacher disagreement.

$\mathcal{R}_{adv}$ penalizes similarities in image generation based on the Jensen-Shannon divergence between the teacher and student distribution,

$$\mathcal{R}_{adv}(\hat{x}) = -JSD(f_t(\hat{x})||f_s(\hat{x})) \tag{5}$$

where the client model is the teacher and the server model is the student model. To do this adaptive teaching in a federated setting, the server shares the gradient $\nabla_{\hat{x}} f_s(\hat{x})$ with the clients for local adaptive extraction. The clients then locally calculate $\nabla_{\hat{x}} \tilde{\ell}(\hat{x},\ \theta_k)$ which is then aggregated at the server for knowledge aggregation in Eq 6.

Note that generative models create synthetic data with objectives to resemble the real data and align with the input distribution by maximizing the likelihood of the data. Unlike synthetic data, the only goal of optimizing *dreams* is to enable efficient knowledge distillation. Therefore, *dreams* do not need to appear like real images. We also visualize *dreams* and compare them against real images in Figure 7.

c.4.2 Collaborative dreaming for knowledge aggregation

Since the data is siloed and lies across multiple clients, we want to extract the collective knowledge from the distributed system. While `FedAvg` aggregates gradients of the model updates from clients, it assumes the same model architecture across clients and thus is not model-agnostic.

We propose a novel mechanism for aggregating the knowledge by collaboratively optimizing *dreams* across different clients. Instead of each client independently synthesizing *dreams* using Eq 3, they now collaboratively optimize them by taking the expectation over each client's local loss w.r.t. the same $\hat{x}$: $\min_{\hat{x}} \mathbb{E}_{k \in K} \left[\tilde{\ell}(\hat{x},\ \theta_k)\right]$ This empirical risk can be minimized by computing the local loss at each client. Therefore, the update rule for $\hat{x}$ can be written as:

$$\hat{x} \leftarrow \hat{x} - \nabla_{\hat{x}} \sum_{k \in K} \frac{1}{|\mathcal{D}_k|} \tilde{\ell}(\hat{x},\ \theta_k)$$

Using the linearity of gradients, we can write it as

$$\hat{x} \leftarrow \hat{x} - \sum_{k \in K} \frac{1}{|\mathcal{D}_k|} \nabla_{\hat{x}} \tilde{\ell}(\hat{x},\ \theta_k) \tag{6}$$

The clients compute gradients locally with respect to the same input and share them with the server, which aggregates the gradients and returns the updated input to the clients. This formulation is the same as the distributed-SGD formulation, but the optimization is performed in the data space instead of the model parameter space. Thus, unlike `FedAvg`, our approach `CoDream` is model-agnostic and allows clients with heterogenous model architecture as shown in Fig 4. Our framework is also compatible with existing cryptographic aggregation techniques, as the aggregation step is linear and only reveals the final aggregated output without exposing individual client gradients.

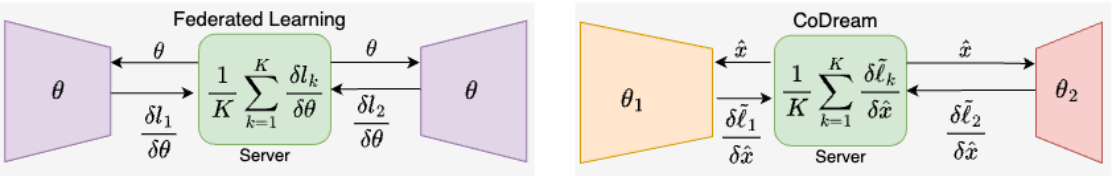


Figure 4: Comparing aggregation framework in FL and `CoDream`. In FL, the server aggregates the gradients of model parameters, whereas, in `CoDream`, aggregation happens in the gradients of the data space, called dreams ($\hat{x}$), allowing for different model architectures. Here K is the number of clients and $l, \tilde{l}$ are loss functions given in Eq 1 and Eq 3.

Collaboratively optimizing representations, known as *dreams* in our approach, is a novel concept that has not been explored before. Our experiments in Section C.6.9 demonstrate that dreams obtained through this approach capture knowledge from all clients and outperform dreams independently synthesized dreams by clients.

C.4.3 Knowledge acquisition

Finally, the extracted knowledge, in the form of collaboratively trained *dreams*, is then acquired by the local client and server models to update their models with the global information. They learn to match the distribution of the ensemble of clients on the distilled samples (denoted by $\hat{\mathcal{D}}$), which are obtained by Eq 6. The clients share soft logits for each *dream*, which are then aggregated by the server to perform knowledge distillation (where clients and server now act as students) on the following training objective:

$$\min_{\theta} \sum_{\hat{x}\in\hat{\mathcal{D}}} \mathrm{KL}\left(\sum_{k} \frac{1}{|\mathcal{D}_k|} f_{\theta_k}(\hat{x}) \,\middle\|\, f_{\theta}(\hat{x})\right) \tag{7}$$

We provide the complete algorithm of `CoDream` in Algorithm 1. Note that the choice of parameters such as local updates M, global updates R, local learning rate η_l, global rate η_g, and the number of clients K typically guide the trade-off between communication efficiency and convergence of the optimization.

Input: Number of client K, local models and data θ_k and $\mathcal{D}_k, k \in K$, local learning rate η_k, global learning rate η_g, local training rounds M, global training epochs R, total number of epochs N.

for $t = 1$ **to** N **do**
  Server initializes a batch of dreams as $\hat{x} \sim \mathcal{N}(0, 1)$;
  for $r = 1$ **to** R **do**
    Server broadcasts current dream $\hat{x}^r$ to all clients
    for *each client* $k \in K$ ***in parallel*** **do**
      $\hat{x}^r_{k,0} := \hat{x}^r$;
      for $m = 1$ **to** M **do**
        // Local knowledge extraction stage (Eq 3)
        $\hat{x}^r_{k,m} \leftarrow \hat{x}^r_{k,m-1} - \eta_k \cdot \nabla_x(\tilde{\ell}(\hat{x}^r_{k,m-1}, \theta_k))$;
      end
      each client shares pseudo-gradient $\nabla \hat{x}^r_k = \hat{x}^r_{k,M} - \hat{x}^r$ with the server;
    end
    // Collaborative knowledge aggregation stage (Eq 6)
    $\hat{x}^{r+1}_S \leftarrow \hat{x}^r + \eta_g \sum_{k \in K} \frac{1}{|\mathcal{D}_k|} \nabla \hat{x}^r_k$;
    // Server aggregates model predictions to get $\hat{\mathcal{D}}$
    $\hat{\mathcal{D}} := \{\hat{x}^{r+1}, \hat{y}^{r+1}_S := \sum_k \frac{1}{|\mathcal{D}_k|} f_{\theta_k}(\{\hat{x}^{r+1}\})\}$;
    // Local knowledge acquisition stage (Eq 7)
    for *each client* $k \in K$ ***in parallel*** **do**
      LocalUpdate($\hat{\mathcal{D}}, \theta_k$); LocalUpdate($\mathcal{D}_k, \theta_k$);
    end
    LocalUpdate($\hat{\mathcal{D}}, \theta_s$);
  end
end

Algorithm 1: `CoDream` Algorithm

Method	MNIST			SVHN			CIFAR10		
	iid($\alpha = \inf$)	$\alpha = 1$	$\alpha = 0.1$	iid($\alpha = \inf$)	$\alpha = 1$	$\alpha = 0.1$	iid($\alpha = \inf$)	$\alpha = 1$	$\alpha = 0.1$
Centralized	85.0$_{(0.9)}$	61.4$_{(7.1)}$	36.9$_{(7.6)}$	80.8$_{(1.3)}$	75.6$_{(1.4)}$	54.6$_{(13.6)}$	65.7$_{(2.9)}$	65.3$_{(0.4)}$	45.5$_{(6.8)}$
Independent	52.4$_{(7.0)}$	36.3$_{(6.2)}$	22.0$_{(4.2)}$	51.3$_{(9.2)}$	42.3$_{(6.4)}$	19.6$_{(9.2)}$	46.4$_{(2.0)}$	39.7$_{(3.4)}$	23.5$_{(5.2)}$
FedAvg	84.7$_{(1.6)}$	60.3$_{(3.4)}$	40.0$_{(6.9)}$	82.9$_{(0.4)}$	79.1$_{(0.9)}$	47.1$_{(23.7)}$	67.2$_{(0.4)}$	62.3$_{(0.9)}$	34.8$_{(8.3)}$
FedProx	78.6$_{(3.5)}$	62.6$_{(3.6)}$	38.1$_{(11.0)}$	**86.9**$_{(0.1)}$	84.3$_{(0.6)}$	**48.7**$_{(26.7)}$	**70.8**$_{(1.8)}$	62.3$_{(2.9)}$	27.1$_{(9.8)}$
Moon	85.1$_{(2.6)}$	66.2$_{(4.4)}$	**42.3**$_{(11.8)}$	80.1$_{(0.1)}$	76.5$_{(1.2)}$	41.7$_{(21.8)}$	66.6$_{(1.4)}$	64.8$_{(0.8)}$	35.5$_{(10.8)}$
AvgKD	61.3$_{(2.3)}$	44.3$_{(4.8)}$	21.4$_{(4.3)}$	75.4$_{(0.7)}$	61.2$_{(4.6)}$	20.7$_{(10.9)}$	54.2$_{(0.9)}$	46.4$_{(3.3)}$	25.9$_{(6.2)}$
SCAFFOLD	**87.5**$_{(0.6)}$	**70.2**$_{(3.6)}$	38.8$_{(13.7)}$	86.0$_{(0.1)}$	**84.5**$_{(0.7)}$	13.5$_{(4.4)}$	73.9$_{(1.5)}$	**67.5**$_{(4.6)}$	22.8$_{(7.8)}$
FedGen	64.5$_{(1.9)}$	51.0$_{(4.3)}$	31.4$_{(7.4)}$	49.7$_{(1.6)}$	44.2$_{(4.1)}$	34.9$_{(19.7)}$	66.2$_{(0.4)}$	62.8$_{(1.8)}$	**40.2**$_{(9.0)}$
CoDream (ours)	80.6$_{(0.5)}$	57.7$_{(3.6)}$	35.7$_{(9.2)}$	81.4$_{(0.1)}$	80.1$_{(0.8)}$	44.5$_{(17.7)}$	69.5$_{(0.3)}$	64.8$_{(0.3)}$	36.6$_{(8.4)}$

Table 1: Performance overview of different techniques with different data settings. A smaller α indicates higher heterogeneity.

C.5 ANALYSIS OF codream

The benefits of CoDream are inherited from using KD, along with additional advantages arising from our specific optimization technique. CoDream extracts the knowledge from clients in *dreams* and shares the updates of these dreams instead of model gradients (∇_θ) as done in FL.

Communication Analysis: We use the following notation: d is the dimension of the inputs or *dreams*, n is the batch size of *dreams* generated, and R is the number of aggregation rounds. Since CoDream communicates input gradients ($\nabla_{\hat{x}}$) instead of model gradients (∇_θ), the total communication is $d \times n\ R$. In FedAvg and its variants, the communication is $|\theta| \times R$. Unlike in FedAvg, the communication of CoDream is independent of the size of the model parameters $|\theta|$ and remains constant even if the model increases in depth and width. Thus, the communication complexity of CoDream does not scale with larger models. For heavily parameterized models, $d \times n \ll |\theta|$. Table 3 provides a comprehensive communication analysis for different model architectures in FedAvg vs CoDream.

Privacy Analysis: Exchange of models between the server and clients can result in potential privacy leakage. Various model inversion and reconstruction attacks [61, 70] have been shown to leak private sensitive information by reconstructing the training data. However, in CoDream, the clients collaborate by sharing the gradients of *dreams'* without even sharing their model parameters. A simple application of data processing inequality shows that dreams obtained from a model provably have lower information about raw data than the model. Further, we visually analyze generated *dreams* in Figure 7. While *dreams* enable knowledge-distillation, they do not resemble real data. Similar to FedAvg, the synchronization step between the clients is a linear operation (weighted aver-

Model	Heterogeneous Clients (Independent clients 1-4)				Method			
	WRN-16-1	VGG-11	WRN-40-1	ResNet-34	Independent	Centralized	AvgKD	CoDream (ours)
iid($\alpha = \inf$)	52.2	55.1	43.5	54.2	$51.6_{(4.5)}$	68.8	$52.9_{(1.4)}$	$69.6_{(1.0)}$
$\alpha = 1$	41.3	38.2	37.1	50.1	$41.7_{(5.1)}$	64.8	$42.4_{(2.9)}$	$60.0_{(1.7)}$
$\alpha = 0.1$	29.1	22.3	33.1	21.5	$27.2_{(4.9)}$	43.0	$30.2_{(3.3)}$	$40.6_{(0.9)}$

Table 2: **Performance comparison with heterogeneous client models**: on CIFAR10 dataset. Left: Accuracy for independent heterogeneous clients with different models; Right: Average client model performance comparison of CoDream with other baselines

age) and hence offers an additional layer of privacy by using secure and robust aggregation [25].

Flexibility of models: Since the knowledge aggregation in CoDream is done by sharing the updates of *dreams* in data space, CoDream is model agnostic and allows for collaboration among clients with different model architectures. We empirically observe no performance drop in collaborative learning with clients of different model architectures.

Customization in sharing knowledge: Additionally, sharing knowledge in the data space enables adaptive optimization, such as synthesizing adversarially robust samples or class-conditional samples for personalized learning. For more details, refer to the Appendix.

C.6 EXPERIMENTS

We conduct a rigorous empirical examination of CoDream, highlighting its capacity to adapt across various model architectures, thereby enhancing the system's flexibility. We perform a rigorous empirical analysis of CoDream and show its ability to be agnostic to model architecture and thus be more flexible. We show the performance analysis of using CoDream in heterogeneous model systems and communication efficiency analysis as compared with FL. Additionally, we conduct ablations to gain a comprehensive understanding of each aspect of CoDream.

Unless stated otherwise, we used ResNet-18 [64] for training the client and server models and set the total number of clients $K = 4$. We conduct our experiments on 3 real-world datasets, including MNIST [93], SVHN [121], and CIFAR10 [90]. To validate the effect of collaboration, we train clients with 50 samples per client for MNIST and 1000 samples per client for CIFAR10 and SVHN datasets. For reference, we compare CoDream to Independent and Centralized training baseline. In the Centralized baseline, all the client data

are aggregated in a single place. In the case of `Independent`, we train models only on the client's local dataset and report average client local accuracy.

To simulate real-world conditions, we perform experiments on both IID and non-IID data. We use Dirichlet distribution $Dir(\alpha)$ to generate non-IID data partition among labels for a fixed number of total samples at each client. The parameter α guides the degree of imbalance in the training data distribution. A small α generates skewed data.

c.6.1 Fast dreaming for knowledge extraction

Despite the impressive results of the original DreamInversion [168], it is found to be extremely slow with 2000 local iterations for a single batch of image generation. The collaborative nature of the knowledge extraction process in `CoDream` makes it further slow. To accelerate this process of generating *dreams*, the Fast-datafree [49] approach learns common features using a meta-generator for initializing *dreams*, instead of initializing with random noise every time. This approach achieves a speedup factor of 10 to 100 while preserving the performance. Thus, to speed up our collaborative process of generating *dreams*, we implement `CoDream-fast` by integrating the Fast-datafree [49] approach on top of our algorithm. However, in each aggregation round, the client now shares both the local generator model and the dreams for secure aggregation by the server. Instead of 2000 global aggregation rounds (R) in `CoDream`, `CoDream-fast` performs only a single global aggregation round with 5 local rounds. We perform all the subsequent experiments using `CoDream-fast`.

In `CoDream-fast`, clients collaboratively train a generator model to learn a good initialization for the meta-features. However, we perform 5 local rounds with only one global aggregation round. We empirically observe that the optimizers' choice for the dreams and the networks was crucial to obtaining good representations. We perform 2 types of training to update the client's local models using SGD with a learning rate of 0.2 and momentum of 0.9. (1) The client models are trained on the local data using cross-entropy loss. (2) To update the models on the global knowledge, each client trains their models on the global dream dataset using knowledge distillation loss.

c.6.2 Flexibility of models: Model-agnostic

Since `CoDream` shares updates in the data space instead of the model space, our approach is model agnostic. We evaluate our approach across heterogeneous

client models having ResNet-34 [65], VGG-11 [147], and Wide-ResNets [169] (WRN-16-1 and WRN-40-1). Table 2 shows the performance of `CoDream` against `Centralized`, `Independent`, and model agnostic FL baselines such as Avg-KD. Note that FedGen is not completely model agnostic as it requires the client models to have a shared feature extractor and thus cannot be applied to our setting. We exclude `FedAvg` as it doesn't support heterogeneous models. Performing FL under both heterogeneous models and non-IID data distribution is a challenging task. Even under this setting, our approach performs better than the baselines.

C.6.3 Communication efficiency

We compare the client communication cost of `CoDream` and `FedAvg` per round for different model architectures in Table 3. In `FedAvg`, the clients share the model with the server, whereas, in `CoDream`, they share the *dreams*(size of data). However, in `CoDream`, each batch of *dreams* is refined for 400 rounds, whereas in `CoDream-fast` there is only a single round of aggregation along with the sharing of a lightweight generator model (as explained in Section C.6.1. The communication of both `CoDream` and `CoDream-fast` is model agnostic and does not scale with large models.

