Bulletin of the Technical Community on

Data Engineering

Dec 2025 Vol. 49 No. 4 IEEE Computer Society

Letters

Special Issue on Privacy-Preserving Technologies for Data Engineering/-Science in the Age of AI

Conference and Journal Notices

The TC on Data Engineering

Membership in the TC on Data Engineering is open to all current members of the IEEE Computer Society who are interested in database systems. The TCDE web page is http://tab.computer.org/tcde/index.html.

The Data Engineering Bulletin

The Bulletin of the Technical Community on Data Engineering is published quarterly and is distributed to all TC members. Its scope includes the design, implementation, modeling, theory and application of database systems and technology.

Letters, conference information, and news should be sent to the Editor-in-Chief. Papers for each issue are solicited by and should be sent to Associate Editor responsible for the issue.

Opinions expressed in contributions are those of the authors and do not necessarily reflect the positions of TC on Data Engineering, IEEE Computer Society, or authors' organizations.

The Data Engineering Bulletin web site is at http://tab.computer.org/tcde/bull_about.html.

Letter from the Editor-in-Chief

Privacy in AI is having its "everything, everywhere" moment. It shows up in procurement checklists, product reviews, incident postmortems, and policy debates, and it is often discussed with the urgency of an existential risk and the practicality of a compliance chore. That contradiction is not a sign of confusion so much as it is a sign that we are living through a transition. The tools and institutions that once made privacy feel manageable were built for a world in which data use looked like a bounded transaction: a service collected information, stored it, and used it for an identifiable purpose. In the AI era, data does not merely support a service; it becomes part of a capability that can be reused, repurposed, and redeployed, long after the original context has faded.

That is why it is becoming harder to believe in privacy as a moment—a notice, a click, a checkbox, a clause. When people are asked to make meaningful decisions through low-signal interfaces, under time pressure, in environments engineered for throughput rather than deliberation, "choice" starts to resemble a ritual rather than an exercise of agency. Even when consent is present, it rarely carries predictive power about what will happen next, because modern AI systems blur the lines between primary use and secondary use. Data flows into pipelines; pipelines feed models; models are integrated into products; products generate logs; logs become training material; and vendors, tools, and internal services become links in a chain that few individuals can see, much less control.

The papers in this issue suggest a sturdier thesis: privacy in AI will be won or lost by infrastructure. By "infrastructure," I mean defaults that do not require constant vigilance from the individual, controls that remain meaningful as systems scale and get outsourced, and accountability that follows data and models across their lifecycle rather than concentrating at the moment of collection. The old world asked whether privacy disclosures were complete and whether consent was obtained. The new world has to ask whether the system is constructed so that restraint is the normal path, misuse is harder than proper use, and the consequences of failure are detectable before they become irreversible.

One way to see the limits of the older framing is to notice how often privacy debates get stuck on a narrow question: did the model memorize sensitive training examples, and can that information be extracted? That question matters, but it is no longer the whole story. The paper *Beyond Data Privacy: New Privacy Risks for Large Language Models* by Du et al, offers a useful reframing by widening the threat model. In practice, privacy risk emerges not only from what was in the training set, but from the way models are deployed, connected, and operated. Prompts and outputs may be logged. Retrieval systems may introduce sensitive context. Tool integrations and plugins may route data to third parties. An application can be carefully designed in one layer and quietly undermine privacy in another. The most unsettling element of this expanded frame is that privacy harm can arise even when nothing is "leaked" in the classic sense, because models can be used to infer, profile, and amplify invasive behavior. In other words, the privacy story shifts from a single vulnerability to an exposure surface spanning the full stack.

Once you accept that, another gap in conventional thinking becomes obvious. We often talk about privacy as if it were mainly a question of who can see the data. Yet modern AI systems increasingly depend on outsourced infrastructure, external platforms, and complex vendor ecosystems, and that raises a second question that is just as fundamental: can we trust what computation happened, where, and under what constraints? The paper *Data Privacy and Computation Integrity in Machine Learning Scenarios* insists that privacy promises become brittle when integrity is treated as an afterthought. If training, storage, or evaluation is delegated to an environment you do not fully control, the question is not merely whether the data was protected in transit or at rest, but whether the computation was performed correctly, completely, and in a way that can be audited. In real deployments, privacy failures are often caused less by cinematic adversaries and more by mundane realities: misconfiguration, partial failures, permissive logging, quiet pipeline drift, and vendor-side changes that are hard to observe from

the outside. Integrity mechanisms, verification patterns, and auditable controls do not make systems perfect, but they shift privacy from hope to evidence. Privacy without integrity is a promise that survives only in the best-case scenario.

If infrastructure is the right lens, then privacy by default must also be understood as a systems design problem rather than an algorithmic checkbox. Federated learning, for instance, is frequently introduced as a simple idea: keep data local and ship updates. But the paper *Privacy-Preserving Federated Large Language Models* by Xu et al is valuable precisely because it refuses to sell simplicity. At the scale of large language models, the tensions among privacy, utility, and efficiency are not abstract. If you push too hard on privacy, performance may degrade or training may become unstable under heterogeneous data. If you prioritize utility, you may invite leakage through updates. If you optimize for efficiency, you may weaken the very redundancy and aggregation that make privacy protections workable. The paper reads as an argument for intellectual honesty: privacy engineering is tradeoff engineering. The goal is not to deny tradeoffs, but to make them explicit, measurable, and governable.

That idea – governable tradeoffs – is where "infrastructure" stops being a metaphor. Governance requires enforcement. Enforcement requires permissioning. And permissioning, in large-scale AI services, is inseparable from identity and access control. It is easy to categorize managed identities and authorization systems as "security plumbing" and treat them as adjacent to privacy. In practice, they are privacy. Access drift is a privacy leak. Ambiguous service identity is a privacy risk. Mis-scoped tokens are privacy incidents waiting to happen. The paper *Hyper-Scale Managed Identities and Access Control*, by Alagenchev et al, underscores how much modern systems depend on reliable identity, scalable authorization, and strong assurance about who is calling what. In the AI era, where models become shared services consumed by many clients, privacy cannot be protected if the system cannot express and enforce who may use which model, with what data, under what constraints. The boundary between privacy and security is not disappearing, but it is becoming increasingly artificial at the level where systems actually fail.

Of course, infrastructure is not only about locks; it is also about measurement. What can be observed can be governed, and what cannot be observed becomes a matter of trust. Yet measurement is only as useful as the reality it reflects. Formal guarantees can be technically correct while socially misleading if they are not aligned with how risk manifests in practice. Two contributions in this issue, in different ways, press toward the same lesson: privacy claims must match real units of harm and real deployment contexts. Mahloujifar et al, in *Optimal Group Privacy for DP-SGD*, push beyond the idea that privacy is only about a single record. Many harms accrue at the level of groups, cohorts, households, or communities; user participation itself can be sensitive; and correlated records can amplify exposure. Meanwhile, Mokhtari et al, in *Rethinking Benchmarks for Differentially Private Image Classification*, challenge the tendency to benchmark methods in settings that are convenient rather than representative of privacy-critical domains. Benchmarks shape what we optimize, what we deploy, and what we celebrate. If they fail to reflect the stakes and constraints of real-world settings, we risk building methods that look strong on paper while leaving the highest-risk contexts underserved.

Stepping back, the connective tissue across these papers is not a single mechanism, but a shared insistence that privacy must move upstream and become operational. The path forward is less about asking individuals to shoulder the burden of prediction and vigilance, and more about building systems in which restraint is the default behavior of the pipeline. It is less about treating compliance as a final gate, and more about structuring interdisciplinary ownership so that privacy requirements shape architecture choices early and remain meaningful as systems evolve. It is less about a narrow focus on training data and more about end-to-end thinking that includes deployment realities, integrations, logging, and misuse. It is less about privacy as a promise and more about privacy as an observable property supported by integrity, identity, and measurement.

That framing also clarifies what progress should look like. Progress is a world in which collecting less

is not an act of heroism, but the easiest path. It is a world in which provenance, retention, and purpose are not buried in policy documents, but expressed in system behavior and enforced by controls. It is a world in which outsourcing does not dissolve responsibility, because privacy and integrity constraints survive across vendor boundaries. It is a world in which "privacy-preserving" is not a marketing adjective, but a claim that can be scrutinized with shared benchmarks, realistic assumptions, and guarantees that align with how harm is experienced.

The call to action, then, is not to wait for a single breakthrough. It is to build the missing privacy supply chain. Researchers can prioritize work that is system-aware and deployment-relevant, connecting formal guarantees to operational realities like identity, access control, logging, and outsourcing. Practitioners can stop treating privacy as something to be validated late and instead embed it into data pipelines, model lifecycle management, and continuous monitoring, with the same seriousness we apply to reliability and safety. Institutions can push accountability upstream, encouraging norms that make provenance, retention, and permissioning central rather than peripheral.

If we do that work, privacy in the AI era stops being a rear-guard action. It becomes a discipline of construction: designing systems that can learn and serve at scale without turning personal data into a permanent liability. The papers in this issue do not claim that future has arrived, but they make it easier to see what it would take to build it. That is the real opportunity before us, and it is also the test that will define what comes next.

Haixun Wang
EvenUp

Letter from the Special Issue Editor

The rapid advancement of Artificial Intelligence (AI) across diverse fields like healthcare, finance, transportation, and entertainment hinges heavily on data engineering/science. This discipline plays a crucial role in developing novel methods for collecting, storing, processing, and analyzing vast amounts of data, often containing sensitive personal information. However, traditional approaches such as access control and anonymization face significant challenges in handling sensitive data in the age of AI. This special issue contributes to the critical discussion on privacy-preserving technologies for data engineering/science in the age of AI and features the following seven articles.

1. *Privacy, Policy, and Compliance in Yet Another 'Age': The Necessity of Interdisciplinary Collaboration for Artificial Intelligence Applications by Bailey Kacsmar.* This article focuses on the existing domains relevant to AI governance, such as privacy, policy, and compliance, and discusses the challenges in different stages of AI and solutions that have been investigated.

2. *Data Privacy and Computation Integrity in Machine Learning Scenarios: Some Issues and Approaches by De Capitani di Vimercati et al.* While the previous paper shows the importance of AI in different fields, this paper focuses on privacy and computation integrity issues arising when data and machine learning models or tasks are shared with external parties. They develop a target-aware data anonymization technique and a solution for generating a privacy-friendly classifier that requires neither sensitive information nor information correlated with it for training a high-accuracy classifier.

3. *Beyond Data Privacy: New Privacy Risks for Large Language Models by Du et al.* The previous two papers set up the background for LLM in terms of their usages and privacy issues of ML. This paper introduces critical privacy vulnerabilities, particularly during their deployment and integration into applications. In the paper, rather than focusing on training-phase data privacy, as most existing work does, the authors explore new risks, including data leakage and malicious exfiltration enabled by LLM' autonomous capabilities. To combat these threats, the paper provides a systematic analysis of these emerging risks and calls for the development of broader, more robust defense strategies.

4. *Privacy-Preserving Federated Large Language Models: Techniques and Trade-offs by Xu et al.* Moving from the centralized LLM, this paper focuses on three specific challenges when combining Federated Learning with LLM, that are maintaining model utility amid statistical and system heterogeneity; ensuring efficiency by alleviating severe communication and computation bottlenecks; and safeguarding privacy against powerful attacks.

   The next three papers focus on specific techniques and advances for addressing various privacy challenges, including access control and differential privacy.

5. *Hyper-Scale Managed Identities and Access Control by Alagenchev et al.* This paper further discusses access control on large-scale settings and focuses on the limitations, such as scalability, adaptability, and fine-grained, context-aware control, of existing access control techniques in dynamic, distributed service-to-service architectures and presents the design and architecture of Hyper-Scale Managed Identities (HSMIs) and their integration with decentralized access control policies.

6. *Optimal Group Privacy for DP-SGD by Mahloujifar et al.* This paper addresses the long-standing challenge of tight privacy accounting for group privacy in Differentially Private Stochastic Gradient Descent (DP-SGD). While individual privacy bounds are well-established, group privacy—which

protects sets of individuals rather than just one—has historically lacked precise bounds. The researchers introduce a novel technique using "dominating pairs of distributions" to achieve tighter group privacy guarantees. They also find that sub-sampling heavily influences group privacy, meaning that two models with identical individual privacy parameters can have vastly different levels of protection for groups depending on their specific hyperparameters.

7. *Rethinking Benchmarks for Differentially Private Image Classification by Mokhtari et al.* This paper focuses on developing a benchmark to evaluate techniques for differentially private machine learning in a variety of settings, including with and without additional data, in convex settings, and on a variety of qualitatively different datasets; and creates a publicly available leaderboard for the community to track progress in differentially private machine learning.

We would like to thank all the authors for their valuable contributions. We also thank Haixun Wang for the opportunity to put together this special issue, and Jieming Shi for his help in its publication.

Shantanu Sharma[1], Xi He[2]
[1] New Jersey Institute of Technology, [2] University of Waterloo

Privacy, Policy, and Compliance in Yet Another 'Age': The Necessity of Interdisciplinary Collaboration for Artificial Intelligence Applications

Bailey Kacsmar
University of Alberta
kacsmar@ualberta.ca

Abstract

The advent of artificial intelligence (AI) interfaces that are accessible to the general populace, in the form of large language model (LLM) chat bots, brought the field of AI to the forefront of conversations on technology laws. These highly visible LLMs train on massive corpora of data from across the internet, amplifying questions as to the privacy implications of AI and the challenges for transparency, explainability, and auditing. While the recent proliferation of LLMs has increased awareness of the use and misuse of far-reaching technologies, governance and privacy for AI must not ignore past lessons on regulating technologies. With this work, we emphasize the existing domains relevant to AI governance and argue that while there are challenges and nuances for privacy, policy, and compliance in AI, it is generally still automation. Rather than focusing on specific algorithms, data set sizes, or parameter settings, we put forth an organization of the different life-stages that an AI deployment goes through as well as the lessons from relevant sub-fields. We highlight how focusing on the impact and consequences, both intentional and unintentional, provides a better grounding for domain experts and provides the path for connecting technical communities and governance communities to make effective regulations and policies for AI.

1 Introduction

The "Internet Age", "Digital Age", or "Information Age", depending on your terminology of choice, refers to the time period of the mid-twentieth century and is largely associated with the impact of information technology and computers on society [25–27]. There is no clear demarcation between the information age and the recently termed "age of AI" corresponding to the rise in prevalence of artificial intelligence (AI). However, we can largely attribute the prevalence of the phrase, "age of AI", to the significant shift in a specific sub-field AI, that of language modeling, between the years of 2019 and 2023. In particular, the release of a chat interface from OpenAI in 2022 accelerated public facing systems explicitly marketed as AI [97], such as AI chat bots incorporated into search engines [72] and social media platforms [15] as well as chat bot specific platforms [38, 57, 114, 149].

Legal questions and consequences in regards to AI are both currently increasing at a notable rate. For instance, proponent organizations for AI use in high-risk settings, like cognitive therapy, have persisted despite persistent and tragic consequences [73, 89, 103]. Efforts towards adoption in high risk domains have given rise to legal questions and legal actions. For example, there are ambiguities in regards to whether therapy-style exchanges between a person and a chatbot will be held to the same standards of protection requirements that an actual cognitive therapist would be held to. Despite such legal questions, chatbots have already been put forth for use as a therapist [128]. There is also already a record breaking large class action lawsuit for copyright violations against organizations which develop

and release generative AI [10]. Further, AI apps have been displaying user interactions with the chatbot that contained personal information such as both medical and legal topics [124]. Recognition of public concern that has arisen from these instances and others has already influenced regulators in regards to broader data protection regulations and privacy, such as in the case of the UK Commissioner's Office guidance about data collection and data transparency [17].

As public access to tools marketed as 'AI' increased rapidly, concerns about potential harms have also increased, as seen in emergent regulations as well as in media articles and statements from advocacy groups [13, 49, 50, 94, 104, 118]. The EU AI Act, which went into force in 2024, with full effect in 2027, was proposed in 2021 [49]. In mid-2025 a collection of organizations brought forth "The People's AI Action Plan" as a counter view to the Trump administration's stance on AI policy [118]. The people's plan emphasizes that AI should be guided and regulated in ways that center social good, public well-being, accountability, equality, and environmental preservation. Also in 2025, the G7 leaders released a statement on AI in which they set out a human-centered view which advocates for many of the same things of the people's plan, such as considerations for energy demands, impact on majority world countries, as well as impact on the workforce [54, 93]. As AI increases in use, mitigating harms associated with it, specifically bias, surveillance, job loss, and misinformation require greater efforts from appropriate governance [94, 104, 151].

Calls to action, from the public and relevant technical experts, in regards to AI regulation is reminiscent of calls in prior "Ages" where concern for the harms from quick adoption of technology were prevalent [75]. For instance, consider the industrial age and cloth-making. Those most impacted by new machines for spinning and making cloth were the industry professionals in the textile industry. These workers would become known as the Luddites for their objections to the factory model. Their concerns echo the modern day in regards to job loss and injury from insufficient labor protections and lack of safety standards which historically led to loss of life and major injuries for early-adopters of the new technologies [102]. While the abysmal safety standards in the early factories are regarded in modern times with great negativity, the consequences of these practices were not unknowable, even at the time. We, in the current age, do not have to make these mistakes. Instead we can recognize the historical patterns of fast adoption of technology leading to harms, whether it is cloth-making, automobiles, or social media [75, 98, 140]. Specifically, we can initiate the development of what prerequisites are needed to ensure an AI deployment can satisfy existing requirements for each application domain, and support research and innovation efforts towards creating those prerequisites. We are not facing a challenge of how to balance regulation and innovation, a not uncommon juxtaposition [61], rather we have the opportunity for regulation to be used to guide innovations we otherwise would not recognize are needed. In this work, we focus on a path towards identifying such innovations, that may address harms to social and functional norms associated with privacy, including both technical solutions as well as policy and compliance efforts.

Contributions. We emphasize that the literature on AI, governance, privacy, and their intersections is vast. Significant bodies of work exist on protections of privacy in AI, largely focused on machine learning (ML) [53, 117], on AI governance [2, 7, 96], and on communication and social implications of AI and automation [12, 36, 77, 81, 109]. With this paper we do not aim to match the depth in each and every one of these disciplines. Rather, we bring them together in this work to highlight the breadth of these areas and how what is known relates to the field of AI, both in terms of possibilities and limitations. To this end, we provide an overview of relevant subdomains including privacy, policy, compliance, and the field of AI. We then walk through the stages of AI to illustrate where challenges exists, what solutions have been investigated, and the different stakeholders and decisions they make throughout the stages of an AI deployment.

Organization. This paper is organized as follows. Section 2 contains an overview of the concept of privacy and particular interpretations of privacy that intersect with the domain of AI. We discuss the meaning of the terms policy and compliance in Section 3 with a focus on highlighting the differences between how the term 'policy' is employed in computing science and engineering research in discussions of regulatory measures and governance versus how policymakers and related decision makers use the term policy. Section 4 is a review of how the term artificial intelligence has evolved over time, with a focus on how the visibility of AI has changed over time. Finally, in Section 5 we present our discussion of each of the above concepts throughout our structure of AI life stages where we discuss how at each stage decisions are made that impact who an AI application impacts with consequences, where existing technical mitigation strategies can be employed, and how these vectors relate to efforts to regulate AI and evaluate AI applications for compliance.

2 Privacy

Conceptually, the notion of *privacy* is incredibly complex. It is influenced by societal and cultural norms, as well as an individuals own understanding, preferences, and expectations in regards to having privacy. Privacy, while complex, is also very intuitive and simple. While these may seem like contradictions, the reality is that the simplicity corresponds to everyone having an intuitive notion, or mental model, of what is meant when they hear privacy. For instance, we have insight into people's mental models of privacy for hundreds of participants across all ages from Oates et al., where their participants each provided illustrations of what privacy meant to them, some of which were turtles, some locks, and some bathrooms [110]. In terms of capturing the complexity of privacy, there have been many efforts at formalizing privacy including Nissenbaum's *Contextual Integrity* [106, 107], Westin's categories of privacy attitudes [87, 147], and Solove's theory of privacy [132].

If we both generalize and simplify these theories of privacy, then we can takeaway that they each formalize some notion of what is being protected and from who, and who gets to decide what is being protected and from who. In the case of contextual integrity, this formalization is done via norms and the appropriateness of information flows, where both norms and information flows have very particular meanings within the theory of contextual integrity. For Westin's categories of privacy attitudes the formalization is done through illustrating what an individual would be willing to share and with who, based upon what category of individual they are within the three categories of a privacy fundamentalist, privacy pragmatist, or unconcerned. Finally, Solove's taxonomy also captures what is being protected, from who, and who gets to decide by synthesizing different notions of privacy and considering the implications of different settings in terms of privacy. While each of these theories, and more [3, 120], could be applied in depth to understanding and formalizing privacy in the context of AI, we focus on the mitigating and protecting efforts, in particular ones that are reflected in proposed and existing regulations. We discuss different approaches in the context of how they are framed as proxies for privacy and the consequences of such framings. We emphasize that while we put forth each privacy formalization in isolation, specifically data protection, consent, and quantifiable measures, none of these will be sufficient on their own and we highlight weaknesses for each.

2.1 Data Protection as Privacy

Data protection is not necessarily privacy protection. However, there are both regulatory frameworks, such as the European Union's General Data Protection Regulation (GDPR) [142] and Canada's Personal Information Protection and Electronic Documents Act (PIPEDA) [113], as well as privacy enhancing technologies, such as differential privacy [45], that work within a worldview where they prioritize protecting data. The existence of data protection laws does not prevent the existence of additional

privacy laws like in the EU [37]. However, even the existence of privacy focused regulations does not prevent the usage of data protection regulations as a proxy for privacy protections. That is, systems, both technical and regulatory, are formulated around the idea of data. Such protections may encompass what is permissible in terms of how data is collected, how it is used, and how it is accessed. However, in the most strict sense, this abstract notion does not necessarily capture the privacy implications, as it does not have a formal connection to how the use or processing of the data impacts the subject of that data. When we use the term data in computing science, engineering, and even within regulatory documents, the term is a useful and convenient abstraction that captures the notion that there is some form of information that is about something. In practice, that something the information is about, is often a person, who is the subject of the data.

Definition 2.1 (Adapted from Def. 2 [76]) *A data subject is an entity whose data, including information about them or generated by their action or inaction, is present in the data set being computed over (e.g., a training set for ML or statistical analysis) and the data describes the subject or their attributes.*

Definition 2.2 (Adapted from Def. 3 [76]) *A data owner is an entity that holds a dataset that is being contributed to some data analysis (e.g, towards training a ML model or statistical analysis) which is made up of data that originates from one or more data subjects and may or may not include the data owner as one of the data subjects.*

In the above two definitions, neither specifies what the data actually is nor its attributes. For instance, whether the data is considered personally identifiable information (PII), pseudonymous, or anonymous is a factor by some regulations for how it is treated [48, 113, 153]. However, the reality is there is no consensus on a rigorous interpretation of these terms such that there are clear delineations as to whether something is, for example, sufficiently anonymous. Different treatments of data based on how it has been classified by regulations may also fail to capture how it might impact the subjects of the data and whether they agreed for their data to be used in such cases. No matter how the data is protected or perturbed, there is always something that is being learned from that data or the analysis would not be done in the first place [153]. As demonstrated in a 2023 study from Kacsmar et al. [76] the necessity of something being learned does not escape members of the populace, who in general are the data subjects whose information is used in the analysis, as highlighted by a participant who stated "At the end of the day, they're still like learning specific things about me" (P7) [77].

When speaking of data and data analysis, not only is something necessarily being learned, but also there are necessarily one or more parties that contribute to the analysis and correspondingly learn the results of the analysis. Who these parties are, which of them learn the outputs of the analysis, and what type of industry they are in; all impact the perceived acceptability for members of the populace [78]. Despite the impact of these factors, regulatory treatments of data as privacy protection and technical protections of data to protect privacy are formulated and presented using the abstraction of "data". Data is what is being protected, and thus is what is being regulated, despite the issues associated with the abstraction as stated by E.M. Renieris, that "Trying to regulate data as such is like trying to regulate technology as if it has a common definition or clear contours-an exercise in futility" [123]. Certain domains, in particular health data, are sometimes treated as special, with their own laws, beyond the general concept of data [138]. However, despite additional protections being placed on certain types of data in a particular jurisdiction, it is still possible for it to fall outside of the scope of the particular laws. For instance, there are cases of health data falling outside of health focused regulations and leading to cases where datasets with health information are being sold [115]. There are even specific cases where the sold datasets include mental health conditions alongside demographic information of the person or individual names and home regions [33]. Thus, the notion of data protection as privacy is lacking

any human-centered considerations for privacy. One common way of incorporating a small amount of human-centered consideration is via our next notion, using consent management as privacy.

2.2 "Consent" Management as Privacy

The plethora of opt-out, click-to-continue, consent windows that have become the norm in our society are an example of *consent management* as privacy, and an illustration of its weaknesses [9]. Specifically, as regulations required companies to communicate when they collect data, what data they collect, and what they collect it for, these companies sent out updates to privacy policies, added cookie banners to their web pages, and other such modifications as the strategy for being compliant with regulations through a consent-based process [67, 112, 135]. However, while there are a multitude of prompts requesting someone agree before they use a service or platform, it is a stretch to refer to the responses to these prompts as consent. The privacy policies and cookie banners, written and provided by legal experts for an organization, are provided to any person wanting to use the system as an all or nothing agreement. If you want to file your taxes using a particular tax software, you are not provided with a way to do so without agreeing to their policy. In short, these consent management strategies, such as privacy policies can be considered an specific example of the general class termed *contracts of adhesion* [69, 82]. A class of contracts that are hardly read and may even include terms and conditions that should not or cannot be enforced. Privacy policies are a core document used to convey how data is used and collected by companies despite it being long established that these documents are hard to read and rarely read [101, 111]. The inaccessible nature of privacy polices, and the inclusion of contentious terms, is even reflected in a decision in Canadian privacy law. Specifically, in Canada (Privacy Commissioner) v. Facebook, Inc. 2024 FCA 140, which decided Facebook failed to meaningfully get consent from its users, it was stated that:

> "Whether consent is meaningful takes into account all relevant contextual factors; the demographics of the users, the nature of the information, the manner in which the user and the holder of the information interact, whether the contract at issue is a one of adhesion, the clarity and length of the contract and its terms and the nature of the default privacy settings" [21].

The challenges associated with using "consent" as a proxy for privacy do not end at inscrutable and lengthy privacy polices. We also have to account for how organizations have been found to use techniques to manipulate people into selecting options that may not be in their best interest. These techniques are broadly referred to as dark patterns and can be found in online shopping sites, mobile apps, and digital interfaces more broadly [18, 63, 99]. This is not to say consent should not be considered. Rather we mention both the issues with privacy policies and manipulative practices to highlight that while systems "request" people indicate agreement to some practice, these people have other priorities beyond reviewing privacy documents. Instead of expending time deciphering legal jargon, people are focused on whatever primary task is associated with the software they are accessing. For instance, consider the company Wealthsimple which purchased another company, SimpleTax [66]. Shortly after this purchase, the prior promise from Simpletax that indicated they would never sell your data was removed, indicating that going forward continuing to use the software to file taxes in the future would require agreeing to their terms and conditions; which no longer promised not to sell your data. However, an individual facing this update would have to decide to leave an ecosystem that they had previously been using for the important task of completing their taxes. The question then becomes, is a person who continues to file their taxes with this software actually consenting to the new terms and conditions, or are they just trying to do what has to be done, and get their taxes in on time?

It may be tempting to evade this problem by stating that a person who chooses to use a platform despite it lacking privacy protections is choosing to act against their own privacy interests. More extreme arguments may take the form that regardless if a person claims to care about privacy, their actions indicate that they do not hold such values. This framing, that people who act against their own privacy interests, while they claim to care about privacy, is referred to as the *privacy paradox* [43, 85]. However, this is an oversimplification of the reality people face. That is, we must consider the infeasibility of being informed via privacy policies, the use of dark patterns, and that people have a different primary task than preserving their privacy which all together means that while these people may act against their own privacy interests, the reality is that it is incredibly hard for them to do otherwise [122]. The natural followup to this conclusion is that since it is too difficult, essentially infeasible, for the populace to preserve their own privacy interests, technologists and regulators must develop solutions to aid in protecting these individuals privacy. Once such solutions emerge, we next must determine a way to decide whether the solutions are sufficient and a way to test that sufficiency.

2.3 Quantifiable Measures as Privacy

Technologists, when making technical systems, develop *quantifiable measures* to evaluate their work. Technical solutions for privacy have several measures that serve as proxies for how much privacy is lost or the maximum amount of privacy that can be protected. For example, there are technical notions of data anonymity, syntactic notions of privacy, which formalize a particular way of enforcing data anonymity. These notions include k-anonymity [126, 127], ℓ-diversity [95], and t-closeness [90]. Consider the following high-level formulations.

Definition 2.3: For *k-anonymity* [126, 127], there must be at least k records that match any subset of potentially identifying values (quasi-identifiers) that are returned for a query on a given dataset.

Definition 2.4: Extending k-anonymity, *ℓ-diversity* [95] adds the requirement that for any sensitive attribute, there should be at least ℓ distinct values represented in the returned response.

Definition 2.5: Once again extending the data protection notion, the property of *t-closeness [90]* adds the requirement that the distribution of the sensitive attribute and the distribution of the whole data sample should differ by no more than a threshold of t.

These three notions, which provide increasingly formal requirements for data protection, have their uses, in particular for data analysis and data release. However, they do not lend themselves nicely to privacy in AI systems. AI system do not have the same form of data release and thus cannot employ the same protections as query-response based data releases.

In the case of AI the quantifiable notions of privacy employed are either semantic or empirical. The prevalent semantic notion of privacy is *differential privacy* [45, 46], which can be applied to data directly or to the output of a function. Since its formulation in 2006, variations on DP as well as particular ways of applying DP to specific ML algorithms have been developed [53]. For reference, we include the formulation of ϵ-Differential Privacy and discuss conceptually the privacy guarantees that it provides.

Definition 2.6: (ϵ-Differential Privacy [45]). A randomized mechanism $M : \mathcal{D} \mapsto \mathcal{F}$ provides *ϵ-differential privacy* iff for all neighbouring inputs D, $D' \in \mathcal{D}$, ie., differing in one element, and all subsets $F \subseteq \mathcal{F}$,

$$Pr[M(D) \in F] \leq e^{\epsilon} Pr[M(D') \in F], \text{ where the probability space is M's coin tosses.}$$

When used correctly, differential privacy (DP) can effectively protect against leaking information as to the presence or absence of a data point in a calculation. The general idea is that when using DP, an adversary observing a differentially private output of a mechanism is sufficiently unlikely to be able to distinguish between a case where a data element was included in the dataset the mechanism computed over versus a case where that same data element was not included in the dataset.

One way of measuring the success of this protection, outside of its theoretical guarantees, is through an empirical measure of the effectiveness of an attack on a conventional ML model versus one trained using DP [71, 74]. An attack where an adversary aims to determine whether a target data element was included in the training dataset or not, is aiming to execute a membership inference (MI) attack [131]. Thus, through testing the attacks, we can generate a measure of information leakage both by evaluating training time attacks and test-time attacks, the latter of which encompasses the area of inference attacks and can be targeted at various deployed ML models [55, 68, 117]. Inference attacks executed by an adversary may target leaking information about the training data as well as model parameters. Assessing the success of inference attacks does not guarantee the non-existence of more sophisticated attacks where additional information leakage may occur. However, by testing models for their susceptibility to known inference attacks, we can get a measure of the minimum amount of privacy leakage that is occurring [92], as long as awareness is maintained that there may be greater privacy leakage than can be assessed via this evaluation. In summary, while quantifiable measures can give us guidelines to work towards, they are not all encompassing and may be vulnerable to yet undiscovered privacy attacks. We also cannot only focus on the risk of privacy attacks and must also maintain an awareness that although quantifiable measures are useful to us as technologists, when measuring social or human-centered notions like privacy, they will never be a perfect proxy. Thus, our measurements require careful evaluation, as with all the prior privacy notions we discussed, to determine whether their protections are appropriate and sufficient for any given deployment.

> Privacy is a multifaceted concept. Even when focusing on AI, each of the different notions of privacy need to be utilized collectively and in context for any technological deployment to include appropriate consideration and protection of privacy.

3 Policy and Compliance

Policy and compliance are entwined within any effort to regulate technologies. Who contributes to formulating regulations and policies impacts the identifications of potential harms and relevant protections [7]. The breadth of expertise required is a factor for addressing challenges associated with innovations on what are often already relatively novel technologies. To construct clear and meaningful governance for AI, policymakers and technologists need to be able to use common language and definitions for AI, for policy, and for harm [62, 86]. Otherwise, challenges will persist due to how each of these parties may regard the consequences of violating societal norms and expectations like privacy [2].

3.1 The Meaning of Policy

Policy is a broad-reaching non-specific term used to capture a lot of different ways of documenting expectations or requirements for individuals, groups, and governments. Policies can be written within a company about both their internal practices and their external practices for employees and users of the products or services they provide. Schools can write policies about expected behaviors and procedures for students, instructors, and administrators. Finally, policy can also be used to refer to contracts, regulations, and laws, including those from governments.

Research in computing and engineering refer to recommendations for policy and policymakers regularly, but generally neglect to distinguish between the different forms policy instruments can take [60]. Consider, for example, the Government of Canada, which provides explanatory documents on their website for "policies, directives, standards and guidelines" that capture what these terms mean within the Canadian government [59]. The term *policy* in these documents refers to mandatory responsibilities that face internally, meaning they apply to officials and deputy heads, with *directives* guiding how these internal actors are to comply with policy. Less stringent documents include guidelines, which are voluntary, providing advice and recommendations rather than requirements. *Laws and legislation*, and in some jurisdictions also regulations, are what define protections for the citizens of a populace, via placing requirements that apply to organizations and individuals within the legal jurisdiction. Overall, despite how it is used within computing science, the term policy itself, does not necessarily correspond to a regulation or legislation, though it may contribute to the development of such things.

Definition 3.1 ([59]) *"A policy is a set of statements of principles, values, and intent that outlines expectations and provides a basis for consistent decision-making and resource allocation in respect to a specific issue".*

Mandatory policy instruments, which are enforced via laws and regulations, and more voluntary policy instruments can have beneficial outcomes for the populace. Determining which type of policy instrument is most appropriate is likely outside of the expertise of many technologists. However, by communicating to 'policymakers' whether a given recommendation requires everyone partake in order to have the desired outcomes or whether it is still beneficial when only a few organizations partake, is a good starting point.

3.2 Compliance

The fundamental notion of *compliance* is for parties to follow the requirements of pertinent laws. Which laws are pertinent is a question not just of the technology being used or the domain it is being used in, but also a question of territorial reach. While some laws may apply, not just within their physical jurisdictions, but also to citizens of its jurisdiction when they are outside of it, many laws are localized within geographical borders [48, 142]. Data technologies, including internet services and AI, cross these jurisdictional and geographical boundaries, as the use of such technologies and their corresponding impacts on people are not bound within territories. This leads to courts making decisions as to which jurisdictions' laws are to be applied in the case of cross-jurisdictional cases [108].

Compliance requirements give rise to concerns about monetary cost as well as how to ensure systems are auditable in a way that is transparent to the entities that enforce compliance [64, 129]. As new requirements come into effect, companies need to take action, with actions corresponding to changes in processes that can require additional infrastructure, training time of staff, and even just loss of efficiency as employees adapt to the new processes. As a consequence of these, the actual actions a company takes in their effort to be compliant may focus on following the exact letter of the law in a way that most minimizes impact on the company's status-quo. The ways companies act to comply with regulations significantly impacts whether the regulations provide protections, in other words, whether the spirit of the law holds up [9].

As companies enact their practices for emergent technologies or novel applications of existing technologies, such as AI, it can become a competition to establish the best practices and standards that may influence the development of formal laws and oversight. These proactive policies may be very sound, however, focusing on them presumes that the organization is able to set aside its own business goals to produce a beneficial policy that could be applicable more broadly; a perhaps unfair expectation to put on companies' teams [151]. Practices where an organization complies in this way, evading the spirit of

the law, has been termed "avoision" by scholars [79, 137]. Yew et al. have already examined the potential for avoision in the context of the EU AI Act where they identified behaviors which currently, plausibly, and technically, comply with the act while leading to consequences that are contrary to the spirit of the regulation [152]. Therefore, to ensure laws and policy maintain the important outcomes that motivated their construction, we cannot understate the importance of including compliance considerations and challenges across all stages of developing, deploying, and moderating AI.

Technologists use policy to refer to both mandatory and guiding responsibilities for private and public organizations. Divergence in meaning when crossing communities causes issues on both sides, whether it is how policy makers speak of AI or how AI experts speak of policy. These misunderstandings only serve to escalate the issues in the already difficult domain of compliance.

4 Artificial Intelligence

The phrase artificial intelligence itself leads to confusion and misunderstandings about AI technologies' capabilities. Ensuring accurate expectations of what a technology is and is not capable of, fostering risk awareness, requires understanding mental models of the technology and its terminology [44]. The literature for artificial intelligence suggests that how AI technologies are presented, including whether it is referred to as machines, as tools, or as companions influences what human traits or mental capacities are attributed to it by the populace [30].

4.1 Conceptual Terminology Over Time

General notions of automation and stories of thinking machines go back far in history, however, we see the prominent emergence of the term *artificial intelligence* in the late 1950s, some time before the 1956 Dartmouth summer research project [35]. The focus of the 1956 conference was on intelligence and articulating the concept of intelligence such that machines may simulate it. This theme, of understanding intelligence and endeavoring to simulate it, remains prominent in the field of AI to this day. Modern descriptions, such as what one finds in a textbook, refer to the field of AI as being *"...concerned with not just understanding but also building intelligent entities–machines that can compute how to act effectively and safely in a wide variety of novel situations"* [125]. While there are many notions and definitions which are used to refer to the field of AI, what we ultimately see is a split where one camp typically corresponds to *"making machines think like humans"* and the other *"making machines act like humans"* [32]. Representatives of these camps are not limited to particular sub-domains of AI, rather these views may be found across the areas, including natural language processing (NLP), computer vision, and robotics.