Model	FedAvg	`CoDream`	`CoDream-fast`
Resent34	166.6 MB	600 MB	23.5MB
Resnet18	89.4 MB	600 MB	23.5MB
VGG-11	1013.6 MB	600 MB	23.5MB
WRN-16-1	1.4 MB	600 MB	23.5MB
WRN-40-1	4.5 MB	600 MB	23.5MB

Table 3: Communication analysis of `FedAvg` vs `CoDream` and `CoDream-fast` per round

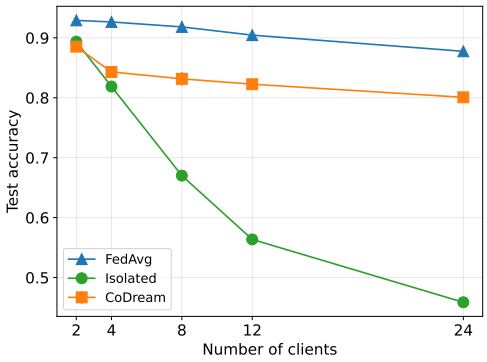
C.6.4 Varying number of clients

A key goal of `CoDream` is to aggregate knowledge from many decentralized clients. We evaluate this by varying the number of clients $K = [2, 4, 8, 12, 24]$, while keeping the total data samples constant. Thus, as K increases, each client contributes fewer local samples.

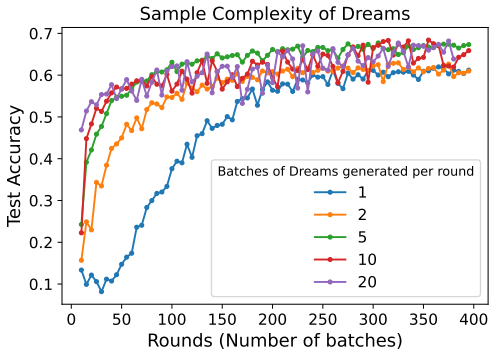
As expected, performance declines with more clients, since each client's knowledge is less representative of the overall distribution. However, Figure 5a

shows this drop is sublinear, making CoDream viable for cross-device federated learning. The gap between CoDream and FedAvg remains similar across different K.

In summary, CoDream sees a graceful decline in accuracy as data gets more decentralized. The framework effectively distills collective knowledge, even when local datasets are small. This scalability demonstrates CoDream's suitability for privacy-preserving collaborative learning from many heterogeneous client devices.



(a) Comparison by varying the number of clients. The performance gap widens between CoDream and independent optimization as we increase the number of clients.



(b) Sample complexity of generated dreams for effective knowledge transfer

c.6.5 Real-world datasets/comparison with FL

We evaluate our method under both IID and non-IID settings by varying $\alpha = 0.1, 0.5$ and report the performances of different methods in Table 1. We compare CoDream against FedAvg, FedProx [101], Moon[98], and Scaffold [82]. We also include other model-agnostic federated baselines such as FedGen[177], which uses a generator model to generate a proxy for locally sensitive data, and AvgKD [3], which alternately shares models with other clients to get averaged model soft predictions across two clients. We extend the AvgKD method for an n-client setting. The results show that our approach CoDream achieves high accuracy(close to centralized) across all datasets and data partitions. Even as α becomes smaller (i.e., data become more imbalanced), CoDream still performs well. Note that CoDream does not beat other state-of-the-art non-iid techniques since it is not designed for the non-iid data challenges. It is analogous to FedAvg

in the data space, and thus, all non-iid tricks can also be applied to `CoDream` to improve its accuracy further.

c.6.6 Analysis of sample complexity of dreams

We plot the accuracy of the server model optimized from scratch against the number of batches of *dreams* it is trained on as shown in Fig 5b. Note that the quality of generated dreams for training increases as training progresses in each round. We also vary the number of batches generated in each round and find 5 batches per round to be an optimum number, after which the marginal gain is very small.

c.6.7 Analysis of sample complexity of dreams

We plot the accuracy of the server model optimized from scratch against the number of batches of *dreams* it is trained on as shown in Fig 5b. Note that the quality of generated dreams for training increases as training progresses in each round. We also vary the number of batches generated in each round and find 5 batches per round to be an optimum number, after which the marginal gain is very small.

c.6.8 Validating knowledge-extraction based on Eq 3

We evaluate whether the knowledge-extraction approach (Sec C.4.1) allows for the effective transfer of knowledge from teacher to student. We first train a teacher model from scratch on different datasets, synthesize samples with our knowledge-extraction approach, and then train a student on the extracted *dreams*. To validate its compatibility within an FL setting where clients have a small local dataset, we reduce the size of the training set of the teacher to reduce its local accuracy and evaluate how this affects student performance. Results in Fig 6 show that the teacher-student performance gap does not degrade consistently even when the teacher's accuracy is low. This result is interesting because the extracted features get worse in quality as we decrease the teacher accuracy, but the performance gap is unaffected.

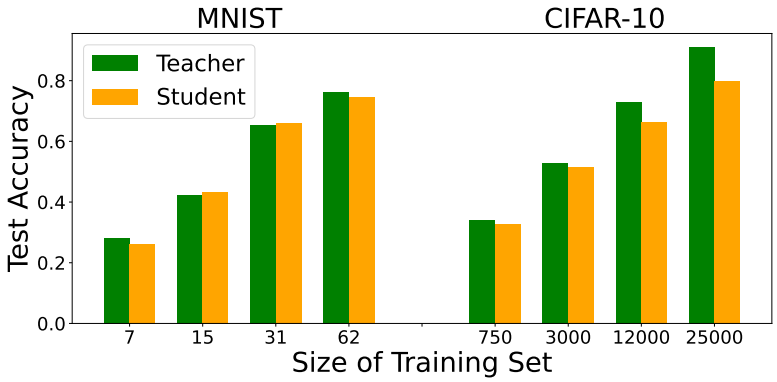


Figure 6: Validating the effectiveness of knowledge transfer from teacher to student: We vary the size of the training dataset (on the x-axis) for the teacher and compare its accuracy with the student trained on dreams generated using Eq 3

c.6.9 Validating collaborative optimization based on Eq 6

We also evaluate the effectiveness of collaborative optimization of *dreams* over multiple clients in aggregating the knowledge. To do this, we compare the performance of collaboratively optimized *dreams* in `CoDream` (using Eq 3) with independently optimized *dreams*. As we can see in table 4 (last row), the aggregation step in Eq 3 not only helps in secure averaging, leading to more privacy but also improves the performance.

Data partition	iid	$\alpha = 1$	$\alpha = 0.1$
`CoDream`	$69.2_{(0.1)}$	$61.6_{(0.5)}$	$45.6_{(1.5)}$
w/o $\mathcal{R}_{adv}$	$65.7_{(0.2)}$	$58.4_{(1.3)}$	$42.0_{(1.4)}$
w/o $\mathcal{R}_{bn}$	$51.2_{(6.1)}$	$33.1_{(7.1)}$	$24.1_{(5.2)}$
w/o collab	$64.4_{(0.5)}$	$58.4_{(1.4)}$	$30.8_{(3.2)}$

Table 4: Ablation of components in `CoDream` on CIFAR10

c.6.10 Contribution of loss components $\mathcal{R}_{bn}$ and $\mathcal{R}_{adv}$ in knowledge extraction

We further explore the impacts of various components of loss function in data generation in Eq. 3. Through leave-one-out testing, we present results by excluding $\mathcal{R}_{bn}$ (w/o $\mathcal{R}_{bn}$) and excluding $\mathcal{R}_{adv}$ (w/o $\mathcal{R}_{adv}$). Table 4 shows removing either component influences the accuracy of the overall model, illustrating the

impact of each part of the loss function plays an important role in generating good quality *dreams*.

C.6.11 Visual representation of dreams

Figure 7 visualizes the *dreams* generated by `CoDream-fast` on CIFAR10. While not visually similar to the original training data, these *dreams* effectively encapsulate collaborative knowledge. The goal is to enable decentralized knowledge transfer, not reconstructing the raw data. Thus, models trained on *dreams* perform well despite their visual differences from the underlying distribution.

Figure 7: Visualization of *dreams* generated on CIFAR10 dataset

C.7 LIMITATIONS AND FUTURE WORK

Despite its promising potential, `CoDream` has some limitations, especially additional computation on client devices. While the number of parameters on the client device remains unchanged, the client device has an additional computation burden. To circumvent this challenge, we implement `CoDream-fast` which uses a meta-generator that learns good initialization for *dreams* instead of random initialization. Further research and optimizations may be needed to address this limitation. Another promising future avenue is new privacy mechanisms catered for `CoDream` that improve the privacy-utility trade-off.

C.8 CONCLUSION

In this paper, we introduce `CoDream`, a collaborative data synthesis approach where clients jointly optimize synthetic *dreams* in a privacy-preserving manner.

It is a model-agnostic learning framework that leverages a knowledge extraction algorithm by performing gradient descent in the input space. We view this approach as a complementary technique to FedAvg, which performs gradient descent over model parameters. Through comprehensive evaluations and ablation studies, we validate the effectiveness of our proposed method.

C.9 OVERALL IMPACT

The proposed CoDream framework significantly advances the landscape of federated learning by introducing key technical innovations with far-reaching implications. Its model-agnostic approach allows clients with diverse architectures to collaboratively optimize data representations, overcoming the need for consensus on model structure. This not only broadens the scope of federated learning but also caters to resource-constrained clients, potentially fostering increased participation in federated ecosystems. The scalability of CoDream, with communication independent of model parameters, addresses concerns related to the size of machine learning models, opening avenues for the deployment of federated learning in scenarios involving large models.

CoDream holds potential across sectors such as healthcare and finance, where data is often decentralized among different entities. By facilitating collaborative data synthesis without centralizing raw information, CoDream supports the development of robust and accurate machine learning models. CoDream's privacy-preserving features, including the two-fold privacy protection and compatibility with secure aggregation, ensure responsible and privacy-aware practices in the context of federated learning. Moreover, by enabling the synthesis of data without direct data sharing, CoDream addresses the concerns related to data ownership and privacy infringement, which are increasingly critical in the era of data-driven technologies. However, CoDream does not fix several issues inherent to collaborative learning such as client dropout, stragglers, formal privacy guarantees, bias, fairness, etc. We believe further research is warranted to explore the effectiveness of CoDream under those constraints.

D COLLABORATOR SELECTION FOR DECENTRALIZED ORCHESTRATION

D.1 INTRODUCTION

Despite the vast amounts of data on the web, individuals can effectively discover content tailored to their specific interests. This capability has been facilitated by services such as hyperlinks, search engines, and recommendation systems, contributing to the web's exponential growth and significant impact. How can we extend these ideas to collaborative machine learning (ML)? We focus on enabling users with similar tasks, constrained by limited data and computational resources, to exchange knowledge in a decentralized ecosystem rich with data and computation. A key enabler for such ecosystems is developing robust communication protocols that allow users to dynamically identify relevant collaborators. Consequently, we investigate the critical challenge of collaborator selection in fully decentralized networks without any central coordination.

The key objective of collaborator selection is to identify a subset of peers for each participant, such that collaborating with these chosen peers enhances the performance of their individual models beyond what they could achieve through isolated training alone. We study this problem in the context of federated averaging (FedAvg) [112] collaboration where each user with a small local dataset averages its model weights with other users to exchange *knowledge* with the *collective*. We systematically compare different strategies for collaborator selection and study their dynamics and evolution of interactions over multiple rounds.

Our experiments in Section D.3 show that when a large enough model is trained by a small number of users over multiple collaboration rounds, collaborator selection is not required because randomly choosing anyone suffices. This observation aligns with the emerging trend of foundation models [23] where training on a big dataset, regardless of distribution, on a sufficiently large model yields a superior performance. However, decentralized systems are characterized by *large number of limited resources* phenomenon, i.e., lots of users and data diversity while limited samples, model size, and communication rounds. In these situations, carefully selecting collaborators results in improved performance over randomly choosing collaborators. In other words, as the

system becomes more decentralized, strategic collaborator selection becomes increasingly important.

In a system of diverse local users, the task of identifying and forming communities comprising users with *similar* data distributions is challenging. This difficulty arises from the lack of knowledge about individual data domains, as users can not disclose their raw data. In the absence of raw data, existing works have used the users' models as a proxy for the similarity of their data distribution by either taking the parametric [152, 104] or functional [126, 104, 58, 48] approaches.

While collaborating with similar users is beneficial, our research highlights a conundrum associated with selecting users based on similarity. Specifically, when a user consistently collaborates with a particular user deemed similar, it fosters an increased similarity with the same user, creating undesirable feedback loops. Consequently, the likelihood of encountering the same user who is considered similar in subsequent rounds also increases. We term this phenomenon as *collaborator collapse* because users get trapped into isolated cliques and do not collaborate outside, leading to suboptimal performances. We identify two kinds of cliques in *collaborator collapse*: 1) diversity collapse: collapsing of users within the same community, leading to lack of diversity between collaborators, and 2) relevance collapse: collapsing with users from different communities leading to sub-optimal training due to lack of relevance between collaborators.

Finally, we revisit the problem of collaborator selection as a consensus problem. We assume each user has an initial *opinion* about their state based on its raw data and model. The idea of consensus is to change user opinions based on the *opinion* of its *neighbors*. First, we show collaborator selection based only on users' opinions, which is not the best strategy. Then we show that consensus-based collaborator selection strategies can mitigate both modes of collapse and perform better than random selection. Change in the *opinion* results in the change of *neighbors*, resulting in an adaptive network.

Our contributions can be summarized as follows:

- **Investigation** of when randomly selecting collaborators is sufficient and when it hampers the performance.
- **Identification** of feedback loop between the model collaboration and the communication topology marked by the emergence *collaborator collapse*.
- **Mitigation** of the feedback loop by introducing *consensus* mechanisms.

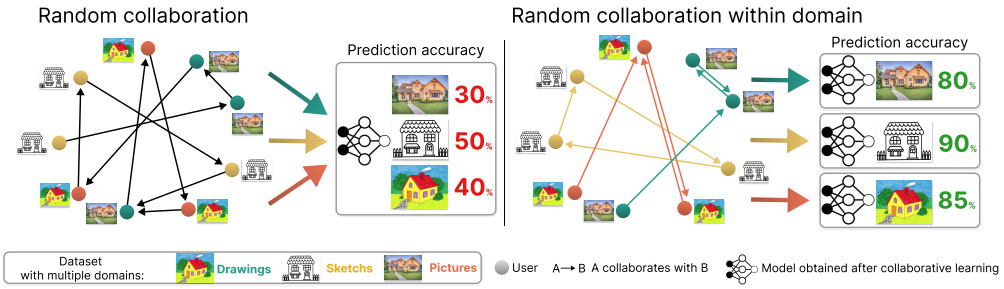


Figure 8: Random Collaboration among (left) all Users versus (right) Users belonging to the Same Domain. Each user has data that belongs to one of three domains from DomainNet: *quickdraw*, *infograph*, or *real*. We find it beneficial for a user to identify and collaborate with users belonging to the same domain.

D.2 RELATED WORK

Federated Learning (FL) [115] enables collaborative training of ML models over decentralized data by a central server that aggregates the ML models from the clients. However, including all clients in the process might hinder global convergence. Therefore, collaborator selection might be done by the server to improve robustness to byzantine [31], efficiency in a resources-constrained environment [122], or better convergence rates [37]. Unlike most work in FL, we primarily focus on collaborator selection in the absence of a centralized orchestrator.

More recently, collaborator selection has gained some attention in the P2P settings. (Meta-)L2C [100] learns (a model encoder) collaboration weights to optimize performance on a local validation set. In [48], collaboration weights are assigned based on the similarity of prediction between users on a publicly visible unlabeled dataset. These techniques make strong assumptions, such as heavy collaboration in the initial rounds or dependency on a validation set.

PAMN [104] uses Monte Carlo to discover potential collaborators before augmenting the candidates list with EM-GMM. To distinguish between users within the same domain and others, they assume the similarities of users within and outside of the domain to follow the Gaussian distribution. In this work, we are not making any assumptions about the distribution of similarities. In FedeRiCo [152] uses EM to find latent variables describing the users' distribution. In PENS [126], users select the best-performing users on their own local data as collaborators.

D.3 EMPIRICAL ANALYSIS

We study two modes of collaborator selection as shown in Fig 8 and how collaborator selection impacts the collaborative learning process. First, we evaluate the performance of random *within-domain* collaboration – when users sample collaborators uniformly at random among users sharing the same domain. We compare it to *unrestricted* random collaboration – random sampling collaborators among all the participants (hereafter referred to as *random collaboration*), for multiple number of collaboration rounds and different users and domains topologies to answer the following question: **Why and when does random collaboration within-domain make sense?** Our analysis in D.3.2 demonstrates that limiting collaboration to users having the same domain improves performance. We further investigate the dynamics of collaborator selection to answer the following question: **How to discover and collaborate with users within the same domain?** Our findings in D.3.3 highlight the emergence of a feedback loop between the choices and the results of collaboration when the similarity between users dictates collaboration choices. When users act greedily, this leads to a lack of diversity in collaboration and ultimately hurts performance. Finally, in D.3.3, we motivate clustering as a solution to ensure diversity of collaboration by considering collaborator selection as a collaborative task rather than an individual one.

D.3.1 Experimental Setup

We consider three multi-domains datasets: *DomainNet* [132], *Camelyon17* [16] and *Digit-Five* [103] using the *ResNet-10* model architecture. We evaluate the performance of every client on their respective domain-specific complete test set. Similar to online learning [144], we are not only interested in final accuracy but also in achieving the best accuracy at each round. Similar to online learning, we evaluate the strategies based on the mean Area Under the Curve (AUC) of users' test accuracy.

We assume peer-to-peer (P2P) collaboration between users, where any two users can communicate directly. However, users are limited to C collaborators per round. At each round, users train their model locally, select C collaborators, fetch their model, and average them with their own. Note that collaboration is not symmetric. The goal for every user is to train a model that is performing the best on their underlying domain distribution. In our experiments, we intentionally set $C = 1$ across all scenarios. This deliberate choice aligns

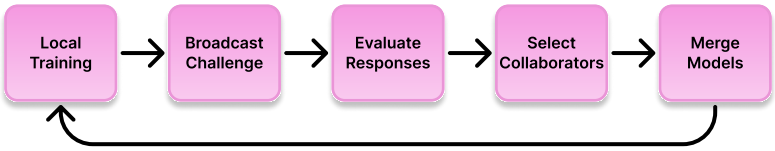


Figure 9: **Protocol for Decentralized Collaborative Learning.** In each round, users begin with local training on their data. Subsequently, each user broadcasts a small set of challenge samples and responds with its model's predictions on the received challenges. Each user then evaluates the responses from other users and selects collaborators for the given round based on their model performance on the challenges. Finally, users merge their model parameters with the selected collaborators' model parameters through a weighted averaging process. These steps are then repeated in each iteration.

with practical considerations, where collaboration is expected to occur amidst millions of potential users with only a fraction of the participants.

Strategies based on similarity to select collaborator rely on the protocol illustrated in 9. More specifically, at each round, after local training, users broadcast a set of challenges. Upon receiving other participants' challenges, users respond with their model predictions on the challenges (soft labels). Subsequently, other users' similarity is inferred from their predictions.

In our experiment, we use similarity based on the functional view of the model. We use raw data as representations exchanged between users and true labels as the clients' predictions to compute similarity. Note this can be implemented without any raw data exchange as it is equivalent to every user collecting other users' models to compute locally the performance of their local data. While this method of computing similarity isn't practical, it serves as a simplification to study the collaborator selection problem. We argue synthetic data [79] or dreams [125, 148] would be a practical approach to representations-based similarity in real-world scenarios, refer to D.4 for a comprehensive discussion.

Baselines and Strategies: We implement several baselines for reference and compare different strategies: *centralized*, all the data is sent to a single node that centrally trains a single model; In *centralized per domain* centralization happens within a domain and a single model is trained per domain; In *isolated*, no collaboration takes place between users, and each user trains models only on its local dataset; In *random*, users select collaborators uniformly at random from the entire user pool; and Conversely, in *random within domain*) where we assume users know their domain and choose collaborators randomly from within their

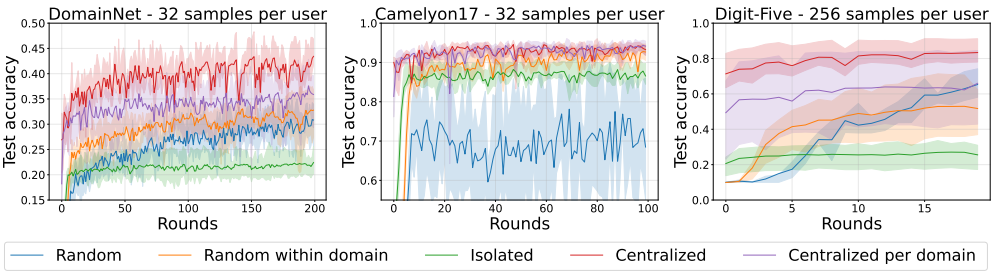


Figure 10: **Accelerated Convergence via Within-Domain Collaboration**. We compare random within-domain collaborator selection strategy with multiple selection baselines described in D.3.1. The within-domain collaboration demonstrates faster convergence compared to random collaboration, particularly in the initial rounds. The duration of this phase, however, depends on the compatibility of the domains. The figure illustrates outcomes from collaborative learning tasks among 12 clients partitioned across 3 domains.

respective domains, promoting collaboration among users with similar data distributions.

D.3.2 When Does Collaborator Selection Matter?

A prevalent approach in decentralized ML is randomly selecting collaborators among all participants [73, 154, 157, 162, 41]. However, as we consider practical considerations such as scalability and heterogeneity, random collaboration is expected to reduce in effectiveness as drift among the models would increase. Indeed, we empirically (see random collaboration in 10) observe that selecting users at random is not optimal and might even be worse than not collaborating. We systematically study when random selection performs better or worse than collaborating within a domain.

1. Number of Users

In decentralized systems, scalability is paramount, as the system should be able to support a large number of users. However, as the number of users increases, the challenge of achieving parameter consensus among models becomes more pronounced. As random collaboration aims at reaching a consensus among the whole pool of users, the convergence rate is expected to decrease as the number of users increases. Limiting collaboration to users within the same domain limits the available knowledge but has the potential to enhance the convergence rate by significantly reducing the number of collaborating peers.

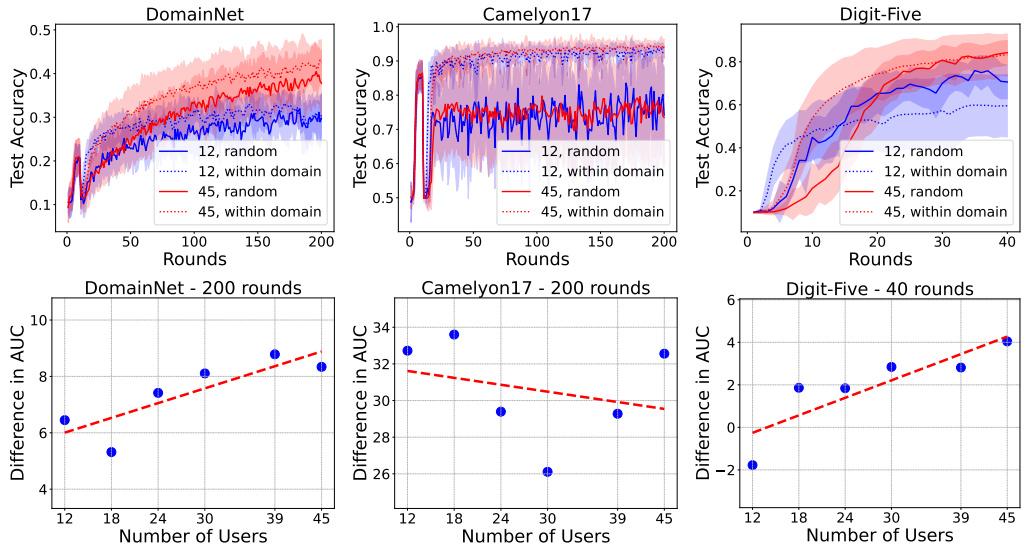


Figure 11: Enhanced Accuracy with Within-Domain Collaboration as the System Scales. (Top:) Comparison of test accuracy between within-domain collaboration and random collaboration with varying numbers of clients (12 and 45 here). (Bottom:) Illustration of the trend as the system scales. On average, the AUC gap between these two strategies widens as the system scales, indicating that within-domain collaboration is a superior strategy.

This insight is verified in 11. In the case of DomainNet and Digit-Five, performance improvement is achieved by confining collaboration to users within the same domain and scales linearly with the number of users. For Camelyon17 as seen in 10 users are not converging when collaborating at random, thus increasing the number of clients has no impact on system-wide random collaboration. In contrast, for domain-specific collaboration, increasing the number of users ultimately slows down convergence, explaining the slight decrease in AUC.

2. Domain Diversity:

In a practical, permissionless decentralized ecosystem, diverse data domains are expected to emerge, increasing overall domain diversity. The heterogeneity of clients' distribution has been extensively examined in Federated Learning (FL) [173, 176] due to its negative impact on training performance. Restricting collaboration within the domain greatly reduces heterogeneity in the data distributions of users collaborating. The performance gap between random

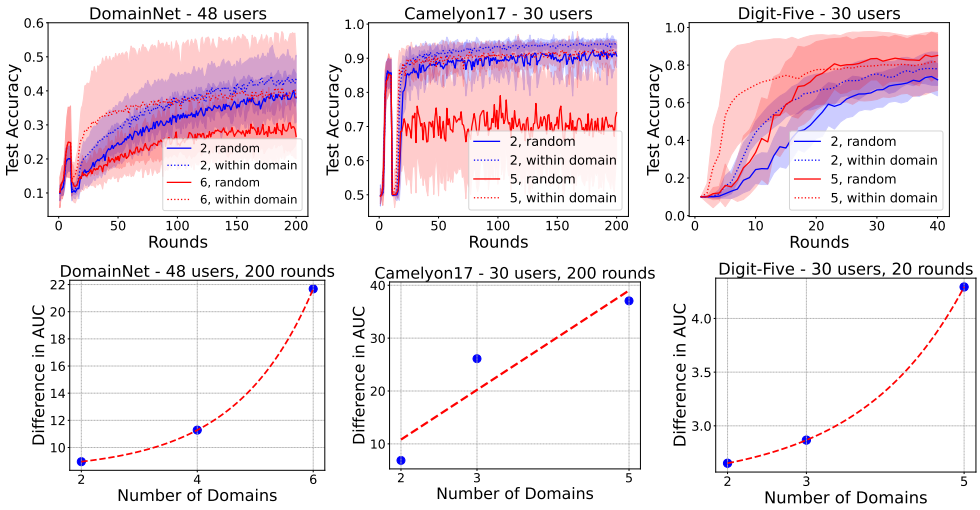


Figure 12: Enhanced Accuracy with Within-Domain Collaboration as Data Heterogeneity Grows. (Top:) Comparison of test accuracy between within-domain collaboration and random collaboration for varying numbers of domains (2 and 6 here). (Bottom:) Illustration of the trend in widening of AUC gap between the two strategies, thus showing within-domain collaboration is a superior collaboration strategy.

and within-domain collaboration is thus expected to increase as the number of domains grows, as seen in 12.

3. Number of Collaboration Rounds:

The availability of communication rounds influences the effectiveness of collaborator selection strategies. Empirical findings in figure 13 demonstrate that increasing the number of communication rounds can enhance the performance of random collaboration, making it outperform clever selection strategy. However, even with an unlimited number of rounds, carefully selecting collaborators can still have its benefits.

Desiderata for Collaborator Selection

So far, we have discussed various practical scenarios that highlight the importance of collaborator selection and provided empirical evidence in support. Now we discuss the desirable properties of a collaborator selection algorithm that guide our subsequent investigations.

1. Better than isolation – Any collaboration scheme should enhance performance compared to individual isolation. If a collaboration scheme fails to

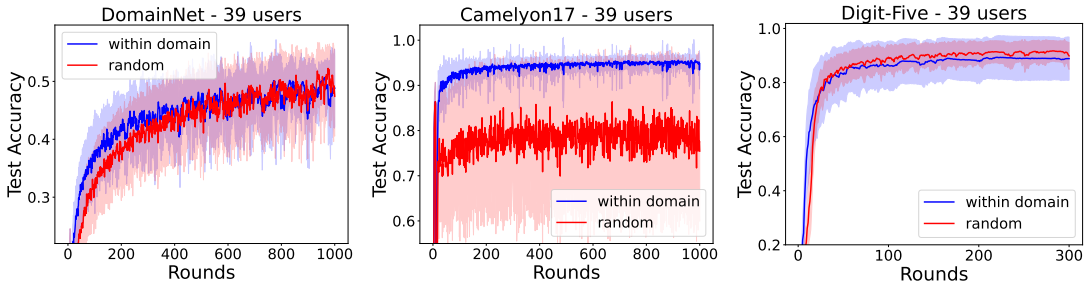


Figure 13: Evolution of within-domain collaboration with increasing number of rounds. We compare the convergence rate of the test accuracy for random and within-domain collaboration for a large number of rounds.

improve performance, participants have little incentive to engage in collaboration. Despite its apparent simplicity, this criterion is not always met by many techniques, including random selection.

2. Independent of metadata – We argue that collaborator selection techniques should be data-driven and agnostic to specific metadata. Relying on metadata like publicly available datasets or domain knowledge (e.g., demographics) to discover relevant collaborators restricts applicability. Techniques depending on public datasets are sensitive to those datasets' distributions and cannot be generalized when diverse data is unavailable. Techniques relying on domain knowledge assume prior distribution knowledge, which is often unrealistic in practical scenarios where data distributions may vary unpredictably.

3. Robust to imperfect information – While data-driven approaches for collaborator selection offer advantages, they can be susceptible to noisy parameter estimation due to imperfections in local datasets or models. An ideal selection scheme should demonstrate consistent performance even in the presence of such noisy sampling.

D.3.3 Dynamics of Collaborator Selection

We view the problem of collaborator selection as designing and evolving the collaboration graph of users in a decentralized manner. However, the interplay between collaborator selection and the collaboration process (weight aggregation) makes the problem challenging. The edges in the collaboration network (the selected neighbors) alter the state of the network nodes (models). Consequently, changes in the models lead to changes in the collaboration network edges, creating interesting dynamics. Adaptive complex systems [156] ex-

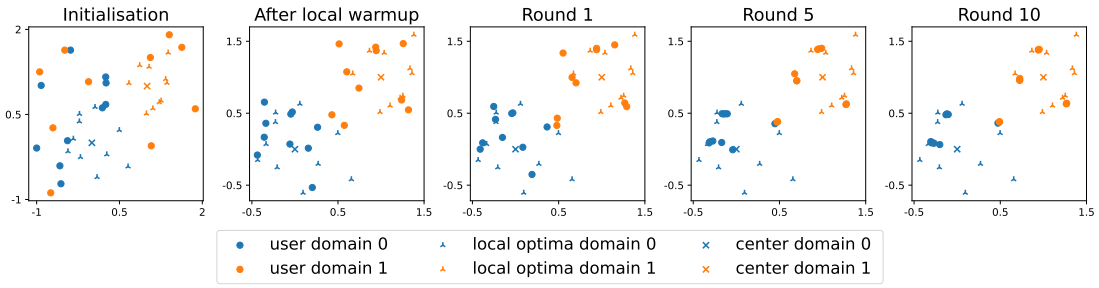


Figure 14: Illustration of greedy collaboration leading to *collaborator collapse* in 2D toy simulation. Users' models are initialized at random. Local training and collaboration attract them toward their local optima and their selected collaborator, respectively. Here, local optima are the models obtained if users train in isolation. They follow a normal distribution per domain centered around the domain center. Users select the closest user as their collaborator. This results in users collapsing into small groups, leading to a loss of domain-relevant knowledge.

hibiting such dynamics and have been studied extensively in their respective domains such as infectious diseases [84], opinion dynamics [133], and spin glass systems [10]. In our case, local training of models (a non-linear function with a non-convex optimization) makes the problem difficult to analyze. Therefore, almost all the works have relied on heuristics. For instance, several works in centralized FL [74, 143, 46] as well as decentralized FL [48, 126, 152, 104] have proposed the idea of selecting users based on similarity between users.

Emergence of Collaborator Collapse:

If collaborators are selected purely based on similarity, a feedback loop can be initiated where the same collaborator may be repeatedly selected. This results in a lack of diversity in collaborator selection, causing users to miss out on knowledge relevant to their task and ultimately limiting their performance. These dynamics are inherent to any similarity metric used between two models.

We perform several experiments and observe the collapse of collaborator diversity when a pair of users initially find each other similar and start collaborating exclusively, enhancing the likelihood of selecting each other repeatedly. This phenomenon happens either within or across domains. In the former scenario, the collapsed users lack diversity and are prone to overfitting their combined local data, hindering their ability to generalize to their actual domain distribution. In the latter case, arguably leads to worse performance as the collapsed users lack diversity and relevance. Similarly to the first scenario, they

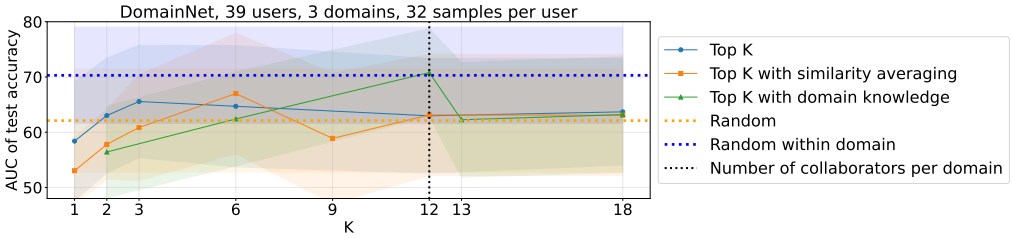


Figure 15: Sensitivity of K in Top K. The performance of top K is highly sensitive to the noise in the similarity metric. Even if we assume the domain sizes are known, finding the optimal K is not trivial. If K is too small, it can result in collaborator collapse, with performance dipping below that of random collaboration. Conversely, exceeding the optimal K threshold leads to inter-domain collaboration, hindering any potential benefits of collaborator selection.

are likely to overfit; however, their local data is even less representative of their respective underlying domain distribution. Both scenarios are observed in 14 where users collapse into smaller groups instead of learning a single model per domain, missing relevant knowledge. Users collapsing between domains (points in the center) find themselves situated far from their domain optimal model.

Validation of Collaborator Collapse using Top-K:

For the examination of these dynamics, we introduce *Top K*, a greedy collaborator selection scheme that follows the protocol introduced in 9. This scheme enables us to regulate the level of diversity in client selection through the parameter K. In each round, every client evaluates and ranks all other users according to their estimated (potentially asymmetric) similarity. From the top K most similar users, C(=1 here) users are then randomly sampled with uniform probability.

In 15, we study the importance of the choice of K[1]. To accomplish this, we define an additional baseline: *Top K with domain knowledge* it assumes users know in which domain they are part of, and sample collaborator from a fixed set of K users within their domain when K is smaller than the domain size. *Similarity*

[1] When domains have the same number of clients, it makes sense for all users to share the same value of K. However, in real-world situations, users are unlikely to be evenly distributed across domains. Consequently, each client would need to determine its optimal diversity threshold independently. This variability in K values contributes to increased heterogeneity within the system.

averaging enhances the robustness of the similarity metric by computing its running average over the last R(=10 here) rounds.