Post-2022 is a world where generative AI, and in particular generative AI based chatbots have become prolific and public facing. OpenAI, the organization that released ChatGPT to the world was founded in 2015 with earlier forms of what came to be known as ChatGPT existing as early as 2018 with GPT-1; though it was far from what its successors would become [97]. Even when the advance to GPT-3 occurred in 2022, visibility was still largely limited as it required API access and was primarily only in the awareness of computing science experts and other technologists. In 2022 though, we see the first user interface form of ChatGPT released to the broader populace. The result has been that ChatGPT and the family of models associated with it known as large-language models (LLMs) have in some ways become synonymous with the term AI itself. That is, as reflected on by Karen Hao in regards to her reporting on the area,

> *"While ChatGPT and other so-called large language models or generative AI applications have now taken the limelight, they are but one manifestation of AI, a manifestation that embodies a particular and remarkable narrow view about the way the world is and the way it should be"* [65].

LLMs more broadly, becoming a center for the AI story is entwined with human perceptions of machines and language, which has been around formally since 1955 when John McCarthy, Marvin L. Minsky, Nathaniel Rochester, and Claude E. Shannon proposed a Dartmouth summer research project on AI. In their proposal, they stated that *"An attempt will be made to find how to make machines use language, form abstractions and concepts, solve kinds of problems now reserved for humans, and improve themselves"* [100]. Further elaborating on their intentions, they outlined seven aspects they determined as the aspects of the *"artificial intelligence problem"*. Among these aspects, in fact the second one, was the problem of how a computer could be programmed to use language. The justification for this problem proposed that a lot of human thought consists of the manipulation of what could be termed language. Thus, one might expect that the ability to acquire and use language would be a core component to achieving machines that can address *"problems now reserved for humans"*, and be a way to achieve "artificial intelligence". All together, the field of AI has advanced much since its nascent naming. However, while the public view of AI has been overrun with an awareness of LLMs, both LLMs and other AI have a frequent reliance on ML.

4.2 Artificial Intelligence and Machine Learning

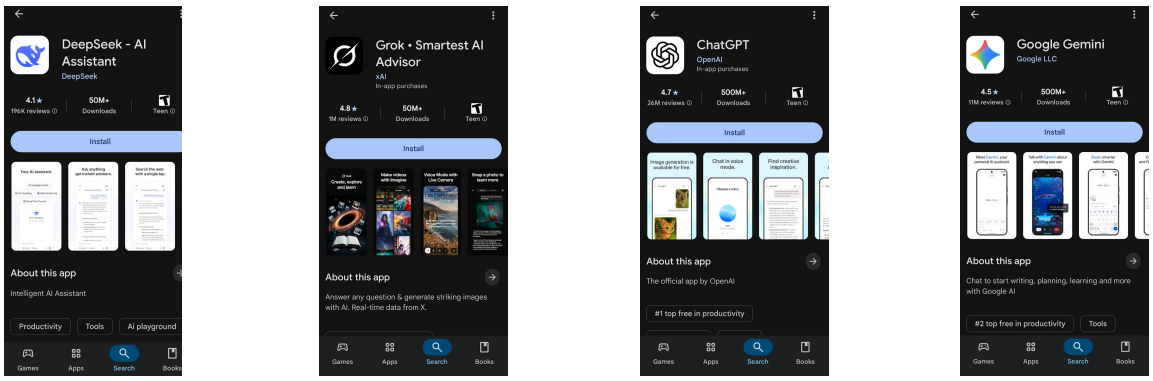
Oftentimes when researchers and technologists say AI what we are talking about is something that uses ML. Modern chat-bots, as well as other modern AI systems, regularly employ ML to achieve the desired functionality. ML is considered a subset of the AI field and the types of ML can be grouped into supervised learning, unsupervised learning, and reinforcement learning [125]. The differences across these types of ML include the goals of the system being produced as well as the way in which data needs to be acquired and prepared for it to be useful to the system. In supervised learning, the goal is to produce a model that is able to perform its task well, such as performing classification, on data it has not seen before. Supervised learning requires training data to be prepared for it with true labels that correspond to the values the model is trying to learn to apply to data. Unsupervised learning takes in training data without labels with a goal of finding patterns that are understandable and of use to humans. Unlike supervised learning, unsupervised learning does not require a corpus of pre-labeled data to train on. Finally, in reinforcement learning the goal is to learn from interacting with an environment such that the system learns a behavior in a way that supports some defined purpose within that environment.

Whether something is AI or ML or even which specific algorithm it uses is not a core issue for this work. However, in the case of policy and compliance there are distinctions associated with the algorithms in use that are of significant importance. For example, different algorithmic approaches, all of which fall under the umbrella of AI, can be either probabilistic or deterministic which has implications for testing, auditing, and transparency. Similarly, model explainability, and whether a model is more or less explainable does not correspond to a particular type of ML. However, some ML algorithms are easier to explain, such as decision trees, whereas others are generally more difficult, such as neural networks [91]. For instance, consider one of the first chat-bots, namely Eliza, which employs rule based systems where outputs are determined via a combination of pattern matching and preset scripts [144, 145]. Possible outputs from Eliza for any given prompt from a user is bounded, as it only outputs things from whatever its current script is set as. While the script can be updated or replaced, ultimately it is possible to reasonably trace why an output was produced. In contrast to this, recent chatbots like DeepSeek [38], Grok [149], ChatGPT [114], and Google Gemini [57] employ large neural networks each of which have

been trained on a large corpus. All together, the amount of data, the parameters and layers of the neural network, along with the different ways the organizations behind these AI models have constructed their output filtration and access controls leads to systems that are infeasible to trace or explain. These chatbots, which are leaders in being hard to explain, are now being deployed en-masse towards having them used in nearly every application one could imagine.

4.3 The Visibility of Artificial Intelligence in Applications

Applications of AI prior to 2022 were largely not marketed as AI to the users of those applications. However, in our post-2022 era, this has completely changed, largely due to the impact of advancements in AI for language, a domain much more accessible, and thus much more marketable to the public than other domains for AI. The goals of language models include both recognition of speech and recognition of text. What is meant by recognition determines a lot. Recognition includes speech-to-text, translation between languages, and predictive text, the last of which is colloquially referred to as auto-complete, an application for AI that has been in wide use, first in web browsers and then in phones, since the early 2000s [56]. AI has also been widely used in recommender systems such as for media consumption on Netflix [4, 14] and YouTube [34], in spam filtration systems [23], in translation [148], in voice assistants like Siri [22], and in fraud detection [8, 105]. Not all AI applications require sophisticated black-box-esque neural networks. Rather, simpler techniques, including linear regression, are also quite effective, including for tasks such as predicting changes in housing markets [19, 24]. Random forest, XGBoost and neural networks, have been employed in efforts for wild fire predictions, further emphasizing that the technical sophistication of the algorithm is not a sole predictor of how useful it is for a task [41]. Even in the case of video games, where there is a greater awareness that AI may be used, the AI includes programmatic solutions in addition to more complex supervised learning, unsupervised learning, and reinforcement learning approaches [150].



(a) DeepSeek (b) Grok (c) ChatGPT (d) Google Gemini

Figure 1: The Google Play store page for the first four results for AI assistant apps on the Google app store as displayed on September 13, 2025.

Notably, the AI in these applications is "hidden" in that it is working in the back-end of the systems. For example the users who access their emails or use their credit cards have no need nor explicit information that the spam folder is produced with the help of AI, or that the flag on their credit card that had a suspicious transaction uses AI. The details of such workings, or even that these systems use

AI at all, stayed with those who made the systems, audited the systems, or otherwise contributed to their production. Those whose primary interactions were based on some other task, such as checking email, did not need nor care to know the details of the automation techniques aiding in their day-to-day interactions. These pre-2022 examples highlight the prolific, but less visible AI of the time.

Post-2022, we see a surge in visibility of AI. Advancement towards producing machines that manipulate language formed through a series of evolutions as to what is the best technique for natural language processing (NLP). These evolutions include, among others, how to process the text by breaking it into parts or n-grams [31], transfer learning [70], and the 'T' in GPT, transformers [141]. After these advancements came forth, we see the transition to AI assistants, AI chatbots, and marketing that uses the term AI. For example, in the Google Play store as of September 13, 2025, there are AI assistants from both long established technology companies as well as more recent ones, which explicitly market their app as "AI", "Intelligent", and "the smartest". We include for reference the app pages of the first four AI assistants shown on our app store in Figure 1. Further demonstrating the extensive way AI has become part of the status-quo in terms of public expectations for technologies, there are even emerging directives from governments on how to adopt AI into their processes [58, 61], a call towards adoption that was not explicit in the regular use of AI predating this point and time.

AI, as with any other technology, cannot be treated as a monolith, nor can it be treated it as a unique innovation that eludes existing regulatory norms on automation. While LLMs have been in the forefront of conversations, AI is a broad field itself with a breadth of applications, each of which have domain-specific norms and standards associated with them throughout any development and deployment.

5 Structuring Life Stages for AI Applications to Identify Privacy Vectors

To determine when and how to use AI, one approach is to consider what the specific technical implementation is, the consequences of its use, and who is impacted by it [96]. In this section we articulate pertinent life stages of AI applications. These stages are identified in part, though consideration for the data science life-cycle. Our stages are formulated to reflect how the processes have changed with advancements of AI as well as to be structured in ways that correspond to policy, regulation, and compliance. Thus our stages deviate from the data science life-cycle. Finally, our stages support the identification of where human actions and decisions are made as well as where technological protections can be implemented when evaluating or assessing novel AI applications.

5.1 Data Science Life-Cycle

The term *data science* broadly refers to an area of analysis which employs statistics, algorithms, and related processes over data sets to extract useful insights. The data science life-cycle as presented by Kelleher and Tierney [80] originates with the CRISP-DM cycle from Chapman et al. [28]. Note there is not just one representation of the cycle, but the CRISP-DM cycle does capture the information relevant to our discussion. The data science cycle allows for moving back and forth between stages, with the stages being defined as 'business understanding', 'data understanding', 'modeling', 'evaluation', and finally 'deployment'. Overall the idea of the data science life cycle is that the likely starting point corresponds to identifying some project objectives and requirements that can be addressed via analysis. Once the project problem is identified, it is now time to collect, clean and prepare, and gain familiarity

with the data for the analysis. The modeling techniques stage could use any type of analysis methods from basic regression through to complex ML techniques depending on the project goal. Once the appropriate modeling technique is identified, it is only a matter of testing and verifying it before finally deploying it to be used for its intended task. We will now use this underlying structure to formulate AI life stages.

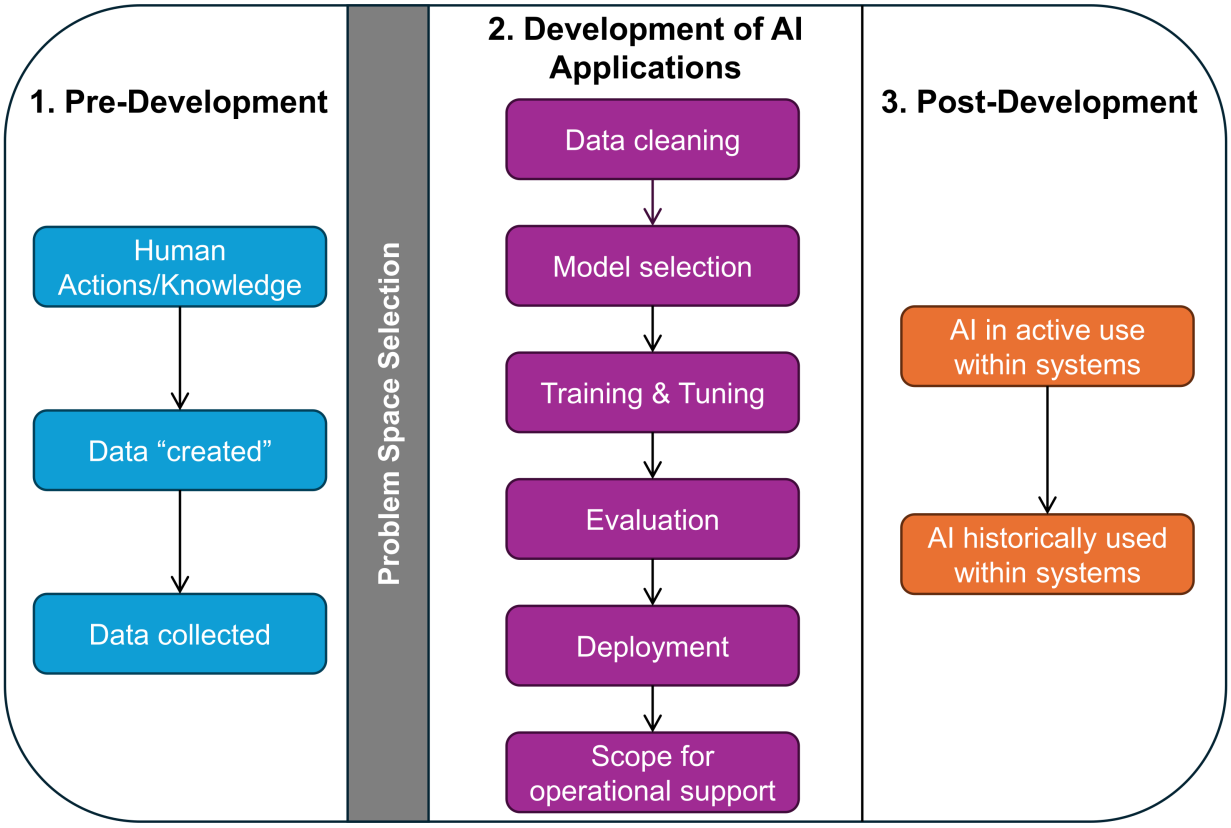


Figure 2: We depict AI application "life" stages of particular relevance for our discussion of privacy, policy, and compliance. These stages span pre-development through to instances where the AI is retired from use. Note that despite our numerical ordering, the process can flow back and forth between stages.

5.2 AI Life Stages

Consider the synthesis of the privacy theories from Section 2 and our claim that the core idea is to define what is being protected, from who, and who gets to decide. Now, by applying this formulation to AI, we can identify the relevant details for each portion of our privacy description. First, in terms of what is being protected, we can formulate it as protecting the training data, protecting the model, protecting the inferences or outputs from the model, and protecting the people impacted by the system at any stage in the AI development process. Second, in terms of who we need to protect against, we may be protecting against the entity that collected all the data (the data owner), those who are the subjects of the data (data subject), individuals within an organization that is involved at any step of the AI process, and even those who just are using the AI system that is produced. Finally, who gets to decide is a reoccurring question across every step of the process. While we can argue that those most

impacted should be the ones that make the decision, the reality is that when these decisions are being made, it can be very far away from those impacted by these decisions. Additionally, those making the decisions, whether company employees or government actors, are powerful and distanced from the people impacted. These decision will also require interdisciplinary expertise to ensure accurate consideration for the functionality of AI as well as the standards within the deployment domain [136]. Through our partitioning of the life stages, we are able to capture the different points in time that people are effected versus when they may actually have any power to intercede on their own behalf or on behalf of others. We provide a visual overview of our AI application life-stages as Figure 2.

5.3 Problem Space Selection

Within our life stages figure, we include *problem space selection* as a literal gray area. Whether the problem space is identified prior to data collection, after data collection, or even after the development of an AI tool, at some point it all comes back to what is the domain and what does it mean to address problems within it. This particular component, while not necessarily a stage itself, is perhaps one of the most critical parts of the AI life-stages. This "stage" is when we can stipulate correctness. For example, before we can measure what "accuracy" is achieved, we need to first ensure that what we have trained the model to assess as being "correct" is actually correct for that application. Furthermore, we may need to account for whether or not the domain has stable or evolving notions of correctness. In the case of fields like medicine or dentistry where practices are continuously advancing or in criminal justice systems where we know historical issues exist, we know that we do not want to mirror the past's standards of correctness [139].

For example, if a classification reflects past ways of referring to someone or something, which are now considered slurs, the model may still probabilistically report that past term since there are a greater number of examples where that was the answer. Reflecting the past probabilistically, whether our social or scientific past, is a risk that may be acceptable for some settings, but certainly not when deploying to any high-impact domains like education or health. In the case of LLMs, they are inherently probabilistic. That is to say, you can give them the same prompt and they could give different outputs. Therefore, LLMs may largely be inappropriate for domains where their probabilistic nature cannot meet the standards of practice for that industry. However, in order to determine that, we cannot make an assessment based on the insights from only technologists, but rather we must engage with domain experts.

5.4 AI Life Stage: Pre-Development

From Daily Life to Data. Throughout any given day, data is generated by our actions and engagements with technology. Every time we access our emails, browse web pages, get captured by video recording doorbells, data is being generated and collected about us. Our day-to-day actions produce immeasurable quantities of data. We define the *pre-development* stage to capture the reality that there are components within the AI life cycle, in particular in relation to data, that may occur well before any consideration or speculation for using AI.

Consider, for example, two "classic" datasets within the field of AI, the Enron dataset and ImageNet [39, 84]. ImageNet came to be through collecting billions of photos from web pages, video clips, and Google's image search database. All of these images had been captured and then shared on the internet by people who had no way of anticipating their images would then be taken in the future, collected together with other images, and labeled by other people to produce a dataset that would then be used to train and evaluate an unknowable multitude of ML models and image analysis algorithms [39]. Similarly, the employees of Enron could not have anticipated their regular email communications would

come to be the Enron dataset, a corpus of text that exists due to a fraud investigation which seized the emails of employees as part of the investigation and then released all the emails to the public. The result was a dataset of communications that has been used by countless researchers in natural language processing [84].

For both the Enron emails and the images within ImageNet, the origin is *human actions or communication of information*. Whether it was taking a photo and sharing it on the internet, sending an email to a spouse, or to a colleague, all of it was captured by researchers and made into datasets. These two datasets, which receive prolific usage, not only did not get consent from the data-subjects, the humans who this data came from, but at the time these people could not even have anticipated their data would be used in this way. Further, we may attempt to argue that such data scraping practices would not be compliant with current regulations. However, the current lawsuit against a company for scraping and utilizing data they gathered from the internet [10] suggests that the practice persists. Data is still being treated as something to be collected for use as a resource to exact value from regardless of the data origins and despite awareness of the financial and personal consequences faced by those whom the data originated with. The extraction and exploitation of our data is so prolific we cannot even begin to understand it with every app we use, every online banking transaction, every credit card purchase, and essentially every action we take connecting back to a digital representation that goes into datasets we know nothing of. Even going so far as data being collected by organizations we have never interacted with such as when data brokers procure data from both public and private sources [20].

From Data to Datasets. After the data was collected, before ImageNet could become what it is today, it first needed to have the mass collection of images be assigned labels, a task that would be relegated to a labor force from across the world which would do the task for low pay through the use of Amazon's Mechanical Turk [5]. The practice of having large scale distributed workforces annotate data quickly and cheaply has become the status-quo in modern day [12, 65, 119]. While the mass collection and annotation of data has become typical, that does not mean overcoming the issues associated with these practices are not the subject of investigation. Attempts have been made to move away from a reliance on large datasets and mitigate the impact they have on people's privacy. For instance, in an effort to go beyond traditional anonymization techniques (recall Section 2.3) the use of synthetic data has been considered. Synthetic data may be generated from sensitive data or from rules and statistics. However, so far synthetic data has not found much success at privacy protection and has not proven better than using traditional anonymization techniques, which themselves are not ideal for training AI [134]. Alternative approaches target the other side of the issue, focusing on transparency and consent, through efforts to formulate how data donation could work [143]. Having a data donation style strategy to address to issues within the pre-development life-stage, while potentially very beneficial, will require a complete overhaul in current practices, re-centering decisions to align with those whom provide the data and those impacted by it.

The people from whom data originates are generally far removed from the decision makers. That is, those impacted are not the same as those who make decisions about data. While data protection laws address some issues, what has become increasingly clear with AI is that the continual treatment of data as something to be possessed or exploited only serves to increase potential harms by distancing these practices from their impact on individuals and society.

5.5 AI Life Stage: Development

The *development* life stage encompasses the aspects of an AI application that are typically determined by technologists, such as algorithmic techniques, parameter configurations, and testing procedures. This includes considering measures for privacy leakage from models via empirical measures of the effectiveness of attacks [71, 74, 92]. However, there are many factors that impact how successful attacks are such as memorization of training data, which is, as one may expect, bad for privacy [133]. However, memorization is not just a bad side effect of the AI training process, but also an important property that corresponds to good performance on data outside of the training sets [51].

This leaves other specialists, who are AI practitioners rather than privacy attack researchers, without clear measures to evaluate against. Despite this, the software developers and engineers are the technical practitioners who code and configure the AI for a system, including the privacy considerations. These considerations can include protecting the data used to train the model as well as protecting against unauthorized access or use of the model. When developing an AI application that requires protecting training data, there are many technical innovations to aid in protecting privacy when training models, including: secure aggregation, differentially private stochastic gradient descent, and differentially private empirical risk minimization among others [1, 16, 29, 42, 53, 116, 130]. However, identifying what the appropriate technique is to use, or even whether there is an appropriate technique is not necessarily within the expertise of the practitioners, and thus there is a need for relevant education, developer tools, and process-oriented support to help practitioners prevent privacy harms being embedded in their deployments [88].

> The technical details, such as the algorithmic techniques and configurations, are determined by technologists within the development stage. However, whether those configurations are appropriate, and how to determine their appropriateness for an application requires insight from outside of the development stage, where domain experts can define what requirements the application has to fulfill and their feasibility.

5.6 AI Life Stage: Post-Development

The *post-development* stage considers the human decisions and actions that impact the consequences and benefits associated with the development of an AI application. After an AI application has been developed for some domain or task, we typically expect it will be deployed. Once the AI system is out in the world, it will either be in a state of active use or it will be retired and withdrawn from use.

First, consider an AI system that is in active use. In this case, maintenance, accounting for changes in what is required from the model, and communicating to those who use the model will require human intervention. Communication is of particular importance, as how people perceive a technology influences their trust in it as much or more than the reality of what the technology can achieve [11, 52, 83, 145]. Mismatched expectations between technological reality and marketing correspond to skewed mental models of AI functionality. Overconfidence in what LLM's can actually do led to a model being used to replace the jobs of people who were staff and volunteers at the National Eating Disorder Association's helpline, at least until the chat bot Tessa had to be removed from use [146]. The chatbot was removed, as it could not actually provide the aid required. This means that people were not acquiring correct mental health resources or appropriate conversational support. Rather, there was an instance where someone who was asking for advice while in eating disorder recovery and the chatbot instead gave information that was essentially suggesting how to continue having an eating disorder. Therefore, the over automation of critical support systems in our society is only leading to greater social and physical

failings for the people who depend on these systems.

Even when an AI application has been retired from use, or removed from use after causing harms like the Tessa chatbot, the impacts of the application can still remain. Consequences from the prior use of an algorithmic deployment, such as an AI application, remain and influence individuals and institutions. This lingering influence is termed an algorithmic imprints by Upol et al. who illustrate its effect in their analysis of an incident of algorithmic deployment on students around the world, focusing on students in Bangladesh [47]. The algorithmic standardization of the results of the General Certificate of Education (GCE) Advanced (A) Level exams in 2020 turned out to be critically flawed and biased, negatively impacting university admissions for students across the world who had taken the exam that year [47, 121]. While the exam grades were ultimately retracted and revised to not use the flawed algorithmic approach, the efforts of the teachers that had to prepare documents and the experience of those students does not go away. Finally, this deployment was inflicted on teachers and students globally by the Office of Qualifications and Exam Regulations in the UK, required significant time investment from the teachers beyond their normal role, and disregarded students preparation efforts. In this particular story, protests and media coverage along with large scale push back eventually corresponded to the retraction of the algorithmic scores, but the power to decide to deploy it and to retract it still remained with the Office of Qualifications and Exam Regulations and not with the students and teachers impacted by it.

There is significant disparity in terms of power as well as pertinent expertise among: those who the data came from, those who decide to use the data for their chosen purpose, those who decide what measure of truth to apply to the data, and those who the resulting AI impacts. Therefore, we must consider whether there should be a way to preserve the right to refuse to participate or be classified by AI.

6 Conclusion

Our understanding of what AI means has changed overtime, as has how we use the term data. The meaning of data changed quickly, but now so too have the consequences of data, such that data protection alone cannot solve the issues of human privacy in our current society. We can make better systems, but even good systems can cause harm without consideration for the full picture of who they impact. The reality is that AI systems necessarily reflect a snapshot in time, the time at which these labels or these notions of correctness were established. Changing these over time requires additional training or additional processes to account for the fact that these models will necessarily not reflect new understandings or new notions of what is acceptable.

Therefore, to advance towards resolving problems at the intersection of AI, privacy, policy, and compliance, we do not need yet another generalized framework or guideline. Even as early as 2021 there were more than 170 guidelines and frameworks in the broad area of responsible and trustworthy AI, including guidelines that synthesized collections of guidelines [6, 40]. Rather, we need to recognize the importance of domain expertise, specifically the expertise required to execute the tasks in the settings where AI applications are being proposed. This means that therapists' expertise should determine the viability of any proposed use of AI in the therapy domain. Dentists should determine the viability of any proposed use of AI in dentistry. Teachers should determine the viability of any proposed use of AI as an educator. This is not to state that these experts should speak on how to implement the AI or whether AI can achieve the standards they identify, but that they are the only ones who can properly determine what a fail state would be for any form of automation working within their area. This is the lesson we can learn from the past ages of industry, we do not need to repeat the mistakes of the past where

"*...workers are time and again ignored by regulators and governments in favor of entrepreneurs and their technologies of disruption*" [102].

To determine what needs to be developed, the technical feasibility of AI, and the appropriate legal consequences of using AI within a particular domain, the fail states must be determine by domain experts. Correspondingly, policy for AI must be guided by domain experts, and when privacy is a factor, privacy experts must contribute as well. We cannot rely on the technology organizations or service providing companies developing AI applications to identify consequences for applications. We must have domain experts collectively work together with law-makers and technologists if we are to develop technology-agnostic requirements on what matters in that domain, what failures cannot happen, and what it means for automation to work correctly.

References

[1] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep Learning with Differential Privacy," in *the 2016 ACM SIGSAC Conference on Computer and Communications Security*. Vienna, Austria: ACM, 2016, pp. 308–318.

[2] N. Agrawal, R. Binns, M. Van Kleek, K. Laine, and N. Shadbolt, "Exploring Design and Governance Challenges in the Development of Privacy-Preserving Computation," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–13.

[3] I. Altman, *The Environment and Social Behavior: Privacy, Personal Space, Territory, and Crowding*. ERIC, 1975.

[4] X. Amatriain, "Beyond Data: From User Information to Business Value Through Personalized Recommendations and Consumer Science," in *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, 2013, pp. 2201–2208.

[5] Amazon, "Amazon Mechanical Turk," 2025, accessed 2025-11-01.

[6] S. Amershi, D. Weld, M. Vorvoreanu, A. Fourney, B. Nushi, P. Collisson, J. Suh, S. Iqbal, P. N. Bennett, K. Inkpen *et al.*, "Guidelines for Human-AI Interaction," in *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 2019.

[7] B. Attard-Frost and K. Lyons, "AI Governance Systems: A Multi-Scale Analysis Framework, Empirical Findings, and Future Directions," *AI and Ethics*, vol. 5, no. 3, pp. 2557–2604, 2025.

[8] J. O. Awoyemi, A. O. Adetunmbi, and S. A. Oluwadare, "Credit Card Fraud Detection Using Machine Learning Techniques: A Comparative Analysis," in *2017 international conference on computing networking and informatics (ICCNI)*. IEEE, 2017, pp. 1–9.

[9] BBC News, "Cookie Banner Frustration to be Tackled by EU," https://www.bbc.com/news/business-38583001, 2017, accessed 2025-09-18.

[10] A. Belanger, "AI Industry Horrified to Face Largest Copyright Class Action Ever Certified," *Ars Technica*, Aug. 2025. [Online]. Available: https://arstechnica.com/tech-policy/2025/08/ai-industry-horrified-to-face-largest-copyright-classaction-ever-certified/

[11] E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell, "On the Dangers of Stochastic Parrots: Can Language Models be Too Big? 🦜," in *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, 2021, pp. 610–623.

[12] E. M. Bender and A. Hanna, *The AI Con: How to Fight Big Tech's Hype and Create the Future We Want*. Random House, 2025.

[13] Y. Bengio, S. Mindermann, D. Privitera, T. Besiroglu, R. Bommasani, S. Casper, Y. Choi, P. Fox, B. Garfinkel, D. Goldfarb *et al.*, "International AI Safety Report," *arXiv preprint arXiv:2501.17805*, 2025.

[14] J. Bennett and S. Lanning, "The Netflix Prize," *KDD Cup and Workshop 2007*, 2007.

[15] M. Bobrowsky, "Meta Will Begin Using AI Chatbot Conversations to Target Ads," The Wall Street Journal, 2025, accessed 2025-10-20.

[16] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical Secure Aggregation for Privacy-Preserving Machine Learning," in *the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 1175–1191.

[17] R. Booth, "Makers of Air Fryers and Smart Speakers Told to Respect Users' Right to Privacy," *The Guardian*, Jun. 2025. [Online]. Available: https://www.theguardian.com/technology/2025/jun/16/air-fryers-smart-tv-speakers-user-data-privacy-ico

[18] C. Bösch, B. Erb, F. Kargl, H. Kopp, and S. Pfattheicher, "Tales from the Dark Side: Privacy Dark Strategies and Privacy Dark Patterns," *Proceedings on Privacy Enhancing Technologies*, vol. 2016, no. 4, pp. 237–254, 2016.

[19] J. Brannlund, H. Y. Lao, M. MacIsaac, and J. Yang, "Predicting Changes in Canadian Housing Markets with Machine Learning," Bank of Canada, Staff Discussion Paper 2023-21, 2023. [Online]. Available: https://www.bankofcanada.ca/wp-content/uploads/2023/09/sdp2023-21.pdf

[20] D. Cameron and D. Mehrotra, "CFPB Quietly Kills Rule to Shield Americans From Data Brokers," Wired, 2025, accessed 2025-05-15.

[21] "Canada (Privacy Commissioner) v. Facebook, Inc., 2024 FCA 140 (CanLII)," https://canlii.ca/t/k6pn1, 2024, acccessed 2025-09-18.

[22] T. Capes, P. Coles, A. Conkie, L. Golipour, A. Hadjitarkhani, Q. Hu, N. Huddleston, M. Hunt, J. Li, M. Neeracher *et al.*, "Siri On-Device Deep Learning-Guided Unit Selection Text-to-Speech System," in *Interspeech*, 2017, pp. 4011–4015.

[23] G. Caruana and M. Li, "A Survey of Emerging Approaches to Spam Filtering," *ACM computing surveys (CSUR)*, vol. 44, no. 2, pp. 1–27, 2008.

[24] K. E. Case and R. J. Shiller, "Forecasting Prices and Excess Returns in the Housing Market," *Real Estate Economics*, vol. 18, no. 3, pp. 253–273, 1990.

[25] M. Castells, *End of Millennium*. John Wiley & Sons, 2010.

[26] ——, *The Power of Identity*. John Wiley & Sons, 2011, vol. 14.

[27] ——, *The Rise of the Network Society*. John wiley & sons, 2011.

[28] P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer, and R. Wirth, "CRISP-DM 1.0," 1999.

[29] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate, "Differentially Private Empirical Risk Minimization," *Journal of Machine Learning Research*, vol. 12, no. Mar, pp. 1069–1109, 2011.

[30] A. Chen, S. S. Kim, A. Dharmasiri, O. Russakovsky, and J. E. Fan, "Portraying Large Language Models as Machines, Tools, or Companions Affects What Mental Capacities Humans Attribute to Them," in *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, 2025, pp. 1–14.

[31] S. F. Chen and J. Goodman, "An Empirical Study of Smoothing Techniques for Language Modeling," *Computer Speech & Language*, vol. 13, no. 4, pp. 359–394, 1999.

[32] T. Colburn, *Philosophy and Computer Science*. Routledge, 2015.

[33] K. Collier, "A Researcher Tried to Buy Mental Health Data. It Was Surprisingly Easy," *NBC News*, Feb. 2023. [Online]. Available: https://www.nbcnews.com/tech/security/researcher-tried-buy-mental-health-data-was-surprisingly-easy-rcna70071

[34] P. Covington, J. Adams, and E. Sargin, "Deep Neural Networks for YouTube Recommendations," in *Proceedings of the 10th ACM conference on recommender systems*, 2016, pp. 191–198.

[35] D. Crevier, *AI: The Tumultuous History of the Search for Artificial Intelligence*. Basic Books, Inc., 1993.

[36] R. Cummings, G. Kaptchuk, and E. M. Redmiles, ""I Need a Better Description": An Investigation Into User Expectations For Differential Privacy," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '21. Association for Computing Machinery, 2021, p. 3037–3052. [Online]. Available: https://doi.org/10.1145/3460120.3485252

[37] N. N. G. de Andrade, "Data Protection, Privacy and Identity: Distinguishing Concepts and Articulating Rights," in *IFIP PrimeLife International Summer School on Privacy and Identity Management for Life*. Springer, 2010, pp. 90–107.

[38] DeepSeek, "DeepSeek," 2025, accessed 13 September 2025. [Online]. Available: https://www.deepseek.com/en

[39] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A Large-Scale Hierarchical Image Database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[40] A. Deshpande and H. Sharp, "Responsible AI Systems: Who are the Stakeholders?" in *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, ser. AIES '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 227–236. [Online]. Available: https://doi.org/10.1145/3514094.3534187

[41] F. Di Giuseppe, J. McNorton, A. Lombardi, and F. Wetterhall, "Global Data-Driven Prediction of Fire Activity," *Nature Communications*, vol. 16, no. 1, p. 2918, 2025.

[42] A. Diaa, L. Fenaux, T. Humphries, M. Dietz, F. Ebrahimianghazani, B. Kacsmar, X. Li, N. Lukas, R. A. Mahdavi, S. Oya, E. Amjadian, and F. Kerschbaum, "Fast and Private Inference of Deep Neural Networks by Co-designing Activation Functions," in *33rd USENIX Security Symposium (USENIX Security 24)*. Philadelphia, PA: USENIX Association, Aug. 2024, pp. 2191–2208. [Online]. Available: https://www.usenix.org/conference/usenixsecurity24/presentation/diaa

[43] T. Dienlin and S. Trepte, "Is the Privacy Paradox a Relic of the Past? An In-Depth Analysis of Privacy Attitudes and Privacy Behaviors," *European Journal of Social Psychology*, vol. 45, no. 3, pp. 285–297, 2015.

[44] H. J. Do, M. Brachman, C. Dugan, Q. Pan, P. Rai, J. M. Johnson, and R. Thawani, "Evaluating What Others Say: The Effect of Accuracy Assessment in Shaping Mental Models of AI Systems," *Proceedings of the ACM on Human-Computer Interaction*, vol. 8, no. CSCW2, pp. 1–26, 2024.

[45] C. Dwork, "Differential Privacy," in *International Colloquium on Automata, Languages, and Programming*. Springer, 2006, pp. 1–12.

[46] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating Noise to Sensitivity in Private Data Analysis," *Journal of Privacy and Confidentiality*, vol. 7, no. 3, pp. 17–51, 2016.

[47] U. Ehsan, R. Singh, J. Metcalf, and M. Riedl, "The Algorithmic Imprint," in *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, ser. FAccT '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 1305–1317.

[48] European Union, "General Data Protection Regulation (GDPR) - Chapter 1 General Provisions Article 4," https://gdpr-info.eu/chapter-1/, accessed 2025-10-28.

[49] ——, "The EU Artificial Intelligence Act," *European Union*, 2024.

[50] Federal Trade Commission, "FTC Launches Inquiry into AI Chatbots Acting as Companions," https://www.ftc.gov/news-events/news/press-releases/2025/09/ftc-launches-inquiry-ai-chatbots-acting-companions, accessed 2025-10-26.

[51] V. Feldman, "Does Learning Require Memorization? A Short Tale about a Long Tail," in *Proceedings of the 52nd annual ACM SIGACT symposium on theory of computing*, 2020, pp. 954–959.

[52] A. Ferrario and M. Loi, "How Explainability Contributes to Trust in AI," in *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, ser. FAccT '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 1457–1466. [Online]. Available: https://doi.org/10.1145/3531146.3533202

[53] F. Fioretto, P. Van Hentenryck *et al.*, *Differential Privacy in Artificial Intelligence: From Theory to Practice*. Now Publishers, Inc., 2025.

[54] G7 Leaders, "G7 Leaders' Statement on AI for Prosperity," https://g7.canada.ca/en/news-and-media/news/g7-leaders-statement-on-ai-for-prosperity/, Jun. 2025.

[55] K. Ganju, Q. Wang, W. Yang, C. A. Gunter, and N. Borisov, "Property Inference Attacks on Fully Connected Neural Networks using Permutation Invariant Representations," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, CCS 2018, October 15-19, 2018*. Toronto, Canada: ACM, 2018, pp. 619–633. [Online]. Available: https://doi.org/10.1145/3243734.3243834

[56] M. Garber, "How Google's Autocomplete Was...Created/Invented/Born," *The Atlantic*, Aug. 2013. [Online]. Available: https://www.theatlantic.com/technology/archive/2013/08/how-googles-autocomplete-was-created-invented-born/278991/

[57] Google, "Gemini," 2025, accessed 13 September 2025. [Online]. Available: https://gemini.google.com/

[58] Government of British Columbia, "Draft Artificial Intelligence Responsible Use Principles," 2024, last updated: November 26, 2024. [Online]. Available: https://digital.gov.bc.ca/ai/draft-responsible-use-principles/

[59] Government of Canada, "Foundation Framework for Treasury Board Policies," 2008, accessed 2025-09-20.

[60] ——, "Introduction to Policy," 2021, accessed 2025-09-20.

[61] ——, "Directive on Automated Decision-Making," 2025, last updated: July 24, 2025. [Online]. Available: https://www.tbs-sct.canada.ca/pol/doc-eng.aspx?id=32592

[62] N. Guha, C. M. Lawrence, L. A. Gailmard, K. T. Rodolfa, F. Surani, R. Bommasani, I. D. Raji, M.-F. Cuéllar, C. Honigsberg, P. Liang *et al.*, "AI Regulation Has Its Own Alignment Problem: The Technical and Institutional Feasibility of Disclosure, Registration, Licensing, and Auditing," *Geo. Wash. L. Rev.*, vol. 92, p. 1473, 2024.