When the value of K is much smaller than the domain size, users tend to *collapse* into small groups, consequently missing out on a significant portion of the knowledge available within their domain as seen in 16. On the other hand, as K increases, collaboration extends beyond the confines of the domain, ultimately hurting the performance. However, the optimal value for K during collaboration may differ from its ground truth value, which is the number of other users within the domain. The empirical value of K is closely intertwined with the noisiness of the similarity metric. In our case, representations, models, and similarity serve as sources of noise. As the similarity metric becomes noisier, the optimal K diminishes as the inherent noise already contributes to increased diversity. This trend is discernible in Fig 15: as similarity becomes more robust and accurate in finding same-domain users, the best K increases from 3 to 12, which corresponds to the number of other users within the same domain.

Any similarity score comparing the knowledge present in any two users should be reduced after a collaboration event takes place between the two users. Thus, in the next selection round, the closest users of the previous round are even closer and, thus, more likely to be selected again.

The observation we have made so far is that the similarity metrics can be imperfect. This imperfectness gets amplified as the users rely on using this similarity metric to select other users and update their model weights with selected users' parameters. This leads to a systemic collapse that either occurs among a small number of collaborators within a domain or users across different domains. Both collapses reduce the performance of the overall system. How can users prevent this collapse? In the following section, we discuss the principle of consensus to regularize the collapse of parameters.

D.3.4 Mitigating the Feedback Loops with Consensus

We revisit the constraints and objectives of decentralized learning to mitigate the problem of feedback loops. We take inspiration from the literature in consensus-based distributed optimization [88] and posit that the goal of all users should be to reach a consensus on model parameters among users in their respective data domain. Hence, the goal can be viewed as a collective one instead of an individual one. This view partly contradicts complete decentralization, where each user can act independently and instead enforces consensus as a mechanism to prevent drift among similar users. We study consensus at two

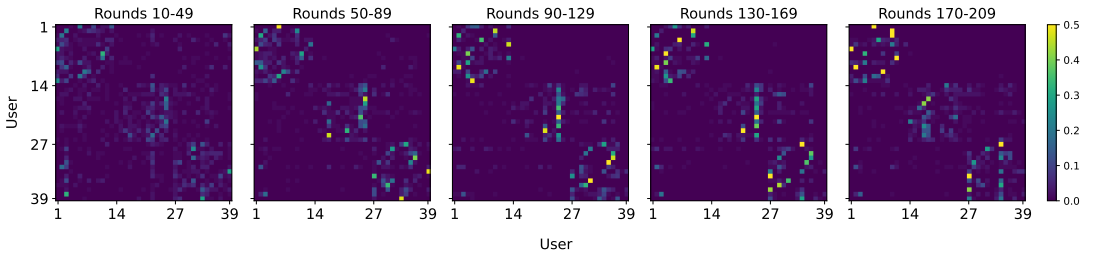


Figure 16: Evolution of collaboration with greedy collaborator selection. Each row represents the collaboration choices of the corresponding user over 40 rounds. Users are ordered by domain, with users 1-13, 14-26, and 27-39 belonging to the first, second, and third domains, respectively. At each round, each user collaborates with their most similar user. As seen, this strategy leads to *collapse* of clients collaborating with a small clique of users within a domain and thus leads to low collaboration diversity.

layers of abstraction – 1) evaluation of similarity and 2) selection of collaborators. Therefore, to address feedback loops, users need to utilize information from their neighbors and not purely rely on their own state of the system.

Improving Similarity via Consensus: In the context of similarity, the main step towards consensus is to not solely rely on one's own similarity with other users but to consider other users' similarities as well. Taking into account the *similarity profile* of other users (their respective similarity with other users) yields two main benefits: Firstly, users have a better sense of the topology of the users landscape. Indeed, by only considering one's own similarities, one can only place other participants along a single axis, whereas if the similarity between participants is known, the placement is multi-dimensional. This additional information can be leveraged when designing a collaborator selection scheme. Secondly, considering the correlation between users' similarity profiles as a similarity metric increases robustness to the noise in the system. This relies on the fact that if two users agree on their similarity estimation for other participants, they are likely to be similar themselves. In this setting, the similarity between two users is based on two times *the number of users in the similarity profile* individual evaluations. As a result, the noise contained in the individual evaluations is averaged out.

Improving Collaborators Selection via Consensus: The performance of each user is influenced not only by their own collaboration choices but also by the

Table 5: **Collaborator selection performance comparison**: We compare performance AUC of mean test accuracy across users for 39 users, 3 domains, and 200 rounds. *Top K*, *ϵ-greedy* and *L2C* are advantaged as the first two necessitate parameter tuning (K and ϵ) and *L2C* relies on extensive collaboration to learn collaboration weights. In *Greedy*, users select the most similar users, whereas in *ϵ-greedy* with probability ϵ, a collaborator is instead selected at random. *Sim sampling* is sampling users based on the softmax of their similarities. *Top K* is selecting a client uniformly at random from the top *K* most similar ones. *Affinity Propagation* clusters clients based on their similarity profile. (*Our implementation).

Selection	Random	Greedy	Sim sampling	ϵ-greedy	Top K	L2C*	Aff. Prop.	Mean-Shift
DomainNet	62.25	52.35	60.98	63.82 $_{\epsilon=0.5}$	64.91 $_{K=6}$	59.86	69.63	69.74
Camelyon17	149.64	182.46	150.10	175.94 $_{\epsilon=0.1}$	182.46 $_{K=1}$	156.01	181.13	179.26
Digit-Five	24.96	27.89	24.88	28.19 $_{\epsilon=0.5}$	29.56 $_{K=6}$	34.05	28.83	28.91

collaboration choices made by their collaborators, and this recursive impact extends further into the collaboration network. Thus establishing consensus across users on communities of collaborators ensures all collaborators within the same communities are collaborating toward the same objective, which is to converge as a group. It ensures that one's collaborators of collaborators are also one's collaborators, creating a one-hop neighborhood of collaboration.

Clustering for Communities Consensus: To empirically evaluate consensus, we design a scheme that relies on the correlation of users' similarity profiles and clustering. At each round, once the pairwise similarity between clients is established through the challenge-response mechanism, users broadcast their similarity profile. Based on the received users' similarity profile, users are clustered using a parameter-free clustering algorithm. Finally, C collaborators are sampled from the set of users in the same cluster. Note the consensus in selection is achieved as we assume full participation. This implies that after receiving similarity profiles from other users, every user has the complete similarity matrix, thus if the clustering procedure is identical between clients, they produce the same communities. In our experiments, we use *affinity propagation* [51] and *mean shift* [36] as parameter-free clustering. Note clustering implicitly sets the diversity threshold.

Consensus through clustering is giving promising results as it is the only selection scheme consistently beating random collaboration by a large margin, as seen in 5 for the three considered datasets. We further evaluate the robustness of clustering-based approaches in 17 and find that in most cases,

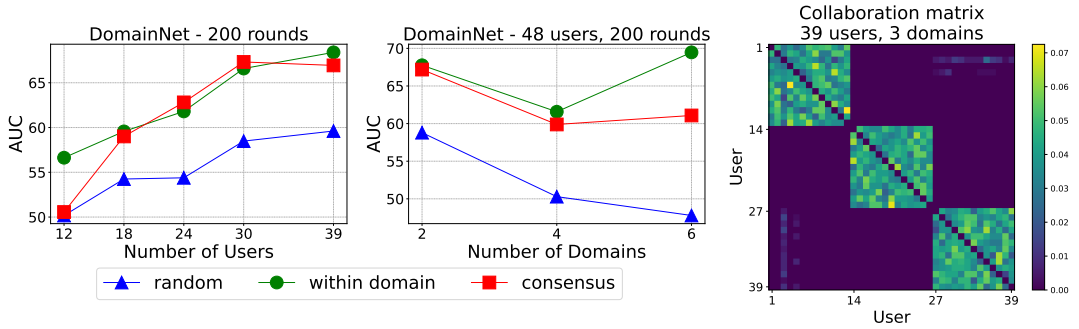


Figure 17: Robustness of clustering-based collaborator selection. Affinity propagation selection scheme is evaluated for different topologies with various numbers of users and domains. The collaboration matrix indicates that collaboration is happening mostly within the domain (clients' ordered w.r.t. domains), reaching performances comparable to those of within-domain collaboration.

affinity propagation can correctly cluster clients per domain, leading to similar performance as random collaboration within the domain.

D.4 DISCUSSION AND FUTURE WORK

In this paper, we delve into collaborator selection by addressing the following key problems: - 1) when to perform selection, 2) diversity and relevance collapse in collaborator selection, and 3) strategies to mitigate collapses. However, we make some simplifying assumptions to study the problem, but further investigation is needed to fully understand their impact. We will now discuss these limitations in detail and their potential consequences.

1. Privacy: Our experiments involve exchanging raw data, simplifying analysis for two reasons: - i) we want a proxy that closely represents user data distribution, and ii) the exchange of raw data can be replaced by users sharing their models that can be evaluated locally to score potential collaborators. As discussed in Section D.3.1, we propose alternatives like using locally generated synthetic data or dreams [168, 148]. We evaluate whether synthetic data adequately serves as a proxy for collaborator selection. Our results show that while viable, this approach incurs slight performance degradation compared to sharing raw data.

2. Asynchronous collaboration: To address the issue of feedback loops, we adopt a synchronous approach by assuming interaction between every pair of users during each communication round. This heavily synchronous system allows us to investigate the robustness of various approaches to user availability. We vary the drop probability of communication edges between users to analyze the impact on performance. We leave the exploration of fault-tolerant and asynchronous designs for future work.

3. Thresholding: While our problem formulation assumes exclusive collaboration with similar users as optimal, this approach might limit the performance by restricting the gathering of common knowledge from dissimilar peers. As observed in D.3.2, while enhancing performance initially, it can cause convergence delays and performance decline with increased system diversity. A more practical collaboration algorithm should dynamically adjust the size of the consensus group based on the number of available communication rounds. However, addressing this problem is non-trivial, as different users may have varying communication budgets, rendering a consensus-based approach incompatible. Furthermore, we only account for a single source of variance across data domains, yet there could be domain shifts along multiple dimensions. In such cases, it's unclear how a consensus-based approach can be effectively applied, as multiple distinct clusters may exist.

While the focus of this work is collaborator discovery, other important characteristics of collaborative learning such as user dropout, byzantine robustness, and stragglers should be studied in future works to have a more holistic understanding of such decentralized systems.

E LARGE POPULATION MODELS FOR DECENTRALIZED PREDICTIONS

E.1 INTRODUCTION

The recent outbreaks of COVID-19 have left an indelible mark on society at a global scale, highlighting the vulnerability of public health [117, 95]. Thus, deepening our insights into how pandemics evolve is imperative, ensuring that our actions are prompt, effective, and grounded in evidence [11, 153]. Given the unprecedented nature of these pandemics, it is challenging to simulate their dynamics. Agent-based modeling has emerged as a pivotal tool for replicating the complex dynamics inherent in the pandemic evolution [140, 2, 8, 85]. Agent-based models (ABMs) are unique in their ability to provide a granular view of disease propagation by analyzing both micro-level interactions and the broader emergent phenomena, making them particularly suited for delineating the effects of potential interventions.

In the past, governments globally adopted varied strategies to curb the spread of infections, particularly during COVID-19 [62, 26, 136]. Some were effective, while others were not [50, 17]. Interventions such as delayed travel bans proved insufficient, allowing rapid global infection spread [155], while prolonged severe lockdowns crippled global economies [167]. Additionally, the deployment of digital initiatives for contact tracing [35, 138, 160, 18] had a limited impact due to low adoption and delays in user quarantine post-exposure [40]. As notified users awaited test results, potential carriers inadvertently continued activities, making this approach largely ineffective in curbing transmission [97, 39]. Pharmaceutical interventions, once viewed as the primary defense against the pandemic, encountered their own set of challenges [163]. The apprehension over the longevity of vaccine-induced immunity and potential side effects further hampered the pace of vaccination drives [151, 124]. Moreover, by the time effective vaccines were produced, many nations had already peaked in infections [128]. Thus, reflecting on these previous strategies to understand what worked and what did not is crucial for developing efficient future pandemic responses.

However, decision-making in such scenarios is challenging due to the multifarious intricacies of complex societies characterized by heterogeneous popula-

tions, diverse behavioral patterns, and differential access to resources [9, 12, 33]. The interplay of various interventions, their mutual impacts, and the factors influencing their effectiveness adds further layers of complexity. In this paper, we address these challenges of modeling real-world simulations in complex societies by considering varied populations with interaction networks spanning across household, occupational, and random graphs. We consider behavioral patterns through app adoption rates, self-quarantine, and compliance probabilities. Further, we model differential access to resources stratified by age or policy choices, for instance, prioritizing higher age group individuals for vaccination and age-based app adoption.

We model the progression of a pandemic over its initial 6 months (180 days) using real-world socio-demographic (census) data from Kings County, Washington, evaluated at a scale of 100,000 agents. Our ABM framework is currently parameterized to King's County demographics and calibrated to the epidemic outcomes. It can easily be re-parameterized for other geographies. All static variables such as age, household, occupation, and (random) number of daily interactions are initialized using the real-world census and mobility data for King's County. We release our data parameters in the code.

For our analysis, we simulate three types of interventions: pharmaceutical, behavioral, and digital, and highlight the effectiveness and potential pitfalls of each approach in controlling future pandemics. We not only assess these individual interventions but also integrate a collective interplay of these interventions, suggesting they are complementary to each other, not alternate. Pharmaceutical interventions include vaccination drives and testing to detect cases. Behavioral interventions include self-quarantine upon testing positive and individual responsiveness and adherence to recommended actions. The lockdowns many countries implemented can be viewed as a prolonged strict self-quarantine. Digital interventions explore tools such as contact tracing apps designed to monitor and curtail spread through tracking interactions. Our extensive analysis suggests relying solely on rapid vaccine development for outbreak control isn't viable. Enhanced preparedness demands an integration of pharmaceutical approaches with contact tracing and behavioral strategies, ensuring a holistic, prompt response.

The following are the major contributions of our paper: (1) We introduce a general pipeline using ABMs that simulates a real-world synergy of interventions at scale, encompassing pharmaceutical, behavioral, and digital strategies. This framework offers extensive detail to capture the complexities observed in the real-world adoption of these interventions. (2) Our user-friendly and flexible framework is designed with a customizable configuration file, enabling

non-technical people like epidemiologists and policy-makers to study the effect of intricate interventions on pandemics. (3) We provide a comprehensive cost analysis of pandemic containment under each intervention strategy. (4) We perform extensive experiments on real-world data from Kings County, Washington for COVID-19. Our findings deepen the understanding of pandemic trends and offer valuable policy recommendations for effective pandemic response. (5) Some of our interesting insights are: (a) The first 100 days of the pandemic are a pivotal threshold in determining the course of a pandemic's trajectory. (b) Pairing delayed vaccination with digital and behavioral interventions proves more impactful than solely pushing for early vaccination, as it not only reduces overall infections and hospitalizations but also delays their peak. (c) With a fixed $0.5M budget, investing in testing with self-quarantine and digital contact tracing is more effective than funding early vaccinations alone.

E.2 RELATED WORK

Agent-based models (ABMs) are discrete simulators that allow entities (agents) with designated characteristics to interact within a given computational environment, replicating complex systems [24, 123, 139, 174, 44, 67]. Recently, ABMs have been widely employed in epidemiology to understand disease progression and the efficacy of interventions by providing relevant information to investigate and predict the behavior of the pandemic [110, 140, 2, 8, 85]. Several studies have utilized ABMs to evaluate the effectiveness of different interventions, such as social distancing, quarantine, lockdown, and vaccination [68, 38, 140]. ABMs have also been used in prior works for addressing policy-related queries like evaluating the importance of test turnaround time versus its sensitivity [92], and the benefits of postponing the vaccine's second dose to focus on the distribution of the first dose [140].

However, the utility of ABMs for practical decision-making depends upon several factors. These include their accuracy in replicating the population behavior [131, 55]. Furthermore, ABMs are conventionally slow. A single forward simulation over a large ABM can take several days [15, 22]. ABM simulations are difficult to scale to large populations [22], and are tough to calibrate with real-world data [131]. Most prior work either studies the effect of only one intervention at a time or simulates very few agents [43]. Real-world deployment of intervention strategies intricately linked to each other should be

scalable to large populations and need to be studied with a combined effect of each of these interventions [85, 38, 68].

Only after overcoming these challenges in ABMs can their insights truly guide strategic pandemic interventions. Our model provides a comprehensive system that simulates interventions with real-world challenges of deployment or adoption, representing them through quantifiable parameters. We adopt a vectorized approach [38] which enables a fast, parallelized simulation, allowing analysis of emergent behavior on a large-scale population for millions of agents in a few seconds. We not only assess the individual interventions but also integrate a holistic interplay of these interventions.

E.3 METHOD

Figure 18 shows the pipeline for different interventions along with the progression of disease stages. We build on existing open-source agent-based modeling frameworks [38, 68], optimizing large-scale simulations through matrix computations, leading to enhanced computational efficiency. Our model addresses the potential variances in the adoption of interventions by employing a stochastic approach, sampling from Gaussian distributions based on a certain compliance level to predict outcomes. We model these interventions with an unprecedented level of detail with aspects that have not been explored in such granular detail in prior research. This comprehensive approach facilitates the examination of the individual impact of each parameter, and also provides insights into the synergistic effects of these interventions on the pandemic response.

In our model, individual agents (and their states) are modeled as tensors. Agents navigate through eleven potential disease stages: susceptible, asymptomatic, presymptomatic (mild or severe), symptomatic (mild or severe), hospitalized, in intensive care, recovered, immunized, or deceased. Infections can propagate during any interaction between susceptible and infected agents. These interactions span three networks: household, occupation, and random encounters, which are represented as sparse adjacency matrices. Each such interaction is a stochastic process with a certain risk of disease transmission. The foundational assumptions about disease progression, transmission dynamics, and network interactions align with prior agent-based models [38, 140]. We will now delve into an in-depth examination of each intervention, detailing various parameters and compliance factors reflective of real-world scenarios.