[63] J. Gunawan, A. Pradeep, D. Choffnes, W. Hartzog, and C. Wilson, "A Comparative Study of Dark Patterns Across Web and Mobile Modalities," in *Proceedings of the ACM on Human-Computer Interaction, Volume 5, Issue CSCW2*, vol. 5, no. CSCW2. New York, NY, USA: Association for Computing Machinery, Oct. 2021. [Online]. Available: https://doi.org/10.1145/3479521

[64] P. Hacker, "Comments on the Final Trilogue Version of the AI Act," *Available at SSRN 4757603*, 2024.

[65] K. Hao, *Empire of AI: Dreams and Nightmares in Sam Altman's OpenAI*. Penguin Random House, 2025.

[66] A. Heydari, "The Canadian Tech Company that Changed its Mind About Using Your Tax Return to Sell Stuff," CBC Radio, 2020, accessed 2025-09-18.

[67] M. Hils, D. W. Woods, and R. Böhme, "Measuring the Emergence of Consent Management on the Web," in *Proceedings of the ACM Internet Measurement Conference*, 2020, pp. 317–332.

[68] B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep Models Under the GAN: Information Leakage from Collaborative Deep Learning," in *the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 603–618.

[69] D. A. Hoffman, "Defeating the Empire of Forms," *Virginia Law Review*, vol. 109, no. 7, pp. 1367–1427, 2023.

[70] J. Howard and S. Ruder, "Universal Language Model Fine-Tuning for Text Classification," *arXiv preprint arXiv:1801.06146*, 2018.

[71] T. Humphries, S. Oya, L. Tulloch, M. Rafuse, I. Goldberg, U. Hengartner, and F. Kerschbaum, "Investigating Membership Inference Attacks Under Data Dependencies," in *2023 IEEE 36th Computer Security Foundations Symposium (CSF)*. IEEE, 2023, pp. 473–488.

[72] L. Jamali, "AI Chatbot to be Embedded in Google Search," BBC, 2025, accessed 2025-10-20.

[73] J. Jargon and S. Kessler, "A Troubled Man, His Chatbot and a Murder-Suicide in Old Greenwich," https://www.wsj.com/tech/ai/chatgpt-ai-stein-erik-soelberg-murder-suicide-6b67dbfb, accessed 2025-10-26.

[74] B. Jayaraman and D. Evans, "Evaluating Differentially Private Machine Learning in Practice," in *the 28th USENIX Security Symposium*, Santa Clara, CA, 2019, pp. 1895–1912.

[75] S. E. Jones, *Against Technology: From the Luddites to Neo-Luddism*. Routledge, 2013.

[76] B. Kacsmar, "Perceptions and Practicalities for Private Machine Learning," Ph.D. dissertation, University of Waterloo, 2023.

[77] B. Kacsmar, V. Duddu, K. Tilbury, B. Ur, and F. Kerschbaum, "Comprehension from Chaos: Towards Informed Consent for Private Computation," in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 210–224. [Online]. Available: https://doi.org/10.1145/3576915.3623152

[78] B. Kacsmar, K. Tilbury, M. Mazmudar, and F. Kerschbaum, "Caring about Sharing: User Perceptions of Multiparty Data Sharing," in *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA: USENIX Association, Aug. 2022, pp. 899–916. [Online]. Available: https://www.usenix.org/conference/usenixsecurity22/presentation/kacsmar

[79] L. Katz, *Ill-Gotten Gains: Evasion, Blackmail, Fraud, and Kindred Puzzles of the Law*. University of Chicago Press, 1996.

[80] J. D. Kelleher and B. Tierney, *Data Science*. MIT press, 2018.

[81] P. G. Kelley, C. Cornejo, L. Hayes, E. S. Jin, A. Sedley, K. Thomas, Y. Yang, and A. Woodruff, "'There will be less privacy, of course': How and why people in 10 countries expect {AI} will affect privacy in the future," in *Nineteenth Symposium on Usable Privacy and Security (SOUPS 2023)*, 2023, pp. 579–603.

[82] F. Kessler, "Contracts of Adhesion–Some Thoughts About Freedom of Contract," *Columbia Law Review*, vol. 43, no. 5, pp. 629–642, 1943.

[83] S. S. Kim, E. A. Watkins, O. Russakovsky, R. Fong, and A. Monroy-Hernández, "Humans, AI, and Context: Understanding End-Users' Trust in a Real-World Computer Vision Application," in *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, 2023, pp. 77–88.

[84] B. Klimt and Y. Yang, "Introducing the Enron Corpus," in *CEAS*, vol. 45, 2004, pp. 92–96.

[85] S. Kokolakis, "Privacy Attitudes and Privacy Behaviour: A Review of Current Research on the Privacy Paradox Phenomenon," *Computers & security*, vol. 64, pp. 122–134, 2017.

[86] P. M. Krafft, M. Young, M. Katell, K. Huang, and G. Bugingo, "Defining AI in Policy versus Practice," in *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, ser. AIES '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 72–78. [Online]. Available: https://doi.org/10.1145/3375627.3375835

[87] P. Kumaraguru and L. F. Cranor, *Privacy Indexes: A Survey of Westin's Studies*. Carnegie Mellon University, School of Computer Science, Institute for ..., 2005.

[88] H.-P. H. Lee, L. Gao, S. Yang, J. Forlizzi, and S. Das, "" I Don't Know If We're Doing Good. I Don't Know If We're Doing Bad": Investigating How Practitioners Scope, Motivate, and Conduct Privacy Work When Developing {AI} Products," in *USENIX Security Symposium*, 2024.

[89] M. Li, W. Bickersteth, N. Tang, L. Cranor, J. Hong, H. Shen, and H. Heidari, "A Closer Look at the Existing Risks of Generative AI: Mapping the Who, What, and How of Real-World Incidents," in *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, vol. 8, no. 2, 2025, pp. 1561–1573.

[90] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy Beyond k-anonymity and l-diversity," in *2007 IEEE 23rd international conference on data engineering*. IEEE, 2006, pp. 106–115.

[91] Z. C. Lipton, "The Mythos of Model Interpretability: In Machine Learning, the Concept of Interpretability is Both Important and Slippery," *Queue*, vol. 16, no. 3, pp. 31–57, 2018.

[92] Y. Liu, R. Wen, X. He, A. Salem, Z. Zhang, M. Backes, E. De Cristofaro, M. Fritz, and Y. Zhang, "{ML-Doctor}: Holistic risk assessment of inference attacks against machine learning models," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 4525–4542.

[93] M. Longiaru, W. Negrón, B. J. Chen, A. Nguyen, S. N. Patel, and D. Calacci, "The 'Privacy' Trap: How "Privacy-Preserving AI Techniques" Mask the New Worker Surveillance and Datafication," https://datasociety.net/library/the-privacy-trap/, accessed 2025-10-26.

[94] L. MacCleery, "The New Surveillance State: Why Data Privacy Is Now Essential to Democracy," *Tech Policy Press*, Jun. 2025. [Online]. Available: https://techpolicy.press/the-new-surveillance-state-why-data-privacy-is-now-essential-to-democracy/

[95] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy Beyond k-anonymity," *Acm transactions on knowledge discovery from data (tkdd)*, vol. 1, no. 1, pp. 3–es, 2007.

[96] M. Mäntymäki, M. Minkkinen, T. Birkstedt, and M. Viljanen, "Defining Organizational AI Governance," *AI and Ethics*, vol. 2, no. 4, pp. 603–609, 2022.

[97] B. Marr, "A Short History Of ChatGPT: How We Got To Where We Are Today," Forbes, 2023, https://www.forbes.com/sites/bernardmarr/2023/05/19/a-short-history-of-chatgpt-how-we-got-to-where-we-are-today/.

[98] J. L. Mashaw and D. L. Harfst, *The Struggle for Auto Safety*. Harvard University Press Cambridge, MA, 1990.

[99] A. Mathur, G. Acar, M. Friedman, E. Lucherini, J. Mayer, M. Chetty, and A. Narayanan, "Dark Patterns at Scale: Findings from a Crawl of 11K Shopping Websites," *Proceedings of the ACM Human-Computer Interaction*, vol. 1, 2019.

[100] J. McCarthy, M. L. Minsky, N. Rochester, and C. E. Shannon, "A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence, August 31, 1955," *AI magazine*, vol. 27, no. 4, pp. 12–12, 2006.

[101] A. M. McDonald and L. F. Cranor, "The Cost of Reading Privacy Policies," *ISJLP*, vol. 4, p. 543, 2008.

[102] B. Merchant, *Blood in the Machine: The Origins of the Rebellion Against Big Tech*. Hachette UK, 2023.

[103] B. Montgomery, "Mother Says AI Chatbot Led her Son to Kill Himself in Lawsuit Against its Maker," https://www.theguardian.com/technology/2024/oct/23/character-ai-chatbot-sewell-setzer-death, accessed 2025-10-26.

[104] Nature Editorial, "Stop Talking About Tomorrow's AI Doomsday When AI Poses Risks Today," *Nature*, vol. 618, pp. 885–886, 2023.

[105] E. W. Ngai, Y. Hu, Y. H. Wong, Y. Chen, and X. Sun, "The Application of Data Mining Techniques in Financial Fraud Detection: A Classification Framework and An Academic Review of Literature," *Decision support systems*, vol. 50, no. 3, pp. 559–569, 2011.

[106] H. Nissenbaum, "Privacy in Context: Technology, Policy, and the Integrity of Social Life," in *Privacy in context*. Stanford University Press, 2009.

[107] ——, "Contextual Integrity Up and Down the Data Food Chain," *Theoretical Inquiries in Law*, vol. 20, no. 1, pp. 221–256, 2019.

[108] M. Nitzberg and J. Zysman, "Algorithms, Data, and Platforms: The Diverse Challenges of Governing AI," *Journal of European Public Policy*, vol. 29, no. 11, pp. 1753–1778, 2022.

[109] S. U. Noble, "Algorithms of Oppression: How Search Engines Reinforce Racism," in *Algorithms of oppression*. New York university press, 2018.

[110] M. Oates, Y. Ahmadullah, A. Marsh, C. Swoopes, S. Zhang, R. Balebako, and L. F. Cranor, "Turtles, Locks, and Bathrooms: Understanding Mental Models of Privacy Through Illustration," *Proceedings on Privacy Enhancing Technologies*, 2018.

[111] J. A. Obar and A. Oeldorf-Hirsch, "The Biggest Lie on the Internet: Ignoring the Privacy Policies and Terms of Service Policies of Social Networking Services," *Information, Communication & Society*, pp. 1–20, 2018.

[112] S. O'Connor, R. Nurwono, A. Siebel, and E. Birrell, "(Un)clear and (In)conspicuous: The Right to Opt-out of Sale Under CCPA," in *Proceedings of the 20th Workshop on Workshop on Privacy in the Electronic Society*. New York, New York: ACM, 2021, pp. 59–72.

[113] Office of the Privacy Commissioner of Canada, "PIPEDA in Brief," https://www.priv.gc.ca/en/privacy-topics/privacy-laws-in-canada/the-personal-information-protection-and-electronic-documents-act-pipeda/pipeda_brief/, 2019, accessed 2019-06-18.

[114] OpenAI, "ChatGPT," 2025, accessed 13 September 2025. [Online]. Available: https://chatgpt.com/

[115] D. Otis, "Private Clinics in Canada are Selling Personal Health Data: Study," *CTV News*, May 2025. [Online]. Available: https://www.ctvnews.ca/health/article/private-clinics-in-canada-are-selling-personal-health-data-study/

[116] N. Papernot, M. Abadi, Úlfar Erlingsson, I. Goodfellow, and K. Talwar, "Semi-Supervised Knowledge Transfer for Deep Learning from Private Training Data," in *the International Conference on Learning Representations*, Toulon, France, 2017.

[117] N. Papernot, P. McDaniel, A. Sinha, and M. P. Wellman, "SoK: Security and Privacy in Machine Learning," in *the 2018 IEEE European Symposium on Security and Privacy*. London, UK: IEEE, 2018, pp. 399–414.

[118] "People's AI Action Plan," https://peoplesaiaction.com/, 2025, accessed on September 10, 2025.

[119] B. Perrigo, "Is 'Sweatshop Data' Really Over?" *TIME*, Jul. 2025. [Online]. Available: https://time.com/7306153/ai-sweatshop-data-over/

[120] S. Petronio, *Boundaries of Privacy: Dialectics of Disclosure*. Suny Press, 2002.

[121] M. Posner, "The Fight to Get AI Right in Bangladesh, the World's Eighth-Most Populous Country," *Khoury News*, Jul. 2025. [Online]. Available: https://www.khoury.northeastern.edu/the-fight-to-get-ai-right-in-bangladesh-the-worlds-eighth-most-populous-country/

[122] K. Renaud, M. Volkamer, and A. Renkema-Padmos, "Why doesn't Jane protect her privacy?" in *International Symposium on Privacy Enhancing Technologies Symposium*. Springer, 2014, pp. 244–262.

[123] E. M. Renieris, *Beyond Data: Reclaiming Human Rights at the Dawn of the Metaverse*. MIT Press, 2023.

[124] K. Robison, "The Meta AI App Lets You 'Discover' People's Bizarrely Personal Chats," *WIRED*, Jun. 2025. [Online]. Available: https://www.wired.com/story/meta-artificial-intelligence-chatbot-conversations/

[125] S. Russell, P. Norvig, and A. Intelligence, "Artificial Intelligence: A Modern Approach," *Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs*, vol. 25, no. 27, pp. 79–80, 1995.

[126] P. Samarati, "Protecting Respondents Identities in Microdata Release," *IEEE transactions on Knowledge and Data Engineering*, vol. 13, no. 6, pp. 1010–1027, 2002.

[127] P. Samarati and L. Sweeney, "Protecting Privacy When Disclosing Information: k-anonymity and its Enforcement Through Generalization and Suppression," technical report, SRI International, 1998.

[128] R. Scammell, "Sam Altman Says Your ChatGPT Therapy Session Might Not Stay Private in a Lawsuit," *Business Insider*, Jul. 2025. [Online]. Available: https://www.businessinsider.com/chatgpt-privacy-therapy-sam-altman-openai-lawsuit-2025-7

[129] M. Shafieinejad, X. He, and B. Kacsmar, "Adopt a PET! An Exploration of PETs, Policy, and Practicalities for Industry in Canada," *arXiv preprint arXiv:2503.03027*, 2025.

[130] R. Shokri and V. Shmatikov, "Privacy-Preserving Deep Learning," in *the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015, pp. 1310–1321.

[131] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership Inference Attacks Against Machine Learning Models," in *the 2017 IEEE Symposium on Security and Privacy*. San Jose, CA, USA: IEEE, 2017, pp. 3–18.

[132] D. J. Solove, *Understanding Privacy*. Harvard university press, 2010.

[133] C. Song, T. Ristenpart, and V. Shmatikov, "Machine Learning Models that Remember Too Much," in *Proceedings of the 2017 ACM SIGSAC Conference on computer and communications security*, 2017, pp. 587–601.

[134] T. Stadler, B. Oprisanu, and C. Troncoso, "Synthetic Data–Anonymisation Groundhog Day," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 1451–1468.

[135] State of California Department of Justice, "California Consumer Privacy Act," https://oag.ca.gov/privacy/ccpa, 2018, accessed 2022-09-04.

[136] K. Tilbury, B. Kacsmar, and J. Hoey, "Towards Safety in Multi-agent Reinforcement Learning through Security and Privacy by Design," in *Coordination and Cooperation for Multi-Agent Reinforcement Learning Methods Workshop*, 2024.

[137] R. Turner, "Reactions of the Regulated: A Federal Labor Law Example," *Lab. Law.*, vol. 17, p. 479, 2001.

[138] US Department of Health and Human Services, "Health Information Privacy," https://www.hhs.gov/hipaa/index.html, accessed 2025-10-25.

[139] S. Vallor, *The AI Mirror: How to Reclaim Our Humanity in an Age of Machine Thinking*. Oxford University Press, 2024.

[140] J. Van Dijck, *The Culture of Connectivity: A Critical History of Social Media*. Oxford University Press, 2013.

[141] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is All You Need," *Advances in neural information processing systems*, vol. 30, 2017.

[142] P. Voigt and A. Von dem Bussche, "The EU General Data Protection Regulation (GDPR)," *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 2017.

[143] R. Wang, R. De Viti, A. Dubey, and E. M. Redmiles, "The Role of Privacy Guarantees in Voluntary Donation of Private Data for Altruistic Goals," *arXiv e-prints*, pp. arXiv–2407, 2024.

[144] J. Weizenbaum, "ELIZA—A Computer Program for the Study of Natural Language Communication Between Man and Machine," *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966.

[145] ——, *Computer Power and Human Reason: From Judgment to Calculation*. WH Freeman & Co, 1976.

[146] C. Westfall, "Non-Profit Helpline Shifts To Chatbots, Then Shuts Down Rogue AI," Forbes, 2023, accessed 2025-07-05.

[147] A. F. Westin, *Privacy and Freedom*. IG Publishing, 1967.

[148] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey *et al.*, "Google's Neural Machine Translation System: Bridging the Gap Between Human and Machine Translation," *arXiv preprint arXiv:1609.08144*, 2016.

[149] xAI, "Grok," 2025, accessed 13 September 2025. [Online]. Available: https://x.ai/grok

[150] G. N. Yannakakis and J. Togelius, *Artificial Intelligence and Games*. Springer, 2018, vol. 2.

[151] R.-J. Yew and B. Judge, "Anti-Regulatory AI: How 'AI Safety' is Leveraged Against Regulatory Oversight," *arXiv preprint arXiv:2509.22872*, 2025.

[152] R.-J. Yew, B. Marino, and S. Venkatasubramanian, "Red Teaming AI Policy: A Taxonomy of Avoision and the EU AI Act," in *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency*, 2025, pp. 404–415.

[153] R.-J. Yew, L. Qin, and S. Venkatasubramanian, "You Still See Me: How Data Protection Supports the Architecture of AI Surveillance," in *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, vol. 7, 2024, pp. 1709–1722.

Data Privacy and Computation Integrity in Machine Learning Scenarios: Some Issues and Approaches

Sabrina De Capitani di Vimercati[1], Sara Foresti[1], Stefano Paraboschi[2], Pierangela Samarati[1]
[1]Università degli Studi di Milano, Italy – Email: *firstname.lastname*@unimi.it
[2]Università degli Studi di Bergamo, Italy – Email: parabosc@unibg.it

Abstract

The increasing popularity of artificial intelligence and the wide adoption of machine learning has raised new privacy and security issues. Indeed, the massive amount of data on which AI and ML rely can include sensitive or company-confidential information. To enable the wide distribution of trained models as a service to interested final users, as well as the adoption of external parties for training models (e.g., to leverage the flexibility and economies of scale offered by cloud solutions), data need to be adequately protected. Also, the correctness and completeness of computations (e.g., for running or training models) delegated to external parties need to be guaranteed. In this paper, we consider some privacy and computation integrity issues arising when data and machine learning models or tasks are shared with external parties, together with possible solutions and research directions.

1 Introduction

The growing adoption of machine learning in our daily lives brings several advantages, such as better decision making and faster problem solving. Machine learning applications rely on the availability of massive amounts of data, together with powerful and efficient computational infrastructures and services used to train models and support increasingly complex analytics tasks. In many scenarios, these data can be provided by different owners, and often contain personal, sensitive, or company-confidential information that cannot be freely shared without proper protection. It is therefore important to ensure that both data confidentiality and privacy of data subjects (i.e., the entities to whom the data refer, such as users or organizations) are adequately protected throughout these data-sharing and analysis processes. Furthermore, privacy risks are not limited to data. When trained models are shared with external parties (e.g., for testing the performance of the model), they may also leak sensitive information about the data used for training. Although the protection of the confidentiality and privacy of data, models, and users is extremely important, this is not sufficient to guarantee that machine learning applications operate correctly, especially when the data sharing and preparation, model training, or inference phase of the machine learning life cycle is delegated to external providers.

Existing solutions for protecting confidentiality when sharing training datasets with external parties (e.g., [6, 7]) are based on cryptographic techniques such as homomorphic encryption that, together with secure multi-party computation, can be used to protect the dataset while used to train a machine learning model. More recent solutions combine cryptographic techniques with, for example, differential privacy, hardware-based trusted execution environments, or secure aggregation protocols used in federated learning (e.g., [12, 27]). Alternative solutions aim to avoid sharing real data by releasing synthetic data, that is, artificially generated data that preserve the statistical properties of the original data (e.g., [8]). While promising, these solutions based on synthetic data do not guarantee full protection (e.g., [2]). Training data can also be subject to attacks aimed at degrading the resulting model or altering its

behavior. One of the well-known threats is data poisoning that happens when an adversary intentionally injects, modifies, or manipulates data in the training dataset. Existing solutions aim at detecting and removing poisoning samples or to make learning algorithms more robust (e.g., [5, 25]). The privacy of data subjects represented in training datasets released to third parties is instead protected through the application of anonymization techniques such as k-anonymity [23] or differential privacy [14] and their variations. Although these techniques help to preserve privacy, they often reduce the accuracy of the resulting machine learning models (e.g., [24]). One of the main challenges is therefore to balance privacy and utility with respect to the analytical task for which the data will be used (e.g., [4, 9, 17]). Public or outsourced models are instead vulnerable to different types of inference attacks, such as membership inference, attribute inference, model inversion, and property inferences as well as model extraction attacks, which allow adversaries to reconstruct an equivalent model (i.e., a model that provides nearly the same results as the original model) and exploit it for further privacy violations (e.g., [22]). Periodic model updates (e.g., incremental learning) can also disclose changes in the underlying data such as the addition or removal of data subjects in unlearning settings. In addition to protecting the privacy of data subjects in the training dataset, also the sensitive information that users provide when using the model need to be protected. Indeed, a privacy-friendly model should not require users to disclose (either releasing directly or exposing indirectly) sensitive information. All these privacy and confidentiality issues become particularly significant in emerging collaborative learning scenarios (e.g., federated learning), where data are not centralized but remain distributed among multiple clients (e.g., [1, 15, 26]). In this context, the clients exchange only model updates (e.g., the gradients or parameter modifications computed during the training performed by the participants). These updates, however, can be exploited for inferring, for example, information about the underlying data (e.g., [12, 27]).

While one may assume an overall proper behavior, the use of external providers in the machine learning life cycle is clearly vulnerable to possible misbehavior by them, which can either be sloppy in their operation, or - even worse - intentionally misbehave, and therefore opportunistic in their responses. Since users as well as organizations are increasing their dependency on data and on the results of machine learning and data analytics tasks for their daily operations, data and computation integrity is paramount. Guaranteeing integrity means that techniques should be adopted to discover possible misbehavior that tampered with data or results of computations. When training datasets are stored and managed by external providers, it is necessary to ensure that no unauthorized modification occur (e.g., data poisoning). When prediction computations are outsourced, users should be able to verify that the results returned by the external provider are both correct and complete. Incorrect or incomplete results may arise from laziness, misconfiguration, or malicious tampering. Ensuring integrity therefore requires solutions that allow data owners and users to assess the integrity of every computation performed by the service provider (e.g., [10]).

To concretely analyze and discuss possible solutions to the confidentiality, privacy, and integrity issues discussed above, we focus on the classification problem since it is at the basis of a wide range of real-world applications (e.g., medical diagnosis, fraud and anomaly detection in financial systems, and spam filtering, just to name a few). Data classification is the process of training a machine learning model that is used for predicting the correct label of a given input data. The model learns patterns from a labeled training dataset (i.e., a dataset where each piece of information is associated with the correct label), and then uses this model to make predictions on new, unseen data. In the remainder of this paper, we address some data privacy, confidentiality, and integrity issues that can arise in the machine learning life cycle. Specifically, at data sharing and preparation time, we focus on the problem of protecting the privacy of data subjects while preserving the utility of the data for the downstream classification task (Section 2). We describe TA_DA, a target-aware data anonymization technique that allows data owners to contribute with their data to a classification task, anonymizing their data while maintaining utility

	Gender	Age	Smoker	Disease	Risk
t_1	F	65	former	emphysema	yes
t_2	M	54	never	cirrhosis	no
t_3	F	61	current	bronchitis	yes
t_4	M	72	never	cirrhosis	no
t_5	F	35	former	arthritis	no
t_6	M	68	former	bronchitis	yes
t_7	M	64	never	diabetes	no
t_8	F	15	never	asthma	no
t_9	F	39	never	osteoporosis	no
t_{10}	M	29	never	celiac	no
t_{11}	M	33	never	arthritis	no
t_{12}	F	54	current	bronchitis	yes
t_{13}	F	75	former	bronchitis	yes

Figure 1: An example of dataset including information about a sample of people

for the downstream task. At the model training time, we focus on the problem of minimizing leakage of sensitive information from the training data used for building a model made publicly accessible as well as from data of users interacting with the model. We describe PriSM, a solution for generating a privacy-friendly classifier that requires neither sensitive information nor information correlated with it for training a high accuracy classifier (Section 3). Finally, we address the problem of ensuring data and computation integrity by enabling the detection of incorrect or incomplete results returned by external providers (Section 4). We describe sentinels and twins that span all three phases of the machine learning data life cycle. In the following, we assume that the training dataset is a relational table R characterized by a set $\{a_1, \ldots, a_n, s, l\}$ of attributes, where s is sensitive and l is the label attribute. Figure 1 illustrates a running example showing a dataset collecting information about chronic disease and smoking habits of a sample of people. In particular, the relation keeps track, for each subject, of their gender (attribute `Gender`), age (attribute `Age`), smoking habits (attribute `Smoker` with values current, former, or never), chronic disease (attribute `Disease`), and risk of suffering from a Chronic Obstructive Pulmonary Disease (binary attribute `Risk`). Attribute `Disease` is considered sensitive, while `Risk` is the label attribute.

2 Target-aware anonymization

When data owners contribute their datasets to collaborative data analytics and machine learning tasks for training more comprehensive and accurate models, their datasets could be revealed to both the party in charge of the training phase and to other contributors. To provide protection to such datasets, solutions based on noise injection (e.g., differential privacy [14]) or on generalization and suppression (e.g., k-anonymity [23]) can be used. While noise injection may significantly distort the data and can compromise the quality of the downstream analysis, generalization preserves the truthfulness of the data. However, generalization needs to be applied with care as it inevitably causes information loss, which can reduce the accuracy of downstream analytical task, especially when applied on attributes that are good predictors of the classification label. Based on this observation, TA_DA (Target-Aware Data Anonymization) [4, 9] relies on generalization for protecting (i.e., anonymizing with k-anonymity [23] and ℓ-diversity [19]) data, while preserving as much as possible the utility of the protected data for the downstream collaborative analytics.

Anonymization with k-anonymity and ℓ-diversity operates by generalizing the values of the quasi-identifier attributes (i.e., attributes that, in combination, could be used to reconstruct the identity of data subjects through linking with external data sources) in the training dataset R in such a way that

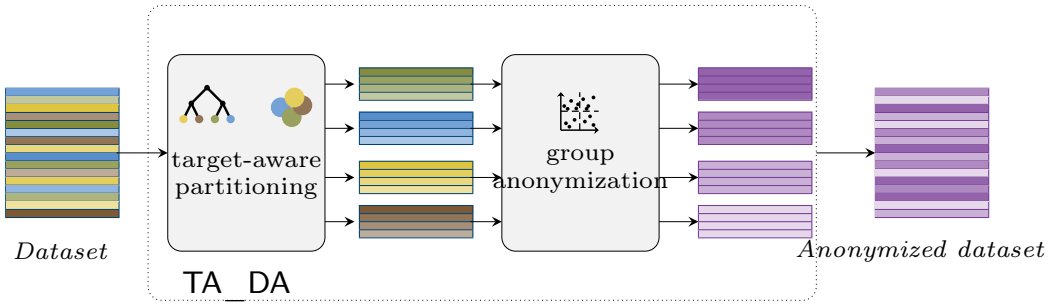


Figure 2: Overall working of TA_DA

each combination of generalized quasi-identifier values appears at least k times in the released dataset, and each group of tuples sharing the same generalized quasi-identifier contains at least ℓ well-represented (e.g., distinct) values for the sensitive attribute. Given a dataset R and privacy parameters k and ℓ, different generalization strategies can enforce k-anonymity and ℓ-diversity. Among all the solutions guaranteeing k-anonymity and ℓ-diversity, TA_DA aims at computing the one that preserves as much as possible those attributes on which the label attribute depend more, favoring generalization on the other attributes of the quasi-identifier. For instance, with reference to the example in Figure 1 where the quasi-identifier includes attributes `Gender` and `Age`, TA_DA would prefer a solution generalizing attribute `Gender` if `Risk` highly depends on `Age`. To achieve this goal, TA_DA operates in two steps (see Figure 2): *target-aware partitioning*, which partitions the tuples in R in groups according to the downstream classification task while satisfying k-anonymity and ℓ-diversity; and *group anonymization*, which anonymizes each partition generated by the previous step.

Target-aware partitioning. The goal of this phase is to partition tuples in the training dataset in such a way that each group includes tuples with similar values for the attributes in the quasi-identifier that are the best predictors of the label attribute (i.e., the attributes on which the label attribute value depends more). The intuition is that keeping in the same group tuples with similar values for an attribute limits the amount of generalization needed on it for achieving k-anonymity. The problem is that the best predictors of the label attribute are not known and must be learned from the dataset itself. TA_DA learns predictors and partitions the dataset according to their values by building a (k, ℓ)-compliant decision tree for the label attribute. A (k, ℓ)-compliant decision tree is a decision tree [16] having leaf nodes including at least k tuples with at least ℓ different values for the sensitive attribute. TA_DA builds a (k, ℓ)-compliant decision tree considering only quasi-identifying attributes for split operations. Other attributes, including the sensitive attribute are not considered. The sensitive attribute is not considered because there is the need to ensure diversity of its values in each generalized group. Other attributes are not affected by generalization, and therefore the anonymization process has no impact on them. Starting from the original training dataset, which corresponds to the root node, TA_DA recursively splits the dataset according to the values of a quasi-identifier attribute, generating children nodes. The split attribute, and its values used for partitioning tuples, is selected to maximize the quality of the decision tree (e.g., maximize information gain to have uniform labels in the leaf nodes), provided the resulting nodes guarantee that the tree is (k, ℓ)-compliant. The recursive split terminates when a stopping condition is satisfied (e.g., uniform values for the label in leaf nodes) or when any split would result in a decision tree that is not (k, ℓ)-compliant. By construction, the split attributes are those on which the label attribute depends more and each group contains tuples with similar values for these attributes, since the tuples in a node satisfy the same decision rule (i.e., the same set of conditions defined over the split attributes along the path from the root to the node). Figure 3 illustrates an

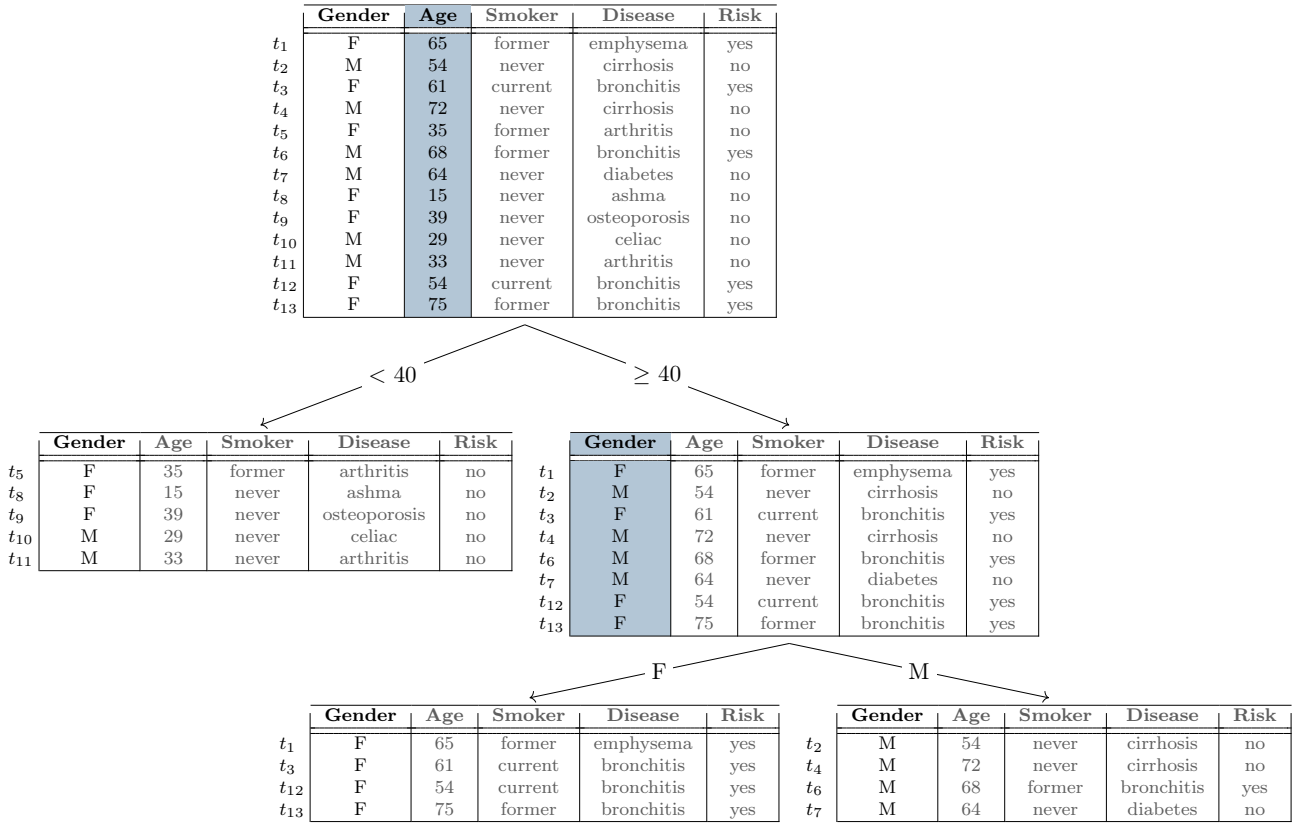


Figure 3: An example of (2,2)-compliant decision tree built over the relation in Figure 1

example of (2,2)-compliant decision tree built over the relation in Figure 1 and considering attributes `Gender` and `Age` in the quasi-identifier. The root node corresponds to the whole dataset that is split over attribute `Age` on condition < 40 (≥ 40, resp.), obtaining two partitions. For the first partition of tuples, representing patients with `Age`< 40, the process stops since all the tuples in the group have the same value for label attribute `Risk`. For the second, representing patients with age `Age`≥ 40, there is a further split on attribute `Gender`. The partitioning process then stops, since there are no other attributes in the quasi-identifier that could be used for further splitting the two leaf nodes. In the figure, attributes with gray values are those that cannot be used for splitting, and the attribute with a light blue background (gray in b/w printout) is, at each level, the one on which a further split is performed. The resulting decision tree is $(2, 2)$-compliant since each leaf node includes at least two tuples and has at least two different values for the sensitive attribute `Disease`.

Group anonymization. The goal of this phase is to independently anonymize each group of tuples corresponding to the leaf nodes of the (k, ℓ)-compliant decision tree computed in the previous phase. Intuitively, by generalizing each leaf node in a (k, ℓ)-compliant decision tree, the resulting dataset would be k-anonymous and ℓ-diverse. While any anonymization algorithm can be used for this second phase, TA_DA relies on Mondrian [18]. Mondrian has the advantage of leveraging a multi-dimensional spatial representation of the dataset similarly to the approach of construction of the (k, ℓ)-compliant decision tree. Each tuple is modeled as a point in a multi-dimensional space having a dimension for each attribute in the quasi-identifier. Mondrian recursively splits the multi-dimensional space in two partitions, in such a way that each partition includes at least k tuples with at least ℓ different values for the sensitive attribute. This process terminates when any further split would generate subspaces with less than k

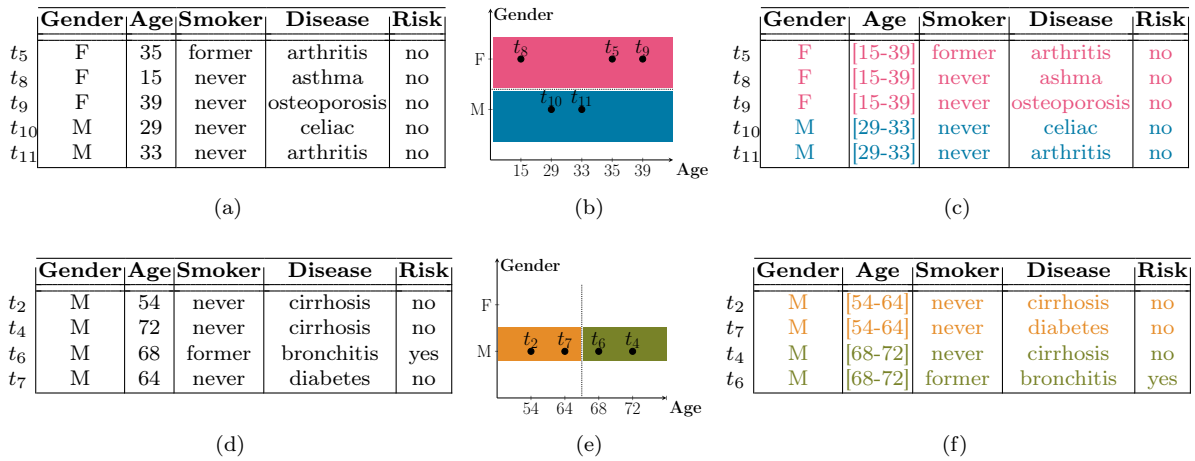


Figure 4: Anonymization of the first (a-c) and third (d-f) leaf nodes of the (2,2)-compliant decision tree shown in Figure 3

	Gender	Age	Smoker	Disease	Risk
t_1	F	[54-75]	former	emphysema	yes
t_3	F	[54-75]	current	bronchitis	yes
t_{12}	F	[54-75]	current	bronchitis	yes
t_{13}	F	[54-75]	former	bronchitis	yes
t_2	M	[54-64]	never	cirrhosis	no
t_7	M	[54-64]	never	diabetes	no
t_4	M	[68-72]	never	cirrhosis	no
t_6	M	[68-72]	former	bronchitis	yes
t_5	F	[15-39]	former	arthritis	no
t_8	F	[15-39]	never	ashma	no
t_9	F	[15-39]	never	osteoporosis	no
t_{10}	M	[29-33]	never	celiac	no
t_{11}	M	[29-33]	never	arthritis	no

Figure 5: An example of (2,2)-anonymous version of the relation in Figure 1

tuples or less than ℓ different values for the sensitive attribute. For all the tuples in each sub-space, the values of the attributes in the quasi-identifier are generalized to the same combination of values, thus guaranteeing that all the generalized tuples in the subspace are indistinguishable according to the quasi-identifier. The anonymized version of the dataset R is then obtained through the union of the anonymized groups of tuples corresponding to the leaves of the (k, l)-compliant decision tree. For instance, with reference to the (2,2)-compliant decision tree in Figure 3, the first leaf node includes five tuples and Mondrian performs a split over attribute `Gender` producing two groups on which attribute `Age` is then generalized. Similarly, for the third leaf node, Mondrian performs a split on attribute `Age`, producing two groups of tuples, in which attribute `Age` is then generalized. For the second leaf, Mondrian cannot perform a split on the attribute `Age` (which is the only attribute with different values for the tuples in the group), since the result would violate ℓ-diversity. Figure 4 illustrates the tuples in the first and third leaf nodes of the tree, their spatial representation, and the corresponding (2,2)-anonymous version. Figure 5 illustrates the resulting generalized table after the partitioning and the group anonymization described.