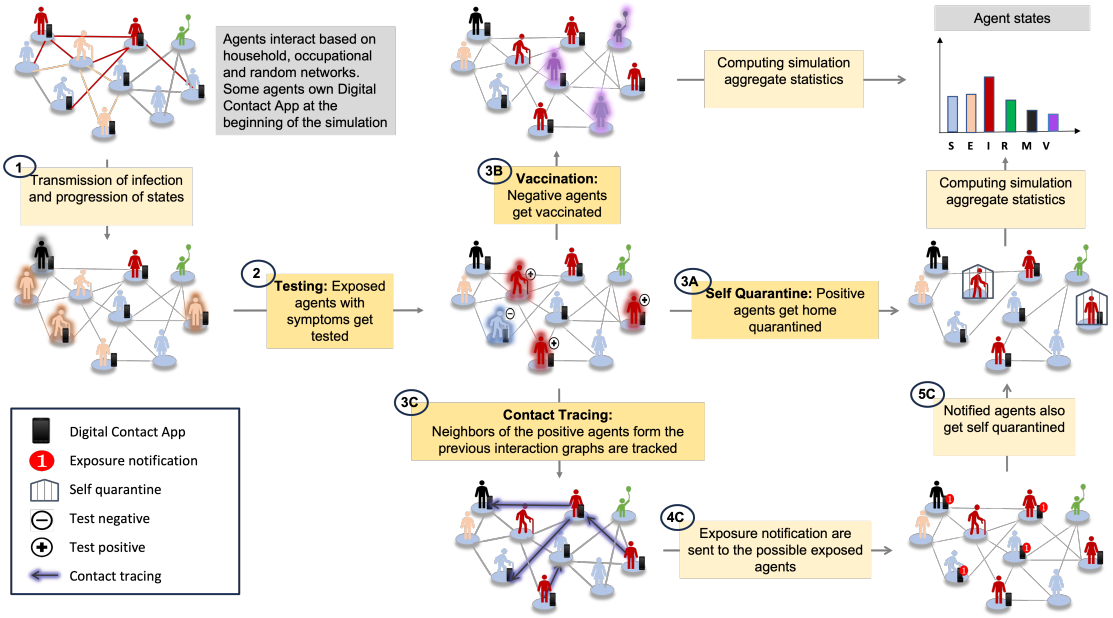


Figure 18: Implementation of different interventions - Testing, Self-quarantine, Vaccination, and Contact Tracing. (1) Infection spreads through the interaction of infected with susceptible agents, and the states of the agents are then updated based on disease progression. (2) Upon experiencing symptoms, exposed agents get themselves tested (3a) If tested positive, agents undergo self-quarantine with compliance. A quarantined agent then engages in no further interactions until the quarantine period ends. The interaction graph of quarantine agents is thus an isolated point (3b) Agents that have not tested positive or are not quarantined get vaccinated. Vaccination reduces the susceptibility of an agent to infection risk (3c) In case of contact tracing: interactions of the positively tested agents (that own app in case of DCT) from the previous interaction graphs of past days are tracked; (4c) exposure notifications are sent to the possibly exposed tracked agents (that own the app in case of DCT); (5c) notified agents then opt for self-quarantine. (Last) After simulating for N days, the aggregate statistics of the agent states are computed. Agent states here are: susceptible (S), exposed (E), infected (I), recovered (R), mortal (M), and vaccinated (V)

Table 6: Description of Testing parameters

Parameter	Explanation
test_start_date	Date on which testing begins
test_true_positive	Prob. of a true positive result
test_false_positive	Prob. of a false positive result
test_results_dates	Potential dates of receiving test results
test_results_dates_probs	Dictionary of probabilities associated with each test result date
test_validity_days	Duration for test results validity
test_cost	Average cost of production of a test

E.3.1 Testing

Agents who are exposed to infection and develop symptoms undergo testing. Every diagnostic test is defined by three primary parameters: specificity, turnaround time, and duration of test validity. The turnaround time accounts for any inherent delays in receiving test results, presented as a dictionary detailing possible result dates and their associated probabilities. The test validity indicates the duration for which the test results are considered relevant. After this period, agents are expected to be retested. Factoring in real-world delays related to the deployment of testing kits, tests can be deployed in the model after some start date, marking the start of distribution of that particular testing method to the public. Table 6 shows the different parameters supporting the testing mechanism.

In our simulations, we employ two types of tests: (i) RT-PCR test, with a specificity of 0.99 and a turnaround time of 1 to 3 steps (1 to 3 days) uniformly distributed [38, 89], and (ii) rapid point-of-care test, which offers slightly reduced specificity of 0.85 with a turnaround time of 0 steps (same day). To cater to varying diagnostic requirements, these parameters can be adjusted, offering flexibility in modeling different test types. By default, our model uses the more reliable RT-PCR test for simulations unless specified otherwise.

E.3.2 Self-quarantine(SQ)

Upon testing positive or receiving exposure notification, an agent undergoes a 14-day self-quarantine adhering to compliance. However, an agent might not consistently adhere to the complete quarantine. To simulate such imperfections, a daily dropout probability of 1% is incorporated to account for potential non-compliance. During the quarantine period, the agent's interaction network effectively becomes isolated, leading to no interactions with other agents.

Table 7: Description of Self-quarantine parameters

Parameter	Explanation
quar_enter_prob	Prob. with which an agent enters self-quarantine after testing positive
quar_break_prob	Daily quarantine dropout probability due to non-compliance
quar_days	Number of self-quarantine days

After successfully completing the quarantine period, the agent's capacity to transmit the infection is nullified, effectively resetting their infectiousness to zero. Table 7 provides a detailed overview of different parameters for modeling self-quarantine.

E.3.3 Vaccination(VACC)

We simulate a two-dose vaccination regimen with an extensive level of granularity. A dose of the vaccine provides a certain probability of becoming immune to infections, depending on whether it is a first or second dose. Vaccines are administered in an age-prioritized fashion, with the oldest individuals receiving their vaccines first, and first-dose candidates given precedence over the second dose. We simulate a probabilistic immunity conferred post-vaccination, where immunity is not immediate but materializes after a stipulated delay post-inoculation. Table 8 illustrates the parameters driving our vaccination models, such as the vaccine's start date, daily production rate, shelf life, and efficacy percentages for both doses. Moreover, it also highlights potential dropouts - those who, after receiving the first dose might choose not to return for the second, capturing a real-world nuance in the vaccination process.

For all the experiments, we assume a 90% efficacy for the first dose and a 95% efficacy for the second dose administered 21 days later. Further, we also do a sensitivity analysis on lower efficacy rates for the first dose of vaccines, including 30%, 50%, and 70%. (in Supplementary). All simulations, unless indicated otherwise, presume vaccination commencement at t=10 with a daily vaccination rate of 0.3% on a population of 100K based on U.S. vaccination rates and patterns observed internationally [38].

E.3.4 Contact Tracing (CT)

We adopt a hybrid contact tracing approach where first exposure notifications are dispatched to contacts of infected app users followed by manual follow-up

Table 8: Description of Vaccine-related parameters

Parameter	Explanation
vacc_start_date	Date on which vaccine drive begins
vacc_daily_prod	No. of vaccine doses produced daily
vacc_shelf_life	Duration before a vaccine dose expires
vacc_dose_delay	Days after which the vaccine dose starts showing effect
vacc_dose1_priority	Indicator if the first dose is prioritized over second in distribution
vacc_dose1_eff	Efficacy of the first vaccine dose
vacc_dose2_gap	Duration between the first and second doses of the vaccine
vacc_dose2_eff	Efficacy of the second vaccine dose
vacc_dose2_drop	Probability of an individual not returning for the second dose
vacc_price	Cost for development of a single vaccine

for non-compliant users and agents not owning the app. Below, we delve into the specifics of digital and manual tracing methods.

Digital Contact Tracing (DCT): At the start of the simulations, agents own a Digital Contact App (DCA) with a fixed adoption rate based on age-stratified data. This app records interactions of an agent across all three networks: household, occupation, and random, within a 7-day window. Note that interactions are logged only if both agents have the app. When an agent with an app tests positive, they can opt to notify exposed contacts via the DCA. Recipients then undergo self-quarantine based on their compliance probability. In our experiments, we simulate DCT assuming an average 40% app adoption rate and 80% compliance rate for self-quarantine.

Manual Contact Tracing (MCT): Manual tracing is similar to its digital counterpart, with a few key differences as illustrated in Figure 19. Unlike DCT, MCT doesn't require smartphone ownership and is unlikely to remember random or casual encounters (like those in public transport or stores). Only contacts within the household and occupational networks are traced through MCT. Manual tracers interview an infected agent to identify and track the potential contacts over the past N (=7) days. However, only a portion of the true interactions are identified based on the likelihood of recalling them (70%). From these, a subset responds based on a set probability. Successfully contacted agents then self-quarantine with a compliance probability of 90%.

In a targeted two-step process of contact tracing, MCT and DCT leverage coupled capabilities of human intervention with digital tools. Performing manual contact tracing of targeted potential infected agents who either did not own the app or ignored the digital notifications can significantly improve the

Table 9: Description of Contact Tracing parameters

Parameter	Explanation
app_adoption_rate	Prob. of agents owning the app at the start of the simulation in DCT
max_contact_days	Number of days for which history of previous interactions are traced (unique for DCT and MCT)
test_inform_prob	Prob. to notify the contacts via DCA/MCT after testing positive (unique for DCT and MCT)
mct_recall_prob	Probab. that an individual recalls their contacts accurately during MCT
mct_reachable_prob	Probability that an individual is reachable for manual contact tracing
sq_comply_prob	Compliance prob. for quarantine upon successful contact tracing (unique for DCT and MCT)

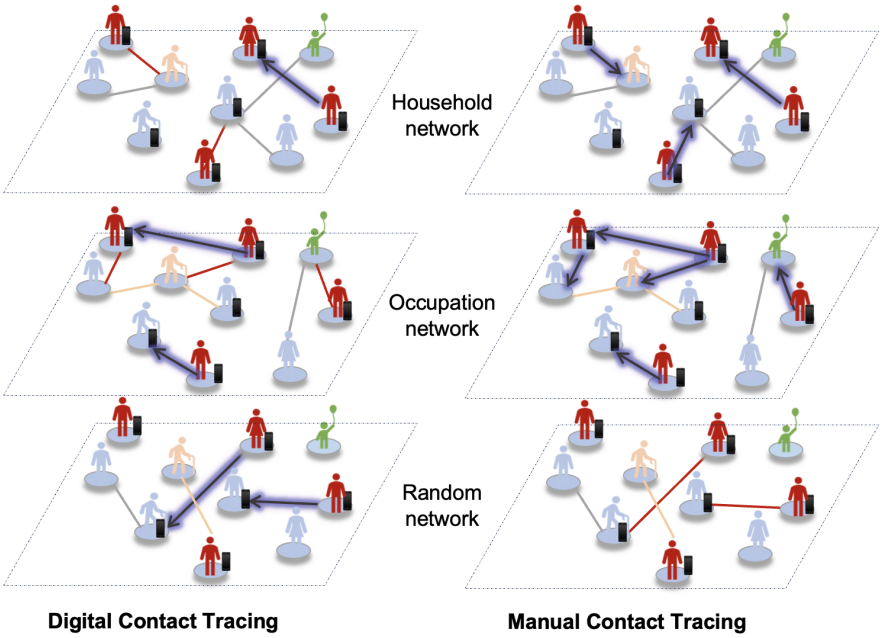


Figure 19: Comparison of Digital vs. Manual Contact Tracing: Digital tracing requires app ownership for both interacting agents but can effectively track unknown or random interactions, while manual tracing captures household and occupational contacts but may miss random interactions

scale/outreach of the tracing efforts to contain the infection spread. Table 9 details the parameters used to model contact tracing.

E.4 RESULTS

We study the impact of different interventions discussed above on disease progression and pandemic evolution in a population with 100,000 agents over a period of 180-time steps. In particular, we simulate Self-Quarantine (SQ), Vaccination (VACC), and Contact Tracing (CT) interventions and evaluate their

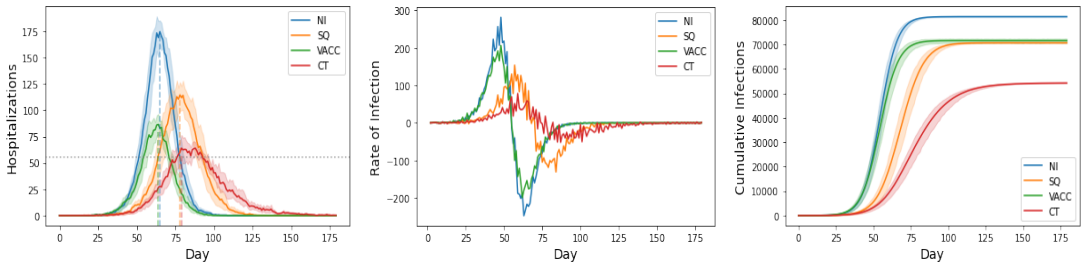


Figure 20: Comparative analysis of the individual impact of different interventions on pandemic progression; No Interventions (NI), Self-Quarantine (SQ), Vaccination (VACC), and Contact Tracing (CT). (a) Peak hospitalizations showcase the strain on healthcare under each scenario, with notable stress in the NI and SQ cases. The dotted line represents the hospital bed availability for Kings County, Washington (b) Daily new infection rates highlight the efficacy of interventions, with CT significantly lowering the infection rate. (c) Cumulative infections over time reveal the pervasive nature of the pandemic in the absence of effective measures and a substantial reduction in total infections under VACC, SQ, and CT.

outcomes. We ran experiments using real-world socio-demographic and geo-census data from Kings County in Washington state. All the results correspond to the mean and standard deviation aggregated over 10 independent runs of the simulation.

We present our results in five sections. Section E.4.1 examines the individual effect of different interventions on the evolution of the pandemic outcomes. Section E.4.2 focuses on the age-stratified analysis for these individual interventions. Section E.4.3 delves into the overall cost analysis of individual interventions, highlighting their financial implications. Section E.4.4 provides insights into the interplay of pharmaceutical, behavioral, and digital interventions, allowing us to study their cumulative effect on the pandemic's trajectory. Section E.4.5 shows the geographical progression of infections in Kings County, WA, where we simulate a combination of all the interventions together and compare the spread with the unmitigated no-intervention (NI) case.

E.4.1 Analysis of individual impact of different interventions

To provide a baseline for the impact of the pandemic, we first investigate an unmitigated scenario in which there are no interventions (NI). We compare this baseline unmitigated scenario with interventions such as self-quarantine (SQ), vaccination (VACC), and contact tracing (CT). Figure 20 details the comparative

analysis of the individual effect of each of these interventions in terms of the number of severely affected individuals who require hospitalization, rate of infection, and cumulative infections. For our analysis, we assume the number of beds per 1k people in Kings County in early 2020 to be 1.57 [106]. On average, 65% of hospital beds are already occupied. So the number of available hospital beds per 100,000 people is 1.57*0.35/1000*100,000 $\approx$ 55.

Our analysis shows that the uncontrolled pandemic (NI) peaks at t=65, with the number of hospitalizations of 175, far exceeding the available capacity of 55 by 218% as depicted in Figure 20(a). This indicates the immense strain on the healthcare system in the no-intervention case, pushing it to the brink of collapse. Figure 20(b) highlights a peak daily infection rate of 281 per 100,000 agents in NI, with a staggering 81% of the population infected by the end as visualized in 20(c).

In the SQ scenario, while the maximum rate of infections dropped by 45% compared to the NI case, hospitalizations still peaked at 115, overshooting the capacity by 109%. This suggests that self-quarantine alone without the support of additional containment strategies is not a viable option. The VACC strategy resulted in hospitalizations still peaking at 57% above the estimated capacity, with daily infection rates nearing those in the NI scenario. For both SQ and VACC strategies, around 70% of the population was infected by the end of the pandemic. Interestingly, despite VACC's high infection rate, we observe fewer individuals needed hospitalization due to enhanced immunity gained by the agents through vaccination.

In the case of contact tracing (CT), a huge reduction in hospitalizations from the NI case is observed, bringing the peak very close (within 16%) to the available capacity. Additionally, CT delayed the peak by 14 days, giving the healthcare system more time to be prepared with the necessary resources. The maximum rate of infections dropped by a massive 72%, with only 54% of the total population infected by the end of the pandemic. Therefore, our experiments indicate that CT with testing is the most effective standalone intervention for pandemic containment.

Notably, irrespective of the specific intervention, the peak consistently occurs within the first 100 days for each individual strategy.

E.4.2 Age stratification of infections for different interventions

Figure 21 depicts the age-stratified cumulative infections in Kings County, Washington. While implementing CT, we simulate the app distribution with an

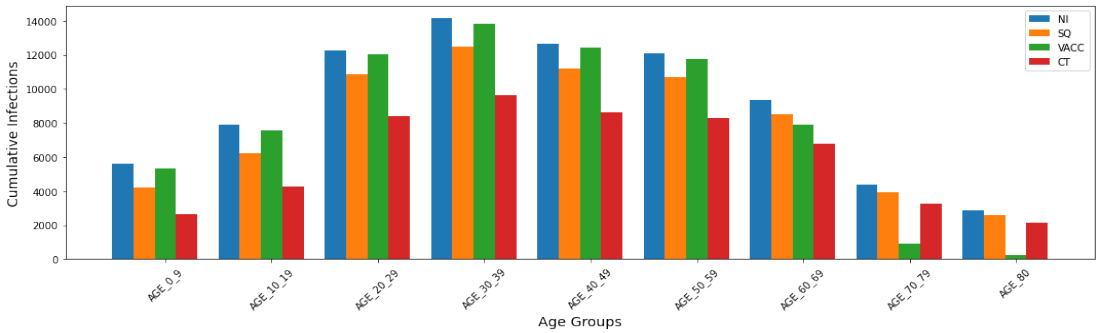


Figure 21: Age-stratified cumulative infections in Kings County, Washington, illustrating the impact of contact tracing (CT), self-quarantine (SQ), and vaccination (VACC) intervention scenarios on different age groups.

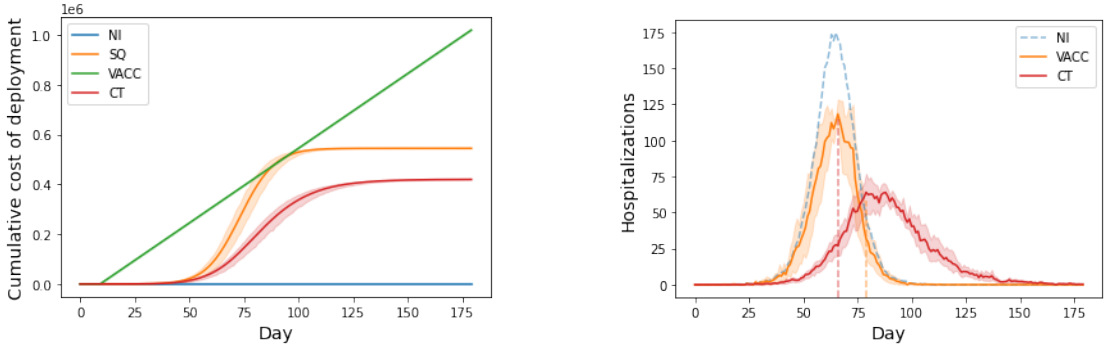
average overall app adoption rate of 40% in an age-stratified manner, where the age groups 20-59 have a higher probability of owning the app. Consequently, we observed a large drop of 26% in cumulative infections for agents in these age groups of 20-59. However, a significant drop (approximately 40%) in cumulative infections in the age group 0-19, was also observed even with relatively low app adoption rates. This is due to effective manual contact tracing implementation within households.

For the VACC intervention case, we prioritize agents in higher age groups for vaccinations. Hence, we observe a reduction of 82% and 92% infections for the age groups 70-79 and 80-89, respectively, compared to the NI case. These drops are substantially high compared to the average drop in infections over all age groups of 14%.

E.4.3 Cost analysis of individual interventions

In this section, we evaluate the economic implications of various interventions in controlling the pandemic. The cost of the no-intervention (NI) case is $0. For the self-quarantine (SQ) and contact tracing (CT) interventions, we account for the cost of tests taken by agents experiencing COVID-19 symptoms. The average cost per test for each case is assumed to be $5 [45]. We assume this is the average cost per testing kit incurred by the government. Similarly, for vaccination (VACC), each dose is priced at an average of $20 [83].

Figure 22a shows the respective costs of each intervention strategy, with their individual impacts elaborated in Section E.4.1. E.4.1. Computing the total expenditure, VACC stands at $1.02M, SQ at $0.54M, and CT at a minimum of $0.42M Beyond the reduction in infections and hospitalizations explored



(a) (b)

Figure 22: (a)Comparison of costs for different intervention strategies. The figure shows contact tracing (CT) is the most cost-effective over both self-quarantine (SQ) and vaccination (VACC); excluding $0 cost for no-intervention (NI). (b)Comparative analysis of hospitalizations under a fixed budget of $0.42M for contact tracing (CT) versus vaccination (VACC). The figure shows CT leads to a significant reduction in hospitalizations compared to VACC along with a pronounced delay in the peak, underlining the superior cost-effectiveness and strategic value of contact tracing in the pandemic's early stages.

in Section E.4.1, it's evident that CT surpasses by being 23% and 59% more cost-effective than SQ and VACC, respectively.

Further, for a fixed budget of $0.42M, our analysis shows that deploying contact tracing with self-quarantine (CT) outperforms an exclusive focus on vaccination. In this context, the CT strategy remains consistent with the previous simulations, with only the daily vaccination production adjusted to fit the budget. As Figure 22b demonstrates, allocating the budget to testing with contact tracing (CT) results in a significant 63% decline in peak hospitalizations against the no-intervention (NI) baseline. Conversely, directing the entire same budget towards vaccinations alone (VACC) yields just a 32% reduction in peak hospitalizations compared to NI. Notably, while the VACC and NI peaks coincide, the CT approach introduces a 13-day delay in the surge of hospitalizations. This 20% temporal divergence is pivotal, offering healthcare systems a crucial extended window for preparation.

In conclusion, for every dollar invested, contact tracing proves to be the more cost-efficient choice compared to vaccination, particularly in the crucial first 100 days. A mere 40% app adoption rate paired with 80% self-quarantine compliance under the CT strategy offers a better return on investment than the same expenditure on vaccination alone.

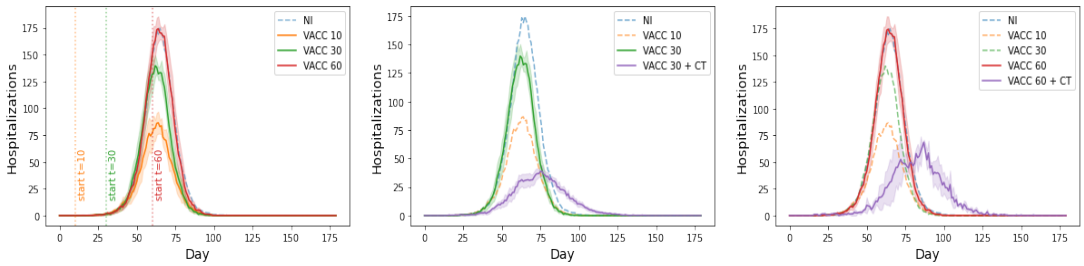


Figure 23: Analysis of interplay of digital and behavioral interventions on delayed vaccination. (a) Illustrates the impact of vaccine deployment speed on hospitalizations. Vaccine rollout delays lead to a consequential rise in hospitalizations with peak incidence remaining consistent. (b) Demonstrates the synergy of contact tracing and varied vaccine deployment timings, emphasizing that combining VACC(t = 30) + CT significantly diminishes hospitalizations and prolongs the time to peak compared to early vaccination alone. (c) Indicates the challenges with vaccine initiation at the pandemic's zenith, stressing that even late vaccine rollouts, when coupled with testing, contact tracing, and self-quarantine, can drastically mitigate infections and allow for a crucial extended immunization period. This highlights the indispensability of integrating behavioral and digital strategies, especially in the pandemic's early days when clinical interventions might not yet be in full swing.

E.4.4 Coupled effect of pharmaceutical, behavioral, and digital interventions

In this section, we study the combined effects of vaccine deployment speed and other pivotal interventions, examining their collective impact on hospitalizations. We observe that regardless of intervention combinations, the peak consistently emerges within the first 100 days, highlighting the significance of timely informed decisions. Figure 23(a) illustrates the relationship between the speed of vaccine deployment at distinct time intervals: $t = 10$, $t = 30$, and $t = 60$ and the subsequent hospitalizations. Compared to starting the vaccinations at t=10, a delayed vaccination drive starting at t=30 and t=60 increases the number of hospitalizations by 61% and 103%, respectively, with all scenarios peaking around the same time.

However, by integrating other digital and behavioral interventions with vaccination, we observe a transformative mitigation effect. Figure 23(b) shows that VACC starting at t = 30 + CT leads to a 55% reduction in hospitalizations compared to only early vaccination starting at t=10 and further a 72% drop in hospitalizations compared with VACC at t=30. Additionally, this amalgamated

approach grants an extra 14-day buffer prior to the hospitalization peak, facilitating a strategic advantage for healthcare system preparedness. The cost analysis for Figure 23(b) is provided in supplementary material.

Further, sole reliance on late vaccination starting at $t = 60$ fails because of the inherent lag in post-inoculation immunity development. However, when this late vaccination is augmented with proactive contact tracing and self-quarantine measures, the results are noteworthy: a 61% reduction in hospitalizations accompanied by an additional 23-day window for effective immunization as in Figure 23(c).

This underscores the potential of non-pharmaceutical interventions (NPIs) like behavioral and digital, not just as alternate measures but as pivotal strategies in pandemic control, especially when vaccination rollout faces delays. Thus, a multifaceted approach combining behavioral, digital, and pharmaceutical measures is pivotal in effectively managing the pandemic, especially during the first 100 days when clinical interventions face delays.

E.4.5 Geographical spread

Figure 24 shows heat maps of infection spread over time in King's County, Washington State at distinct time intervals: $t = 50$, $t = 70$, and $t = 120$. We compare the unmitigated (NI) scenario in Figure 24 (Top) against an integrated strategy that combines all interventions vaccinations, testing, contact tracing, and self-quarantine in Figure 24 (Bottom). We observe that in the case of no intervention, the infection spreads aggressively, already infecting a substantial 25% of the population by day 50, 76% by t=70, and 81% by t=120. In stark contrast, Figure 24 (Bottom) captures the attenuated spread when a comprehensive set of interventions is deployed. Infections stand at a mere 5% population infected by day 50, 19% by day 70, and only 36% by t=120.

E.5 DISCUSSION

In this paper, we highlight the potential of agent-based models to simulate highly complex environments through multiple intertwined interventions. We discussed intricate modeling of pharmaceutical, behavioral, and digital interventions and how their holistic understanding is important when creating pandemic policies. While we take one step towards bridging the gap between understanding the emerging trends and policy-making, we posit that there

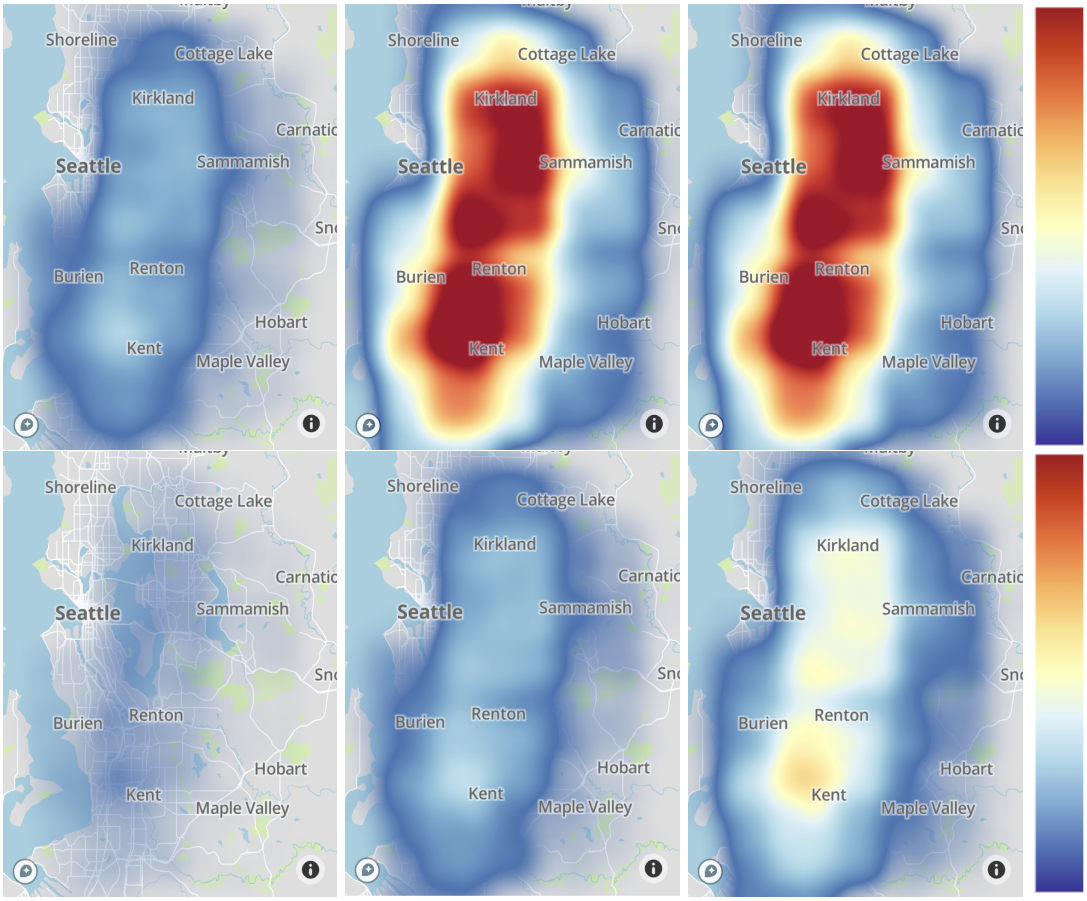


Figure 24: Geographical progression of infections in Kings's County, Washington, at different time intervals. (Top) In case of no intervention, the infection spreads to 25% of the population by t=50, 76% by t=70, and 81% by t=120. (Bottom) In the case of combined digital, behavioral, and pharmaceutical interventions, infection spreads slowly to only 5% of the population by t=50, 19% by t=70, and only 36% by t=120.

can be additional implicit factors stemming from complex interventions and their ripple effects that frequently go unnoticed, yet significantly influence the trajectory of the pandemic. For instance, financial interventions like severance funds or government aid [75] provided to the unemployed to stay at home could alter mobility patterns [14, 7], influencing the pandemic's course. Additionally, our cost analysis focuses predominantly on explicit monetary costs. Some interventions, while not incurring direct costs, may lead to broader economic implications [111, 158, 145]. For instance, extensive lockdowns, a common strategy during COVID-19, triggered a severe global economic downturn. This

collapse was characterized by soaring unemployment rates [5, 13], halted international trade [159, 63, 107], and suspended supply chains [120, 56, 109, 130]. These influencing factors further raise pivotal questions for policy-making: Does the amount of unemployment aid play a larger role in pandemic control than the speed of its provision? Are short-term lockdowns (<31 days) the solution [165, 127, 21], or is there an optimal percentage of people returning to offices (RTO) [166, 52] that can help control the pandemic and also not harm the economy at a global scale? Answering these questions can provide insights into the efficacy and promptness of policy interventions. Therefore, modeling these latent factors is a vital future direction in aiming for a comprehensive understanding of a pandemic's broader impacts.

Further, effectively using ABMs for real-world decisions requires meticulously recreating population details, demanding ample data. However, most of this data is siloed across diverse institutions and individuals [146] and may also be private [136]. Future endeavors can also explore private[28, 59] and collaborative machine-learning approaches[149, 134] for learning and calibrating these models.

E.6 CONCLUSION

In this paper, we emphasize the capabilities of agent-based models in understanding the complex dynamics of pandemics and simulating the potential impact of different policy interventions. By simulating interventions with their real-world deployment challenges, we analyze emergent behaviors on populations at scale. Our approach goes beyond merely evaluating standalone interventions by capturing the comprehensive interplay of combined strategies. From our experiments, several critical findings emerged. The initial 100 days of a pandemic largely shape its course and underline the need for swift and informed decisions from the beginning. While vaccines play a pivotal role in reducing individual susceptibility, achieving community-wide immunity is a gradual process due to the time-consuming nature of mass vaccination rollouts. Our research emphatically highlights the indispensability of sustained interventions alongside vaccinations. Notably, we observed contact tracing's efficacy for not only reducing the cumulative infections from 81% (in the absence of intervention) to 54% but also delaying the infection peak by 14 days. Our analysis further shows that the same amount of dollars spent on extensive testing with contact tracing and self-quarantine proves to be more cost-effective than

spending on vaccinations alone. Future global health crises thus necessitate a balanced, multi-pronged response.

E.7 ETHICS STATEMENT

Our research emphasizes the responsibility to consider the societal implications of pandemic response strategies. By employing Agent-Based Models (ABM) to simulate various interventions, we aim to provide decision-makers with evidence-based insights while stressing the need for swift and informed action, particularly in the critical initial phase of a pandemic. Furthermore, our findings advocate for a balanced approach to pandemic response, highlighting the complementary roles of pharmaceutical, behavioral, and digital interventions. This study accentuates the need for ongoing dialogue and collaboration among researchers, policymakers, and communities to address complex challenges in shaping effective and equitable responses to global health crises responsibly.

F | CONCLUSION

The digital age offers an unparalleled opportunity to tap into the wealth of data dispersed across various entities. Developing decentralized learning methods becomes imperative, not only to unlock the collective wisdom inherent in distributed networks but also to safeguard privacy and respect the autonomy of individual entities. This thesis addresses these challenges by introducing innovative algorithms and systems to facilitate collective intelligence while ensuring data privacy and preserving the decentralized nature of digital ecosystems. Specifically, it addresses key research challenges on collaborative training mechanisms with private data silos and heterogeneous resources, self-coordination in decentralized systems, and the value of crowdsourced prediction for population-level outcomes.

However, to realize the full potential of Decentralized AI, a holistic view of the vision must encompass several key components: privacy, distributed data, computing, orchestration, verifiability, and incentives. These elements are deeply intertwined, presenting interesting challenges at their intersections. While the decentralized nature of the learning process fosters collaboration and scalability, it also introduces vulnerabilities that malicious actors can exploit. Several known attack vectors during both model training and aggregation threaten the privacy of local data contributions. These challenges are particularly crucial in decentralized AI settings where no central authority oversees the process. Given that the primary objective of decentralization is to foster collaboration among entities with distinct assets and objectives, motivating contributors to engage and participate in the decentralized system is thus paramount. Therefore, incentive mechanisms must be developed to encourage user involvement while ensuring fairness, transparency, and value attribution. Addressing these challenges forms the basis for future work in advancing this vision of decentralized AI.

This thesis advocates for accelerating AI research to coordinate distributed data, training algorithms, orchestration, and insights across the ecosystem. By promoting collaboration, innovation, and equitable access to knowledge, this work lays the foundation for a future where trust and inclusivity drive the evolution of decentralized AI systems, characterized by privacy-preserving

measures, incentivized participation, and orchestrated collaboration. It aims to empower individuals, unlock innovation, and build a future where AI is owned by everyone and works for everyone.