Summary and other issues. Combining target-aware partitioning with group anonymization, TA_DA enforces anonymization taking into consideration the downstream classification task. Indeed, TA_DA identifies and limits the amount of generalization on quasi-identifier attributes on which the label

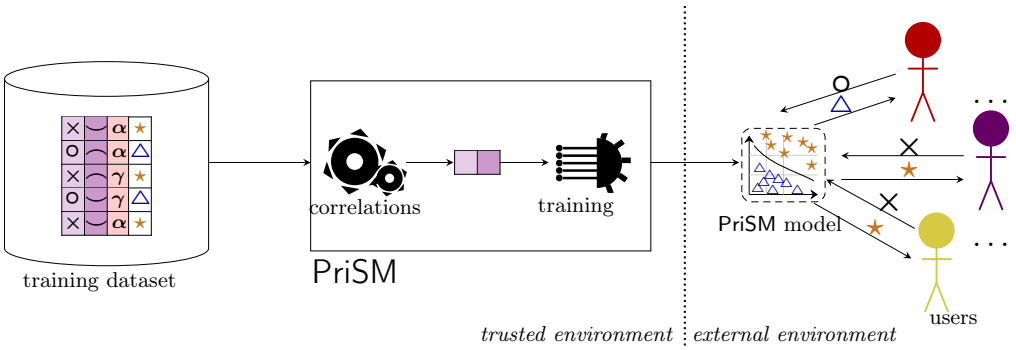


Figure 6: Overall working of PriSM

attribute depends more, thus producing a generalized table that satisfies k-anonymity and ℓ-diversity and that also preserves utility for data classification. Hence, a classifier trained over data anonymized using TA_DA can provide higher accuracy compared to a classifier trained over data anonymized with a classical (non target-aware) anonymization algorithm, as also demonstrated by the experimental analysis in [4]. Note that, besides classification, TA_DA can be used for *clustering* (a non-supervised learning task [21]) as downstream analytics. To this purpose, the target-aware partitioning phase needs to be revised to group tuples according to the same strategy adopted by clustering (e.g., minimize intra-cluster distance or similarity), while at the same time guaranteeing k-anonymity and ℓ-diversity. Besides the consideration of different downstream analytics, TA_DA could be extended to consider different privacy definitions, like differential privacy, with the goal of limiting noise and driving its injection based on the downstream machine learning task.

3 Privacy-friendly training

Allowing users (beyond the data or model owners) to rely on machine learning models trained using large and valuable data collections represents a clear advantage, especially for those who would not have the resources for collecting data and/or training models. However, this increased accessibility to machine learning models also increases the risk of exposing sensitive data in the training dataset. As a matter of fact, even if the training dataset is not released, its content might partially be exposed if the model is publicly released, or even through its simple use (e.g., observing the behavior of the model and its output). To address this challenge, we illustrate PriSM (Privacy-friendly Support vector Machine) [3], which aims at building a privacy-preserving classifier that guarantees protection of the sensitive attributes in the training dataset. For enabling final users to take advantage of machine learning models, while protecting the data used for training, PriSM removes from the training dataset not only sensitive attributes but also those (combination of) attributes that could indirectly expose them (e.g., through inferences). This strategy has a twofold advantage as it protects the sensitive information in the training dataset as well as the privacy of the users, who would not need to disclose sensitive information for obtaining predictions from the model. Intuitively, if an attribute is not considered in training, it will not be used (and therefore asked) for prediction. Given attributes (or sets thereof) that are considered sensitive, PriSM operates in two steps (Figure 6): it first identifies (sets of) other attributes that could leak the sensitive attribute values, and then trains the classifier in such a way to protect the sensitive attribute while limiting the impact on the accuracy of the model.

Sensitive correlation discovery. The goal of the first phase is to identify sets of attributes that could

leak the values of the sensitive attribute. A *sensitive correlation* is a set X of non-label attributes (i.e., $X \subset R \setminus \{l\}$, with l the label attribute) that can be used to predict the sensitive attribute $\in R$ (i.e., the values of s can be inferred from the values of the attributes in X and knowing X could lead to knowledge also of s). For instance, in the dataset in Figure 1, `Age` and `Smoker` can be used to infer (or reduce the uncertainty about) sensitive attribute `Disease`. Intuitively, a correlation between a set X of attributes and a sensitive attribute can be assessed by training a classifier that uses the sensitive attribute as target. A set X is a sensitive correlation if the classifier achieves a prediction accuracy for s higher than a predefined threshold. Since this approach is impractical in real-world scenarios, PriSM uses a correlation coefficient as a proxy for such evaluation. If the coefficient is greater than a given threshold, the correlation is considered sensitive and must be protected. Aimed at limiting information loss and at maximizing classification precision, PriSM distinguishes critical values from non-critical values in the domain of the sensitive attribute (e.g., a critical value for the attribute `Disease` could be emphysema, while a non-critical one could be flu), and considers sensitive only those correlations that can be exploited to infer critical values. As a matter of fact, not all values of the sensitive attribute are critical, and focusing only on the critical ones allows PriSM to be more precise in identifying sensitive correlations, excluding from consideration those correlations that are not considered sensitive, and hence possibly improve accuracy in classification. PriSM verifies the ability to leverage non-sensitive attributes to predict each critical value individually (assuming each value to represent a different class) as well as the set of all critical values together (assuming a binary classifier distinguishing critical from non-critical values), to better capture different sensitive correlations. Indeed, correlations discovered considering each critical value singularly taken may not be discovered when the critical values are considered together and vice versa. To identify attribute sets that could leak critical values of the sensitive attribute, PriSM leverages the natural monotonicity of sensitive correlations. If a subset X of attributes leaks the sensitive attribute, then any superset of X should be excluded from training, as it could expose the sensitive attribute. PriSM therefore examines subsets of non-sensitive (and non-label) attributes in R ordered by increasing cardinality, omitting any superset of already identified sensitive correlations. As an example, consider the dataset in Figure 1. PriSM first evaluates the correlation between each single attribute and the sensitive attribute `Disease`. Suppose that `Smoker` permits to infer critical values of `Disease`, while `Age` and `Gender` do not. In the second iteration, PriSM then checks only the correlation between the pair of attributes `Age` and `Gender` since all pairs including `Smoker`, although sensitive, are implicitly represented by the singleton sensitive correlation involving `Smoker`.

Classifier training. The goal of this second phase is to train a classifier that excludes from training the sensitive attribute as well as any sensitive correlation that could leak its critical values, while maximizing the predictive accuracy of the model. To this purpose, PriSM selects a subset of attributes in R to be used for training while ensuring that, for each sensitive correlation, at least one attribute involved in that correlation is excluded. PriSM extends classical Support Vector Machine (SVM) classifier, proposing an approach that aims at minimizing misclassification while excluding the sensitive attribute and at least one attribute from each sensitive correlation identified in the first phase. In addition, PriSM follows a parsimony principle, aimed at using at most a predefined number of attributes as predictors. This constraint has two advantages: it reduces the disclosure of unnecessary attributes and improves the efficiency of the training process.

Summary and other issues. Excluding the sensitive attribute and sensitive correlations from training, PriSM provides a privacy-preserving classifier able to protect the privacy of data subjects represented in the training dataset, as well as final users who do not need to release their potentially sensitive information for obtaining a prediction. While imposing constraints on the attributes to be used for training, PriSM maintains high accuracy in predicting the correct value for the label attribute target of the classification task, as demonstrated by experimental results on both real-world and synthetic

datasets [3]. Besides maintaining high accuracy in classification, PriSM is also characterized by a limited performance overhead in training a privacy-preserving classifier compared to a traditional classifier. It is interesting to note that, although PriSM has been specifically designed to define privacy-preserving SVM classifiers, the identification of sensitive correlations is independent of the underlying classifier. The first phase of the approach identifies sensitive correlations analyzing the dataset, independently from the specific classifier for which such dataset will be used. The first phase of the approach can then be used in combination with any classification model (e.g., non-linear or non-binary classifiers), as well as with any other machine learning model. The working of the classification/machine learning approach then needs to be revised for excluding from training the sensitive attribute as well as sensitive correlation maintaining, at the same time, high accuracy of prediction results.

4 Probabilistic integrity controls

Delegating the training or deployment of a machine learning model to an external party (e.g., a computational provider or a set of workers), for cost efficiency or to enable access to a wider set of final users, introduces potential risks to the correctness and completeness of the resulting computations. Indeed, workers may be lazy or malicious and partially or entirely omit the computation, returning an empty or randomly generated result. Lazy workers omit computations for economic reasons, with the goal of saving on computational resources elaborating a subset of the tuples in the input dataset while being payed for the elaboration of the whole dataset. Malicious workers instead intentionally misbehave, partially omitting the computation or returning incorrect result on purpose (e.g., to influence decision making processes). Integrity verification techniques verify the *correctness*, *completeness*, and *freshness* of computation results. Correctness refers to the verification that the computation has been executed in accordance with the algorithm defined by the data owner. Completeness ensures that the computation has been executed over the entire dataset, with no tuples omitted. Freshness consists of verifying that the computation has been performed over the most recent version of the dataset. Integrity verification techniques fall into two main categories: *deterministic* approaches, providing integrity guarantees with full confidence, and *probabilistic* approaches, providing such guarantees with a certain degree of confidence [11].

Deterministic techniques associate each computation result with a *Verification Object* (VO), which enables the recipient to verify the integrity of the returned result. The VO is constructed using an authenticated data structure (e.g., a Merkle hash tree or a skip list [13, 20]) built by the data owner over the dataset. The recipient of a computation then uses the VO, possibly together with information provided by the data owner, to verify the integrity of the result. Deterministic techniques provide full confidence in the integrity of computation results defined over the attribute(s) on which the authenticated data structure is built, but they do not offer guarantees for computations involving other attributes.

Probabilistic techniques provide probabilistic integrity guarantees, meaning that they provide, with a probabilistic degree of confidence, guarantee of a computation result to satisfy integrity property when passing integrity checks. These techniques are typically based on the injection of *control tuples* into the dataset. Their main advantage with respect to deterministic techniques is their broader applicability, as they are not restricted to computations defined over specific attributes. However, the integrity guarantee remains probabilistic because an integrity violation can be detected only if it affects the injected control tuples.

Probabilistic techniques are based on the injection of fake tuples (*sentinels*) with known results, or on the replication of a subset of the tuples (*twins*) in the dataset. A sentinel is a tuple generated by the client with known computation result: a result different from the expected one signals an integrity violation. Sentinels should be indistinguishable from genuine tuples, to prevent workers from selectively

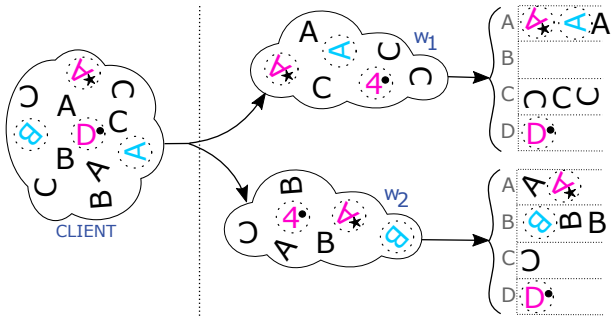


Figure 7: Sentinels and twins for integrity verification

processing them while omitting non-sentinel tuples. A twin is a tuple assigned to multiple workers: inconsistency in the results of twins signals an integrity violation. Twins are allocated to different workers to prevent them from detecting control tuples and selectively omitting their processing, while returning a coordinated result, to bypass integrity verification checks.

To assess the effectiveness of the combined adoption of sentinels and twins, several aspects must be investigated: How should sentinels be distributed among classes? Is it preferable to employ more twin pairs or more replicas of the same job? How many sentinels and twins are required to achieve a given integrity guarantee? In the following, we provide an answer to these questions, based on the analysis in [10]. Figure 7 illustrates the considered scenario, characterized by a client distributing classification jobs (i.e., the computation of the class associated with each tuple in the dataset) to a set of workers. In the figure, control jobs are circled, with twin pairs characterized by the same symbol. When omitting a classification job, a worker can return a random result (i.e., a randomly extracted a class), or opportunistically select the class that maximizes the probability of the omission to go undetected. In the following, we will refer to a worker omitting jobs as a lazy worker and, taking a safe approach, assume that lazy workers return an opportunistic answer for omitted jobs. We do not distinguish between lazy and malicious workers as the impact on integrity guarantees provided by sentinels and twins is the same for any omission, independently from the reason why the classification job has been omitted.

Distribution of sentinels in classes. To maximize the effectiveness of sentinels, it is important to properly tune their distribution in classes and their allocation to workers. The client has full control over how sentinels are distributed in the classes, since these jobs are generated ad hoc. Although the client may choose to take the distribution of the original dataset into account when distributing sentinels in classes, their effectiveness is maximized when they are uniformly distributed, that is, when injecting the same number of sentinels in each class. Otherwise, a lazy worker could exploit knowledge of the input data distribution in its opportunistic behavior. If sentinels are distributed according to genuine data distribution, by returning the most frequent class for each omitted classification job a lazy worker would correctly guess the sentinel's expected class for a high percentage (frequency of the most frequent class) of sentinels. Consider, as an example, a data distribution following Zipf's law with $\alpha = 1$ (Figure 8(a)), the injection of sentinels according to the same distribution as the data would result in 48% of them with label c_1 (see Figure 8(b), reporting sentinels added to each class on top of the corresponding bar). A lazy worker returning c_1 as a default value for omitted jobs would correctly guess almost half of the sentinels. Similarly, if the client distributes sentinels according to the normalized inverse of the input distribution, a lazy worker could simply return the least frequent class (D in the above example, see Figure 8(c)) for every omitted job, again correctly guessing the correct result of a high percentage of sentinels (48% in the example above). In contrast, when sentinels are uniformly distributed, lazy workers cannot leverage any knowledge of class frequencies to reduce the risk of the client detecting omissions.

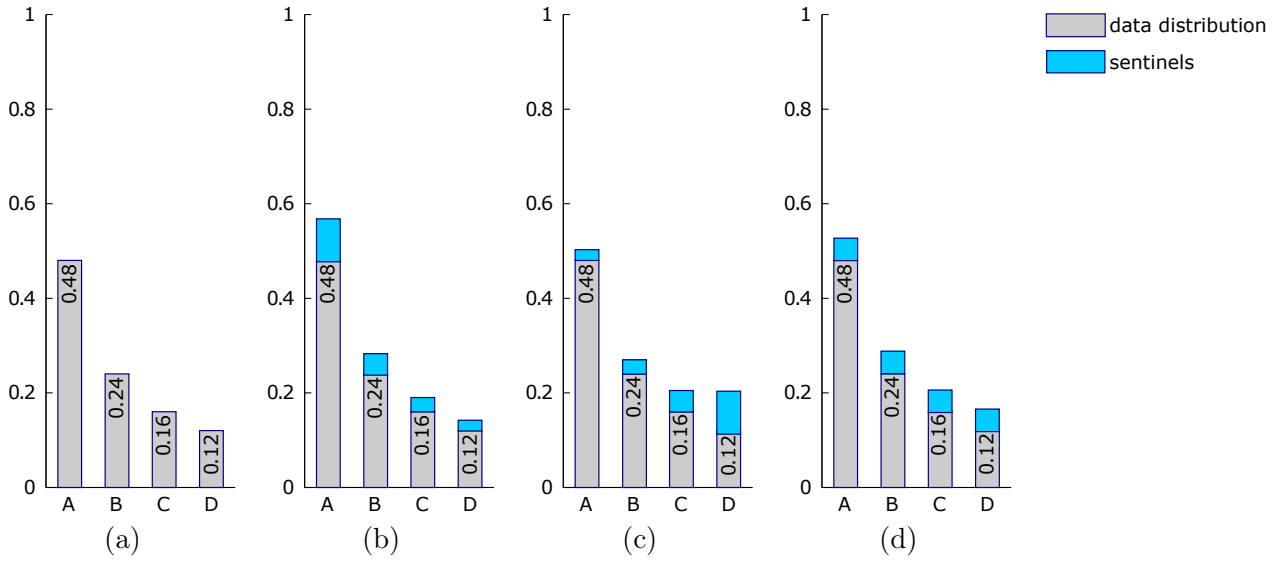


Figure 8: An example of probability mass function following a Zipf's law with $\alpha = 1$ (a) and sentinels distributed according to the data distribution (b), the normalized inverse data distribution (c), and uniformly (d)

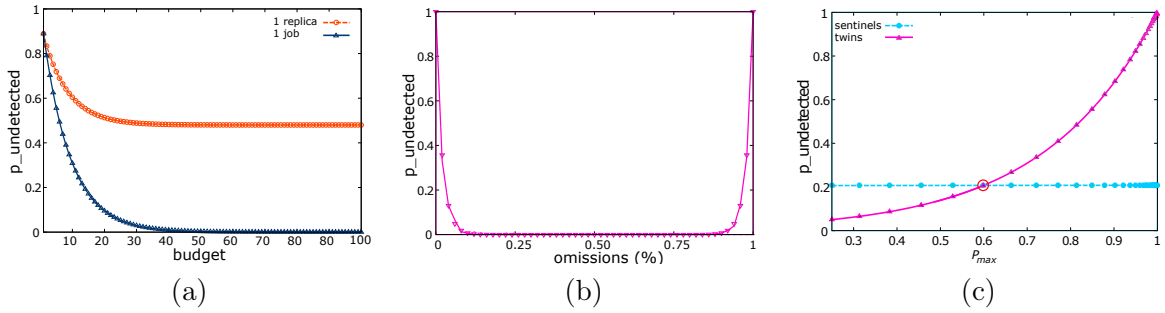


Figure 9: Probability that omissions go undetected using only twins and varying the number of twin pairs and the number of replicas (a), using only twins and varying the percentage of omissions (b), using sentinels and twins varying the frequency of the most frequent class (c)

With reference to the example above, a uniform distribution would result in distributing 25% of the sentinels in each class (Figure 8(d)), with therefore 25% probability for the lazy worker of correctly guessing omitted sentinels, independently from the data distribution.

Twin pairs vs replicas. Each job can be replicated multiple times, with each replica allocated to a different worker. Considering a fixed budget that the client can spend for integrity verification, it is worth noting that managing twins in pairs is substantially more effective, in terms of probability of detecting omission, than producing many replicas of the same classification job. In other words, it is preferable to generate a single additional copy for a larger number of different classification jobs rather than replicating one job many times. This is mainly due to the fact that the client cannot control the distribution of twins into classes, as jobs to be twinned are randomly extracted from the input dataset. Twins are naturally distributed according to the same distribution as the input dataset. Assuming lazy workers to return, for omitted jobs, the most frequent class (which minimizes classification error), the

probability of a lazy worker of correctly guessing all the replicas of a classification job is the probability of such a class. Therefore, investing all the budget replicating a single job is less effective than replicating different jobs. Indeed, especially if the original data distribution is very unbalanced, the probability for lazy workers of correctly guessing omitted replicas is very high. This behavior is confirmed in Figure 9(a), which reports the probability that an omission (50 omitted jobs by 49 workers out of 100 lazy workers) goes undetected as a function of either the number of distinct replicated jobs (continuous blue curve with triangles) or the number of replicas of a single job (dashed orange curve with circles). As visible from the figure, the probability of undetected omissions rapidly approaches zero as the number of replicated jobs increases, whereas it remains significantly higher (never going below the frequency of the most frequent class) when increasing the number of replicas for a single replicated job.

Sentinels and twins balance. Given a budget (in terms of additional jobs) for integrity verification, it is necessary to balance the adoption of sentinels and twins aiming at maximizing the probability of detecting omissions by workers. While in general twins are roughly twice as effective as sentinels (with one additional job, the client controls the behavior or two workers), a few sentinels are necessary to avoid extreme omissions to go undetected. As a matter of fact, the effectiveness of twins decreases in case of extreme omissions. Intuitively, if both the workers omitting the two replicas of a same job return the most frequent class, the omission goes undetected (the results are coherent, even if wrong). Then, if a large majority of workers omit their jobs, the probability of passing twins control is high, even when injecting a large number of twins. Figure 9(b) illustrates the probability of an omission to go undetected varying the percentage of omitted jobs, assuming to twin 5% of the jobs and that all workers are lazy. As visible from the figure, the probability quickly decreases when the percentage of omissions becomes non negligible, but it increases when the percentage of omissions becomes close to 1. Therefore, a few sentinels are always necessary to prevent extreme omissions to go undetected. The preference between the adoption of sentinels only or twins (with a few sentinels), based on their effectiveness, depends on the distribution in classes of the input dataset. More specifically, sentinels are preferred if $P_{max} > 0.5 \cdot (1 + c)$, with P_{max} the probability of the most frequent class and c the number of classes. If the data distribution is highly unbalanced, sentinels are more effective as the most effective strategy for the lazy worker (i.e., return the most frequent class for omitted jobs) implies a correct guess of the job result. Note that this threshold provides a nice and easy to use indication for the client. Once defined which between twins and sentinels is better to use, the client can size the number of control jobs based on either the client's budget or the threshold of probability of omissions going undetected. Figure 9(c) illustrates the probability of an omission to go undetected when using only sentinels (dashed blue line with bullets) and when using only twins (continuous pink line with triangles), varying the probability P_{max} of the most frequent class, assuming 10 workers, 4 of which are lazy and considering a probability of omitting jobs of 20%. As visible from the figure, the probability of omissions to go undetected is lower when using twins if P_{max} is small, it is lower when using sentinels when P_{max} grows.

Summary and other issues. While the results illustrated above permit to reason about the distribution of sentinels and twins among workers to verify their behavior, they are based on the assumption that (lazy) workers do not collude to maximize their probability of going undetected when omitting classification jobs. Colluding workers can indeed identify twin pairs assigned to them and hence return a coherent result, thus passing twin integrity check. In scenarios where workers can collude, twins need to be distributed in such a way to minimize the probability that a job and its replica are both allocated to colluding workers. Alternatively, twin jobs should be generated in such a way to make twin pairs unrecognizable as such. We also note that sentinels and twins have been primarily designed to verify precise computations (i.e., deterministic jobs, like the run of an algorithm), while real world scenarios need to account for computations characterized by an acceptable (limited) amount of errors.

5 Conclusions

We discussed some privacy and integrity issues that arise when external, potentially untrusted, parties are involved in machine learning tasks (e.g., for training or deploying a machine learning model). In fact, when releasing data to external parties for training models or releasing models for external use, it is crucial to maintain confidentiality of potentially sensitive or company-confidential information and to protect the privacy of the data subjects represented in the dataset or of the users interacting with the deployed model. Furthermore, if the party responsible for training or executing the model behaves lazily or maliciously, the resulting computation may be incomplete or incorrect, with potentially severe integrity issues. To address these concerns, we discussed solutions for constructing privacy-preserving machine learning models that preserve high utility. In addition, we presented mechanisms for verifying the correctness and completeness of results returned by potentially untrusted parties. The solutions discussed provide a foundation for designing privacy-aware and integrity-preserving machine learning techniques in (distributed) environments.

Acknowledgements

This work was supported in part by the EC under projects GLACIATION (101070141) and EdgeAI (101097300), by the Italian MUR under PRIN project POLAR (2022LA8XBH), and by project SERICS (PE00000014) under the MUR NRRP funded by the EU - NGEU. Project EdgeAI is supported by the Chips Joint Undertaking and its members including top-up funding by Austria, Belgium, France, Greece, Italy, Latvia, Netherlands, and Norway under grant agreement No. 101097300. Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union, the Chips Joint Undertaking, or the Italian MUR. Neither the European Union, nor the granting authority, nor Italian MUR can be held responsible for them.

References

[1] F. Ahmed, D. Sánchez, Z. Haddi, and J. Domingo-Ferrer, "MemberShield: A framework for federated learning with membership privacy," *Neural Networks*, vol. 181, January 2025.

[2] M. Annamalai, A. Gadotti, and L. Rocher, "A linear reconstruction approach for attribute inference attacks against synthetic data," in *Proc. of USENIX SEC*, PA, USA, August 2024.

[3] M. Barbato, A. Ceselli, S. De Capitani di Vimercati, S. Foresti, and P. Samarati, "PriSM: A Privacy-friendly Support vector Machine," in *Proc. of ESORICS*, Tolouse, France, September 2025.

[4] S. Barezzani, S. De Capitani di Vimercati, S. Foresti, V. Ghirimoldi, and P. Samarati, "TA_DA: Target-Aware Data Anonymization," *IEEE Transactions on Privacy*, vol. 2, pp. 15–26, 2025.

[5] B. Biggio and F. Roli, "Wild patterns: Ten years after the rise of adversarial machine learning," in *Proc. of ACM CCS*, Toronto, Canada, October 2018.

[6] H. Chabanne, A. De Wargny, J. Milgram, C. Morel, and E. Prouff, "Privacy-preserving text classification on deep neural network," *Neural Processing Letters*, vol. 57, no. 2, 2025.

[7] C. Chen, L. Wei, J. Xie, and Y. Shi, "Privacy-preserving machine learning based on cryptography: A survey," *ACM TKDD*, vol. 19, no. 4, May 2025.

[8] P. Coscia, S. Ferrari, V. Piuri, and A. Salman, "Synthetic and (Un) secure: evaluating generalized membership inference attacks on image data," in *Proc. of SECRYPT*, Bilbao, Spain, June 2025.

[9] S. De Capitani di Vimercati, S. Foresti, V. Ghirimoldi, and P. Samarati, "DT-Anon: Decision tree target-driven anonymization," in *Proc. of DBSec*, San Jose, CA, USA, July 2024.

[10] S. De Capitani di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, R. Sassi, and P. Samarati, "Sentinels and twins: Effective integrity assessment for distributed computation," *IEEE TPDS*, vol. 34, no. 1, pp. 108–122, January 2023.

[11] S. De Capitani di Vimercati, S. Foresti, and P. Samarati, *Query Integrity in Smart Environments*. Springer Nature Switzerland, 2025.

[12] R. de Laage, P. Yuhala, F. Wicht, P. Felber, and C. Cachin, "Practical secure aggregation by combining cryptography and trusted execution environments," in *Proc. of DEBS*, Gothenburg, Sweden, June 2025.

[13] P. Devanbu, M. Gertz, C. Martel, and S. Stubblebine, "Authentic third-party data publication," in *Proc. of DBSec*, Schoorl, The Netherlands, August 2000.

[14] C. Dwork, "Differential privacy," in *Proc. of International colloquium on automata, languages, and programming*, Venice, Italy, July 2006.

[15] A. R. Elkordy, Y. H. Ezzeldin, S. Han, S. Sharma, C. He, S. Mehrotra, S. Avestimehr *et al.*, "Federated analytics: A survey," *APSIPA Transactions on Signal and Information Processing*, vol. 12, no. 1, 2023.

[16] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, 2006.

[17] K. LeFevre, D. DeWitt, and R. Ramakrishnan, "Workload-aware anonymization," in *Proc. of KDD*, Philadelphia, PA, USA, August 2006.

[18] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan, "Mondrian multidimensional k-anonymity," in *Proc. of ICDE*, Atlanta, GE, USA, April 2006.

[19] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "ℓ-diversity: Privacy beyond k-anonymity," *ACM TKDD*, vol. 1, no. 1, March 2007.

[20] W. Pugh, "Skip lists: a probabilistic alternative to balanced trees," *Communications of ACM*, vol. 33, no. 6, pp. 668–676, 1990.

[21] X. Ran, Y. Xi, Y. Lu, X. Wang, and Z. Lu, "Comprehensive survey on hierarchical clustering algorithms and the recent developments," *Artificial Intelligence Review*, vol. 56, no. 8, pp. 8219–8264, August 2023.

[22] M. Rigaki and S. Garcia, "A survey of privacy attacks in machine learning," *ACM CSUR*, vol. 56, no. 4, pp. 1–34, 2023.

[23] P. Samarati, "Protecting respondents identities in microdata release," *IEEE TKDE*, vol. 13, no. 6, pp. 1010–1027, November/December 2001.

[24] N. Senavirathne and V. Torra, "On the role of data anonymization in machine learning privacy," in *Proc. of IEEE TrustCom*, Guangzhou, China, December 2020.

[25] Z. Tian, L. Cui, J. Liang, and S. Yu, "A comprehensive survey on poisoning attacks and countermeasures in machine learning," *ACM CSUR*, vol. 55, no. 8, December 2022.

[26] J. Vaidya, H. Yu, and X. Jiang, "Privacy-preserving SVM classification," *Knowledge and Information Systems*, vol. 14, pp. 161–178, 2008.

[27] R. Xu, B. Li, C. Li, J. Joshi, S. Ma, and J. Li, "TAPFed: Threshold secure aggregation for privacy-preserving federated learning," *IEEE TDSC*, vol. 21, no. 5, pp. 4309–4323, September/October 2025.

Beyond Data Privacy: New Privacy Risks for Large Language Models

Yuntao Du† , Zitao Li‡, Ninghui Li†, Bolin Ding‡
† Department of Computer Science, Purdue University
{ytdu, ninghui}@purdue.edu
‡ Alibaba
{zitao.l, bolin.ding}@alibaba-inc.com

Abstract

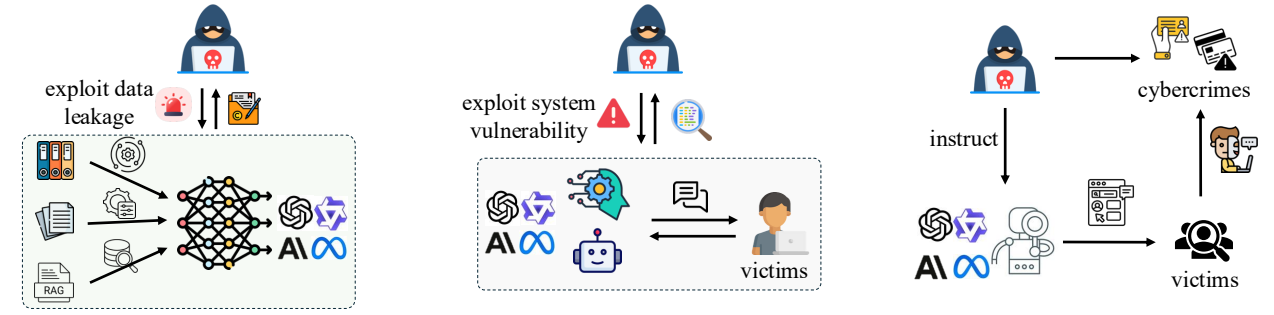
Large Language Models (LLMs) have achieved remarkable progress in natural language understanding, reasoning, and autonomous decision-making. However, these advancements have also come with significant privacy concerns. While significant research has focused on mitigating the data privacy risks of LLMs during various stages of model training, less attention has been paid to new threats emerging from their deployment. The integration of LLMs into widely used applications and the weaponization of their autonomous abilities have created new privacy vulnerabilities. These vulnerabilities provide opportunities for both inadvertent data leakage and malicious exfiltration from LLM-powered systems. Additionally, adversaries can exploit these systems to launch sophisticated, large-scale privacy attacks, threatening not only individual privacy but also financial security and societal trust. In this paper, we systematically examine these emerging privacy risks of LLMs. We also discuss potential mitigation strategies and call for the research community to broaden its focus beyond data privacy risks, developing new defenses to address the evolving threats posed by increasingly powerful LLMs and LLM-powered systems.

1 Introduction

Recent advancements in deep learning, particularly in natural language processing, have led to the development of large language models (LLMs). Over the past few years, LLMs have demonstrated impressive capabilities in understanding and generating human language. These models are rapidly growing in size and effectiveness, yielding breakthroughs and attracting increasing research and social attention. Beyond natural language understanding, their emergent abilities [173] have enabled them to achieve unparalleled performance on complex tasks. As a result, LLMs are no longer standalone models but are increasingly integrated as core decision-making components in larger systems, such as interactive chatbots [14, 127, 186] and autonomous agents [132, 144, 187].

However, this rapid development comes with growing concerns about its privacy implications. As a primary source of privacy risk, LLMs are trained on vast, internet-scale corpora that often contain sensitive personal information and copyrighted content. These data privacy risks are amplified when models are fine-tuned on private, proprietary datasets. Studies [32, 107, 109] have shown that LLMs can memorize and inadvertently leak training data across various model learning stages, raising issues related to training data extraction [25], copyright infringement [172], and test set contamination [129].

Beyond the risks of training data leakage, privacy threats also emerge from the integration of LLMs into larger, more complex systems, which we refer to as LLM-powered systems. These systems, especially those that use LLMs as decision-making engines in agent-based applications [36, 169], introduce new vulnerabilities and expand the potential attack surface for privacy violations. For instance, a user may share personal information with an LLM-based chatbot in order to receive personalized responses or



(a) Data privacy risks of various learning stages (i.e., pre-training, fine-tuning, and in-context learning).

(b) Privacy risks in LLM-powered systems, such as side-channel and information exfiltration.

(c) Privacy risks from malicious use of LLMs, such as automated profiling and social engineering.

Figure 1: Illustrations of different types of privacy risks posed by large language models.

suggestions. However, this information could be exfiltrated through side channels [23] or unintentionally disclosed by the model itself [158]. Such risks are not inherent to the LLM alone but emerge from the architecture of interactions between users, models, and other system components. As LLM-powered applications become increasingly widespread in both daily life and professional domains, these privacy risks become more prominent and urgent.

A third category of privacy threats arises from the advanced reasoning and autonomous decision-making capabilities of LLMs, which create new opportunities for malicious exploitation. These capabilities enable adversaries to automate sophisticated attacks at unprecedented scale and speed, substantially lowering the barrier to entry for cyberattacks. For instance, an attacker could instruct an LLM to infer sensitive attributes, such as a user's demographics, from their public online posts, leading to de-anonymization and other severe cybercrimes [49, 101]. Similarly, LLMs can be leveraged to launch large-scale, highly personalized social engineering campaigns, resulting in significant financial and societal consequences [106].

Positioning and Contribution Existing research has predominantly focused on training data privacy issues of LLMs. In contrast, less attention has been paid to the privacy threats posed by LLM-powered systems and the malicious use of LLMs. The privacy risks arising from LLM-powered systems and their malicious use represent a paradigm shift. These risks are not rooted in sensitive training data, but in the increasingly powerful autonomy of LLMs. As a result, existing data privacy frameworks may not always be well-suited to analyze or mitigate these emerging threats. This paper aims to bridge this gap by providing a comprehensive study of the new threat landscape introduced by LLMs. We systematically examine the privacy risks that arise from these models and their applications, and discuss potential mitigation strategies, calling for research efforts and greater public awareness to address these emerging privacy challenges.

Related Work Several studies [48, 94, 96, 99, 153, 176] have surveyed the data privacy risks of LLMs and explored mitigation strategies, However, these studies do not cover the emerging privacy threats posed by the increasing integration of LLM-powered systems and the potential malicious use of LLMs, which we identify as critical new privacy risks. Another line of research [18, 30, 97, 180] has examined the privacy implications of LLMs during user interactions. Our work builds on and extends these prior studies, providing a systematic and comprehensive analysis of the privacy risks posed by LLMs.

Roadmap In Section 2, we introduce the necessary background on LLMs and their key developmental trends. In Section 3, we discuss the primary data privacy risks associated with LLMs at different stages of training. In Section 4, we analyze the privacy threats of LLM-powered systems and discuss potential mitigations. In Section 5, we examine the privacy threats arising from the malicious use of LLMs.

Finally, we conclude and discuss future directions in Section 8.

2 Background on Large Language Models

A language model (LM) is a type of machine learning model for natural language processing. In general, an LM estimates the generative likelihood of sequences of words by predicting the probabilities of future or missing tokens[1]. In recent years, large language models (LLMs), trained on massive datasets for token prediction, have achieved unprecedented performance across a wide range of applications [205]. The scaling of these models has also unlocked remarkable emergent abilities not present in their smaller counterparts [173], including in-context learning [45], analogical reasoning [171], and the capacity to power autonomous agents [44].

With the rapid development of LLMs, LLM agents have emerged and become increasingly popular. These are intelligent entities powered by LLMs that are capable of autonomously carrying out complex tasks—such as conducting in-depth research or managing computer operations—while adapting to specific user needs [169]. This shift represents not only a major technological advancement but also a reimagining of human–machine interaction. Their impressive capabilities have already been applied in a wide range of domains, from chatbots [1] to professional tools like programming assistants [36].

Trends on LLMs Despite these advancements, the rapid evolution of LLMs also introduces new privacy risks. We identify three key trends that are closely related to growing privacy challenges:

- *Trained on Sensitive Data.* LLMs are trained on vast amounts of diverse data, which may include sensitive or copyrighted information. Moreover, the advanced capabilities of these models also enable them to access and utilize sensitive data through fine-tuning or in-context learning, especially when building proprietary models or personalized LLM-based applications [203]. This further increases the potential privacy risk of sensitive data that is trained on LLMs.

- *Incorporating into Popular Applications.* LLMs are increasingly embedded into core components of widely used software and professional tools. For example, they serve critical roles in domains such as code generation for software development, document analysis in legal and medical contexts, and as reasoning engines for autonomous agents that interact with external systems and platforms. As LLMs become ubiquitous in these daily applications, the surface area for privacy risks expands significantly, as adversaries can exploit these models in a wide range of sensitive contexts.

- *Growing Capability and Accessibility.* LLMs are rapidly evolving beyond text-only capabilities. Recent advances in vision-language models (VLMs) enable multimodal reasoning over both text and images [128, 162]. At the same time, access to powerful commercial and open-source models has become dramatically easier and much more affordable [108]. This combination of greater capability and accessibility empowers adversaries, giving them opportunities to exploit these advanced, autonomous systems to perform privacy-infringing attacks.