BIBLIOGRAPHY

[1] Safe paths: A privacy-first approach to contact tracing. *MIT News*, 2022.

[2] Matthew Abueg, Robert Hinch, Neo Wu, Luyang Liu, William Probert, Austin Wu, Paul Eastham, Yusef Shafi, Matt Rosencrantz, Michael Dikovsky, et al. Modeling the effect of exposure notification and non-pharmaceutical interventions on covid-19 transmission in washington state. *NPJ digital medicine*, 4(1):49, 2021.

[3] Andrei Afonin and Sai Praneeth Karimireddy. Towards model agnostic federated learning using knowledge distillation. *arXiv preprint arXiv:2110.15210*, 2021.

[4] Andrei Afonin and Sai Praneeth Karimireddy. Towards model agnostic federated learning using knowledge distillation. In *International Conference on Learning Representations*, 2022.

[5] Muneeb Ahmad, Yousaf Ali Khan, Chonghui Jiang, Syed Jawad Haider Kazmi, and Syed Zaheer Abbas. The impact of covid-19 on unemployment rate: An intelligent based unemployment rate prediction in selected countries of europe. *International Journal of Finance & Economics*, 28(1):528–543, 2023.

[6] Samiul Alam, Luyang Liu, Ming Yan, and Mi Zhang. Fedrolex: Model-heterogeneous federated learning with rolling sub-model extraction. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[7] Laura Alessandretti. What human mobility data tell us about covid-19 spread. *Nature Reviews Physics*, 4(1):12–13, 2022.

[8] Alberto Aleta, David Martin-Corral, Ana Pastore y Piontti, Marco Ajelli, Maria Litvinova, Matteo Chinazzi, Natalie E Dean, M Elizabeth Halloran, Ira M Longini Jr, Stefano Merler, et al. Modelling the impact of testing, contact tracing and household quarantine on second waves of covid-19. *Nature Human Behaviour*, 4(9):964–971, 2020.

[9] MA Alsalem, AH Alamoodi, OS Albahri, KA Dawood, RT Mohammed, Alhamzah Alnoor, AA Zaidan, AS Albahri, BB Zaidan, FM Jumaah, et al.

Multi-criteria decision-making for coronavirus disease 2019 applications: a theoretical analysis review. *Artificial Intelligence Review*, 55(6):4979–5062, 2022.

[10] Daniel J Amit, Hanoch Gutfreund, and Haim Sompolinsky. Spin-glass models of neural networks. *Physical Review A*, 32(2):1007, 1985.

[11] Jimoh Amzat, Kafayat Aminu, Victor I Kolo, Ayodele A Akinyele, Janet A Ogundairo, and Maryann C Danjibo. Coronavirus outbreak in nigeria: Burden and socio-medical response during the first 100 days. *International Journal of Infectious Diseases*, 98:218–224, 2020.

[12] Federica Angeli and Andrea Montefusco. Sensemaking and learning during the covid-19 pandemic: A complex adaptive systems perspective on policy decision-making. *World Development*, 136:105106, 2020.

[13] Anzhelika Antipova. Analysis of the covid-19 impacts on employment and unemployment across the multi-dimensional social disadvantaged areas. *Social Sciences & Humanities Open*, 4(1):100224, 2021.

[14] Nikolaos Askitas, Konstantinos Tatsiramos, and Bertrand Verheyden. Estimating worldwide effects of non-pharmaceutical interventions on covid-19 incidence and population mobility patterns using a multiple-event study. *Scientific reports*, 11(1):1972, 2021.

[15] Joseph Aylett-Bullock, Carolina Cuesta-Lazaro, Arnau Quera-Bofarull, Miguel Icaza-Lizaola, Aidan Sedgewick, Henry Truong, Aoife Curran, Edward Elliott, Tristan Caulfield, Kevin Fong, et al. June: open-source individual-based epidemiology simulation. *Royal Society open science*, 8(7):210506, 2021.

[16] Peter Bandi, Oscar Geessink, Quirine Manson, Marcory Van Dijk, Maschenka Balkenhol, Meyke Hermsen, Babak Ehteshami Bejnordi, Byungjae Lee, Kyunghyun Paeng, Aoxiao Zhong, et al. From detection of individual metastases to classification of lymph node status at the patient level: the camelyon17 challenge. *IEEE Transactions on Medical Imaging*, 2018.

[17] Nicolas Banholzer, Eva Van Weenen, Adrian Lison, Alberto Cenedese, Arne Seeliger, Bernhard Kratzwald, Daniel Tschernutter, Joan Puig Salles, Pierluigi Bottrighi, Sonja Lehtinen, et al. Estimating the effects of non-pharmaceutical interventions on the number of new infections with covid-19 during the first epidemic wave. *PLoS one*, 16(6):e0252827, 2021.

[18] Alain Barrat, Ciro Cattuto, Mikko Kivelä, Sune Lehmann, and Jari Saramäki. Effect of manual and digital contact tracing on covid-19 outbreaks: A study on empirical contact data. *Journal of the Royal Society Interface*, 18(178):20201000, 2021.

[19] Aurélien Bellet, Rachid Guerraoui, Mahsa Taziki, and Marc Tommasi. Personalized and private peer-to-peer machine learning. In *International Conference on Artificial Intelligence and Statistics*, pages 473–481. PMLR, 2018.

[20] Frédéric Berdoz, Abhishek Singh, Martin Jaggi, and Ramesh Raskar. Scalable collaborative learning via representation sharing. *arXiv preprint arXiv:2211.10943*, 2022.

[21] Mauro Bisiacco and Gianluigi Pillonetto. Covid-19 epidemic control using short-term lockdowns for collective gain. *Annual Reviews in Control*, 52:573–586, 2021.

[22] Keith R Bisset, Jiangzhuo Chen, Xizhou Feng, VS Anil Kumar, and Madhav V Marathe. Epifast: a fast algorithm for large scale realistic epidemic simulations on distributed memory systems. In *Proceedings of the 23rd international conference on Supercomputing*, pages 430–439, 2009.

[23] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[24] Eric Bonabeau. Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the national academy of sciences*, 99(suppl_3):7280–7287, 2002.

[25] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1175–1191, 2017.

[26] Jan M Brauner, Sören Mindermann, Mrinank Sharma, David Johnston, John Salvatier, Tomáš Gavenčiak, Anna B Stephenson, Gavin Leech, George Altman, Vladimir Mikulik, et al. Inferring the effectiveness of government interventions against covid-19. *Science*, 371(6531):eabd9338, 2021.

[27] Christopher Briggs, Zhong Fan, and Peter Andras. Federated learning with hierarchical clustering of local updates to improve training on non-iid data. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–9. IEEE, 2020.

[28] Zhiqi Bu, Hua Wang, Zongyu Dai, and Qi Long. On the convergence and calibration of deep learning with differential privacy. *arXiv preprint arXiv:2106.07830*, 2021.

[29] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006.

[30] Sebastian Caldas, Jakub Konečny, H Brendan McMahan, and Ameet Talwalkar. Expanding the reach of federated learning by reducing client resource requirements. *arXiv preprint arXiv:1812.07210*, 2018.

[31] Xiaoyu Cao, Minghong Fang, Jia Liu, and Neil Zhenqiang Gong. Fltrust: Byzantine-robust federated learning via trust bootstrapping. *arXiv preprint arXiv:2012.13995*, 2020.

[32] Hongyan Chang, Virat Shejwalkar, Reza Shokri, and Amir Houmansadr. Cronus: Robust and heterogeneous collaborative learning with black-box knowledge transfer. *arXiv preprint arXiv:1912.11279*, 2019.

[33] Serina Chang, Emma Pierson, Pang Wei Koh, Jaline Gerardin, Beth Redbird, David Grusky, and Jure Leskovec. Mobility network models of covid-19 explain inequities and inform reopening. *Nature*, 589(7840):82–87, 2021.

[34] Hong-You Chen and Wei-Lun Chao. Fed{be}: Making bayesian model ensemble applicable to federated learning. In *International Conference on Learning Representations*, 2021.

[35] Hao-Yuan Cheng, Shu-Wan Jian, Ding-Ping Liu, Ta-Chou Ng, Wan-Ting Huang, Hsien-Ho Lin, et al. Contact tracing assessment of covid-19 transmission dynamics in taiwan and risk at different exposure periods before and after symptom onset. *JAMA internal medicine*, 180(9):1156–1163, 2020.

[36] Yizong Cheng. Mean shift, mode seeking, and clustering. *IEEE transactions on pattern analysis and machine intelligence*, 17(8):790–799, 1995.

[37] Yae Jee Cho, Jianyu Wang, and Gauri Joshi. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies, 2020.

[38] Ayush Chopra, Esma Gel, Jayakumar Subramanian, Balaji Krishnamurthy, Santiago Romero-Brufau, Kalyan S Pasupathy, Thomas C Kingsley, and Ramesh Raskar. Deepabm: scalable, efficient and differentiable agent-based simulations via graph neural networks. *arXiv preprint arXiv:2110.04421*, 2021.

[39] Mohammad Jabed Morshed Chowdhury, Md Sadek Ferdous, Kamanashis Biswas, Niaz Chowdhury, and Vallipuram Muthukkumarasamy. Covid-19 contact tracing: challenges and future directions. *Ieee Access*, 8:225703–225729, 2020.

[40] Emma L Davis, Tim CD Lucas, Anna Borlase, Timothy M Pollington, Sam Abbott, Diepreye Ayabina, Thomas Crellen, Joel Hellewell, Li Pi, et al. Contact tracing is an imperfect tool for controlling covid-19 transmission and relies on population adherence. *Nature communications*, 12(1):5412, 2021.

[41] Martijn De Vos, Sadegh Farhadkhani, Rachid Guerraoui, Anne-Marie Kermarrec, Rafael Pires, and Rishi Sharma. Epidemic learning: Boosting decentralized learning with randomized communication. *Advances in Neural Information Processing Systems*, 36, 2024.

[42] Enmao Diao, Jie Ding, and Vahid Tarokh. Hetero{fl}: Computation and communication efficient federated learning for heterogeneous clients. In *International Conference on Learning Representations*, 2021.

[43] Frank Dignum. *Social Simulation for a Crisis*. Springer, 2021.

[44] Nedialko B Dimitrov and Lauren Ancel Meyers. Mathematical approaches to infectious disease prediction and control. In *Risk and optimization in an uncertain world*, pages 1–25. INFORMS, 2010.

[45] Zhanwei Du, Abhishek Pandey, Yuan Bai, Meagan C Fitzpatrick, Matteo Chinazzi, Ana Pastore y Piontti, Michael Lachmann, Alessandro Vespignani, Benjamin J Cowling, Alison P Galvani, et al. Comparative cost-effectiveness of sars-cov-2 testing strategies in the usa: a modelling study. *The Lancet Public Health*, 6(3):e184–e191, 2021.

[46] Moming Duan, Duo Liu, Xinyuan Ji, Renping Liu, Liang Liang, Xianzhang Chen, and Yujuan Tan. Fedgroup: Efficient federated learning via decomposed similarity-based clustering. In *2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*, pages 228–237. IEEE, 2021.

[47] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. *Advances in Neural Information Processing Systems*, 33:3557–3568, 2020.

[48] Dongyang Fan, Celestine Mendler-Dünner, and Martin Jaggi. Collaborative learning via prediction consensus, 2023.

[49] Gongfan Fang, Kanya Mo, Xinchao Wang, Jie Song, Shitao Bei, Haofei Zhang, and Mingli Song. Up to 100x faster data-free knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 6597–6604, 2022.

[50] Seth Flaxman, Swapnil Mishra, Axel Gandy, H Juliette T Unwin, Thomas A Mellan, Helen Coupland, Charles Whittaker, Harrison Zhu, Tresnia Berah, Jeffrey W Eaton, et al. Estimating the effects of non-pharmaceutical interventions on covid-19 in europe. *Nature*, 584(7820):257–261, 2020.

[51] Brendan J Frey and Delbert Dueck. Clustering by passing messages between data points. *science*, 315(5814):972–976, 2007.

[52] Elpidio Maria Garzillo, Arcangelo Cioffi, Angela Carta, and Maria Grazia Lourdes Monaco. Returning to work after the covid-19 pandemic earthquake: a systematic review. *International journal of environmental research and public health*, 19(8):4538, 2022.

[53] Avishek Ghosh, Jichan Chung, Dong Yin, and Kannan Ramchandran. An efficient framework for clustered federated learning. *Advances in Neural Information Processing Systems*, 33:19586–19597, 2020.

[54] Jack Goetz and Ambuj Tewari. Federated learning via synthetic data, 2020.

[55] Elizabeth R Groff, Shane D Johnson, and Amy Thornton. State of the art in agent-based modeling of urban crime: An overview. *Journal of Quantitative Criminology*, 35:155–193, 2019.

[56] Dabo Guan, Daoping Wang, Stephane Hallegatte, Steven J Davis, Jingwen Huo, Shuping Li, Yangchun Bai, Tianyang Lei, Qianyu Xue, D'Maris Coffman, et al. Global supply-chain effects of covid-19 control measures. *Nature human behaviour*, 4(6):577–587, 2020.

[57] Gauri Gupta, Ritvik Kapila, Ayush Chopra, and Ramesh Raskar. First 100 days of pandemic; an interplay of pharmaceutical, behavioral and digital interventions – a study using agent based modeling, 2024.

[58] Gauri Gupta, Ritvik Kapila, Keshav Gupta, and Ramesh Raskar. Domain generalization in robust invariant representation. *arXiv preprint arXiv:2304.03431*, 2023.

[59] Gauri Gupta, Krithika Ramesh, Anwesh Bhattacharya, Divya Gupta, Rahul Sharma, Nishanth Chandran, and Rijurekha Sen. End-to-end privacy preserving training and inference for air pollution forecasting with data from rival fleets. *Cryptology ePrint Archive*, 2023.

[60] Otkrist Gupta and Ramesh Raskar. Distributed learning of deep neural network over multiple agents. *Journal of Network and Computer Applications*, 116:1–8, 2018.

[61] Niv Haim, Gal Vardi, Gilad Yehudai, Ohad Shamir, and Michal Irani. Reconstructing training data from trained neural networks. *Advances in Neural Information Processing Systems*, 35:22911–22924, 2022.

[62] Nina Haug, Lukas Geyrhofer, Alessandro Londei, Elma Dervic, Amélie Desvars-Larrive, Vittorio Loreto, Beate Pinior, Stefan Thurner, and Peter Klimek. Ranking the effectiveness of worldwide covid-19 government interventions. *Nature human behaviour*, 4(12):1303–1312, 2020.

[63] Kazunobu Hayakawa and Hiroshi Mukunoki. The impact of covid-19 on international trade: Evidence from the first shock. *Journal of the Japanese and International Economies*, 60:101135, 2021.

[64] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. arxiv 2015. *arXiv preprint arXiv:1512.03385*, 14, 2015.

[65] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[66] Lie He, An Bian, and Martin Jaggi. Cola: Decentralized linear learning. *Advances in Neural Information Processing Systems*, 31, 2018.

[67] Herbert W Hethcote. The mathematics of infectious diseases. *SIAM review*, 42(4):599–653, 2000.

[68] Robert Hinch, William JM Probert, Anel Nurtay, Michelle Kendall, Chris Wymant, Matthew Hall, Katrina Lythgoe, Ana Bulas Cruz, Lele Zhao, Andrea Stewart, et al. Openabm-covid19—an agent-based model for non-pharmaceutical interventions against covid-19 including contact tracing. *PLoS computational biology*, 17(7):e1009146, 2021.

[69] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

[70] Briland Hitaj, Giuseppe Ateniese, and Fernando Perez-Cruz. Deep models under the gan: information leakage from collaborative deep learning. In *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, pages 603–618, 2017.

[71] Samuel Horváth, Stefanos Laskaridis, Mario Almeida, Ilias Leontiadis, Stylianos Venieris, and Nicholas Donald Lane. FjORD: Fair and accurate federated learning under heterogeneous targets with ordered dropout. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[72] Kevin Hsieh, Amar Phanishayee, Onur Mutlu, and Phillip Gibbons. The non-iid data quagmire of decentralized machine learning. In *International Conference on Machine Learning*, pages 4387–4398. PMLR, 2020.

[73] Chenghao Hu, Jingyan Jiang, and Zhi Wang. Decentralized federated learning: A segmented gossip approach. *arXiv preprint arXiv:1908.07782*, 2019.

[74] Li Huang, Andrew L Shea, Huining Qian, Aditya Masurkar, Hao Deng, and Dianbo Liu. Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records. *Journal of biomedical informatics*, 99:103291, 2019.

[75] JPMorgan Chase Institute. The first 100 days and beyond.

[76] Márk Jelasity, Spyros Voulgaris, Rachid Guerraoui, Anne-Marie Kermarrec, and Maarten van Steen. Gossip-based peer sampling. *ACM Trans. Comput. Syst.*, 25(3):8–es, 8 2007.

[77] Eunjeong Jeong, Seungeun Oh, Hyesung Kim, Jihong Park, Mehdi Bennis, and Seong-Lyun Kim. Communication-efficient on-device machine learning: Federated distillation and augmentation under non-iid private data. *arXiv preprint arXiv:1811.11479*, 2018.

[78] Yihan Jiang, Jakub Konečnỳ, Keith Rush, and Sreeram Kannan. Improving federated learning personalization via model agnostic meta learning. *arXiv preprint arXiv:1909.12488*, 2019.

[79] James Jordon, Jinsung Yoon, and Mihaela Van Der Schaar. Pate-gan: Generating synthetic data with differential privacy guarantees. In *International conference on learning representations*, 2018.

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

[81] Shivam Kalra, Junfeng Wen, Jesse C Cresswell, Maksims Volkovs, and Hamid R Tizhoosh. Proxyfl: decentralized federated learning through proxy model sharing. *arXiv preprint arXiv:2111.11343*, 2021.

[82] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pages 5132–5143. PMLR, 2020.

[83] Jennifer Kates, Cynthia Cox, and Josh Michaud. How much could covid-19 vaccines cost the us after commercialization. *KFF. March*, 10, 2023.

[84] Matt J Keeling and Ken TD Eames. Networks and epidemic models. *Journal of the royal society interface*, 2(4):295–307, 2005.

[85] Cliff C Kerr, Robyn M Stuart, Dina Mistry, Romesh G Abeysuriya, Katherine Rosenfeld, Gregory R Hart, Rafael C Núñez, Jamie A Cohen, Prashanth Selvaraj, Brittany Hagedorn, et al. Covasim: an agent-based model of covid-19 dynamics and interventions. *PLOS Computational Biology*, 17(7):e1009149, 2021.

[86] Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. Tighter theory for local sgd on identical and heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*, pages 4519–4529. PMLR, 2020.

[87] Jakub Konečnỳ, H Brendan McMahan, Daniel Ramage, and Peter Richtárik. Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527*, 2016.

[88] Lingjing Kong, Tao Lin, Anastasia Koloskova, Martin Jaggi, and Sebastian Stich. Consensus control for decentralized deep learning. In *International Conference on Machine Learning*, pages 5686–5696. PMLR, 2021.

[89] Polychronis Kostoulas, Paolo Eusebi, and Sonja Hartnack. Diagnostic accuracy estimates for covid-19 real-time polymerase chain reaction and lateral flow immunoassay tests with bayesian latent-class models. *American journal of epidemiology*, 190(8):1689–1695, 2021.

[90] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

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

[92] Daniel B Larremore, Bryan Wilder, Evan Lester, Soraya Shehata, James M Burke, James A Hay, Milind Tambe, Michael J Mina, and Roy Parker. Test sensitivity is secondary to frequency and turnaround time for covid-19 screening. *Science advances*, 7(1):eabd5393, 2021.

[93] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[94] Gihun Lee, Minchan Jeong, Yongjin Shin, Sangmin Bae, and Se-Young Yun. Preservation of the global knowledge by not-true distillation in federated learning. In *Advances in Neural Information Processing Systems*, 2021.

[95] Rhonda K Lewis, Pamela P Martin, and Bianca L Guzman. Covid-19 and vulnerable populations, 2022.

[96] Chengxi Li, Gang Li, and Pramod K Varshney. Decentralized federated learning via mutual knowledge transfer. *IEEE Internet of Things Journal*, 9(2):1136–1147, 2021.

[97] Jinfeng Li and Xinyi Guo. Covid-19 contact-tracing apps: A survey on the global deployment and challenges. *arXiv preprint arXiv:2005.03599*, 2020.

[98] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10713–10722, 2021.

[99] Shuangtong Li, Tianyi Zhou, Xinmei Tian, and Dacheng Tao. Learning to collaborate in decentralized learning of personalized models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9766–9775, 2022.

[100] Shuangtong Li, Tianyi Zhou, Xinmei Tian, and Dacheng Tao. Learning to collaborate in decentralized learning of personalized models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9756–9765, 2022.

[101] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020.

[102] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. *arXiv preprint arXiv:1907.02189*, 2019.

[103] Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. Fedbn: Federated learning on non-iid features via local batch normalization. *arXiv preprint arXiv:2102.07623*, 2021.

[104] Zexi Li, Jiaxun Lu, Shuang Luo, Didi Zhu, Yunfeng Shao, Yinchuan Li, Zhimeng Zhang, Yongheng Wang, and Chao Wu. Towards effective clustered federated learning: A peer-to-peer framework with adaptive neighbor matching. *IEEE Transactions on Big Data*, 2022.

[105] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. *Advances in Neural Information Processing Systems*, 33:2351–2363, 2020.

[106] Samantha Liss and Nami Sumida. How hospital capacity varies dramatically across the country.

[107] Xuepeng Liu, Emanuel Ornelas, and Huimin Shi. The trade impact of the covid-19 pandemic. *The World Economy*, 45(12):3751–3779, 2022.

[108] Guodong Long, Ming Xie, Tao Shen, Tianyi Zhou, Xianzhi Wang, and Jing Jiang. Multi-center federated learning: clients clustering for better personalization. *World Wide Web*, 26(1):481–500, 2023.

[109] Ghazi M Magableh. Supply chains and the covid-19 pandemic: A comprehensive framework. *European Management Review*, 18(3):363–382, 2021.

[110] Madhav Marathe and Anil Kumar S Vullikanti. Computational epidemiology. *Communications of the ACM*, 56(7):88–96, 2013.

[111] Martin McKee and David Stuckler. If the world fails to protect the economy, covid-19 will damage health not just now but also in the future. *Nature Medicine*, 26(5):640–642, 2020.

[112] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[113] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017.

[114] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data, 2023.

[115] H Brendan McMahan, Eider Moore, Daniel Ramage, and Blaise Agüera y Arcas. Federated learning of deep networks using model averaging. *arXiv preprint arXiv:1602.05629*, 2:2, 2016.

[116] Yiqun Mei, Pengfei Guo, Mo Zhou, and Vishal Patel. Resource-adaptive federated learning with all-in-one neural composition. In *Advances in Neural Information Processing Systems*, 2022.

[117] Youssef Miyah, Mohammed Benjelloun, Sanae Lairini, Anissa Lahrichi, et al. Covid-19 impact on public health, environment, human psychology, global socioeconomy, and education. *The Scientific World Journal*, 2022, 2022.

[118] Alessio Mora, Irene Tenison, Paolo Bellavista, and Irina Rish. Knowledge distillation for federated learning: a practical guide. *arXiv preprint arXiv:2211.04742*, 2022.

[119] Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks. 2015.

[120] Saira Naseer, Sidra Khalid, Summaira Parveen, Kashif Abbass, Huaming Song, and Monica Violeta Achim. Covid-19 outbreak: Impact on global economy. *Frontiers in public health*, 10:1009393, 2023.

[121] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[122] Takayuki Nishio and Ryo Yonetani. Client selection for federated learning with heterogeneous resources in mobile edge. In *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pages 1–7, 2019.

[123] Kerri-Ann Norton, Chang Gong, Samira Jamalian, and Aleksander S Popel. Multiscale agent-based and hybrid modeling of the tumor immune microenvironment. *Processes*, 7(1):37, 2019.

[124] Anna Odone, Daria Bucci, Roberto Croci, Matteo Riccò, Paola Affanni, and Carlo Signorelli. Vaccine hesitancy in covid-19 times. an update from italy before flu season starts. *Acta Bio Medica: Atenei Parmensis*, 91(3):e2020031, 2020.

[125] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2(11):e7, 2017.

[126] Noa Onoszko, Gustav Karlsson, Olof Mogren, and Edvin Listo Zec. Decentralized federated learning of deep neural networks on non-iid data. *arXiv preprint arXiv:2107.08517*, 2021.

[127] Tamer Oraby, Michael G Tyshenko, Jose Campo Maldonado, Kristina Vatcheva, Susie Elsaadany, Walid Q Alali, Joseph C Longenecker, and Mustafa Al-Zoughool. Modeling the effect of lockdown timing as a covid-19 control measure in countries with differing social contacts. *Scientific reports*, 11(1):3354, 2021.

[128] Abhishek Pandey, Pratha Sah, Seyed M Moghadas, Sandip Mandal, Sandip Banerjee, Peter J Hotez, and Alison P Galvani. Challenges facing covid-19 vaccination in india: Lessons from the initial vaccine rollout. *Journal of Global Health*, 11, 2021.

[129] Christodoulos Pappas, Dimitris Chatzopoulos, Spyros Lalis, and Manolis Vavalis. Ipls: A framework for decentralized federated learning. In *2021 IFIP Networking Conference (IFIP Networking)*, pages 1–6. IEEE, 2021.

[130] Sanjoy Kumar Paul, Priyabrata Chowdhury, Md Abdul Moktadir, and Kwok Hung Lau. Supply chain recovery challenges in the wake of covid-19 pandemic. *Journal of Business Research*, 136:316–329, 2021.

[131] Lorenzo Pellis, Frank Ball, Shweta Bansal, Ken Eames, Thomas House, Valerie Isham, and Pieter Trapman. Eight challenges for network epidemic models. *Epidemics*, 10:58–62, 2015.

[132] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1406–1415, 2019.

[133] Antonio F Peralta, János Kertész, and Gerardo Iñiguez. Opinion dynamics in social networks: From models to data. *arXiv preprint arXiv:2201.01322*, 2022.

[134] Maarten G Poirot, Praneeth Vepakomma, Ken Chang, Jayashree Kalpathy-Cramer, Rajiv Gupta, and Ramesh Raskar. Split learning for collaborative deep learning in healthcare. *arXiv preprint arXiv:1912.12115*, 2019.

[135] Yuben Qu, Haipeng Dai, Yan Zhuang, Jiafa Chen, Chao Dong, Fan Wu, and Song Guo. Decentralized federated learning for uav networks: Architecture, challenges, and opportunities. *IEEE Network*, 35(6):156–162, 2021.

[136] Ramesh Raskar, Isabel Schunemann, Rachel Barbar, Kristen Vilcans, Jim Gray, Praneeth Vepakomma, Suraj Kapa, Andrea Nuzzo, Rajiv Gupta, Alex Berke, et al. Apps gone rogue: Maintaining personal privacy in an epidemic. *arXiv preprint arXiv:2003.08567*, 2020.

[137] Mohammad Rasouli, Tao Sun, and Ram Rajagopal. Fedgan: Federated generative adversarial networks for distributed data, 2020.

[138] Joren Raymenants, Caspar Geenen, Jonathan Thibaut, Klaas Nelissen, Sarah Gorissen, and Emmanuel Andre. Empirical evidence on the efficiency of backward contact tracing in covid-19. *Nature Communications*, 13(1):4750, 2022.

[139] Theresa Reiker, Monica Golumbeanu, Andrew Shattock, Lydia Burgert, Thomas A Smith, Sarah Filippi, Ewan Cameron, and Melissa A Penny. Machine learning approaches to calibrate individual-based infectious disease models. *medRxiv*, pages 2021–01, 2021.

[140] Santiago Romero-Brufau, Ayush Chopra, Alex J Ryu, Esma Gel, Ramesh Raskar, Walter Kremers, Karen Anderson, Jayakumar Subramanian, Balaji Krishnamurthy, Abhishek Singh, et al. The public health impact of delaying a second dose of the bnt162b2 or mrna-1273 covid-19 vaccine. *medRxiv*, pages 2021–02, 2021.

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

[142] Felix Sattler, Tim Korjakow, Roman Rischke, and Wojciech Samek. Fedaux: Leveraging unlabeled auxiliary data in federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[143] Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE transactions on neural networks and learning systems*, 32(8):3710–3722, 2020.

[144] Shai Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012.

[145] Yunfeng Shang, Haiwei Li, and Ren Zhang. Effects of pandemic outbreak on economies: evidence from business history context. *Frontiers in public health*, 9:146, 2021.

[146] Viktoriia Shubina, Sylvia Holcer, Michael Gould, and Elena Simona Lohan. Survey of decentralized solutions with mobile devices for user location tracking, proximity detection, and contact tracing in the covid-19 era. *Data*, 5(4):87, 2020.

[147] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[148] Abhishek Singh, Gauri Gupta, Ritvik Kapila, Yichuan Shi, Alex Dang, Sheshank Shankar, Mohammed Ehab, and Ramesh Raskar. Codream: Exchanging dreams instead of models for federated aggregation with heterogeneous models. *arXiv preprint arXiv:2402.15968*, 2024.

[149] Abhishek Singh, Gauri Gupta, Charles Lu, Yogesh Koirala, Sheshank Shankar, Mohammed Ehab, and Ramesh Raskar. Co-dream: Collaborative data synthesis with decentralized models. In *ICML Workshop on Localized Learning (LLW)*, 2023.

[150] Rui Song, Dai Liu, Dave Zhenyu Chen, Andreas Festag, Carsten Trinitis, Martin Schulz, and Alois Knoll. Federated learning via decentralized dataset distillation in resource-constrained edge environments. *arXiv preprint arXiv:2208.11311*, 2022.

[151] Kanta Subbarao. The success of sars-cov-2 vaccines and challenges ahead. *Cell host & microbe*, 29(7):1111–1123, 2021.

[152] Yi Sui, Junfeng Wen, Yenson Lau, Brendan Leigh Ross, and Jesse C Cresswell. Find your friends: Personalized federated learning with the right collaborators. *arXiv preprint arXiv:2210.06597*, 2022.

[153] Derrick Y Tam, David Naimark, Madhu K Natarajan, Graham Woodward, Garth Oakes, Mirna Rahal, Kali Barrett, Yasin A Khan, Raphael Ximenes, Stephen Mac, et al. The use of decision modelling to inform timely policy decisions on cardiac resource capacity during the covid-19 pandemic. *Canadian Journal of Cardiology*, 36(8):1308–1312, 2020.

[154] Zhenheng Tang, Shaohuai Shi, Bo Li, and Xiaowen Chu. Gossipfl: A decentralized federated learning framework with sparsified and adaptive communication. *IEEE Transactions on Parallel and Distributed Systems*, 34(3):909–922, 2022.

[155] Gerard J Tellis, Nitish Sood, and Ashish Sood. Price of delay in covid-19 lockdowns: Delays spike total cases, natural experiments reveal. *USC Marshall School of Business Research Paper*, 2020.

[156] Stephen Thurner, Rudolf Hanel, and Peter Klimek. *Introduction to the Theory of Complex Systems*. Oxford University Press, 2019.

[157] Aleksei Triastcyn, Matthias Reisser, and Christos Louizos. Decentralized learning with random walks and communication-efficient adaptive optimization. In *Workshop on Federated Learning: Recent Advances and New Challenges (in Conjunction with NeurIPS 2022)*, 2022.

[158] Jasper Verschuur, Elco E Koks, and Jim W Hall. Global economic impacts of covid-19 lockdown measures stand out in high-frequency shipping data. *PloS one*, 16(4):e0248818, 2021.

[159] Jasper Verschuur, Elco E Koks, and Jim W Hall. Observed impacts of the covid-19 pandemic on global trade. *Nature Human Behaviour*, 5(3):305–307, 2021.

[160] Lander Willem, Steven Abrams, Pieter JK Libin, Pietro Coletti, Elise Kuylen, Oana Petrof, Signe Møgelmose, James Wambua, Sereina A Herzog, Christel Faes, et al. The impact of contact tracing and household bubbles on deconfinement strategies for covid-19. *Nature communications*, 12(1):1524, 2021.

[161] Tobias Wink and Zoltan Nochta. An approach for peer-to-peer federated learning. In *2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*, pages 150–157. IEEE, 2021.

[162] Lars Wulfert, Navidreza Asadi, Wen-Yu Chung, Christian Wiede, and Anton Grabmaier. Adaptive decentralized federated gossip learning for resource-constrained iot devices. In *Proceedings of the 4th International Workshop on Distributed Machine Learning*, pages 27–33, 2023.

[163] Y. Xiao, Y. Zhang, D. Geng, D. Cong, K. Shi, and R. J. Knapp. Challenges of drug development during the covid-19 pandemic: key considerations for clinical trial designs. *British Journal of Clinical Pharmacology*, 87:2170–2185, 2020.

[164] Bangzhou Xin, Wei Yang, Yangyang Geng, Sheng Chen, Shaowei Wang, and Liusheng Huang. Private fl-gan: Differential privacy synthetic data generation based on federated learning. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2927–2931, 2020.

[165] Woraphon Yamaka, Siritaya Lomwanawong, Darin Magel, and Paravee Maneejuk. Analysis of the lockdown effects on the economy, environment, and covid-19 spread: Lesson learnt from a global pandemic in 2020. *International Journal of Environmental Research and Public Health*, 19(19):12868, 2022.

[166] Longqi Yang, David Holtz, Sonia Jaffe, Siddharth Suri, Shilpi Sinha, Jeffrey Weston, Connor Joyce, Neha Shah, Kevin Sherman, Brent Hecht, et al. The effects of remote work on collaboration among information workers. *Nature human behaviour*, 6(1):43–54, 2022.

[167] Moshe Yanovskiy and Yehoshua Socol. Are lockdowns effective in managing pandemics? *International Journal of Environmental Research and Public Health*, 19:9295, 07 2022.

[168] Hongxu Yin, Pavlo Molchanov, Jose M Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free

knowledge transfer via deepinversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8715–8724, 2020.

[169] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[170] Jie Zhang, Chen Chen, Bo Li, Lingjuan Lyu, Shuang Wu, Shouhong Ding, Chunhua Shen, and Chao Wu. Dense: Data-free one-shot federated learning. *Advances in Neural Information Processing Systems*, 35:21414–21428, 2022.

[171] Lin Zhang, Li Shen, Liang Ding, Dacheng Tao, and Ling-Yu Duan. Fine-tuning global model via data-free knowledge distillation for non-iid federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10174–10183, 2022.

[172] Jiaqi Zhao, Hui Zhu, Fengwei Wang, Rongxing Lu, Zhe Liu, and Hui Li. Pvd-fl: A privacy-preserving and verifiable decentralized federated learning framework. *IEEE Transactions on Information Forensics and Security*, 17:2059–2073, 2022.

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

[174] Stephan Zheng, Alexander Trott, Sunil Srinivasa, David C Parkes, and Richard Socher. The ai economist: Taxation policy design via two-level deep multiagent reinforcement learning. *Science advances*, 8(18):eabk2607, 2022.

[175] Yanlin Zhou, George Pu, Xiyao Ma, Xiaolin Li, and Dapeng Wu. Distilled one-shot federated learning. *arXiv preprint arXiv:2009.07999*, 2020.

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

[177] Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. Data-free knowledge distillation for heterogeneous federated learning. In *International Conference on Machine Learning*, pages 12878–12889. PMLR, 2021.