Taken together, these trends create a new privacy landscape that both amplifies data privacy concerns and introduces new privacy threats. The following sections will analyze these risks in detail, exploring their implications and potential mitigation strategies.

3 Data Privacy Risks in LLMs

LLMs themselves pose a significant privacy risk, as they, like other machine learning models, have been shown to memorize elements of their training data [22, 25]. These data privacy risks can arise at multiple

[1]A token refers to the smallest semantic unit processed by the model, which can be a character, subword, or word.

stages of the model learning process. From a parametric training perspective, LLMs typically undergo both a large-scale *pre-training stage* and subsequent *fine-tuning stages*. While LLMs are pre-trained and fine-tuned on massive corpora, and most state-of-the-art systems [14, 19, 53, 127, 186] do not disclose the provenance of their training data, concerns regarding potential privacy breaches have become increasingly pronounced. Moreover, due to their generative nature, LLMs support *in-context* learning, which provides a simple yet powerful mechanism for adapting inference behavior without modifying model parameters. These unique characteristics not only enhance the utility of LLMs but also broaden the avenues through which sensitive information may be exposed. In the following discussion, we examine two of the most prominent privacy threats to LLMs, membership inference and training data extraction, along with their privacy implications and potential enhancements.

3.1 Membership Inference Attack

Closely connected with Differential Privacy (DP) [54, 95], Membership Inference Attack (MIA) [152] has become a widely adopted approach for privacy auditing of machine learning (ML) models [119, 157]. MIA assesses how much a trained ML model reveals about its training data by determining whether specific query instances (or documents) were included in the dataset. In the context of LLMs, membership inference attacks have been applied at different stages to explore the privacy risk.

- *Pre-training Stage.* Early studies [110, 150, 181, 201] propose various membership signals that are derived from LLM's outputs to distinguish members from non-members. However, subsequent studies [111, 197] have identified fundamental flaws in the evaluation methodologies of prior studies. Particularly, the use of temporal data to separate members from non-members introduces subtle distributional shifts between the two groups, resulting in unreliable attack performance that does not accurately reflect true privacy leakage. Under more rigorous setups, existing MIAs [27, 37, 83, 198] perform no better than random guessing when targeting pre-training data of LLMs. This ineffectiveness is largely due to the fact that each data point is typically used only once during pre-training [118, 166], and the vast diversity of training corpora further dilutes the influence of a single example [51, 68].

- *Fine-tuned Stage.* Following the pre-training stage, the fine-tuning stage requires substantially fewer resources and focuses on adapting pre-trained models to domain-specific downstream tasks. However, fine-tuning datasets frequently contain personally identifiable information (PII) [32], copyrighted material [103], or sensitive organizational data [11]. MIAs against fine-tuned LLMs leverage techniques such as prompt calibration [61], hypothesis testing [113], and ensemble methods [199]. Moreover, recent work [58] has shown that human preference data used for alignment tuning (via Direct Preference Optimization (DPO) [185]) is also vulnerable to MIAs. Compared to attacks on pre-trained LLMs, MIAs against fine-tuned models are markedly more effective. This increased vulnerability arises because fine-tuning datasets are considerably smaller, and fine-tuning often involves multiple training epochs. These factors increase the model's memorization of sensitive data, posing heightened privacy threats in fine-tuned LLMs.

- *In-context Learning Stage.* Fine-tuning LLMs for specific domains involves non-trivial computation via parameter updates. In-Context Learning (ICL) [45] has emerged as a popular, more efficient adaptation paradigm, as it does not require modifying model parameters. In ICL, private data are provided as demonstrations within the prompt itself to guide the model's inference for a specific task. These demonstrations can be manually prepared [19] or dynamically retrieved from a private knowledge base using Retrieval-Augmented Generation (RAG) systems [15, 93, 151]. The vulnerability of ICL to MIAs has been explored using methods such as prompt injection [13],

analysis of semantic similarity [59], or measuring contextual influence [59]. Since ICL relies on only a few demonstration examples, each one has a significant impact on the model's output and performance, making these attacks highly effective at identifying private data.

Growing Threats While current MIAs have been effective at assessing privacy risks for fine-tuned and in-context data in LLMs, they have shown limited success against pre-trained models. However, this does not mean that pre-trained models are free from privacy risks. Two emerging research directions make MIAs more powerful, presenting a growing threat to the privacy of pre-training data. The first direction shifts the focus from analyzing individual data instances to examining larger collections of data. Recent studies [107] show that instead of detecting membership at the sentence level, aggregating membership signals across multiple sentences within a document can reliably reveal whether that dataset was included in training. This has been extended to paragraphs and entire collections, showing that these larger structures are also vulnerable to MIAs [138]. The second direction involves more advanced and computationally intensive attacks. Recent research [65] shows that training hundreds of shadow models [152] to exploit behavioral discrepancies can significantly enhance MIA effectiveness. These stronger attacks, building on prior successes against classifiers [21, 194], further elevate the risks to pre-trained data privacy. Together, these emerging research trends indicate that future improvements in MIAs could amplify the privacy risks of pre-trained LLMs.

Privacy Implications Effective MIAs on LLMs have serious trustworthy risks, such as the leakage of copyrighted content [52, 172] or test set contamination from evaluation benchmarks [129]. Moreover, MIAs can serve as a fundamental building block for more sophisticated attacks, such as training data extraction [25, 26], or as a core component in data auditing systems [74, 96, 107]. The issue is also highly relevant to recent ongoing lawsuits alleging unauthorized data use in model training, such as The New York Times vs. OpenAI [136], Getty Images vs. Stability AI[39] and Doe vs. GitHub[57].

Potential Mitigation Strategies Machine learning with differential privacy (DP-SGD [8] and PATE [131]) is an effective defense mechanism against privacy attacks, including MIAs. Studies [98, 135, 191] have applied DP-SGD to fine-tune LLMs on sensitive domains, which may degrade model utility under reasonable privacy budgets. For in-context learning scenarios, various DP-based techniques have been proposed, such as differential private prompt tuning [50, 70] and differential private synthetic text generation [161, 179]. In addition to the theoretical privacy guarantees that DP offers, many empirical privacy methods [71, 199] have shown effectiveness against MIAs. For example, LoRA [71], a widely used efficient fine-tuning method for LLMs, demonstrates better privacy preservation than full fine-tuning. However, these empirical defenses may be compromised when facing stronger MIAs [65], and there remains a notable trade-off between privacy and utility, especially for pre-trained LLMs.

3.2 Training Data Extraction

Training data extraction refers to the risk of partially or fully reconstructing samples from the training dataset by interacting with a trained LLM [22, 25, 78, 122, 146]. This threat goes beyond merely identifying whether a particular instance was part of the training set (as in membership inference); it involves recovering the training data itself, posing a more severe privacy threat. Similar to MIAs, data extraction attacks have been studied at various stages of the LLM training pipeline:

- *Pre-training Stage.* Research has shown that large amounts of data can be extracted from GPT-2 [140] by repeatedly querying it with different prompt prefixes [25]. Additionally, studies [84, 105] have demonstrated that LLMs can unintentionally leak personally identifiable information (PII), such as a person's full name, address, and phone number. More recently, studies [66] have used a probabilistic extraction approach to successfully recover pieces of copyrighted content, such as excerpts from books, from open-weight models.

- *Fine-tuned Stage.* Unlike the pre-training stage, where most data extraction attacks are conducted in a black-box setting, data extraction during fine-tuning often assumes that both the original and fine-tuned model weights are publicly available. For example, the popular reasoning model DeepSeek-R1 [43] is trained from DeepSeek-Base [42], both of which have open weights; however, the data used to train R1 has not been made public. Recent studies [116] propose efficient data selection strategies that can identify potential training data from a large data pool by matching the gradients from the base model to the fine-tuned model, demonstrating the feasibility of extracting fine-tuning data.

- *In-context Learning Stage.* Studies have shown that it is possible to recover the prompt used for in-context learning via model inversion [117] or prompt stealing [147, 160]. The privacy risks are heightened when using Retrieval-Augmented Generation (RAG) systems for LLMs, where attackers try to extract text data from a private database of RAG models through black-box access. Various attacks have been proposed to extract data from RAG systems, including adversarial prompt injection [75, 139, 170], agent-based attacks [81], and backdoor attacks [134].

Different approaches have been proposed to evaluate the effectiveness of data extraction attacks. The most widely used metric is eidetic memorization (or verbatim extraction) [25] and its variations [22, 78, 82, 164, 196]. This metric requires the model to reproduce the memorized data exactly when given an appropriate prompt. To relax this strict requirement, other studies propose approximate memorization metrics [78, 82], which measure the string or semantic similarity between the model's output and the training data as memorization efficacy.

Privacy Implications Data extraction from LLMs raises serious privacy and copyright concerns. Extraction not only produces a "copy" of training data but also reveals that a model has memorized such data internally. This evidence is central to ongoing legal debates about whether training an LLM on copyrighted material constitutes fair use. Furthermore, it poses a significant risk to the personal or proprietary data used in LLM-based applications such as RAG systems, underscoring the profound privacy challenges in deploying LLMs.

Growing Threats Recent studies [34, 66] highlight that since LLMs are probabilistic, the memorization should be assessed probabilistically. These studies introduce probabilistic discoverable extraction [66], which quantifies the likelihood that a model, under a given decoding scheme, will reproduce a verbatim target suffix when prompted with a specific prefix. This approach not only refines the understanding of memorization in LLMs but also shows how easily certain pieces of data can be extracted from these models. Further research [33] argues that measuring memorization solely by average extraction rates is insufficient. Instead, it should focus on identifying specific pieces of copyrighted or private text that are most likely to be memorized by the model. By enhancing extraction methods and pinpointing highly memorized fragments, these approaches increase the efficacy of data extraction, highlighting the potential for more targeted privacy violations.

4 Privacy Risks in LLM-Powered Systems

The integration of LLMs as core components in larger, complex systems introduces new vectors for privacy risks. For example, LLM-based chatbots, such as ChatGPT [127] and Claude [14], have become the primary interface through which users interact with LLMs. In these interactions, users often share personal narratives and sensitive details to seek advice or obtain personalized responses. This turns their conversation histories into a rich repository of private information, including personal preferences, habits, and even users' secrets [18, 85, 114]. Like other computer systems, these applications are vulnerable to side-channel attacks, where adversaries exploit indirect information leaks from the system to steal

data. Furthermore, the unique features of LLM-based applications, such as reasoning and memory mechanisms, provide additional attack surfaces. Below, we detail two prominent threats associated with LLM-powered systems: side channel attacks and information exfiltration.

4.1 Side Channel Attacks

Side-channel attacks [3, 40, 92] exploit indirect leakage of information through system behaviors such as timing, memory usage, and input/output patterns. These attacks can be especially severe in the context of LLM-based chatbots, which contain vast amounts of users' private conversation histories. Existing side-channel attacks against LLM-based chatbots can be categorized into three different types:

- *Inference Timing Attacks.* Inference timing attacks target the time it takes for an LLM to generate a response. To improve inference efficiency, modern LLMs are often optimized using data-dependent inference techniques, such as speculative decoding [112, 159], where a smaller, faster "draft" model predicts multiple future tokens, and the larger, main model verifies them in a single step. However, these optimizations introduce new vulnerabilities, as demonstrated in recent work [23, 155, 175, 193]. Specifically, the vulnerability arises from the number of drafted tokens that the main model accepts. If many tokens are accepted, the response is faster; if most are rejected, the system slows down. This acceptance rate depends on the predictability of the text. Attackers can craft specific inputs to measure these timing differences and infer the predictability of a user's hidden conversation history. By analyzing these server response time patterns, attackers can infer the topic or even specific characteristics of a user's private conversations without ever seeing the actual content.

- *Cache Timing Attacks.* Cache timing attacks exploit variations in how data is stored and accessed in a system's memory. In the context of LLM-based chatbots, inference services deployed on cloud resources need to handle a high volume of real-time requests while maintaining high throughput and low latency. One common optimization technique is prompt caching [137], where the attention key-value (KV) cache is reused across requests. In this method, the KV cache for a prompt is stored, and if a subsequent prompt shares a matching prefix with a cached prompt, the cached KV data for the prefix can be quickly retrieved. This results in faster processing times, specifically reducing the time to generate the first response token. However, the use of prompt caching introduces observable variations in response times based on the private input. When a prompt matches a cached prefix, the response is faster due to the cache hit, whereas non-matching prompts result in slower response times. By analyzing these timing differences, the attacker can learn the prefixes of other users' private inputs, potentially allowing them to identify or reconstruct the victim's entire prompt with high confidence [63, 156, 178, 208].

- *Keylogging Attacks.* Remote keylogging attacks focus on capturing the keystrokes entered by users during their interactions with chatbots. Unlike traditional keyloggers that require local access to the user's device, recent attacks [177] show it is possible to conduct this remotely by analyzing the timing and length of network packets exchanged between the user and the chatbot. By observing patterns in the size and timing of these encrypted packets, attackers can infer the length of the tokens being transmitted. Leveraging the predictable structure of language, these patterns can be used to reconstruct a user's input without any access to their device.

Privacy Implications Side-channel attacks exploit indirect signals such as inference latency, cache behavior, and packet length to infer sensitive attributes, without requiring the adversary to compromise the chatbot or the user directly. This makes them especially dangerous, as private information can be extracted passively, often without the awareness of either party.

4.2 Information Exfiltration

Information exfiltration refers to the unauthorized transfer of sensitive data from one context to another. In LLM-based (agent) applications, this occurs when attackers steal private information either through unintended leakage by the model itself or by maliciously manipulating the system to reveal user data, opening a new and highly exploitable attack surface [200]. We categorize existing approaches to information exfiltration into the following major categories.

- *Unintended Disclosure.* LLMs often lack awareness of privacy norms and the contextual boundaries of information flows [124]. As a result, they may inadvertently disclose sensitive information to inappropriate recipients. An LLM agent involved in a multi-round conversation may unintentionally repeat or expose previously shared user information, even when it is contextually irrelevant [38, 115, 148, 204]. For instance, it is undesirable for an LLM assistant to reveal that "John is talking to a few companies about switching jobs" while drafting an email to John's current manager, particularly without his consent. This risk increases when LLMs are tasked with complex operations that involve integrating multiple sources of user data, such as combining financial, location, and preference information for personalized recommendations [148]. LLMs struggle to track which information is appropriate to share, which makes these disclosures particularly insidious.

- *Leakage During Model Reasoning.* Recent advances in reasoning techniques encourage LLMs to generate explicit "thinking traces" or intermediate reasoning steps before producing final answers [154, 189]. While this improves task performance, studies show that reasoning traces themselves may leak sensitive user data, either accidentally or via targeted prompt injections [62]. For instance, a model assisting with medical scheduling could inadvertently include a patient's health condition in its hidden reasoning, which may later surface in outputs. This creates a difficult trade-off: increasing computational effort can make an agent's final answer more cautious, but it also encourages more verbose reasoning, thereby enlarging the attack surface.

- *Memory Leakage.* To improve personalization, many commercial LLM-powered chatbots, such as ChatGPT [127] and Gemini [162], have introduced long-term memory features that persist user information across sessions. These memories may include personal details such as location, occupation, or user preferences, stored explicitly in textual form to improve future responses. While convenient, such memory is highly sensitive and attractive to adversaries. Recent studies demonstrate that attackers can exfiltrate this data via carefully designed prompt injection attacks [133, 145]. For example, malicious content embedded in a piece of code or blog post could instruct LLM to reveal stored user memories, potentially encoding them into hidden channels (e.g., URLs or snippets of code) or misleading agents to perform actions like visiting websites that acquire user information, transferring the data to a remote adversary.

- *Insecure Tool Usage.* Tools refer to functions that LLM-based agents use to interact with external data or perform actions that modify the environment, such as writing files, clicking links on web pages, or generating and executing code. While the open-source community has made significant strides in developing secure Model Context Protocols (MCP) to ensure consistent and secure interactions with external data, these tools still pose substantial privacy risks. A recent study [35] demonstrated that MCP servers could be exploited as trojans to compromise user privacy. For example, a malicious weather MCP server, disguised as benign functionality, exploited legitimate banking tools to discover and extract user account balances. Although many vulnerabilities have been recognized [55, 88, 182], the increasing capabilities of agents with more tools at their disposal may lead to more potential privacy vulnerabilities.

- *Compromised Execution Environment.* The agent's execution environment can also be manipulated to exfiltrate sensitive information. For example, browser-based agents are highly susceptible to malicious prompt injections embedded in web pages [29, 100] or triggered by pop-ups [202], leading to the leakage of private user data. A recent study [87] further emphasizes that agents performing GUI-based tasks are particularly vulnerable, due to the misalignment between LLM behavior in conversational settings and their behavior in agent-based, browser-use contexts.

- *Leakge via Share Link.* Users may share their conversations with a chatbot (ChatGPT) through a built-in "Share" button on commercial chatbot platforms, allowing only those with the link to access the conversation. However, it has been shown [6] that these links can be discovered through search engines (Google Search). This can unintentionally expose users' conversation histories to the public. Furthermore, deleting a conversation from your ChatGPT history does not remove the public share link or prevent it from appearing in search engine results. Recognizing the potential for privacy breaches, OpenAI has addressed this issue by providing users with the option to control whether their chats are visible in search engine results, offering more control over users' privacy.

Privacy Implications Information exfiltration can occur even without the presence of an active attacker and can be unintentionally exposed through a model's internal reasoning traces or persistent memory. In this way, features originally intended to improve functionality (tool usage and share links), transparency (reasoning), and personalization (memory) become high-value attack surfaces. The ultimate consequence is a profound erosion of user trust, as interacting with a seemingly helpful LLM agent creates a persistent, exploitable record of their most sensitive information.

Vulnerability Detection and Mitigation Strategies To counter the threat of information exfiltration, research efforts are focused on both detecting vulnerabilities and developing active defenses. On the detection front, some works measure an LLM's capacity for privacy reasoning in ambiguous contexts to identify risks of unintended information disclosure [149, 190]. Another approach [16, 206] examines whether agents interacting with web interfaces adhere to the principle of data minimization, introducing benchmarks to systematically evaluate their compliance. In parallel, other efforts aim to build direct defenses against malicious attacks. This includes designing robust countermeasures for prompt injection attacks, which are a primary vector for information exfiltration [41, 168]. Despite these advancements, a comprehensive mitigation strategy capable of defending against the full spectrum of emerging exfiltration threats remains an open challenge [67, 200].

5 Privacy Risks from Malicious Use of LLMs

The increasingly impressive capabilities of LLMs have demonstrated remarkable potential across diverse fields, such as software engineering [187], human behavior simulation [132], and even assisting scientific discovery [17]. However, this progress presents a dual-use dilemma, as the very capabilities driving these innovations can also be misused for malicious purposes [24, 28, 56, 91, 183]. Specifically, LLMs amplify the risk of privacy violations in two ways:

- *Scaling Sophisticated Attacks.* LLMs can automate and execute privacy attacks that were previously prohibitive due to their complexity or high cost. By either assisting human adversaries or operating independently, they can enable privacy breaches at an unprecedented scale.

- *Democratizing Attack Capabilities.* LLMs lower the barrier for malicious actors by making powerful attack tools accessible to people with little to no expertise. This "democratization" allows individuals with limited knowledge to launch attacks that previously required specialized skills.

In this section, we introduce two emerging privacy risks of the malicious use of LLMs: automated profile inference and automated social engineering.

5.1 Automated Profile Inference

Individuals constantly generate digital footprints through their online activities, encompassing activities from social media comments and posts to shared photos and videos. While some of this data is inherently private (browse history), a vast amount of these activities (posts and comments) is publicly accessible. However, the public availability of this information does not eliminate privacy risks. An adversary can aggregate these seemingly innocuous public activities to construct a detailed personal profile, a process known as profiling [20, 47]. For instance, analyzing a Reddit user's most frequented subreddits could reveal their hobbies, while geotags in posted images could disclose their travel patterns or home location. Profiling is widely recognized as a privacy violation by legitimate privacy frameworks like GDPR[142], CCPA [126], and HIPAA [9].

Profiling based on unstructured and noisy data requires significant expertise and is considered too resource-intensive for large-scale privacy breaches [46]. The emergence of LLMs fundamentally alters this landscape. By leveraging their sophisticated understanding and reasoning capabilities, LLMs can automate the inference process, systematically analyzing vast digital footprints to infer sensitive attributes with minimal human intervention. This automation dramatically amplifies the threat, enabling profiling attacks at an unprecedented scale. A growing body of work has demonstrated the feasibility of LLM-driven profiling attacks [49, 158, 165]. In the following, we categorize these attacks along two primary axes: (i) the data modality they target and (ii) their level of automation.

Profiling Across Data Modalities With the increasing capabilities of LLMs in understanding different data modalities, various profiling attacks have been proposed by analyzing a user's activities across multiple types of data:

- *Profiling from Textual Activities.* Early LLM-based profiling attack [158] assumes that an adversary can access and scrape the public activities (posts and comments) of a pseudonymous user from the Internet. The adversary then instructs LLMs with prompts to infer predefined sensitive attributes (eight types of PIIs), within these textual activities. The results showed that powerful models like GPT-4 [127] can achieve performance comparable to human analysts, even when the humans have the advantage of accessing additional contextual information, which LLMs do not have.

- *Profiling from Visual Activities.* With the rise of Vision-Language Models (VLMs) [128, 162], research has expanded to include profiling from images and videos, which are ubiquitous on social media platforms like TikTok and Instagram. Specifically, one study [165] designed carefully crafted prompts using chain-of-thought reasoning [174] and automated zooming techniques to direct VLMs to focus on potentially sensitive details in the photos, thus enhancing privacy-infringing inferences. Another significant privacy risk arises from directly inferring a user's possible geo-location from their pictures [73, 80, 102, 188]. Research has shown that VLMs can outperform even the professional human players in GeoGuessr [2], which raises serious concerns regarding geographic privacy. However, these models are not infallible; they often exhibit significant regional biases, such as a tendency to over-predict well-known landmarks or locations heavily represented in their training data [73].

Different Levels of Automation in Profiling The privacy risks associated with the malicious use of LLMs depend heavily on the degree of automation involved in the attack. A highly automated and practical attack poses a much greater real-world privacy threat, as it reduces the need for human

adversaries, making it more cost-efficient and scalable. We categorize existing approaches into two types: semi-automated and fully automated, depending on their level of automation:

- *Semi-Automated Profiling.* The majority of current research falls into this category, where the core inference task is automated, but significant human effort is still required for data preparation and defining attack objectives. These systems are powerful in controlled settings but face two major limitations in real-world scenarios: (i) Reliance on curated data. Many studies [104, 158, 165, 188] focus on clean, curated textual or image data that is deliberately designed to contain sensitive information, allowing LLMs and VLMs to infer personal attributes. However, in real-world scenarios, user activities are typically noisy and may not be directly related to personal attributes. As a result, the performance of these semi-automated methods would likely degrade significantly when faced with raw, unfiltered activities. (ii) Predefined attribute targets. These attacks are typically configured to search for a fixed set of sensitive attributes (age, gender, location), which assumes the adversary already knows what to profile from users. However, in the real world, adversaries do not always know what sensitive attributes are present in a user's activities. This lack of predefined knowledge prevents the attacker from designing specific strategies to target particular attributes, further limiting the applicability of such attacks.

- *Fully-Automated Profiling.* To address the limitations of previous approaches, recent work has focused on developing end-to-end automated profiling systems. One example is AutoProfiler [49], an agent-based profiling framework that automatically scrapes, collects, and analyzes potentially sensitive activities from raw, noisy user data. By coordinating with four specialized LLM agents, AutoProfiler fully automates the process of inferring sensitive attributes. This eliminates the need for background knowledge or profiling expertise, making it highly scalable and suitable for deployment on web-scale platforms. Despite its weaker assumptions, the results show that the inferred attributes extend beyond PIIs, uncovering significant amounts of sensitive information. The move toward full automation has profound implications. It means that adversaries no longer need specialized expertise or prior knowledge to launch sophisticated, large-scale profiling attacks.

Privacy Implications Automated profiling inference can result in serious privacy breaches. One of the most well-known risks is de-anonymization [120, 121]. Study [49] shows that some Reddit users can be de-anonymized by inferring personal attributes from their public activities and comparing these with publicly available profiles, such as LinkedIn. The risk of de-anonymization increases when adversaries gain access to multiple profile databases or cross-reference a user's activities to construct more comprehensive profiles. In addition, sensitive information extracted from these online activities can also be exploited for severe cybercrimes like doxing and cyberbullying. We refer to [46, 49] for a deeper discussion of the consequences of exposing sensitive personal data.

Growing Threats Existing attacks exploit the in-context learning (ICL) capability of off-the-shelf LLMs to perform profiling tasks. While this approach is highly efficient and accessible, its performance could be suboptimal, as these models are not specifically designed for profiling. For example, studies show that even state-of-the-art VLMs are outperformed in geo-location identification tasks by PIGEON [64], an image model purpose-built for geolocation. This trend suggests that adversaries may design specialized profiling models that surpass generic LLMs, thereby enhancing attack effectiveness and posing even more severe privacy risks.

Challenges in Evaluation While various methodologies have been proposed to assess the profiling abilities of LLMs [49, 73, 158, 183], there is still no widely acknowledged benchmark to comprehensively evaluate the associated privacy risks. This issue partially stems from a fundamental ethical dilemma: creating a robust benchmark would require a large dataset of real users' activities with labeled, sensitive, ground-truth attributes. To address this, some researchers have proposed using synthetic datasets

generated by LLM agents [79, 192]. However, the behaviors and data produced by these agents may not accurately reflect the complexities of real human activity, limiting their validity and reliability [49]. In addition, evaluation becomes more complex for fully automated systems that perform open-ended inference without predefined attribute targets. Forcing the model to choose from a candidate list simplifies evaluation but fails to measure the model's true, unconstrained inference capabilities. Therefore, evaluating the profiling abilities of LLMs remains an open question. Designing effective evaluation approaches is a critical step toward understanding and mitigating these emerging privacy threats.

5.2 Automated Social Engineering

A social engineering attack exploits the psychological manipulation of human behavior to extract sensitive information, gain access to personal devices, share credentials, or perform other malicious activities that compromise digital security [125]. There are different types of social engineering, such as phishing, vishing, pretexting, and baiting. Over the past decades, social engineering attacks have resulted in numerous incidents, causing severe financial losses and privacy breaches [60, 123]. Most social engineering attacks follow four main stages [141]: (i) *Investigation.* The attacker gathers information about the target, often from public social media, job platforms, and online sources, to identify vulnerabilities. (ii) *Planning.* Based on the gathered information, the attacker develops a strategy, selecting tactics like phishing or impersonation to exploit weaknesses. (iii) *Contact.* The attacker establishes trust with the target, persuading them to take harmful actions such as clicking a malicious link or disclosing sensitive information. (iv) *Execution.* The attacker extracts sensitive data, installs malware, or otherwise compromises the target's system.

Social engineering attacks typically required significant human effort and expertise, and their success rates were often limited by defense mechanisms and human vigilance. For example, phishing emails could be easily recognized by telltale signs like grammatical errors or implausible scenarios [89]. However, the advent of LLMs has introduced a new dimension to social engineering threats, which we refer to as automated social engineering. Unlike traditional methods, LLM-driven attacks can be personalized and executed at scale. These models can automate and enhance all four major stages of a social engineering attack, increasing both effectiveness and efficiency, as detailed below.

Automated Investigation The purpose of this phase is to gather sufficient information about a target to personalize the attack and make it more convincing [123]. Adversaries may directly employ automated profiling strategies (as described in the previous section) to collect personal information. In addition, they may launch proactive information-gathering attempts by manipulating LLM-based chatbots to elicit sensitive details. In such scenarios, a chatbot convinces the user that certain personal information is required to complete a task. Because users often perceive LLMs as helpful assistants, they may willingly provide sensitive details, believing them to be necessary [12, 86, 90, 195]. Attackers can exploit this trust by embedding hidden, privacy-invasive prompts into a chatbot's behavior [158]. For example, a chatbot tasked with creating a travel itinerary might subtly request additional personal details—such as financial information or contact numbers—under the guise of improving the service. The risk is further amplified in multi-agent LLM systems, where multiple agents collaborate by asking for complementary pieces of information and together constructing a detailed personal profile of the victim [209]. These LLM-based information collection strategies dramatically reduce the cost and time required for reconnaissance while producing highly detailed and actionable intelligence about targets.

LLM-Aided Planning In this stage, LLMs could serve as powerful reasoning and analysis engines to help attackers design persuasive attack strategies. Specifically, LLMs can (i) propose tailored attack vectors—such as spear-phishing campaigns or impersonation scenarios, (ii) generate dialogue templates to sustain orchestrated interactions that gradually build trust [163], and (iii) dynamically adapt strategies, for example by suggesting follow-up messages when a target hesitates or fails to respond.

This capability transforms attack planning from a manual, experience-driven art into an automated process. Sophisticated, customized attack blueprints can be generated in minutes, removing the need for an experienced human attacker.

LLM-Enhanced Contact LLMs can be exploited not only to enhance interactions through existing contact channels but also to create entirely new avenues for reaching targets.

First, LLMs enhance traditional methods like phishing by generating persuasive, context-aware emails with remarkable speed. Studies show an LLM could draft a highly convincing spear-phishing email in just five minutes, a task that takes a human team several hours [31, 76, 77]. LLMs can also sustain convincing, real-time conversations that gradually build trust. When paired with generative deepfake technologies for images, video, or audio, impersonations become nearly indistinguishable from legitimate contacts [10, 89, 167]. This allows a single attacker to maintain persistent, personalized engagement across multiple platforms and scale their outreach to thousands of potential victims simultaneously.

Second, LLMs open new avenues for attack by exploiting the growing use of chatbots for emotional and psychological support [72, 207]. In this scenario, attackers deploy malicious chatbots that impersonate trusted friends or companions to establish a deep emotional connection with a victim. The proliferation of third-party platforms like the OpenAI GPT Store [4] and FlowGPT [5] makes it easy to distribute these deceptive chatbots to a wide audience. Once an emotional connection is established, adversaries can manipulate victims into disclosing sensitive information, transferring money under fraudulent pretenses, or even engaging in harmful behaviors [7].

LLM-Aided Execution Once trust is established and sensitive data is obtained, LLMs can assist attackers in carrying out malicious actions. These include: (i) leveraging stolen credentials to gain unauthorized access to systems [184], (ii) automating financial fraud, such as wire transfer scams [106], and (iii) orchestrating follow-on attacks, including malware distribution or pivoting to additional targets within a compromised network [143]. By reducing the need for manual effort, LLMs enable end-to-end, scalable, and highly sophisticated attack pipelines.

Privacy and Security Implications Automated social engineering represents a multifaceted threat to both privacy and security. It dramatically increases the risk of large-scale data leakage and financial loss. Attackers can harvest sensitive personal information, financial details, and corporate credentials with unprecedented efficiency [76]. The real-world consequences are staggering; in one recent incident, fraudsters used a combination of phishing and video-based deepfake impersonation to deceive an employee into authorizing a fraudulent $25 million transfer [106]. Beyond financial loss, certain strategies exploit users' trust or emotional reliance, inflicting psychological harm that can result in profound emotional distress. Thus, automated social engineering not only increases the efficiency of attacks but also expands the pool of potential victims, thereby amplifying the societal impact of privacy breaches.

Growing Threats With the rapid development of LLMs, automated social engineering attacks may become even more sophisticated and hard to detect. Multi-modal LLMs, for example, can generate coordinated text, audio, and video content to produce highly immersive impersonations that are nearly indistinguishable from authentic human interactions. Another concern lies in the emergence of autonomous agents capable of orchestrating end-to-end attack campaigns. Such agents could handle reconnaissance, planning, multi-turn conversations, and final exploitation without any human oversight [89]. These advancements suggest that future LLM-driven social engineering would progress beyond opportunistic scams toward coordinated, persistent, and large-scale operations capable of evading even advanced detection and defense systems.

Vulnerability Detection and Mitigation Strategies Several studies have examined the capabilities of LLMs in conducting social engineering and their impact on human users [69, 101, 184]. For example, a recent work [130] proposed embedding trigger–tag associations into vanilla LLMs through various insertion strategies. When the model is instructed to generate phishing emails, detectable tags are inserted into the output, enabling more effective detection of LLM-generated phishing content. However,

such safety enhancements for LLMs are limited in their real-world applicability. Adversaries can easily bypass them by locally deploying open-source and unconstrained LLMs without these safeguards. Thus, the challenge extends beyond detecting LLM-generated social engineering content to also identifying autonomous malicious activities carried out by LLM agents.

6 Conclusion

The rapid development and integration of LLMs into digital infrastructure and daily life have introduced a new frontier of privacy risks. In this paper, we systematically examined emerging threats of LLMs across three dimensions: (i) data privacy risks across various learning stages of LLMs; (ii) privacy risks in LLM-powered applications, including side channels and information exfiltration; and (iii) malicious use of LLMs, such as automated profiling and social engineering. We then discuss the real-world privacy implications of these threats and highlight the limitations of existing mitigation strategies. This paper helps to illuminate the privacy risks introduced by LLMs and advocates for greater social awareness of these challenges. We also call for research efforts that broaden their focus beyond data privacy and design new defenses to address these privacy threats.

References

[1] Chatgpt. https://chatgpt.com/.

[2] GeoGuessr. https://www.geoguessr.com/.

[3] Side-channel attack. https://en.wikipedia.org/wiki/Side-channel_attack.

[4] GPT Store. https://gptstore.ai/, 2023.

[5] FlowGPT. https://flowgpt.com/, 2025.

[6] OpenAI Is Pulling Shared ChatGPT Chats From Google Search. https://www.searchenginejournal.com/openai-is-pulling-shared-chatgpt-chats-from-google-search 552671/, 2025.

[7] Parents of teenager who took his own life sue OpenAI. https://www.bbc.com/news/articles/cgerwp7rdlvo, 2025.

[8] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.

[9] Accountability Act. Health insurance portability and accountability act of 1996. *Public law*, 104:191, 1996.

[10] Lin Ai, Tharindu Kumarage, Amrita Bhattacharjee, Zizhou Liu, Zheng Hui, Michael Davinroy, James Cook, Laura Cassani, Kirill Trapeznikov, Matthias Kirchner, et al. Defending against social engineering attacks in the age of llms. *arXiv preprint arXiv:2406.12263*, 2024.

[11] NIST AI. Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile, 2024.

[12] Mutahar Ali, Arjun Arunasalam, and Habiba Farrukh. Understanding Users' Security and Privacy Concerns and Attitudes Towards Conversational AI Platforms. In *2025 IEEE Symposium on Security and Privacy (SP)*, pages 298–316, 2025.

[13] Maya Anderson, Guy Amit, and Abigail Goldsteen. Is my data in your retrieval database? membership inference attacks against retrieval augmented generation. *arXiv preprint arXiv:2405.20446*, 2024.

[14] Anthropic. Claude.ai. `https://claude.ai`.

[15] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. 2024.

[16] Eugene Bagdasarian, Ren Yi, Sahra Ghalebikesabi, Peter Kairouz, Marco Gruteser, Sewoong Oh, Borja Balle, and Daniel Ramage. Airgapagent: Protecting privacy-conscious conversational agents. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 3868–3882, 2024.

[17] Daniil A Boiko, Robert MacKnight, and Gabe Gomes. Emergent autonomous scientific research capabilities of large language models. *arXiv preprint arXiv:2304.05332*, 2023.

[18] Hannah Brown, Katherine Lee, Fatemehsadat Mireshghallah, Reza Shokri, and Florian Tramèr. What does it mean for a language model to preserve privacy? In *Proceedings of the 2022 ACM conference on fairness, accountability, and transparency*, pages 2280–2292, 2022.

[19] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[20] Firearms Bureau of Alcohol, Tobacco and Explosives. Criminal Profilers.

[21] Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. Membership inference attacks from first principles. In *2022 IEEE symposium on security and privacy (SP)*, pages 1897–1914. IEEE, 2022.

[22] Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*, 2023.

[23] Nicholas Carlini and Milad Nasr. Remote timing attacks on efficient language model inference. *arXiv preprint arXiv:2410.17175*, 2024.

[24] Nicholas Carlini, Milad Nasr, Edoardo Debenedetti, Barry Wang, Christopher A Choquette-Choo, Daphne Ippolito, Florian Tramèr, and Matthew Jagielski. LLMs unlock new paths to monetizing exploits. *arXiv preprint arXiv:2505.11449*, 2025.

[25] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX security symposium (USENIX Security 21)*, pages 2633–2650, 2021.

[26] Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramer, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 5253–5270, 2023.

[27] Hongyan Chang, Ali Shahin Shamsabadi, Kleomenis Katevas, Hamed Haddadi, and Reza Shokri. Context-aware membership inference attacks against pre-trained large language models. *arXiv preprint arXiv:2409.13745*, 2024.

[28] Yaqub Chaudhary and Jonnie Penn. Large language models as instruments of power: New regimes of autonomous manipulation and control. *arXiv preprint arXiv:2405.03813*, 2024.

[29] Chaoran Chen, Zhiping Zhang, Bingcan Guo, Shang Ma, Ibrahim Khalilov, Simret A Gebreegziabher, Yanfang Ye, Ziang Xiao, Yaxing Yao, Tianshi Li, et al. The Obvious Invisible Threat: LLM-Powered GUI Agents' Vulnerability to Fine-Print Injections. *arXiv preprint arXiv:2504.11281*, 2025.

[30] Chaoran Chen, Daodao Zhou, Yanfang Ye, Toby Jia-jun Li, and Yaxing Yao. Clear: Towards contextual llm-empowered privacy policy analysis and risk generation for large language model applications. In *Proceedings of the 30th International Conference on Intelligent User Interfaces*, pages 277–297, 2025.

[31] Fengchao Chen, Tingmin Wu, Van Nguyen, Shuo Wang, Hongsheng Hu, Alsharif Abuadbba, and Carsten Rudolph. Adapting to Cyber Threats: A Phishing Evolution Network (PEN) Framework for Phishing Generation and Analyzing Evolution Patterns using Large Language Models. *arXiv preprint arXiv:2411.11389*, 2024.

[32] Xiaoyi Chen, Siyuan Tang, Rui Zhu, Shijun Yan, Lei Jin, Zihao Wang, Liya Su, Zhikun Zhang, XiaoFeng Wang, and Haixu Tang. The janus interface: How fine-tuning in large language models amplifies the privacy risks. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 1285–1299, 2024.

[33] A Feder Cooper, Aaron Gokaslan, Amy B Cyphert, Christopher De Sa, Mark A Lemley, Daniel E Ho, and Percy Liang. Extracting memorized pieces of (copyrighted) books from open-weight language models. *arXiv preprint arXiv:2505.12546*, 2025.

[34] A. Feder Cooper and James Grimmelmann. The Files are in the Computer: Copyright, Memorization, and Generative AI. *arXiv preprint arXiv:2404.12590*, 2024.

[35] Nicola Croce and Tobin South. Trivial Trojans: How Minimal MCP Servers Enable Cross-Tool Exfiltration of Sensitive Data. *arXiv preprint arXiv:2507.19880*, 2025.

[36] Cursor. Cursor Team. `https://www.cursor.com/`.

[37] Debeshee Das, Jie Zhang, and Florian Trantèr. Blind Baselines Beat Membership Inference Attacks for Foundation Models. In *2025 IEEE Security and Privacy Workshops*, pages 118–125, 2025.

[38] Saswat Das, Jameson Sandler, and Ferdinando Fioretto. Disclosure Audits for LLM Agents. *arXiv preprint arXiv:2506.10171*, 2025.

[39] Cerys Wyn Davies and Gill Dennis. Getty Images v Stability AI: the implications for UK copyright law and licensing. `https://www.pinsentmasons.com/out-law/analysis/getty-images-v-stability-ai-implications-copyright-law-licensing`.

[40] Edoardo Debenedetti, Giorgio Severi, Nicholas Carlini, Christopher A Choquette-Choo, Matthew Jagielski, Milad Nasr, Eric Wallace, and Florian Tramèr. Privacy side channels in machine learning systems. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 6861–6848, 2024.

[41] Edoardo Debenedetti, Ilia Shumailov, Tianqi Fan, Jamie Hayes, Nicholas Carlini, Daniel Fabian, Christoph Kern, Chongyang Shi, Andreas Terzis, and Florian Tramèr. Defeating prompt injections by design. *arXiv preprint arXiv:2503.18813*, 2025.

[42] DeepSeek-AI. DeepSeek-V3 Technical Report. 2024.

[43] DeepSeek-AI. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. 2025.

[44] Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web. *Advances in Neural Information Processing Systems*, 36:28091–28114, 2023.

[45] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.

[46] David M Douglas. Doxing: a conceptual analysis. *Ethics and information technology*, 18(3):199–210, 2016.

[47] John E Douglas, Robert K Ressler, Ann W Burgess, and Carol R Hartman. Criminal profiling from crime scene analysis. *Behavioral Sciences & the Law*, 4(4):401–421, 1986.

[48] Elias Dritsas, Maria Trigka, and Phivos Mylonas. A Survey on Privacy-Enhancing Techniques in the Era of Artificial Intelligence. In *Novel & Intelligent Digital Systems Conferences*, pages 385–392, 2024.

[49] Yuntao Du, Zitao Li, Bolin Ding, Yaliang Li, Hanshen Xiao, Jingren Zhou, and Ninghui Li. Automated Profile Inference with Language Model Agents. In *Workshop on AI Agents: Capabilities and Safety, Conference on Language Modeling (COLM)*, 2025.

[50] Haonan Duan, Adam Dziedzic, Nicolas Papernot, and Franziska Boenisch. Flocks of stochastic parrots: Differentially private prompt learning for large language models. *Advances in Neural Information Processing Systems*, 36:76852–76871, 2023.

[51] Michael Duan, Anshuman Suri, Niloofar Mireshghallah, Sewon Min, Weijia Shi, Luke Zettlemoyer, Yulia Tsvetkov, Yejin Choi, David Evans, and Hannaneh Hajishirzi. Do Membership Inference Attacks Work on Large Language Models? In *Conference on Language Modeling (COLM)*, 2024.

[52] André Vicente Duarte, Xuandong Zhao, Arlindo L Oliveira, and Lei Li. DE-COP: Detecting Copyrighted Content in Language Models Training Data. In *International Conference on Machine Learning*, pages 11940–11956, 2024.

[53] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[54] Cynthia Dwork. Differential Privacy. In *Automata, Languages and Programming*, pages 1–12, 2006.

[55] Junfeng Fang, Zijun Yao, Ruipeng Wang, Haokai Ma, Xiang Wang, and Tat-Seng Chua. We Should Identify and Mitigate Third-Party Safety Risks in MCP-Powered Agent Systems. *arXiv preprint arXiv:2506.13666*, 2025.

[56] Richard Fang, Rohan Bindu, Akul Gupta, Qiusi Zhan, and Daniel Kang. LLM agents can autonomously hack websites. *arXiv preprint arXiv:2402.06664*, 2024.

[57] Jose Florinio Farcon. Attribution Or Attrition? Doe 1 V. Github, Inc. As A Case For A Robust, Horizontal, Moral Right Of Attribution In Gen AI. 2024.

[58] Qizhang Feng, Siva Rajesh Kasa, SANTHOSH KUMAR KASA, Hyokun Yun, Choon Hui Teo, and Sravan Babu Bodapati. Exposing Privacy Gaps: Membership Inference Attack on Preference Data for LLM Alignment. In *International Conference on Artificial Intelligence and Statistics*, pages 5221–5229. PMLR, 2025.

[59] James Flemings, Bo Jiang, Wanrong Zhang, Zafar Takhirov, and Murali Annavaram. Estimating Privacy Leakage of Augmented Contextual Knowledge in Language Models. *arXiv preprint arXiv:2410.03026*, 2025.

[60] World Economic Forum. AI could empower and proliferate social engineering cyberattacks, 2024.

[61] Wenjie Fu, Huandong Wang, Chen Gao, Guanghua Liu, Yong Li, and Tao Jiang. Membership Inference Attacks against Fine-tuned Large Language Models via Self-prompt Calibration. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[62] Tommaso Green, Martin Gubri, Haritz Puerto, Sangdoo Yun, and Seong Joon Oh. Leaky Thoughts: Large Reasoning Models Are Not Private Thinkers. *arXiv preprint arXiv:2506.15674*, 2025.

[63] Chenchen Gu, Xiang Lisa Li, Rohith Kuditipudi, Percy Liang, and Tatsunori Hashimoto. Auditing Prompt Caching in Language Model APIs. *arXiv preprint arXiv:2502.07776*, 2025.

[64] Lukas Haas, Michal Skreta, Silas Alberti, and Chelsea Finn. Pigeon: Predicting image geolocations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12893–12902, 2024.

[65] Jamie Hayes, Ilia Shumailov, Christopher A Choquette-Choo, Matthew Jagielski, George Kaissis, Katherine Lee, Milad Nasr, Sahra Ghalebikesabi, Niloofar Mireshghallah, Meenatchi Sundaram Mutu Selva Annamalai, et al. Strong Membership Inference Attacks on Massive Datasets and (Moderately) Large Language Models. *arXiv preprint arXiv:2505.18773*, 2025.

[66] Jamie Hayes, Marika Swanberg, Harsh Chaudhari, Itay Yona, Ilia Shumailov, Milad Nasr, Christopher A. Choquette-Choo, Katherine Lee, and A. Feder Cooper. Measuring memorization in language models via probabilistic extraction. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 9266–9291, 2025.

[67] Pengfei He, Yue Xing, Shen Dong, Juanhui Li, Zhenwei Dai, Xianfeng Tang, Hui Liu, Han Xu, Zhen Xiang, and Charu C Aggarwal. Comprehensive Vulnerability Analysis is Necessary for Trustworthy LLM-MAS. *arXiv preprint arXiv:2506.01245*, 2025.

[68] Yu He, Boheng Li, Liu Liu, Zhongjie Ba, Wei Dong, Yiming Li, Zhan Qin, Kui Ren, and Chun Chen. Towards label-only membership inference attack against pre-trained large language models. In *USENIX Security*, 2025.

[69] Fred Heiding, Simon Lermen, Andrew Kao, Bruce Schneier, and Arun Vishwanath. Evaluating Large Language Models' Capability to Launch Fully Automated Spear Phishing Campaigns. In *ICML 2025 Workshop on Reliable and Responsible Foundation Models*.

[70] Junyuan Hong, Jiachen T Wang, Chenhui Zhang, Zhangheng LI, Bo Li, and Zhangyang Wang. Dp-opt: Make large language model your privacy-preserving prompt engineer. In *International Conference on Learning Representations*, 2024.

[71] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[72] Yining Hua, Hongbin Na, Zehan Li, Fenglin Liu, Xiao Fang, David Clifton, and John Torous. A scoping review of large language models for generative tasks in mental health care. *npj Digital Medicine*, 8(1):230, 2025.

[73] Jingyuan Huang, Jen-tse Huang, Ziyi Liu, Xiaoyuan Liu, Wenxuan Wang, and Jieyu Zhao. Vlms as geoguessr masters: Exceptional performance, hidden biases, and privacy risks. *arXiv preprint arXiv:2502.11163*, 2025.

[74] Zonghao Huang, Neil Zhenqiang Gong, and Michael K Reiter. A general framework for data-use auditing of ML models. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 1300–1314, 2024.

[75] Bo Hui, Haolin Yuan, Neil Gong, Philippe Burlina, and Yinzhi Cao. PLeak: Prompt Leaking Attacks against Large Language Model Applications. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 3600–3614, 2024.

[76] IBM. AI vs. human deceit: Unravelling the new age of phishing tactics. `https://www.ibm.com/think/x-force/ai-vs-human-deceit-unravelling-new-age-phishing-tactics`, 2023.

[77] IBM. With generative AI, social engineering gets more dangerous—and harder to spot. `https://www.ibm.com/think/insights/generative-ai-social-engineering`, 2025.

[78] Daphne Ippolito, Florian Tramèr, Milad Nasr, Chiyuan Zhang, Matthew Jagielski, Katherine Lee, Christopher A. Choquette-Choo, and Nicholas Carlini. Preventing Generation of Verbatim Memorization in Language Models Gives a False Sense of Privacy. In *Proceedings of the 16th International Natural Language Generation Conference*, pages 28–53, 2023.

[79] Shalini Jangra, Suparna De, Nishanth Sastry, and Saeed Fadaei. Protecting Vulnerable Voices: Synthetic Dataset Generation for Self-Disclosure Detection. *arXiv preprint arXiv:2507.22930*, 2025.

[80] Neel Jay, Hieu Minh Nguyen, Trung-Dung Hoang, and Jacob Haimes. Evaluating Precise Geolocation Inference Capabilities of Vision Language Models. 2025.

[81] Changyue Jiang, Xudong Pan, Geng Hong, Chenfu Bao, Yang Chen, and Min Yang. Feedback-Guided Extraction of Knowledge Base from Retrieval-Augmented LLM Applications. 2025.

[82] Nikhil Kandpal, Eric Wallace, and Colin Raffel. Deduplicating Training Data Mitigates Privacy Risks in Language Models. In *International Conference on Machine Learning*, pages 10697–10707, 2022.

[83] Gyuwan Kim, Yang Li, Evangelia Spiliopoulou, Jie Ma, Miguel Ballesteros, and William Yang Wang. Detecting training data of large language models via expectation maximization. *arXiv preprint arXiv:2410.07582*, 2024.

[84] Siwon Kim, Sangdoo Yun, Hwaran Lee, Martin Gubri, Sungroh Yoon, and Seong Joon Oh. ProPILE: probing privacy leakage in large language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 2023.

[85] Isadora Krsek, Anubha Kabra, Yao Dou, Tarek Naous, Laura A. Dabbish, Alan Ritter, Wei Xu, and Sauvik Das. Measuring, Modeling, and Helping People Account for Privacy Risks in Online Self-Disclosures with AI. *Proc. ACM Hum. Comput. Interact.*, 9(2):1–31, 2025.

[86] Isadora Krsek, Anubha Kabra, Yao Dou, Tarek Naous, Laura A. Dabbish, Alan Ritter, Wei Xu, and Sauvik Das. Measuring, Modeling, and Helping People Account for Privacy Risks in Online Self-Disclosures with AI. *Proc. ACM Hum.-Comput. Interact.*, 2025.

[87] Priyanshu Kumar, Elaine Lau, Saranya Vijayakumar, Tu Trinh, Elaine T Chang, Vaughn Robinson, Shuyan Zhou, Matt Fredrikson, Sean M Hendryx, Summer Yue, et al. Aligned LLMs are not aligned browser agents. In *The Thirteenth International Conference on Learning Representations*, 2025.

[88] Sonu Kumar, Anubhav Girdhar, Ritesh Patil, and Divyansh Tripathi. Mcp guardian: A security-first layer for safeguarding mcp-based ai system. *arXiv preprint arXiv:2504.12757*, 2025.

[89] Tharindu Kumarage, Cameron Johnson, Jadie Adams, Lin Ai, Matthias Kirchner, Anthony Hoogs, Joshua Garland, Julia Hirschberg, Arslan Basharat, and Huan Liu. Personalized Attacks of Social Engineering in Multi-turn Conversations–LLM Agents for Simulation and Detection. *arXiv preprint arXiv:2503.15552*, 2025.

[90] Jabari Kwesi, Jiaxun Cao, Riya Manchanda, and Pardis Emami-Naeini. Exploring User Security and Privacy Attitudes and Concerns Toward the Use of {General-Purpose}{LLM} Chatbots for Mental Health. In *34th USENIX Security Symposium (USENIX Security 25)*, pages 6007–6024, 2025.

[91] Marcus Law. Scam email cyber attacks increase after rise of ChatGPT. *Technology*, 2023.

[92] Liran Lerman, Gianluca Bontempi, and Olivier Markowitch. Side channel attack: an approach based on machine learning. *Center for Advanced Security Research Darmstadt*, 29:29–41, 2011.

[93] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.

[94] Haoran Li, Yulin Chen, Jinglong Luo, Yan Kang, Xiaojin Zhang, Qi Hu, Chunkit Chan, and Yangqiu Song. Privacy in Large Language Models: Attacks, Defenses and Future Directions. 2023.

[95] Ninghui Li, Wahbeh Qardaji, Dong Su, Yi Wu, and Weining Yang. Membership privacy: A unifying framework for privacy definitions. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 889–900, 2013.

[96] Qinbin Li, Junyuan Hong, Chulin Xie, Jeffrey Tan, Rachel Xin, Junyi Hou, Xavier Yin, Zhun Wang, Dan Hendrycks, Zhangyang Wang, et al. LLM-PBE: Assessing Data Privacy in Large Language Models. *Proceedings of the VLDB Endowment*, 17(11):3201–3214, 2024.

[97] Tianshi Li, Sauvik Das, Hao-Ping (Hank) Lee, Dakuo Wang, Bingsheng Yao, and Zhiping Zhang. Human-Centered Privacy Research in the Age of Large Language Models. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pages 581:1–581:4, 2024.

[98] Xuechen Li, Florian Tramer, Percy Liang, and Tatsunori Hashimoto. Large Language Models Can Be Strong Differentially Private Learners. In *International Conference on Learning Representations*, 2022.

[99] Yiming Li, Shuo Shao, Yu He, Junfeng Guo, Tianwei Zhang, Zhan Qin, Pin-Yu Chen, Michael Backes, Philip Torr, Dacheng Tao, et al. Rethinking data protection in the (generative) artificial intelligence era. *arXiv preprint arXiv:2507.03034*, 2025.

[100] Zeyi Liao, Lingbo Mo, Chejian Xu, Mintong Kang, Jiawei Zhang, Chaowei Xiao, Yuan Tian, Bo Li, and Huan Sun. EIA: ENVIRONMENTAL INJECTION ATTACK ON GENERALIST WEB AGENTS FOR PRIVACY LEAKAGE. In *The Thirteenth International Conference on Learning Representations*.

[101] Zilong Lin, Jian Cui, Xiaojing Liao, and XiaoFeng Wang. Malla: Demystifying real-world large language model integrated malicious services. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 4693–4710, 2024.

[102] Feiran Liu, Yuzhe Zhang, Xinyi Huang, Yinan Peng, Xinfeng Li, Lixu Wang, Yutong Shen, Ranjie Duan, Simeng Qin, Xiaojun Jia, Qingsong Wen, and Wei Dong. The Eye of Sherlock Holmes: Uncovering User Private Attribute Profiling via Vision-Language Model Agentic Framework. abs/2505.19139, 2025.

[103] Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Yuguang Yao, Chris Yuhao Liu, Xiaojun Xu, Hang Li, et al. Rethinking machine unlearning for large language models. *arXiv preprint arXiv:2402.08787*, 2024.

[104] Yupei Liu, Yuqi Jia, Jinyuan Jia, and Neil Zhenqiang Gong. Evaluating LLM-based Personal Information Extraction and Countermeasures. In *34st USENIX Security Symposium (USENIX Security 25)*, 2025.

[105] Nils Lukas, Ahmed Salem, Robert Sim, Shruti Tople, Lukas Wutschitz, and Santiago Zanella-Béguelin. Analyzing Leakage of Personally Identifiable Information in Language Models. In *44th IEEE Symposium on Security and Privacy*, pages 346–363, 2023.

[106] Kathleen Magramo. British engineering giant Arup revealed as 25 million deepfake scam victim, 2024.

[107] Pratyush Maini, Hengrui Jia, Nicolas Papernot, and Adam Dziedzic. LLM Dataset Inference: Did you train on my dataset? *Advances in Neural Information Processing Systems*, 37:124069–124092, 2024.

[108] Nestor Maslej, Loredana Fattorini, C. Raymond Perrault, Yolanda Gil, Vanessa Parli, Njenga Kariuki, Emily Capstick, Anka Reuel, Erik Brynjolfsson, John Etchemendy, Katrina Ligett, Terah Lyons, James Manyika, Juan Carlos Niebles, Yoav Shoham, Russell Wald, Tobi Walsh, Armin Hamrah, Lapo Santarlasci, Julia Betts Lotufo, Alexandra Rome, Andrew Shi, and Sukrut Oak. Artificial Intelligence Index Report 2025. *arXiv preprint arXiv:2504.07139*, 2025.

[109] Justus Mattern, Fatemehsadat Mireshghallah, Zhijing Jin, Bernhard Schölkopf, Mrinmaya Sachan, and Taylor Berg-Kirkpatrick. Membership Inference Attacks against Language Models via Neighbourhood Comparison. In *Findings of the Association for Computational Linguistics*, pages 11330–11343, 2023.

[110] Matthieu Meeus, Shubham Jain, Marek Rei, and Yves-Alexandre de Montjoye. Did the neurons read your book? document-level membership inference for large language models. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 2369–2385, 2024.

[111] Matthieu Meeus, Igor Shilov, Shubham Jain, Manuel Faysse, Marek Rei, and Yves-Alexandre de Montjoye. Sok: Membership inference attacks on llms are rushing nowhere (and how to fix it). In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 385–401, 2025.

[112] Xupeng Miao, Gabriele Oliaro, Zhihao Zhang, Xinhao Cheng, Zeyu Wang, Zhengxin Zhang, Rae Ying Yee Wong, Alan Zhu, Lijie Yang, Xiaoxiang Shi, et al. Specinfer: Accelerating large language model serving with tree-based speculative inference and verification. In *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 932–949, 2024.

[113] Fatemehsadat Mireshghallah, Archit Uniyal, Tianhao Wang, David K Evans, and Taylor Berg-Kirkpatrick. An empirical analysis of memorization in fine-tuned autoregressive language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1816–1826, 2022.

[114] Niloofar Mireshghallah, Hyunwoo Kim, Xuhui Zhou, Yulia Tsvetkov, Maarten Sap, Reza Shokri, and Yejin Choi. Can LLMs Keep a Secret? Testing Privacy Implications of Language Models via Contextual Integrity Theory. In *The Twelfth International Conference on Learning Representations*.

[115] Niloofar Mireshghallah, Hyunwoo Kim, Xuhui Zhou, Yulia Tsvetkov, Maarten Sap, Reza Shokri, and Yejin Choi. Can LLMs Keep a Secret? Testing Privacy Implications of Language Models via Contextual Integrity Theory. In *The Twelfth International Conference on Learning Representations*, 2024.

[116] John X Morris, Junjie Oscar Yin, Woojeong Kim, Vitaly Shmatikov, and Alexander M Rush. Approximating Language Model Training Data from Weights. *arXiv preprint arXiv:2506.15553*, 2025.

[117] John Xavier Morris, Wenting Zhao, Justin T Chiu, Vitaly Shmatikov, and Alexander M Rush. Language Model Inversion. In *The Twelfth International Conference on Learning Representations*, 2024.

[118] Niklas Muennighoff, Alexander Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A Raffel. Scaling data-constrained language models. *Advances in Neural Information Processing Systems*, 36:50358–50376, 2023.

[119] Sasi Kumar Murakonda and Reza Shokri. ML privacy meter: Aiding regulatory compliance by quantifying the privacy risks of machine learning. *arXiv preprint arXiv:2007.09339*, 2020.

[120] Arvind Narayanan and Vitaly Shmatikov. Robust de-anonymization of large sparse datasets. In *29th IEEE Symposium on Security and Privacy*, pages 111–125, 2008.

[121] Arvind Narayanan and Vitaly Shmatikov. De-anonymizing social networks. In *30th IEEE symposium on security and privacy*, pages 173–187, 2009.

[122] Milad Nasr, Javier Rando, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A. Feder Cooper, Daphne Ippolito, Christopher A. Choquette-Choo, Florian Tramèr, and Katherine Lee. Scalable Extraction of Training Data from Aligned, Production Language Models. In *The Thirteenth International Conference on Learning Representations*, 2025.

[123] Anam Naz, Muhammad Sarwar, Muhammad Kaleem, Muhammad Azhar Mushtaq, and Salman Rashid. A comprehensive survey on social engineering-based attacks on social networks. *International Journal of Advanced and Applied Sciences*, 11(4):139–154, 2024.

[124] Helen Nissenbaum. Privacy as contextual integrity. *Wash. L. Rev.*, 79:119, 2004.

[125] Information Technology Laboratory (NIST). Social Engineering, 2024.

[126] State of California Legislature. California Consumer Privacy Act of 2018. *Public law*, 2018.

[127] OpenAI. GPT-4 Technical Report, 2023.

[128] OpenAI. GPT-4V(ision) System Card, 2023.

[129] Yonatan Oren, Nicole Meister, Niladri S Chatterji, Faisal Ladhak, and Tatsunori Hashimoto. Proving test set contamination in black-box language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[130] Yan Pang, Wenlong Meng, Xiaojing Liao, and Tianhao Wang. Paladin: Defending LLM-enabled Phishing Emails with a New Trigger-Tag Paradigm. *arXiv preprint arXiv:2509.07287*, 2025.

[131] Nicolas Papernot, Martín Abadi, Úlfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. In *International Conference on Learning Representations*, 2017.

[132] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pages 1–22, 2023.

[133] Atharv Singh Patlan, S Ashwin Hebbar, Pramod Viswanath, and Prateek Mittal. Context manipulation attacks: Web agents are susceptible to corrupted memory. In *ICML 2025 Workshop on Computer Use Agents*.

[134] Yuefeng Peng, Junda Wang, Hong Yu, and Amir Houmansadr. Data extraction attacks in retrieval-augmented generation via backdoors. *arXiv preprint arXiv:2411.01705*, 2024.

[135] Natalia Ponomareva, Hussein Hazimeh, Alex Kurakin, Zheng Xu, Carson Denison, H Brendan McMahan, Sergei Vassilvitskii, Steve Chien, and Abhradeep Guha Thakurta. How to dp-fy ml: A practical guide to machine learning with differential privacy. *Journal of Artificial Intelligence Research*, 77:1113–1201, 2023.

[136] Audrey Pope. NYT v. OpenAI: The Times's About-Face. `https://harvardlawreview.org/blog/2024/04/nyt-v-openai-the-timess-about-face/`.

[137] Reiner Pope, Sholto Douglas, Aakanksha Chowdhery, Jacob Devlin, James Bradbury, Jonathan Heek, Kefan Xiao, Shivani Agrawal, and Jeff Dean. Efficiently scaling transformer inference. *Proceedings of machine learning and systems*, 5:606–624, 2023.

[138] Haritz Puerto, Martin Gubri, Sangdoo Yun, and Seong Joon Oh. Scaling up membership inference: When and how attacks succeed on large language models. *arXiv preprint arXiv:2411.00154*, 2024.

[139] Zhenting Qi, Hanlin Zhang, Eric P. Xing, Sham M. Kakade, and Himabindu Lakkaraju. Follow My Instruction and Spill the Beans: Scalable Data Extraction from Retrieval-Augmented Generation Systems. In *The Thirteenth International Conference on Learning Representations*, 2025.

[140] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[141] Tejal Rathod, Nilesh Kumar Jadav, Sudeep Tanwar, Abdulatif Alabdulatif, Deepak Garg, and Anupam Singh. A comprehensive survey on social engineering attacks, countermeasures, case study, and research challenges. *Information Processing & Management*, 62(1):103928, 2025.

[142] Protection Regulation. Regulation (EU) 2016/679 of the European Parliament and of the Council. *Regulation (EU)*, 679, 2016.

[143] MIT Technology Review. Cyberattacks by AI agents are coming. `https://www.technologyreview.com/2025/04/04/1114228/cyberattacks-by-ai-agents-are-coming/`, 2025.

[144] Erik Schluntz and Barry Zhang. Building effective agents. `https://www.anthropic.com/research/building-effective-agents`, 2024. Anthropic.

[145] Gregory Schwartzman. Exfiltration of personal information from ChatGPT via prompt injection. *arXiv preprint arXiv:2406.00199*, 2024.

[146] Avi Schwarzschild, Zhili Feng, Pratyush Maini, Zachary Lipton, and J Zico Kolter. Rethinking llm memorization through the lens of adversarial compression. *Advances in Neural Information Processing Systems*, 37:56244–56267, 2024.

[147] Zeyang Sha and Yang Zhang. Prompt Stealing Attacks Against Large Language Models. 2024.

[148] Yijia Shao, Tianshi Li, Weiyan Shi, Yanchen Liu, and Diyi Yang. PrivacyLens: Evaluating Privacy Norm Awareness of Language Models in Action. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

[149] Yijia Shao, Tianshi Li, Weiyan Shi, Yanchen Liu, and Diyi Yang. Privacylens: Evaluating privacy norm awareness of language models in action. *Advances in Neural Information Processing Systems*, 37:89373–89407, 2024.

[150] Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[151] Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. Replug: Retrieval-augmented black-box language models. *arXiv preprint arXiv:2301.12652*, 2023.

[152] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18, 2017.

[153] Victoria Smith, Ali Shahin Shamsabadi, Carolyn Ashurst, and Adrian Weller. Identifying and Mitigating Privacy Risks Stemming from Language Models: A Survey. 2023.

[154] Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

[155] Mahdi Soleimani, Grace Jia, In Gim, Seung-seob Lee, and Anurag Khandelwal. Wiretapping LLMs: Network side-channel attacks on interactive LLM services. *Cryptology ePrint Archive*, 2025.

[156] Linke Song, Zixuan Pang, Wenhao Wang, Zihao Wang, XiaoFeng Wang, Hongbo Chen, Wei Song, Yier Jin, Dan Meng, and Rui Hou. The early bird catches the leak: Unveiling timing side channels in llm serving systems. *arXiv preprint arXiv:2409.20002*, 2024.

[157] Shuang Song and David Marn. Introducing a New Privacy Testing Library in TensorFlow. 2022.

[158] Robin Staab, Mark Vero, Mislav Balunović, and Martin Vechev. Beyond Memorization: Violating Privacy via Inference with Large Language Models. In *The Twelfth International Conference on Learning Representations*, 2024.

[159] Mitchell Stern, Noam Shazeer, and Jakob Uszkoreit. Blockwise parallel decoding for deep autoregressive models. *Advances in Neural Information Processing Systems*, 31, 2018.

[160] Yicong Tan, Xinyue Shen, Yun Shen, Michael Backes, and Yang Zhang. On the Effectiveness of Prompt Stealing Attacks on In-the-Wild Prompts. In *IEEE Symposium on Security and Privacy*, pages 392–410, 2025.

[161] Xinyu Tang, Richard Shin, Huseyin A Inan, Andre Manoel, Fatemehsadat Mireshghallah, Zinan Lin, Sivakanth Gopi, Janardhan Kulkarni, and Robert Sim. Privacy-preserving in-context learning with differentially private few-shot generation. In *International Conference on Learning Representations*, 2024.

[162] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[163] Knostic Team. Jailbreaking Social Engineering via Adversarial Digital Twins, 2024.

[164] Kushal Tirumala, Aram H. Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. Memorization Without Overfitting: Analyzing the Training Dynamics of Large Language Models. In *Advances in Neural Information Processing Systems*, 2022.

[165] Batuhan Tömekçe, Mark Vero, Robin Staab, and Martin Vechev. Private attribute inference from images with vision-language models. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, pages 103619–103651, 2024.

[166] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[167] Nikolaos Tsinganos, Panagiotis Fouliras, and Ioannis Mavridis. Leveraging Dialogue State Tracking for Zero-Shot Chat-Based Social Engineering Attack Recognition. *Applied Sciences*, 13, 2023.

[168] Eric Wallace, Kai Xiao, Reimar Leike, Lilian Weng, Johannes Heidecke, and Alex Beutel. The instruction hierarchy: Training llms to prioritize privileged instructions. *arXiv preprint arXiv:2404.13208*, 2024.

[169] Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Jirong Wen. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345, 2024.

[170] Yidan Wang, Yanan Cao, Yubing Ren, Fang Fang, Zheng Lin, and Binxing Fang. PIG: Privacy Jailbreak Attack on LLMs via Gradient-based Iterative In-Context Optimization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pages 9645–9660, July 2025.

[171] Taylor Webb, Keith J Holyoak, and Hongjing Lu. Emergent analogical reasoning in large language models. *Nature Human Behaviour*, 7(9):1526–1541, 2023.

[172] Boyi Wei, Weijia Shi, Yangsibo Huang, Noah A Smith, Chiyuan Zhang, Luke Zettlemoyer, Kai Li, and Peter Henderson. Evaluating Copyright Takedown Methods for Language Models. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

[173] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.

[174] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems*, 2022.

[175] Jiankun Wei, Abdulrahman Abdulrazzag, Tianchen Zhang, Adel Muursepp, and Gururaj Saileshwar. Privacy risks of speculative decoding in large language models. *arXiv preprint arXiv:2411.01076*, 2024.

[176] Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, et al. Ethical and social risks of harm from language models. *arXiv preprint arXiv:2112.04359*, 2021.

[177] Roy Weiss, Daniel Ayzenshteyn, and Yisroel Mirsky. What was your prompt? a remote keylogging attack on {AI} assistants. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 3367–3384, 2024.

[178] Guanlong Wu, Zheng Zhang, Yao Zhang, Weili Wang, Jianyu Niu, Ye Wu, and Yinqian Zhang. I know what you asked: Prompt leakage via kv-cache sharing in multi-tenant llm serving. In *Proceedings of the 2025 Network and Distributed System Security (NDSS) Symposium*, 2025.

[179] Chulin Xie, Zinan Lin, Arturs Backurs, Sivakanth Gopi, Da Yu, Huseyin A Inan, Harsha Nori, Haotian Jiang, Huishuai Zhang, Yin Tat Lee, et al. Differentially private synthetic data via foundation model apis 2: Text. In *International Conference on Machine Learning*, pages 54531–54560. PMLR, 2024.

[180] Qinge Xie, Karthik Ramakrishnan, and Frank Li. Evaluating privacy policies under modern privacy laws at scale: An {LLM-Based} automated approach. In *34th USENIX Security Symposium (USENIX Security 25)*, pages 5797–5816, 2025.

[181] Roy Xie, Junlin Wang, Ruomin Huang, Minxing Zhang, Rong Ge, Jian Pei, Neil Gong, and Bhuwan Dhingra. ReCaLL: Membership Inference via Relative Conditional Log-Likelihoods. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8671–8689, 2024.

[182] Wenpeng Xing, Zhonghao Qi, Yupeng Qin, Yilin Li, Caini Chang, Jiahui Yu, Changting Lin, Zhenzhen Xie, and Meng Han. MCP-Guard: A Defense Framework for Model Context Protocol Integrity in Large Language Model Applications. *arXiv preprint arXiv:2508.10991*, 2025.

[183] Jiacen Xu, Jack W. Stokes, Geoff McDonald, Xuesong Bai, David Marshall, Siyue Wang, Adith Swaminathan, and Zhou Li. AutoAttacker: A Large Language Model Guided System to Implement Automatic Cyber-attacks. 2024.

[184] Minrui Xu, Jiani Fan, Xinyu Huang, Conghao Zhou, Jiawen Kang, Dusit Niyato, Shiwen Mao, Zhu Han, Kwok-Yan Lam, et al. Forewarned is forearmed: A survey on large language model-based agents in autonomous cyberattacks. *arXiv preprint arXiv:2505.12786*, 2025.

[185] Shusheng Xu, Wei Fu, Jiaxuan Gao, Wenjie Ye, Weilin Liu, Zhiyu Mei, Guangju Wang, Chao Yu, and Yi Wu. Is DPO Superior to PPO for LLM Alignment? A Comprehensive Study. In *Forty-first International Conference on Machine Learning*, 2024.

[186] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 Technical Report. *arXiv preprint arXiv:2407.10671*, 2024.

[187] John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. SWE-agent: Agent-Computer Interfaces Enable Automated Software Engineering, 2024.

[188] Yifan Yang, Siqin Wang, Daoyang Li, Shuju Sun, and Qingyang Wu. GeoLocator: A Location-Integrated Large Multimodal Model for Inferring Geo-Privacy. *Applied Sciences*, 14(16), 2024.

[189] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. ReAct: Synergizing Reasoning and Acting in Language Models. In *The Eleventh International Conference on Learning Representations*, 2023.

[190] Ren Yi, Octavian Suciu, Adria Gascon, Sarah Meiklejohn, Eugene Bagdasarian, and Marco Gruteser. Privacy Reasoning in Ambiguous Contexts. *arXiv preprint arXiv:2506.12241*, 2025.

[191] Da Yu, Saurabh Naik, Arturs Backurs, Sivakanth Gopi, Huseyin A Inan, Gautam Kamath, Janardhan Kulkarni, Yin Tat Lee, Andre Manoel, Lukas Wutschitz, et al. Differentially Private Fine-tuning of Language Models. In *International Conference on Learning Representations*, 2022.

[192] Hanna Yukhymenko, Robin Staab, Mark Vero, and Martin Vechev. A synthetic dataset for personal attribute inference. *Advances in Neural Information Processing Systems*, 37:120735–120779, 2024.

[193] Wojciech Zaremba, Evgenia Nitishinskaya, Boaz Barak, Stephanie Lin, Sam Toyer, Yaodong Yu, Rachel Dias, Eric Wallace, Kai Xiao, Johannes Heidecke, et al. Trading inference-time compute for adversarial robustness. *arXiv preprint arXiv:2501.18841*, 2025.

[194] Sajjad Zarifzadeh, Philippe Liu, and Reza Shokri. Low-Cost High-Power Membership Inference Attacks. In *International Conference on Machine Learning*, pages 58244–58282, 2024.

[195] Hang Zeng, Xiangyu Liu, Yong Hu, Chaoyue Niu, Fan Wu, Shaojie Tang, and Guihai Chen. Automated Privacy Information Annotation in Large Language Model Interactions. *arXiv preprint arXiv:2505.20910*, 2025.

[196] Chiyuan Zhang, Daphne Ippolito, Katherine Lee, Matthew Jagielski, Florian Tramèr, and Nicholas Carlini. Counterfactual Memorization in Neural Language Models. In *Advances in Neural Information Processing Systems*, 2023.

[197] Jie Zhang, Debeshee Das, Gautam Kamath, and Florian Tramèr. Position: Membership Inference Attacks Cannot Prove That a Model was Trained on Your Data. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 333–345, 2025.

[198] Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Frank Yang, and Hai Li. Min-K%++: Improved Baseline for Pre-Training Data Detection from Large Language Models. In *The Thirteenth International Conference on Learning Representations*, 2025.

[199] Kaiyuan Zhang, Siyuan Cheng, Hanxi Guo, Yuetian Chen, Zian Su, Shengwei An, Yuntao Du, Charles Fleming, Ashish Kundu, Xiangyu Zhang, and Ninghui Li. SOFT: Selective Data Obfuscation for Protecting LLM Fine-tuning against Membership Inference Attacks. In *34st USENIX Security Symposium (USENIX Security 25)*, 2025.

[200] Kaiyuan Zhang, Zian Su, Pin-Yu Chen, Elisa Bertino, Xiangyu Zhang, and Ninghui Li. LLM Agents Should Employ Security Principles. *arXiv preprint arXiv:2505.24019*, 2025.

[201] Weichao Zhang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. Pretraining Data Detection for Large Language Models: A Divergence-based Calibration Method. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5263–5274, 2024.

[202] Yanzhe Zhang, Tao Yu, and Diyi Yang. Attacking vision-language computer agents via pop-ups. *arXiv preprint arXiv:2411.02391*, 2024.

[203] Zhehao Zhang, Ryan A Rossi, Branislav Kveton, Yijia Shao, Diyi Yang, Hamed Zamani, Franck Dernoncourt, Joe Barrow, Tong Yu, Sungchul Kim, et al. Personalization of large language models: A survey. *arXiv preprint arXiv:2411.00027*, 2024.

[204] Zhiping Zhang, Bingcan Guo, and Tianshi Li. Privacy Leakage Overshadowed by Views of AI: A Study on Human Oversight of Privacy in Language Model Agent. *arXiv preprint arXiv:2411.01344*, 2024.

[205] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

[206] Arman Zharmagambetov, Chuan Guo, Ivan Evtimov, Maya Pavlova, Ruslan Salakhutdinov, and Kamalika Chaudhuri. Agentdam: Privacy leakage evaluation for autonomous web agents. *arXiv preprint arXiv:2503.09780*, 2025.

[207] Xi Zheng, Zhuoyang Li, Xinning Gui, and Yuhan Luo. Customizing Emotional Support: How Do Individuals Construct and Interact With LLM-Powered Chatbots. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, 2025.

[208] Xinyao Zheng, Husheng Han, Shangyi Shi, Qiyan Fang, Zidong Du, Xing Hu, and Qi Guo. Inputsnatch: Stealing input in llm services via timing side-channel attacks. *arXiv preprint arXiv:2411.18191*, 2024.

[209] Noé Zufferey, Sarah Abdelwahab Gaballah, Karola Marky, and Verena Zimmermann. "AI is from the devil." Behaviors and Concerns Toward Personal Data Sharing with LLM-based Conversational Agents. *Proceedings on Privacy Enhancing Technologies*, 2025(3):5–28, 2025.

Privacy-Preserving Federated Large Language Models: Techniques and Trade-offs

Runhua Xu[†], Guoan Wan[†] and James Joshi[‡]

[†] School of Computer Science and Engineering, Beihang University, Beijing, China

[‡] School of Computing and Information, University of Pittsburgh, Pittsburgh, PA, USA

{runhua,gawan}@buaa.edu.cn,jjoshi@pitt.edu

Abstract

The remarkable capabilities of Large Language Models (LLMs) rely on massive, diverse datasets. This dependence creates a fundamental tension in privacy-sensitive domains such as healthcare and finance, where data is siloed and tightly regulated. Federated Learning (FL) offers a privacy-by-design approach that enables collaborative fine-tuning of foundation models on decentralized data. However, combining FL with LLMs—forming Federated LLMs (FedLLMs)—introduces a critical utility–efficiency–privacy trilemma. This study systematically analyzes this trilemma by outlining three core challenges: (1) maintaining model utility amid statistical and system heterogeneity; (2) ensuring efficiency by alleviating severe communication and computation bottlenecks, even with Parameter-Efficient Fine-Tuning (PEFT); and (3) safeguarding privacy against powerful attacks. We formalize these interrelated challenges, examine their trade-offs, review existing defense mechanisms and optimization strategies, and conclude by outlining key open issues and future research directions.

1 Introduction

The remarkable capabilities of large language models (LLMs) have transformed natural language processing, yet their efficacy fundamentally depends on access to massive, diverse training datasets. This dependence creates a critical tension in high-impact domains, such as healthcare, finance, and enterprises, where the most valuable data remain highly privacy-sensitive and organizationally siloed. Simultaneously, emerging and fast-evolving regulatory frameworks, including the General Data Protection Regulation (GDPR), the Health Insurance Portability and Accountability Act (HIPAA), and the European Union Artificial Intelligence Act (EU AI Act), impose stringent data protection and privacy requirements, making traditional centralized training increasingly untenable.

In response, federated learning (FL) addresses the challenges faced by centralized training by enabling collaborative training without a need to bring all raw data in one place, allowing multiple parties to jointly train a shared global model while keeping data localized [10]. The central principle of FL is data minimization. Under this principle, only model updates traverse the network, thereby establishing a privacy-by-design architecture better aligned with regulatory data protection and privacy requirements. The convergence of FL with LLMs represents the emerging frontier of privacy-preserving artificial intelligence (AI). Given the prohibitive cost of training billion-parameter models from scratch, we focus on federated fine-tuning of pre-trained foundation models that use distributed private datasets. Large-scale FL deployments, including Google's Gboard with differentially private learning [15] and the SWIFT consortium for cross-border fraud detection [7], indicate operational feasibility at scale. As shown in Figure 1, the current design space of federated LLMs still faces three key challenges: utility, efficiency, and privacy.

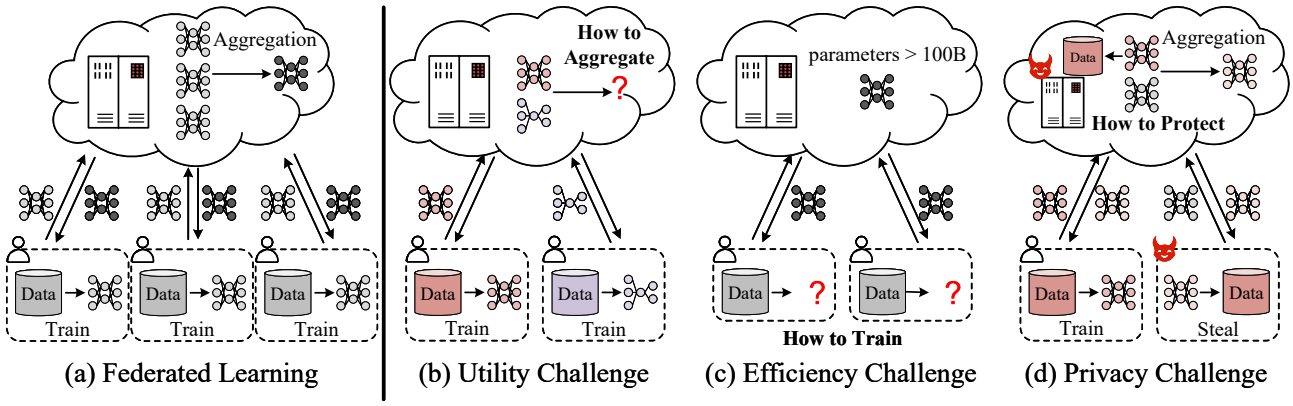


Figure 1: Key Challenges in Privacy-Preserving Federated Large Language Models

Utility Challenge. Federated LLM deployments span cross-device and cross-silo contexts, with pronounced data and model heterogeneity. FedLLM-Bench shows that client sampling strongly affects convergence, with importance-weighted sampling requiring about 2.3× fewer rounds than random sampling [5]. Under a pre-trained global model, utility should be evaluated along two dimensions: aggregation utility, ensuring heterogeneity-aware participation and update fusion, and continual learning utility, which focuses on improving the global model without catastrophic forgetting amid shifting client distributions. Routine data operations such as profiling, schema harmonization, quality control, and maintaining consistent tokenization, vocabulary, and encoding across nodes remain essential for preserving semantic coherence.

Efficiency Challenge. Billion-parameter LLMs induce extreme communication pressure. For example, transmitting full GPT-3 gradients (175B parameters [14]) at FP32 is roughly 700 GB per client per round, rendering naive federated training infeasible. Parameter-Efficient Fine-Tuning (PEFT) provides a practical route: DoRA [2] and VeRA [3] approach full-tuning performance with approximately 100× fewer trainable parameters, and prompt engineering can further reduce on-device computation and communication. Yet these choices introduce new concerns, including guarantees on convergence under non-independent and non-identically distributed (non-IID) client updates and the way PEFT modules interact with privacy amplification and compression.

Privacy Challenge. Although FL keeps raw data local, model updates can leak sensitive information. The DAGER attack [4] demonstrates exact gradient inversion for LLMs, reconstructing training sequences with ROUGE > 0.99 for batches up to 128 tokens. Such near-perfect reconstruction necessitates stronger protection; however, differential privacy may reduce accuracy due to noise injection, whereas cryptographic methods often impose substantial computational overhead. In multi-jurisdictional deployments, designers must also reconcile heterogeneous regulations while maintaining data sovereignty, which further constrains feasible mechanisms and the design of secure aggregation and auditing pipelines.

This comprehensive study provides a systematic analysis of techniques to navigate these challenges. We first delineate three core difficulties in federated LLMs, as reflected in Figure 1(b)–(d): *Aggregation Utility in Heterogeneous Settings*; *Federated Client-Efficient Training of LLMs*, and *Privacy Attacks and Defense Mechanisms*. Next, we formalize the *utility–efficiency–privacy trilemma* that is related to these challenges, specifying threat models, deployment assumptions (cross-device and cross-silo), and evaluation metrics that will guide the subsequent discussion. Finally, we discuss open challenges and future directions.

Table 1: List of abbreviations for partial terms.

Abbreviation	Full Term
CCPA	California Consumer Privacy Act
CKKS	Cheon–Kim–Kim–Song (approximate homomorphic encryption scheme)
DP	Differential Privacy
DP-SGD	Differentially Private Stochastic Gradient Descent
GDPR	General Data Protection Regulation
HIPAA	Health Insurance Portability and Accountability Act
LoRA	Low-Rank Adaptation
PEFT	Parameter-Efficient Fine-Tuning
PIPL	Personal Information Protection Law (China)
RAG	Retrieval-Augmented Generation
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
SGX	(Intel) Software Guard Extensions
SWIFT	Society for Worldwide Interbank Financial Telecommunication

2 Aggregation Utility in Heterogeneous Settings

In federated LLMs, aggregation utility reflects accuracy, stability, and fairness across clients, depending on how heterogeneity is represented and reconciled [5]. In cross-device deployments, strict limits on memory, computation, and bandwidth lead to model heterogeneity[1, 28]. In cross-silo collaborations, differing objectives, label spaces, and domains create data heterogeneity[16, 18].

2.1 Cross-Device Model Heterogeneity and Utility-Oriented Aggregation

Cross-device FL operate under uneven device memory, compute throughput, and network quality. Clients may load different subsets of the model, employ mixed precision, or attach lightweight adaptation modules, which breaks the assumption that every participant optimizes an identical parameterization. Benchmark and toolkit studies in federated LLM fine-tuning show that naive aggregation under such heterogeneity degrades convergence and yields uneven user quality [5]. Open implementations also document mixed local configurations that complicate server-side fusion [1].

The first family of approaches includes those that align parameters before averaging so that the server only fuses structurally compatible updates. When a device trains only a subset of layers, selective or masked aggregation updates those layers while leaving absent components unchanged [6]. When clients expose compact modules such as low-rank adapters, server-side merging that preserves the intended subspace geometry improves stability compared to element-wise averaging [16]. These choices reduce undesirable interference between clients that optimize different slices of the model thereby protecting aggregation utility.

The second family of approaches includes those that are architecture agnostic and rely on distillation to bridge model mismatches. Each client, regardless of backbone or head configuration, produces predictions on a proxy or public corpus. The server distills these signals into a shared student that captures improvements discovered on heterogeneous devices [29]. In cross-device deployments, this reduces payload size, removes strict layer alignment, and remains robust under device churn. Toolkits that support both parameter aggregation and prediction distillation demonstrate that these paths can be combined within one workflow [6].

Personalization improves perceived utility without fragmenting the deployable model. A shared representation is maintained at the server, while devices attach small local heads or adapters that encode preferences and context [27]. Evidence from federated LLM toolchains shows that this separation reduces

cross-device update conflicts and improves user-level metrics across wide variation in hardware and availability [1].

In short, structure-aware averaging, architecture-agnostic distillation, and lightweight personalization form a complementary toolkit for cross-device FedLLMs. These methods enhance average accuracy, stabilize convergence, and reduce performance degradation on under-resourced clients by aligning aggregation with each device's training capabilities.

2.2 Cross-Silo Data Heterogeneity and Utility-Oriented Aggregation

Cross-silo FL approaches connect institutions or organizations that pursue different objectives and curate domain-specific datasets. Partnering institutions may use labels following distinct taxonomies, optimize different task mixes, or operate under diverse risk considerations and compliance requirements. If the server aggregates updates without accounting for these differences, the global model can drift toward the majority objective and underperform on minority tasks. Evidence from federated LLM benchmarks shows that utility is sensitive to participation policies and weighting under such heterogeneity, and that naive averaging widens performance gaps across silos [5].

Data-aware aggregation aligns each silo's contribution with relevance to the shared objective. Importance-weighted participation reduces the rounds needed to reach target quality when some participants provide more informative gradients for the current global state [5]. In multilingual settings, clustering institutions by linguistic proximity and aggregating within clusters before global fusion preserves domain signal and lifts perplexity and accuracy, especially for low-resource groups [34]. In multimodal collaborations, separating client-agnostic knowledge from client-specific signals and distilling them into a shared student can help stabiliz training when distributions differ substantially [35].

Personalization further raises perceived utility while maintaining a single deployable backbone. A practical design keeps a shared representation at the server and allows silos to attach small local heads or adapters that encode institutional constraints and preferences [27]. Recent work for federated LLM tooling have shown that this separation reduces inter-silo conflict during aggregation and improves user-level metrics because global updates carry a common structure while local components handle idiosyncrasies [6].

Optimization and sampling policies are key issues when participation in FL training is intermittent and objectives differ. Proximal regularization reduces client drift under objective mismatch [23]. Control variates reduce variance in aggregated updates and enhance stability under uneven connectivity [24]. Adaptive server optimizers help maintain steady progress with non-IID updates [25]. Normalizing local steps improves fairness when silos contribute different amounts of work per round [26]. Priority-based client-selection accelerates convergence to target quality and complements the aggregation strategies above [5].

Overall, cross-silo aggregation that accounts for data heterogeneity, combined with personalization and principled optimization and sampling, expands the utility frontier. The global model benefits from shared structure while each institution retains control over local adaptations that reflect task priorities and regulatory context.

2.3 Utility-Oriented Continual Global Learning under Joint Model and Data Heterogeneity

Federated LLMs begin from a strong pre-trained global model rather than random initialization. This starting point creates a utility risk: if collaborative updates erode base capabilities, usefulness declines for all participants. Measurable catastrophic forgetting appears during continual adaptation and manifests as losses on previously mastered domains after task- or domain-specific tuning [43]. Privacy-compatible

rehearsal methods further confirm the phenomenon and propose mitigations [44]. Therefore, continual global learning is essential to preserve and expand utility as new sites, skills, and distributions enter the federation.

Joint heterogeneity arises because devices expose different trainable slices while institutions contribute data with divergent label spaces and objectives. Layered aggregation improves utility when it respects both dimensions. Within groups that share similar adapter layouts, structure-preserving fusion of low-rank updates avoids distortions caused by element-wise averaging [16]. Across groups that differ by language or domain, cluster-wise aggregation retains local signal before global fusion and benefits minority clients [34]. Benchmarks that vary participation and task composition report high sensitivity of utility to these choices [5].

Continual learning mechanisms should be integrated with aggregation strategies to prevent regression of the shared model. Weight consolidation protects parameters important for earlier skills and reduces interference during new updates [30]. Federated transfer decomposes knowledge into global and task-specific components so that the server accumulates stable competence while clients keep sparse local parameters [33]. Tooling that supports selective synchronization and modular adapters enables lifecycle operations such as freezing, merging, or retiring client-side modules without destabilizing the shared backbone [6].

When original data cannot be retained, synthetic rehearsal generated by the model can maintain earlier capabilities with low leakage. Yet, it requires careful filtering and scheduling to avoid drift [44]. The broader literature warns that uncontrolled reliance on model-generated data may degrade generality, which motivates audits and curriculum policies for any replay pipeline [45]. In a federated context, these safeguards should be coupled with per-skill evaluation so that global updates are admitted only when they do not reduce established competencies [5].

Based on the above discussion, we can see that a utility-oriented approach can align aggregation with model slices and data groups, protect previously acquired skills through consolidation or parameter isolation, and maintain them with privacy-compatible rehearsal and module lifecycle management.

3 Federated Client-Efficient Training of LLMs

LLM capability rises predictably with increases in parameters, data, and compute, as captured by empirical scaling laws. Compute-optimal analyses further show that sufficiently trained smaller models can outperform under-trained larger models, although the capability still lies at the scael of billion parameter. In a federated setting, transmitting and updating all weights at these scales is impractical and increases bandwidth and memory costs, which degrades training efficiency [14]. In this section, we therefore focuse on parameter-efficient adaptation as the organizing principle. Clients learn compact adjustments on top of frozen backbones and keep base weights in a compressed form so that the device and network constraints are met while task quality is maintained [12]. In the remainder of this section, we examine federated specializations of these adaptations, introduce randomized or compressed update schemes that further reduce traffic, and outline alternatives that avoid weight updates through retrieval, prompting, and agent coordination [9].

3.1 Federated Specializations of PEFT: Methods and Evidence

A growing body of work tailors PEFT to the specifics of FL optimization, with the goal of reducing communication and memory footprints while keeping round time stable under data and model variation and under intermittent participation. These specializations differ in how they partition knowledge between global and client-specific components and in how they regularize the server-side fusion of

client updates. The resulting designs are attractive for production because they align with governance requirements that discourage wholesale model sharing.

FedTune provides a systematic comparison of prompt-, adapter-, and bias-level tuning for pre-trained Transformers in FL settings[22]. It reports fast convergence with substantially reduced communication and, in many cases, performance surpassing local-only baselines, indicating that cross-client collaboration remains effective even when only a tiny subset of parameters is updated.

For multilingual FLs, FedLFC freezes the backbone and trains Low-Rank Adaptation (LoRA) modules per language family, then aggregates at the family level to reflect linguistic proximity [34]. This design improves perplexity and accuracy—especially for low-resource languages—while keeping the adapter budget small and respecting language-specific heterogeneity. Family-wise structuring also reduces negative interference during aggregation and facilitates controlled personalization at deployment time without retraining the backbone.

In multi-modal heterogeneous scenarios, FedDAT introduces a dual-adapter teacher with mutual knowledge distillation; here, a global adapter captures client-agnostic information, while a local adapter encodes client-specific signals [35]. Separating these roles stabilizes aggregation and improves results on vision–language benchmarks compared with centralized PEFT-to-FL pipelines, suggesting that adapter topology is as important as adapter size. The formulation naturally extends to other cross-modality or cross-domain FL setting where shared and private factors must be disentangled.

Beyond module choice, aggregating PEFT parameters requires care to avoid structural drift. FLoRA addresses inconsistencies in naive averaging of LoRA factors by preserving their intended low-rank structure during fusion, yielding consistent quality gains across client–server compositions [16]. At the system layer, SLORA exploits structured sharing across layers to further reduce trainable budgets without sacrificing accuracy, which is useful when client memory ceilings bind participation. Together, these results indicate that PEFT is not only communication-efficient but also amenable to FL-aware rules that directly prioritize global utility under heterogeneous clients.

3.2 Randomization and Compression for Communication Efficiency

Randomized and compressed updates offer an orthogonal means to reduce communication overhead, and they can be layered over adapter- or LoRA-based clients without changing learning objectives. The central idea is to approximate dense updates with sketches or quantized representations that preserve informative directions for aggregation, while variance control and bias correction restore convergence guarantees. This family of methods is particularly an attractive option when bandwidth fluctuates or when client links are asymmetric across rounds.

Ferret, proposed in [36], performs a first-order full-parameter tuning with low-dimensional projections and shared randomness to reconstruct updates at the server. By decoupling local optimization from transmitted dimensionality, Ferret combines the benefits of full parameter training with communication comparable to compressed methods, and it exhibits favorable convergence relative to zeroth-order alternatives. The projection–reconstruction pipeline also interacts well with straggler mitigation because it decimates payloads without altering local objectives, thereby keeping device-side software simple.

Classical quantization and error-feedback mechanisms further lower bandwidth and correct compression bias. QSGD quantizes gradients under variance control [37]. EF SGD feeds back the compression error to recover the trajectory of the uncompressed method [38]. In heterogeneous deployments, combining PEFT with randomized projections yields additive gains. Clients keep tiny trainable modules and transmit sketched deltas when links are constrained, while the server aggregates in a manner consistent with reliability and fairness policies.

3.3 Alternatives to Weight Updates: RAG, Prompts, and Agents

Some alternative approaches to improving FL efficiency is to avoid gradient updates altogether. These approaches leverage local data and global coordination while keeping base model weights fixed [9, 12], which aligns with privacy and governance constraints in both cross-device and cross-silo settings [11, 19]. They are particularly attractive when regulatory or operational constraints limit the sharing of model parameters, even in modified form.

Retrieval-Augmented Generation (RAG) maintains private corpora locally and coordinates retrieval over distributed indices, after which a frozen LLM conditions on retrieved evidence for generation [39]. Synchronization shifts from heavyweight deltas to lightweight retrieval statistics and index metadata, reducing update traffic and simplifying audits because content remains under local control. In practice, FL can align retrieval strategies with local curation standards, seamlessly integrating with evaluation protocols that prioritize accuracy and source attribution.

Prompt-based collaboration includes exchanging soft prompts or prefixes rather than gradients, yielding huge communication savings and natural personalization under tenant or domain structure [40, 41]. Prompts can be shared, clustered, or composed to reflect organizational boundaries without modifying backbone weights, and they can be rotated or gated to manage risk. Because prompts are small, they are amenable to secure aggregation and differential privacy, which further broadens the set of compliant deployment regimes.

Agent orchestration frameworks coordinate tool use, retrieval, and prompting policies across sites, leveraging local data without weight updates and facilitating policy-compliant workflows [42]. By turning adaptation into planning and tool selection rather than gradient descent, agents avoid heavy synchronization while still exploiting situational context and institutional knowledge. In hybrid pipelines, agents can call RAG for evidence, choose prompts for control, and fall back to PEFT only when sustained drift necessitates weight changes.

Overall, these alternatives coexist with parameter-efficient adaptation and randomized compression to enlarge the feasible region of FedLLM design. By reducing or eliminating gradient traffic, they broaden participation, lower costs, and sustain utility under heterogeneous constraints, while leaving room for targeted weight updates when enduring domain shifts demand persistent changes.

4 Privacy Attacks and Defense Mechanisms

Understanding adversarial capabilities against FedLLM systems is crucial for designing effective defenses and ensuring realistic privacy guarantees. Recent studies have revealed advanced attack vectors that can extract sensitive information from model updates in FL process, highlighting the need for thorough analysis of attack methods, success conditions, and layered defense strategies.

4.1 Gradient Inversion Attacks: From Theory to Practice

Gradient inversion attacks pose one of the most serious threat to privacy in federated learning, as they can reconstruct the entire training dataset from observed gradients. These attacks rely on the fact that gradients encode rich information about the data that produced them. For a neural network with parameters θ and loss function ℓ, the gradient $g = \nabla_\theta \ell(\theta; x)$ is a deterministic function of both the parameters and the input data x. Under sufficient constraints, this relationship can be inverted to recover x from g.

Traditional gradient inversion methods frame reconstruction as an optimization problem: find an input $\hat{x}$ that minimizes $\|\nabla_\theta \ell(\theta; \hat{x}) - g\|^2$. Early works have demonstrated successful reconstructions for simple networks and small batches but have struggled with the high-dimensional, discrete nature of

language data. The non-convex optimization landscape and the discrete token space of language models pose fundamental challenges, limiting reconstruction accuracy to semantic similarity rather than exact recovery.

The DAGER (Differentially Private Aggregated Gradient Extraction with Restoration) method [4] marks a major breakthrough in gradient inversion. DAGER achieves exact reconstruction of training sequences—recovering text with ROUGE-1 and ROUGE-2 scores above 0.99—for batch sizes up to 128 tokens. The implications for federated learning are severe. An honest-but-curious server observing gradient updates can reconstruct entire training texts, including medical records, financial documents, or private communications. The attack remains effective even when gradients are aggregated over multiple training steps, provided the batch size stays within feasible limits. Moreover, DAGER shows resilience to common defenses: gradient clipping only slightly reduces reconstruction quality, while compression techniques such as top-k sparsification offer limited protection unless applied aggressively [4].

4.2 Membership and Property Inference in FL Settings

While gradient inversion attacks aim to reconstruct specific training examples, membership and property inference attacks target different forms of privacy leakage that can be equally harmful in practice.

Membership inference attacks determine whether particular data points were used during training without necessarily reconstructing their content. In FL settings, these attacks exploit the distributed nature of learning to achieve higher success rates than those on centralized models. FedMIA [8] leverages the "all for one" principle inherent in federated learning: each client's update encodes information about all its local training samples simultaneously. By analyzing how model updates affect predictions on candidate data points across multiple rounds, FedMIA achieves 63–68% attack success rates compared with 45–52% for similar centralized attacks.

The attack methodology integrates multiple signals to improve inference accuracy. Temporal analysis tracks how prediction confidence on target examples evolves, with members typically showing steadily increasing confidence. Update correlation analysis measures alignment between model updates and gradients computed on target examples, with stronger correlations indicating membership. Influence estimation assesses how removing hypothetical examples would alter model updates, based on the insight that true members measurably influence parameter changes. Combined through ensemble methods, these signals enable robust membership detection even when individual indicators are noisy.

Property inference attacks extract statistical characteristics of training datasets rather than details about specific samples. In FL settings, such attacks can expose sensitive information about participating institutions/clients. A hospital's model updates might reveal unusual disease prevalence patterns, demographic distributions, or treatment protocols that constitute valuable competitive intelligence; financial institutions' updates could disclose customer segment traits, risk profiles, or business strategies embedded in their data.

FL setting paradoxically makes property inference both easier and harder than centralized training does. The isolation of client updates provides clearer signals about individual dataset properties since they are not diluted by mixing with others' data. However, limited visibility—observing only periodic model updates instead of continuous dynamics—reduces available information for adversaries. Recent studies show that sophisticated attackers can overcome this constraint by correlating observations across rounds and exploiting the temporal consistency of dataset properties[8, 15].

4.3 Multi-Layered Defense Strategies

Defending against this range of attacks requires comprehensive strategies that integrate multiple protection mechanisms, each addressing distinct threat vectors while collectively providing defense-in-

depth.

Differential privacy offers the strongest theoretical safeguard against inference attacks by ensuring that model updates reveal only limited information about the training data. DP-SGD's noise injection fundamentally restricts what adversaries can learn, with formal guarantees that hold regardless of their capabilities or auxiliary knowledge. Empirical results show that applying differential privacy with $\epsilon = 1.0$ reduces DAGER's reconstruction quality from ROUGE > 0.99 to ROUGE < 0.3, effectively preventing meaningful text recovery. Similarly, differential privacy provides provable bounds on membership inference advantage, limiting adversarial success to near-random levels under reasonable privacy parameters.

However, differential privacy alone is insufficient. It does not protect against Byzantine attacks where malicious clients craft updates exploiting the noise distribution. Moreover, strong differential privacy often incurs unacceptable utility loss for some applications, making complementary defenses necessary to achieve practical protection with lower performance costs.

Secure aggregation protocols defend against server-side threats by ensuring servers see only aggregated updates rather than individual contributions. Cryptographic secure aggregation [11] uses secret sharing or homomorphic encryption to compute aggregates without exposing individual model updates. This approach strongly protects against honest-but-curious servers attempting gradient inversion or membership inference on single clients. However, its benefits diminish when few clients are aggregated since limited aggregation yields minimal privacy amplification. Additionally, secure aggregation cannot prevent malicious clients from analyzing the global model to infer information about others.

The most effective defense strategy combines multiple mechanisms targeting different threats. A robust configuration might use LoRA for parameter efficiency (reducing attack surface), DP-SGD with a moderate privacy budget for inference resistance, secure aggregation to hide individual updates from the server, robust aggregation to filter Byzantine inputs, and gradient compression to limit information leakage. While no single method provides complete protection against all attacks, this layered approach substantially increases the attack difficulty while preserving practical utility and efficiency.

5 The Privacy-Utility-Efficiency Trilemma

The techniques surveyed in previous sections do not operate independently but rather interact within a complex optimization landscape characterized by fundamental trade-offs that constrain achievable system configurations. Understanding these trade-offs through the lens of a privacy-utility-efficiency trilemma provides essential guidance for practical privacy-preserving FedLLM system design and helps explain why no single solution dominates across all deployment scenarios.

5.1 Mathematical Formalization and Constraint Analysis

The privacy–utility–efficiency trilemma can be formally defined by three interdependent metrics that together determine the feasible operating region of PP-FedLLM systems. The privacy level $\mathcal{P}$ measures protection against information leakage, quantified by the differential privacy parameter ϵ (where smaller values indicate stronger privacy) or by cryptographic security parameters in encryption-based methods. Model utility $\mathcal{U}$ reflects the performance of the trained model, typically evaluated using task-specific metrics such as accuracy, F1 score, or perplexity on held-out test sets. System efficiency $\mathcal{E}$ represents computational and communication costs, measured by training time, number of communication rounds, bandwidth usage, or total computational operations.

The trilemma manifests as fundamental constraints on the achievable region in $(\mathcal{P}, \mathcal{U}, \mathcal{E})$ space. These constraints arise from information-theoretic limits, computational complexity barriers, and statistical requirements that cannot be overcome through engineering alone.

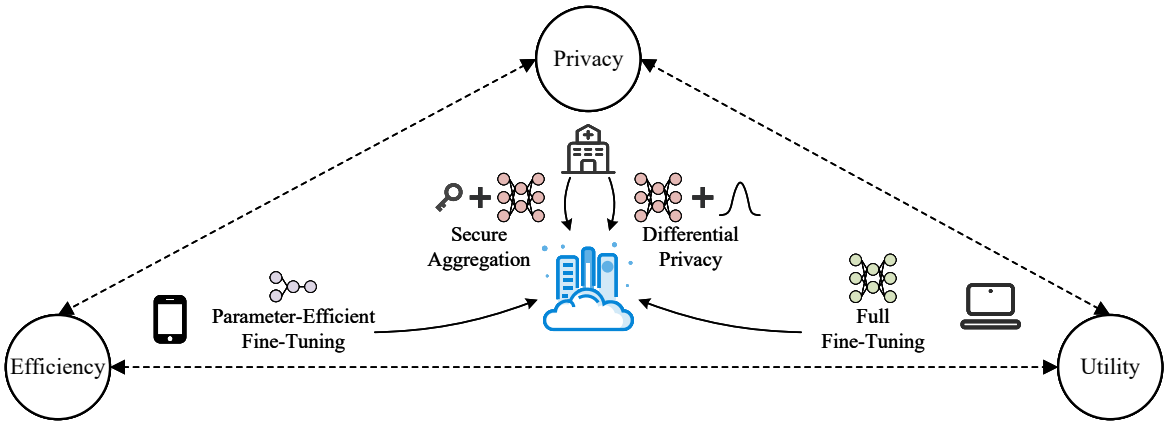


Figure 2: The illustration of privacy-utility-efficiency trilemma in privacy-preserving federated LLM.

The privacy-utility trade-off reflects an information-theoretic reality: protecting privacy requires limiting information flow about training data, but model learning fundamentally requires extracting information from that same data. Differential privacy makes this trade-off explicit through noise injection that provides privacy at the cost of accuracy. The privacy–efficiency trade-off arises from the computational complexity of privacy-preserving mechanisms. For example, homomorphic encryption perfectly preserves utility—the encrypted computation yields the same results as plaintext computation—but incurs a 50–300% computational overhead and a 70–200% increase in communication due to larger ciphertexts [46]. Cryptographic theory establishes fundamental lower bounds indicating that these overheads cannot be fully eliminated, only reduced through improved implementations [47].

The utility–efficiency trade-off is most evident in the choice between full fine-tuning and parameter-efficient methods. Full fine-tuning offers optimal task adaptation but requires updating and transmitting all model parameters, whereas PEFT methods such as LoRA achieve 95–98% of full fine-tuning performance while using 100–1000 times fewer parameters.

5.2 Empirical Characterization Through Systematic Evaluation

Recent systematic evaluations provide quantitative characterization of these trade-offs in practical systems. The FedLLM-Bench evaluation [5] tests configurations across the trilemma space, revealing consistent patterns that guides system design. Their experiments with LoRA (rank 8) on a 7B parameter model demonstrate that this configuration achieves 96.3% of full fine-tuning performance while reducing communication cost by 267 times and computation cost by 4.7 times. This highly favorable utility-efficiency trade-off that is primarily the reason for LoRA's widespread adoption. Adding differential privacy with $\epsilon = 4.0$ to this LoRA-based system reduces accuracy by an additional 4.2% while providing formal privacy guarantees, illustrating how privacy protection compounds with efficiency optimizations to impact utility. Applying selective homomorphic encryption to the most sensitive 10% of LoRA parameters adds 38% communication overhead and 42% computation time while maintaining perfect utility for those protected parameters.

These empirical observations show that the trade-offs are neither linear nor independent. Privacy mechanisms interact with efficiency optimizations in complex ways: the utility loss from differential privacy is amplified when paired with aggressive parameter reduction, as the smaller parameter space offers less redundancy to absorb noise. Conversely, cryptographic methods that preserve utility become more practical when combined with PEFT, since the reduced parameter count keeps encryption overhead manageable. Understanding these interactions is essential for effectively navigating the trilemma.

5.3 Hybrid Solutions: Synergistic Combinations for Practical Deployment

The most successful practical deployments employ hybrid solutions that combine multiple techniques synergistically to achieve favorable positions within the trilemma space. These combinations leverage the strengths of different approaches while mitigating their individual weaknesses through careful system design.

PEFT serves as the foundational enabler for most hybrid solutions, dramatically improving the efficiency baseline creating headroom for adding privacy protections. The combination of LoRA with differential privacy exemplifies this synergy. LoRA's parameter reduction from 7 billion to 21 million parameters (for a 7B model with rank 8) not only reduces communication cost by 333 times but also fundamentally changes how employing differential privacy affects the model. The DP noise added to protect privacy has smaller impact on a lower-dimensional parameter space, as the relative signal-to-noise ratio improves. Empirical studies show that LoRA with $\epsilon = 1.0$ differential privacy achieves accuracy within 5% of non-private full fine-tuning, whereas applying the same privacy budget to full fine-tuning degrades accuracy by 12-15%. This dramatic difference demonstrates how efficiency optimizations can indirectly improve the privacy-utility trade-off.

5.4 Application-Driven Trade-off Navigation

Optimal navigation of the trilemma depends heavily on application-specific requirements, threat models, and deployment constraints. Different domains emphasize different aspects of the trilemma, resulting in distinct solution strategies.

Healthcare applications operating under strict regulations such as HIPAA typically prioritize privacy above all else. A multi-hospital consortium analyzing electronic health records may accept significant utility loss to ensure strong privacy guarantees. Efficiency takes a back seat to meeting privacy requirements, though PEFT remains crucial for practical deployment.

Financial fraud detection systems face a different set of constraints, in which both false negatives (missed fraud) and false positives (legitimate transactions flagged as fraudulent) incur substantial costs. These systems prioritize utility while maintaining sufficient privacy protection against realistic threats. The Swift consortium's use of cryptographic safeguards for highly sensitive features combined with moderate differential privacy for others reflects this balance [7].

Enterprise knowledge management systems encounter yet another trade-off landscape. When deploying federated RAG across corporate departments, the main concern often lies in preserving departmental autonomy and intellectual property rather than individual privacy. Such systems may rely on trusted execution environments (TEE) that offer strong practical protection with minimal overhead, accepting the trust assumptions inherent in hardware-based security.

6 Open Challenges and Future Directions

Despite remarkable progress in making FedLLMs practical, fundamental challenges remain that will determine whether federated learning achieves its potential as a transformative paradigm for privacy-preserving AI or remains limited to specialized applications.

6.1 Foundational Challenges Across Three Dimensions

Deploying FedLLMs involves three interrelated challenges that define the technical landscape of this survey: maintaining utility amid statistical and system heterogeneity, ensuring efficiency under tight communication and computation constraints, and preserving privacy against advanced inference and reconstruction attacks.

6.1.1 Utility: Statistical and System Heterogeneity

Real-world federated networks exhibit significant statistical heterogeneity that violates the Independent and Identically Distributed (IID) assumption of classical learning [18, 21]. In healthcare, institutions serving different populations adopt distinct treatment patterns and documentation practices. In finance, regional regulations and local payment behaviors lead to divergent transaction distributions. This non-IID structure drives client gradients in conflicting directions, slowing convergence and causing uneven model quality across cohorts. FedLLM Bench reports that vanilla FedAvg may require two to three times more rounds to reach a target accuracy under realistic heterogeneity than under IID conditions, and that performance disparities across client groups persist even after convergence [5]. More research is needed to develop heterogeneity-aware aggregation and fairness-aware objectives for FedLLMs, and to standardize non-IID benchmarks with group-level reporting and convergence criteria.

This necessitates more targeted research into advanced heterogeneity-aware aggregation techniques to combat gradient divergence and stabilize convergence. Future work should also design robust fairness-oriented objectives—such as optimizing worst-case client performance or deploying personalized models—to address disparities in model quality. Moreover, there is an urgent need to standardize non-IID benchmarks that realistically capture data skew and to establish group-level metrics for properly evaluating algorithmic fairness and robustness.

System heterogeneity amplifies these effects [28]. Clients vary in memory, compute power, network bandwidth, and availability. Cross-device deployments must accommodate everything from high-end smartphones with neural processing units to entry-level devices with limited memory. Network quality ranges from gigabit fiber to intermittent 3G connections, leading to significant differences in latency and reliability. Synchronous aggregation that waits for all selected clients can be delayed by stragglers, extending round time. Asynchronous aggregation reduces waiting but introduces staleness when slower clients train on outdated parameters. Both approaches require designs that stabilize optimization under delayed or partially aligned updates [20]. More research is needed on staleness-robust asynchronous methods with convergence guarantees under bounded delay, and on adaptive client selection/model sizing that respects device and network constraints.

Future research should develop communication-efficient, staleness-robust asynchronous methods that offer formal convergence guarantees under bounded update delays. This includes exploring adaptive client selection strategies that prioritize available clients without introducing bias and dynamically adjusting model components or quantization to accommodate heterogeneous device constraints. Such system-aware techniques are crucial for stabilizing optimization and mitigating the effects of stragglers and partial client participation.

6.1.2 Efficiency: Communication and Computational Bottlenecks

At modern LLM scales, communication and memory dominate the costs in federated settings. Transmitting full updates for GPT-3 with 175 billion parameters at FP32 requires about 700 GB per client per round, and even FP16 still needs around 350 GB—far beyond typical network capacities [14]. Computation is similarly limiting: fine-tuning LLaMA2 70B demands roughly 512 GB of GPU memory for standard training with the Adam optimizer, accounting for weights, gradients, optimizer states, and activations [13]. Most edge participants cannot host such models, motivating approaches that reduce bytes per round and device memory while maintaining stable training.

Extending from adaptation to foundation model pretraining further amplifies these constraints. Photon demonstrates feasibility for 7B-parameter models across sixteen high-performance computing centers, but scaling to 100B and beyond requires orders of magnitude more tokens and wall-clock time. Autoregressive dependencies restrict parallelism, while heterogeneous pretraining corpora increase opti-

mization drift among participants. Promising directions include hierarchical federation with intra-cluster optimization, bounded staleness asynchrony to handle stragglers, curriculum-style token scheduling by domain or quality, and phase-aware privacy budgets during early training. These remain open challenges that need to be addressed to ensure that federated pretraining at a foundation model is practical for deployment at scale.

6.1.3 Privacy: Threats and Adaptive Adversaries

Keeping data local does not prevent information leakage from model updates. Membership inference attacks, which determine whether a record was used in training, become stronger because each client update reflects many local examples. FedMIA reports success rates of 63–68% under realistic conditions [8]. Gradient inversion attacks pose an even greater threat. DAGER reconstructs LLM training sequences with ROUGE scores above 0.99 for batches of up to 128 tokens by searching the discrete token space [4]. These findings show that data localization alone cannot ensure privacy.

Noise-based defenses such as Differential Privacy (DP) are not a cure-all and can introduce new attack surfaces. Adversaries may shift from direct gradient inspection to exploiting side-channel vulnerabilities—like timing variations or resource usage patterns—or use active probing to maximize information leakage within a fixed privacy budget. Advanced threat models also include colluding clients and Sybil attacks, which can evade basic anomaly detection and amplify targeted or poisoning attacks. Effective mitigation therefore requires a defense-in-depth strategy that combines training-time privacy accounting with strong system-level hardening. This holistic approach should establish a hardware-rooted chain of trust to ensure platform integrity through remote attestation. Key defensive measures should be applied throughout the pipeline: traffic shaping and batched reporting to mask timing side channels; robust aggregation protocols with gradient clipping and cross-client correlation checks; and identity management via public key infrastructure (PKI) to enforce rate limits and defend against Sybils. These controls should be unified under a secure aggregation framework that enables auditable, per-round privacy accounting.

This highlights several critical research needs: (i) developing end-to-end privacy guarantees that cover the full pipeline—from data collection and training to aggregation and inference logging—is essential to meet regulatory compliance with auditable accounting; (ii) integrated objectives that jointly optimize privacy with security and fairness are required, particularly when facing heterogeneous clients and adversaries (iii) new machine unlearning protocols suitable for federated and DP-constrained settings are necessary to address the "right-to-be-forgotten" and facilitate data-correction requests without costly full retraining [48].

6.2 Data Engineering Challenges in Federated LLM

While most existing research on federated LLMs focuses on training algorithms and privacy mechanisms, the data engineering pipeline poses equally critical yet underexplored challenges [17, 18]. Unlike centralized machine learning, where data scientists can directly inspect, clean, and preprocess data, federated settings impose fundamental constraints that complicate traditional data engineering practices.

6.2.1 Data Quality Assessment Without Centralization

In traditional centralized ML pipelines, data quality assessment is performed through exploratory data analysis, profiling, and visualization of the entire dataset. However, in privacy-preserving FL, the orchestrating server cannot directly access raw data, making it impossible to assess data quality through conventional means. This creates several critical challenges:

Distributed Data Profiling. Without centralized access, evaluating data quality metrics—such as missing value rates, class distributions, feature correlations, and outliers—requires privacy-preserving distributed algorithms [17]. Recent research has proposed differentially private data profiling methods, but the added noise can obscure real data quality issues. For example, a hospital with systematically miscoded diagnoses might go unnoticed if privacy noise hides the anomalous patterns.

Heterogeneous Data Schemas. Real-world federated deployments often involve participants with heterogeneous data schemas, especially in healthcare, where institutions use different Electronic Health Record (EHR) systems. Even when training on the same task (e.g., clinical note classification), hospitals may differ in field names, coding systems (ICD-9 vs. ICD-10), and data granularity. Automated schema matching and harmonization in federated settings remain open challenges, as current solutions either demand extensive manual alignment or cause information loss.

Data Drift Detection. In production federated systems, data distributions at client nodes can shift over time due to changing user behavior, seasonal effects, or systematic changes in data collection practices. Detecting such concept drift without centralizing data requires new distributed monitoring methods. For LLMs in particular, vocabulary drift—such as the emergence of new terms and evolving language usage—introduces additional challenges that existing federated learning frameworks do not adequately address.

6.2.2 Data Governance and Compliance

The decentralized nature of federated learning introduces complex data governance challenges that go beyond technical privacy mechanisms [19]. Each participating organization must maintain sovereignty over its data while contributing to a collaborative model, requiring new governance frameworks that balance autonomy with coordination.

Multi-Jurisdictional Compliance. Federated LLM deployments across multiple countries must navigate a complex landscape of data protection laws, including the GDPR (Europe), CCPA (California), PIPL (China), and HIPAA (healthcare). Each jurisdiction imposes distinct requirements for data localization, consent management, and breach notification. For instance, the GDPR's right to erasure ("right to be forgotten") poses technical challenges in federated settings: if a user requests deletion of their data, how can we ensure their contribution is removed from a model trained across hundreds of devices? Current federated unlearning methods remain immature and computationally costly.

Data Lineage and Provenance Tracking. In regulated industries, maintaining detailed records of data lineage—tracking how data moves through ML pipelines and influences model predictions—is often a compliance requirement. In FL settings, this task becomes far more complex: the global model results from aggregated updates across multiple sources, each with its own preprocessing pipeline and data quality controls. Blockchain-based methods have been proposed to establish immutable audit trails for FL training, but they introduce additional computational overhead and privacy risks, as even encrypted metadata can reveal participation patterns.

Dynamic Participant Management. Deployable FL systems must manage participants joining, leaving, or being excluded for poor data quality or malicious behavior. This requires governance mechanisms to (1) assess new participants' data quality and security practices, (2) fairly attribute credit for model improvements among contributors, and (3) manage intellectual property rights in the jointly trained model. Multidisciplinary approaches to address these challenges. Existing frameworks offer limited support for these governance processes, especially when determining fair compensation in cases where participants contribute varying amounts or qualities of data.

6.2.3 Real-Time Federated Learning and Streaming Data

Most existing federated LLM studies assume static datasets and batch training. However, many real-world applications—such as mobile keyboard prediction, content recommendation, and real-time fraud detection—demand continuous learning from streaming data [20, 21]. This creates several data engineering challenges:

Online Data Preprocessing. Traditional ML pipelines perform extensive data preprocessing (tokenization, normalization, feature engineering) as a separate batch step before training. In FL settings that need to use streaming data, preprocessing must occur online at each client [6], requiring careful coordination to maintain consistency. For LLMs, this involves keeping tokenizer vocabularies synchronized as new terms appear, handling out-of-vocabulary words, and determining when to update preprocessing pipelines without disrupting existing models. Further research is needed to develop lightweight on-device tokenization algorithms and decentralized vocabulary synchronization protocols that are communication-efficient and resilient to network dropouts.

Temporal Data Alignment. In cross-device FL (e.g., training across millions of mobile phones), devices may go offline for long periods, causing temporal misalignment where some clients train on outdated data while others use fresh data [20]. For time-sensitive tasks such as news classification or trend detection, this temporal skew can severely degrade model performance. Designing aggregation algorithms that appropriately weight contributions by data freshness remains an open challenge [21]. This requires research into novel staleness-aware aggregation functions that explicitly model temporal dependencies and can dynamically discount or re-weight client updates based on their data timestamps.

Incremental Model Updates. Streaming data demands incremental model updates instead of full retraining. For LLMs, this is especially difficult due to catastrophic forgetting—the tendency of neural networks to lose previously learned knowledge when exposed to new data [43]. Federated continual learning must balance plasticity (learning new information) and stability (preserving prior knowledge) across distributed nodes while maintaining privacy guarantees. A key research and development challenge is to develop federated continual learning (FCL) strategies—such as parameter isolation or rehearsal-based methods—that prevent catastrophic forgetting while preserving privacy and minimizing communication overhead.

6.3 Future Directions

Federated Data Marketplaces. One promising research direction is to design innovative economic mechanisms and technical infrastructure for federated data marketplaces to enable future data economy where participants can discover collaboration opportunities, negotiate data-sharing terms, and receive fair compensation for their contributions. This involves addressing technical challenges (e.g., privacy-preserving dataset search, contribution valuation) and creating governance structures that incentivize high-quality and trustworthy participation.

Adaptive Preprocessing Pipelines. Complementary to this, developing adaptive preprocessing techniques that automatically adjust to heterogeneous data distributions and evolving characteristics in FL settings is crucial. This includes automated feature engineering, dynamic tokenizer updates for LLMs, and context-aware normalization strategies that respect local data properties while maintaining global consistency.

Privacy-Preserving Data Quality Tools. To support both the economic models of data marketplaces and the technical demands of adaptive pipelines, there is a significant need to develop privacy-preserving versions of standard data engineering tools (profilers, validators, schema matchers) with formal privacy guarantees. This research must extend beyond purely technical solutions—such as differentially private exploratory data analysis or secure multi-party computation for joint schema inference—to also address

critical human factors. This includes designing usable interfaces and auditable governance workflows that build trust and incentivize high-quality, trustworthy participation from data providers.

Standardization and Interoperability. Significant effort is also needed to establish standardized interfaces and protocols for federated data engineering, similar to how the OMOP Common Data Model standardizes clinical data. This includes standardized APIs for data quality reporting, common metadata schemas for describing federated datasets, and interoperable governance frameworks that span multiple organizations and jurisdictions.

Addressing these challenges represents critical research opportunities that could greatly accelerate the real-world deployment of privacy-preserving federated LLMs.

7 Conclusion

The convergence of LLMs and FL provides a crucial pathway to unlock the value of decentralized data in privacy-sensitive domains such as healthcare and finance. This paper presents a systematic analysis of the fundamental utility–efficiency–privacy trilemma inherent in designing Federated LLM (FedLLM) systems. We outline key challenges, including managing statistical heterogeneity, mitigating communication bottlenecks, and defending against advanced privacy attacks such as gradient inversion. By analyzing these trade-offs, reviewing existing techniques, and exploring practical applications, we highlight that no single solution suffices. Instead, effective and resilient FedLLM deployment requires a comprehensive and integrated approach. Addressing open research directions—such as developing heterogeneity-aware aggregation methods, robust privacy accounting frameworks, and standardized data engineering practices—is essential for the continued trustworthy and effective advancement of this transformative technology.

References

[1] Ye, R., Wang, W., Chai, J., Li, D., Li, Z., Xu, Y., Du, Y., Wang, Y. & Chen, S. OpenFedLLM: Training Large Language Models on Decentralized Private Data via Federated Learning. *Proceedings Of The 30th ACM SIGKDD Conference On Knowledge Discovery And Data Mining*. pp. 6137-6147 (2024)

[2] Liu, S., Wang, C., Yin, H., Molchanov, P., Wang, Y., Cheng, K. & Chen, M. DoRA: Weight-Decomposed Low-Rank Adaptation. *Proceedings Of The 41st International Conference On Machine Learning*. pp. 32100-32121 (2024)

[3] Kopiczko, D., Blankevoort, T. & Asano, Y. VeRA: Vector-based Random Matrix Adaptation. *The Twelfth International Conference On Learning Representations*. pp. 1-14 (2024)

[4] Petrov, I., Dimitrov, D., Baader, M., Müller, M. & Vechev, M. DAGER: Exact Gradient Inversion for Large Language Models. *Advances In Neural Information Processing Systems*. (2024)

[5] Ye, R., Ge, R., Zhu, X., Chai, J., Du, Y., Liu, Y., Wang, Y. & Chen, S. FedLLM-Bench: Realistic Benchmarks for Federated Learning of Large Language Models. *Advances In Neural Information Processing Systems*. (2024)

[6] Kuang, W., Qian, B., Li, Z., Chen, D., Gao, D., Pan, X., Xie, Y., Li, Y., Ding, B. & Zhou, J. FederatedScope-LLM: A Comprehensive Package for Fine-tuning Large Language Models in Federated Learning. *Proceedings Of The 30th ACM SIGKDD Conference On Knowledge Discovery And Data Mining*. pp. 5260-5271 (2024)

[7] Dave, V. & Santhanagopalan, A. Google Cloud and Swift pioneer advanced AI and federated learning tech to help combat payments fraud. (Google Cloud Blog,2024,12), Accessed October 12, 2025

[8] Zhu, G., Li, D., Gu, H., Yao, Y., Fan, L. & Han, Y. FedMIA: An Effective Membership Inference Attack Exploiting "All for One" Principle in Federated Learning. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*. pp. 20643-20653 (2025)

[9] Dettmers, T., Pagnoni, A., Holtzman, A. & Zettlemoyer, L. QLoRA: Efficient Finetuning of Quantized LLMs. *Advances In Neural Information Processing Systems*. (2023)

[10] McMahan, H., Moore, E., Ramage, D., Hampson, S. & Arcas, B. Communication-Efficient Learning of Deep Networks from Decentralized Data. *Proceedings Of The 20th International Conference On Artificial Intelligence And Statistics*. pp. 1273-1282 (2017)

[11] Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H., Patel, S., Ramage, D., Segal, A. & Seth, K. Practical Secure Aggregation for Privacy-Preserving Machine Learning. *Proceedings Of The 2017 ACM SIGSAC Conference On Computer And Communications Security*. pp. 1175-1191 (2017)

[12] Hu, E., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L. & Chen, W. LoRA: Low-Rank Adaptation of Large Language Models. *The Tenth International Conference On Learning Representations*. pp. 1-14 (2022)

[13] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Canton Ferrer, C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Koura, P., Lachaux, M., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E., Subramanian, R., Tan, X., Tang, B., Taylor, R., Williams, A., Kuan, J., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S. & Scialom, T. Llama 2: Open Foundation and Fine-Tuned Chat Models. *ArXiv Preprint ArXiv:2307.09288*. (2023)

[14] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I. & Amodei, D. Language Models are Few-Shot Learners. *Advances In Neural Information Processing Systems*. pp. 1877-1901 (2020)

[15] Xu, Z., Zhang, Y., Andrew, G., Choquette-Choo, C., Kairouz, P., McMahan, H., Rosenstock, J. & Zhang, Y. Federated Learning of Gboard Language Models with Differential Privacy. *Proceedings Of The 61st Annual Meeting Of The Association For Computational Linguistics (Volume 5: Industry Track)*. pp. 629-639 (2023)

[16] Wang, Z., Shen, Z., He, Y., Sun, G., Wang, H., Lyu, L. & Li, A. FLoRA: Federated Fine-Tuning Large Language Models with Heterogeneous Low-Rank Adaptations. *Advances In Neural Information Processing Systems*. (2024)

[17] Gonzalez Zelaya, C. Towards Explaining the Effects of Data Preprocessing on Machine Learning. *Proceedings Of The 2019 IEEE 35th International Conference On Data Engineering*. pp. 2086-2090 (2019)

[18] Li, Q., Diao, Y., Chen, Q. & He, B. Federated Learning on Non-IID Data Silos: An Experimental Study. *Proceedings Of The 2022 IEEE 38th International Conference On Data Engineering*. pp. 965-978 (2022)

[19] Habu, J., Dhabariya, A., Pal, B. & Abubakar, F. Decentralized Data Governance and Regulatory Compliance in Federated Learning and Edge Computing for Healthcare. *Research Square*. (2025)

[20] Chen, Z., Liao, W., Hua, K., Lu, C. & Yu, W. Towards asynchronous federated learning for heterogeneous edge-powered internet of things. *Digital Communications And Networks*. pp. 317-326 (2021)

[21] Lu, Z., Pan, H., Dai, Y., Si, X. & Zhang, Y. Federated Learning With Non-IID Data: A Survey. *IEEE Internet Of Things Journal*. pp. 19188-19209 (2024)

[22] Chen, J., Xu, W., Guo, S., Wang, J., Zhang, J. & Wang, H. FedTune: A Deep Dive into Efficient Federated Fine-Tuning with Pre-trained Transformers. *ArXiv Preprint ArXiv:2211.08025*. (2022)

[23] Li, T.; Sahu, A.K.; Talwalkar, A.; Smith, V. Federated Optimization in Heterogeneous Networks. *Proceedings Of The 2020 Conference On Machine Learning And Systems*. (2020)

[24] Karimireddy, S.P.; Kale, S.; Mohri, M.; Reddi, S.; Stich, S.; Suresh, A.T. SCAFFOLD: Stochastic Controlled Averaging for Federated Learning. *Proceedings Of The 37th International Conference On Machine Learning*. (2020)

[25] Reddi, S.; Charles, Z.; Zaheer, M.; Garrett, Z.; Rush, K.; Konečný, J.; Kumar, S.; McMahan, H.B. Adaptive Federated Optimization. *arXiv preprint arXiv:2003.00295*. (2020)

[26] Wang, J.; Liu, Q.; Liang, H.; Joshi, G.; Poor, H.V. Tackling The Objective Inconsistency Problem In Heterogeneous Federated Optimization. *Advances In Neural Information Processing Systems*. (2020)

[27] Arivazhagan, M. G.; Aggarwal, V.; Singh, A. K.; Choudhary, S. Federated Learning with Personalization Layers. *ArXiv Preprint ArXiv:1912.00818*. (2019).

[28] Diao, E.; Ding, J.; Tarokh, V. HeteroFL: Computation And Communication Efficient Federated Learning For Heterogeneous Clients. *International Conference On Learning Representations*. (2021)

[29] Li, D.; Wang, J. FedMD: Heterogenous Federated Learning Via Model Distillation. *Proceedings Of The 33rd Conference On Neural Information Processing Systems Workshops*. (2019)

[30] Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; et al. Overcoming Catastrophic Forgetting In Neural Networks. *Proceedings Of The National Academy Of Sciences*. (2017)

[31] Li, Z.; Hoiem, D. Learning Without Forgetting. *European Conference On Computer Vision*. pp. 614–629 (2016)

[32] Farajtabar, M.; Azizan, N.; Mott, A.; Li, A. Orthogonal Gradient Descent For Continual Learning. *Proceedings Of The 23rd International Conference On Artificial Intelligence And Statistics*. (2020)

[33] Yoon, J.; Jeong, W.; Ju, J.; Hwang, S.J.; Yang, E. Federated Continual Learning With Weighted Inter-Client Transfer. *Proceedings Of The 38th International Conference On Machine Learning*. (2021)

[34] Guo, Zhihan, et al. Fedlfc: Towards efficient federated multilingual modeling with lora-based language family clustering. *Findings of the Association for Computational Linguistics: NAACL 2024*. 2024

[35] Chen, H.; Zhang, Y.; Krompass, D.; Gu, J.; Tresp, V. FedDAT: An Approach for Foundation Model Finetuning in Multi-Modal Heterogeneous Federated Learning. *Proceedings of the AAAI Conference on Artificial Intelligence.* (2024)

[36] Shu, Y.; Hu, W.; Ng, S. K.; Low, B. K. H.; Yu, F. R. Ferret: Federated Full-Parameter Tuning at Scale for Large Language Models. In *Proceedings of the 42nd International Conference on Machine Learning.* (2025)

[37] Alistarh, D.; Grubic, D.; Li, J.; Tomioka, R.; Vojnovic, M. QSGD: Communication-Efficient SGD Via Gradient Quantization And Encoding. *Advances In Neural Information Processing Systems.* pp. 1709–1720 (2017)

[38] Karimireddy, S.P.; Rebjock, Q.; Stich, S.; Jaggi, M. Error Feedback Fixes SignSGD And Other Gradient Compression Schemes. *Proceedings Of The 36th International Conference On Machine Learning.* pp. 3252–3261 (2019)

[39] Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-T.; Rocktäschel, T.; Riedel, S.; Kiela, D. Retrieval-Augmented Generation For Knowledge-Intensive NLP. *Advances In Neural Information Processing Systems.* (2020)

[40] Lester, B.; Al-Rfou, R.; Constant, N. The Power of Scale for Parameter-Efficient Prompt Tuning. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing.* pp. 3045–3059 (2021)

[41] Li, X.L.; Liang, P. Prefix-Tuning: Optimizing Continuous Prompts For Generation. *Proceedings Of The 59th Annual Meeting Of The Association For Computational Linguistics.* pp. 4582–4597 (2021).

[42] Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; Cao, Y.ReAct: Synergizing Reasoning and Acting in Language Models. In *Proceedings of the 11th International Conference on Learning Representations.* (2023)

[43] Shi, H.; Xu, Z.; Wang, H.; Qin, W.; Wang, W.; Wang, Y.; Wang, Z.; Ebrahimi, S.; Wang, H. Continual Learning of Large Language Models: A Comprehensive Survey.*ACM Computing Surveys.* (2025).

[44] Huang, J.; Cui, L.; Wang, A.; Yang, C.; Liao, X.; Song, L.; Yao, J.; Su, J. Mitigating Catastrophic Forgetting in Large Language Models with Self-Synthesized Rehearsal. *Proceedings Of The 62nd Annual Meeting Of The Association For Computational Linguistics.* pp. 1450-1466 (2024)

[45] Shumailov, I.; Shumailov, R.; Papernot, N.; et al. AI models collapse when trained on recursively generated data. *Nature*. 630, pp. 971–978 (2024)

[46] Zhang, C., Li, S., Xia, J., Wang, W., Yan, F., Liu, Y. BatchCrypt: Efficient Homomorphic Encryption for Cross-Silo Federated Learning. *Proceedings of the 2020 USENIX Annual Technical Conference.* pp. 493–506 (2020)

[47] Chen, W.-N., Özgür, A., Cormode, G., Bharadwaj, A. The Communication Cost of Security and Privacy in Federated Frequency Estimation. *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics.* pp. 4247–4274 (2023)

[48] Tang, L.; Joshi, J. Towards Privacy-Preserving and Secure Machine Unlearning: Taxonomy, Challenges and Research Directions. *Proceedings of the IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications*. pp. 280–291 (2024)

Hyper-Scale Managed Identities and Access Control

Ivan Alagenchev[1], Shobhit Trehan[2], Kang Gui[1], Dragos Avadanei[2],
Raghavendra Kammara[1], Will Bartlett[1], Rajat Jain[2], Raghav Kaushik[2]
[1]Microsoft Identity
[2]Microsoft Cloud and AI

Abstract

Modern distributed systems demand scalable, low-latency, and flexible authorization, yet conventional directory-based Identity and Role-Based Access Control (RBAC) frameworks remain overly centralized and inflexible, falling short of the requirements posed by dynamic, distributed service-to-service (S2S) architectures. These legacy models lack the scalability, adaptability, and fine-grained, context-aware control essential for decentralized authentication and authorization at hyper-scale. This paper presents the design and architecture of Hyper-Scale Managed Identities (HSMIs) and their integration with decentralized access control policies. We analyze the limitations of traditional identity models and outline the HSMI architecture, including processes for attestation, credential issuance, binding, and the enforcement of access control policies. In addition, we introduce Attested Credential Release and its associated security guarantees. The structure of authorization tokens is presented, emphasizing their role in enabling hyper-scale, ultra-low latency scenarios. Further, we detail a capability-based authorization model that utilizes authorization tokens, attribute tokens, and namespacing to facilitate decentralized, high-performance access control.

1 Introduction

Internal services within cloud platforms require secure, reliable mechanisms to communicate with each other. For example, within Azure, data services such as Azure SQL [19] must interact with Azure Storage [20] to manage database files. Although both components belong to the same provider and operate within a shared trust boundary, robust authentication between services remains essential. This necessity stems from the principle that, even in integrated environments, each service must safeguard against lateral compromise should a breach occur elsewhere. Consequently, cloud architectures typically avoid distinguishing between first-party and third-party callers, instead enforcing authentication to ensure that vulnerabilities in one service do not proliferate across the platform.

The authentication model within cloud services reflects the complexity of their internal structure. Rather than treating a service as a monolithic entity, authentication is applied at the data-plane level: individual SQL instances and Storage containers each authenticate independently. With millions of such instances and containers operating simultaneously in large-scale environments, establishing and maintaining secure authentication relationships presents significant challenges. One commonly employed approach is the use of Shared Access Signatures (SAS) [17], whereby the Storage service uses account-specific keys to derive container-level SAS credentials. The many-to-many mapping between SQL instances and storage containers, necessitate millions of unique SAS credentials distributed across the platform. To mitigate risks, storage access keys are regularly rotated, and SAS credentials regenerated; however, this practice creates substantial operational overhead and places significant demands on the Storage service itself.

Reliance on static access keys for authentication, while straightforward, introduces limitations in both security and scalability. Static credentials behave like passwords—susceptible to theft, reuse, and

misuse—and pose ongoing security risks due to the challenges of managing and rotating them effectively. To address these issues, cloud platforms have adopted mechanisms such as Managed Identities(MIs) [14]. Under this model, a central authentication service—Microsoft Entra [18] (formerly Azure Active Directory), for instance, issues identities to individual service instances. These identities are then used to obtain short-lived access tokens, which replace static keys and offer enhanced security thanks to their limited validity. By shifting the authentication burden to a centralized authority, the MI model reduces operational complexity for service teams and mitigates the risks associated with static credentials.

Despite the advantages of Managed Identities, this approach is not without its own scalability challenges. The central authentication service must maintain an extensive directory of service identities, effectively representing a closed world of continuously changing authenticating entities. As the number of service instances scales into the billions, directory management becomes increasingly difficult, and practical limitations emerge. Furthermore, the certificate-based process for establishing MIs retains some of the operational drawbacks seen in access key rotation, such as management overhead and scheduling for periodic renewal.

In response to these persistent challenges, we propose a novel authentication method designed to enhance both security and scalability. Our approach reimagines service authentication by basing it on the health of the hosting Virtual Machine [21] (VM), leveraging hardware roots of trust such as Trusted Execution Environments [16] (TEEs)—for example, Trusted Platform Module (TPMs) or enclaves—to attest to VM integrity. Rather than verifying identities against a directory, the central authentication service validates health reports signed by keys rooted in the TEE, rendering the process stateless and highly scalable. This technique not only mitigates the risks associated with static secrets and directory-based bottlenecks but also introduces a dynamic, context-aware security posture. We illustrate the improvement in security posture through an example. In the previous approaches discussed above, suppose a malicious actor compromises a cloud VM, for instance, by installing malware on the VM. The malware would be potentially undetected at least until all secrets on the VM are rotated—this could mean days or weeks of compromise. In contrast, with the dynamic approach based on secure booting, the presence of the malware would be measured by the TPM and hence fail an attestation check, reducing the period of compromise.

The authenticating party (SQL in the above example) also needs the right permissions to access resources (Storage in the above example). Hence, there is the need to solve an authorization problem based on the same stateless infrastructure. As an example, data services like Azure SQL manage data across millions of storage accounts, with each individual storage account capable of hosting thousands of containers. Given that data service instances (for example, SQL instances) maintain complex many-to-many relationships with storage containers to optimize both resource utilization and packing, the volume of mappings between SQL instances and storage containers increases substantially. Modelling this authorization pattern through centralized directory services for resolution would necessitate billions of role assignments per subscription and surpass the practical limits for both role assignment storage and evaluation. We also describe our Attribute Based Access Control solution to the authorization problem.

The proposed architecture involves calling multiple components, which on the face of it, poses a performance challenge, since calling multiple components in the data path is expensive. However, we use the notion of tokens and prefetching to ensure that the data path in the hot case does not call all components. This allows us to approach our goal of sub-millisecond latencies in the 99th percentile (P99), which is comparable to the SAS and MSI methods described above. Therefore, the improved security and scalability of HSMI does not carry a performance price.

Azure is currently piloting this technology in production, with plans for broad deployment across its ecosystem. We believe this evolution marks a significant step forward for service-to-service authentication in cloud environments, setting a new standard for secure and scalable operations.

2 Background and Terminology

2.1 Microsoft Entra

Microsoft Entra [18] (formerly Azure Active Directory) is a family of identity and network access products. It lets organizations implement a security strategy and create a trust fabric that verifies identities, validates access conditions, checks permissions, encrypts connection channels, and monitors for compromise.

2.2 Azure SQL

Azure SQL [19] is a Platform as a Service (PaaS) offering built upon the SQL Server engine, hosted on Azure Service Fabric. This architecture features separated compute and storage layers, with Azure Storage used for the persistence of database files, such as log (ldf) and data (mdf) files. The service incorporates a control plane, hereafter referred to as the SQL Control Plane, which oversees the lifecycle management of various databases. Additionally, the service comprises a data plane, representing the sqlservr.exe processes running on compute nodes within the Service Fabric cluster. In this context, the term 'SQL Instance' denotes a single sqlservr.exe process. Although the Azure SQL service supports multiple storage configurations, the approach discussed in this paper is exemplified by a SQL database that utilizes Azure Blob Storage containers to host its data. Azure SQL runs on Azure Service Fabric [22], Microsoft's distributed systems platform for deploying and managing microservices and containers across a cluster of machines. In this context, a cluster is the overall set of machines that run Service Fabric, while a node refers to an individual compute unit (typically a virtual machine in Azure) within that cluster.

2.3 Managed Identities (MI)

Azure provides automatically managed identities [14] that can be assigned to compute resources such as Virtual Machines, Virtual Machine Scale Sets, Azure Kubernetes Service clusters, or supported application hosting platforms. These identities allow services to authenticate to Azure resources without the need for developers to provision or manage secrets, credentials, certificates, or keys, thereby reducing the risk of credential exposure and simplifying secure communication between services.

2.4 SPIFFE (Secure Production Identity Framework for Everyone)

An open-source framework [10], the Secure Production Identity Framework for Everyone (SPIFFE), provides a standard for securely issuing and managing identities for services in a cloud-native or distributed system to enable secure service-to-service authentication and workload identity across heterogeneous environments. A SPIFFE ID is a unique identity assigned to a workload (e.g., a service or process). A SPIFFE Verifiable Identity Document (SVID) is the cryptographic credential—typically an X.509 certificate or JWT—that proves possession of a SPIFFE ID and is used by workloads to authenticate to each other securely.

2.5 Hyper-Scale Managed Identities (HSMIs)

A novel identity framework designed for cloud-native environments that decouples identity from centralized directories and leverages hardware attestation for scalable authentication. Each HSMI is uniquely identified by its SPIFFE identifier and we use the verb 'SPIFFE identifier' or 'HSMI identifier' interchangeably in this paper.

2.6 Trusted Execution Environments (TEEs)

A Trusted Execution Environment [16] is a segregated area of memory and CPU that's protected from the rest of the CPU by using encryption. Any code outside that environment can't read or tamper with the data in the TEE. Authorized code can manipulate the data inside the TEE.

2.7 Resource Management Authorities (RMAs)

Resource Management Authorities (RMA)s are compute orchestration engines such as Kubernetes and Service Fabric, a commonly used orchestration engine within Azure. Typically, RMAs manage VM capacity by bundling them into clusters. Within each cluster, a process or container running the application (SQL in our example) is launched. RMAs attest to a centralized credential authority regarding the validity of the resources they manage. Consequently, the credential authority issues HSMI credentials. Traditionally, identity providers (IdPs) issue identity tokens for entities within their administrative domain; however, in this model, the resource providers do not issue identity tokens themselves. Instead, responsibility for token issuance is delegated to a centralized service, which retains control over the signing keys trusted by the broader application ecosystem. Thus, these resource providers manage the lifecycle of the resources under their supervision and are referred to as Resource Management Authorities (RMAs) to distinguish them from traditional IdPs. RMAs assign identifiers to each application instance. Resource identifiers follow a hierarchical format compatible with SPIFFE standards, with each identifier prefixed by the RMA's reserved namespace. Notably, scaling the RMAs to handle the number of resources they manage is a non-optional requirement. Furthermore, RMAs possess an intimate awareness of partitioning opportunities and constraints that their topology may impose, which is often more nuanced than the topology managed by Microsoft Entra. The hierarchical structure encodes resource-specific metadata, thereby enabling logical segmentation of the resource domain into distinct security zones. In addition, it facilitates efficient bulk operations such as credential and token revocation.

```
spiffe://<internal-prefix>/s/10ef5b45-a7e5-4f96-9d11-90e8b5e06a87/rg/test-eus-rg/sf/test-eus-cluster
    /7af6ddcc-8407-427d-ac61-5a47a0ea8e00/SqlApplicationType/SqlApplicationName
```

In the above example, let's consider a SQL instance (process) running on an Azure Service Fabric (RMA) cluster, the resource identifier encodes the subscription '10ef5b45-a7e5-4f96-9d11-90e8b5e06a87', the resource group 'test-eus-rg' and the cluster 'test-eus-cluster', followed by a unique identifier of the cluster i.e. '7af6ddcc-8407-427d-ac61-5a47a0ea8e00'. These values represent the service fabric cluster where this SQL instance is being hosted. The next two values 'SqlApplicationType' and 'SqlApplicationName' encode the SQL instance and represent the sqlservr.exe process running on the compute instance. This resource identifier uniquely identifies an HSMI.

2.8 Attribute-Based Access Control (ABAC)

An access control model that evaluates permissions based on attributes of the subject, resource, and environment, enabling more flexible and context-aware authorization decisions [15]. There are three types of attributes we highlight—Principal Attributes, Resource Attributes and Environmental Attributes. In a distributed setting, Principal Attributes describe the identity which is requesting access e.g., tenant, environment or service role. In our approach, HSMI principals are directory-less, and hence, we describe principal attributes with an attribute token bound to particular HSMI. We introduce this in Section 2.11. Environmental Attributes represent the contextual information about the access request e.g. time of day, IP address, device posture etc. Lastly, Resource Attributes are associated with the resource which is being accessed e.g. Azure SQL might access Azure Storage as a resource and resource attribute in this flow could be Storage Blob Container Metadata—a key value pair that stores properties of the

container. As an example, let's consider the storage account container 'mycontainer' in the storage account 'mystorageaccount', and that the container hosts data for a SQL Server, managed by SQL Control Plane, with a server identifier 'b3f2c9d4-8a7e-4f1a-9d3b-7e6c2a1f5e8d'. The resource attributes, stored in the metadata can be—attribute key:readAccessGroups and attribute value: b3f2c9d4-8a7e-4f1a-9d3b-7e6c2a1f5e8d. This example is for illustration and we explain how to interpret and use these values in Section 3.

2.9 Attribute Authority

The Attribute Authority manages and issues cryptographically signed attribute tokens that bind attribute values to their respective HSMI instances. Attribute tokens are signed by a trusted central signing authority, ensuring that attribute modifications are prevented even in the event of a compromised HSMI instance. Authorized control planes, such as the SQL control plane, that manages the lifecycle of SQL databases running on hosting platforms or RMAs like Service Fabric, possess the ability to request attribute tokens for HSMI identities under their management from the attribute authority. Management of the attribute authority service is conducted by the authorization authority, guaranteeing consistency and robust security.

2.10 Attribute Namespace

Hyperscale identity systems require a decentralized approach to attribute management to avoid bottlenecks and single points of failure. To facilitate this, the Attribute Authority employs a namespacing model that enables decentralized attribute management by control planes. The control plane owns the attribute names and is responsible for defining the association between attributes and HSMI identities. Each attribute is uniquely identified within a namespace owned by the control plane. This design supports horizontal scaling and reduces state management overhead while maintaining clear ownership boundaries. Namespace ownership is non-transferable and ensures that only authorized control planes can attest to attributes within their designated scope. An example of a namespace owned by the SQL Control Plane in East US region, for principal attributes assigned to SQL databases in that region is—`Microsoft.AttributeAttestation/AttributeNamespaces/SqlEus`

2.11 Attribute Tokens

Attribute tokens encapsulate principal attributes—such as tenant, environment, and service role—in a cryptographically signed format. Issued by the Attribute Authority, these tokens function as inputs to Attribute-Based Access Control (ABAC) systems. Unlike traditional Microsoft Entra directory-based group memberships, attribute tokens are both portable and verifiable. The Attribute Authority addresses a fundamental challenge presented by directory-less identity architectures: securely binding attributes to HSMI principals that lack directory representation. As HSMIs do not inherently support principal attributes necessary for ABAC grouping, the Attribute Authority furnishes a secure mechanism for delegating attribute association to authorized control planes, while preserving robust cryptographic security guarantees. The format of such an attribute token is:

```
{
   "sub": "<HSMI_ID>",
   "xms_attr": {
      "Microsoft.AttributeAttestation/AttributeNamespaces/{namespace}/Attributes/{attribute1}": <
          Value1>,
      "Microsoft.AttributeAttestation/AttributeNamespaces/{namespace}/Attributes/{attribute2}": <
          Value2>,
   },
   "xms_acb": "<ACB_value_to_bind_to_auth_token>",
```

```
}
```

Let's consider the same example as Section 2.8, a SQL Server, managed by SQL Control Plane, with a server identifier 'b3f2c9d4-8a7e-4f1a-9d3b-7e6c2a1f5e8d'. This SQL Server will host multiple SQL instances in the data plane scattered across various Compute Nodes and Azure Service Fabric clusters. We take the example of a single instance named 'SqlApplicationName', also referred in Section 2.7. The below attribute token encodes namespaced attributes assigned to the HSMI for this instance. The 'sub' claim identifies the HSMI, and 'xms_attr' claim holds namespaced attribute key-value pairs. This attribute value helps SQL Control plane maintain a logical grouping over all the SQL Instances hosted on various RMAs. We further explain how to interpret these values in Section 3 and Section 5. Additionally, we explain the 'xms_acb' claim in Section 6.1.

```
{
   "sub": "<internal-prefix>/s/10ef5b45-a7e5-4f96-9d11-90e8b5e06a87/rg/test-eus-rg/sf/test-eus-
       cluster/7af6ddcc-8407-427d-ac61-5a47a0ea8e00/SqlApplicationType/SqlApplicationName",
   "xms_attr": {
      "Microsoft.AttributeAttestation/AttributeNamespaces/SqlEus/Attributes/readAccessGroups": "
          b3f2c9d4-8a7e-4f1a-9d3b-7e6c2a1f5e8d",
   },
   "xms_acb": "<ACB_value_to_bind_to_auth_token>",
}
```

2.12 Policy Decision Point - Remote (PDP-R)

A distributed authorization service that evaluates access policies and issues capability tokens based on principal attributes and resource permissions. It is designed to evaluate access control decisions remotely, rather than locally within the resource itself.

3 Architecture Overview

To understand the architecture of HSMIs, we consider an example flow of the Azure SQL service provisioning a database. In contemporary platforms such as Microsoft Azure, services like Azure SQL Database interact directly with Azure Storage to manage data and log files, thus ensuring transactional consistency and enabling crash recovery in the event that the compute VM crashes, without replication. Authentication is enforced at the data-plane, with each SQL instance and storage container handling authentication and authorization autonomously. SQL service instances are instantiated by the Data Control Plane on a hosting platform such as Service Fabric (the Resource Management Authority or RMA). Logical grouping of SQL instances is orchestrated at the server level, with each group distinguished by a unique serverId, typically in GUID format. These instances may operate across multiple clusters or migrate between them, while storage containers are shared among instances that belong to the same logical server group. In this example, access control can leverage the serverId as an attribute, linking instances and containers within the same group.

As seen in Figure 1, the process commences with the regional SQL Control Plane claiming an attribute namespace by interfacing with the Attribute Authority. For example a regional SQL Control Plane in East US might claim the namespace `Microsoft.AttributeAttestation/AttributeNamespaces/SqlEus`. Once this namespace is registered, only the originating control plane is authorized to obtain attributes from it, thereby establishing strict authorization boundaries. Storage subscriptions are maintained by the control plane, which provisions a conditional role assignment on each such subscription—only those identities possessing attributes from a specific namespace are permitted access to storage containers tagged with matching metadata attributes within a subscription scope. This configuration is a necessary

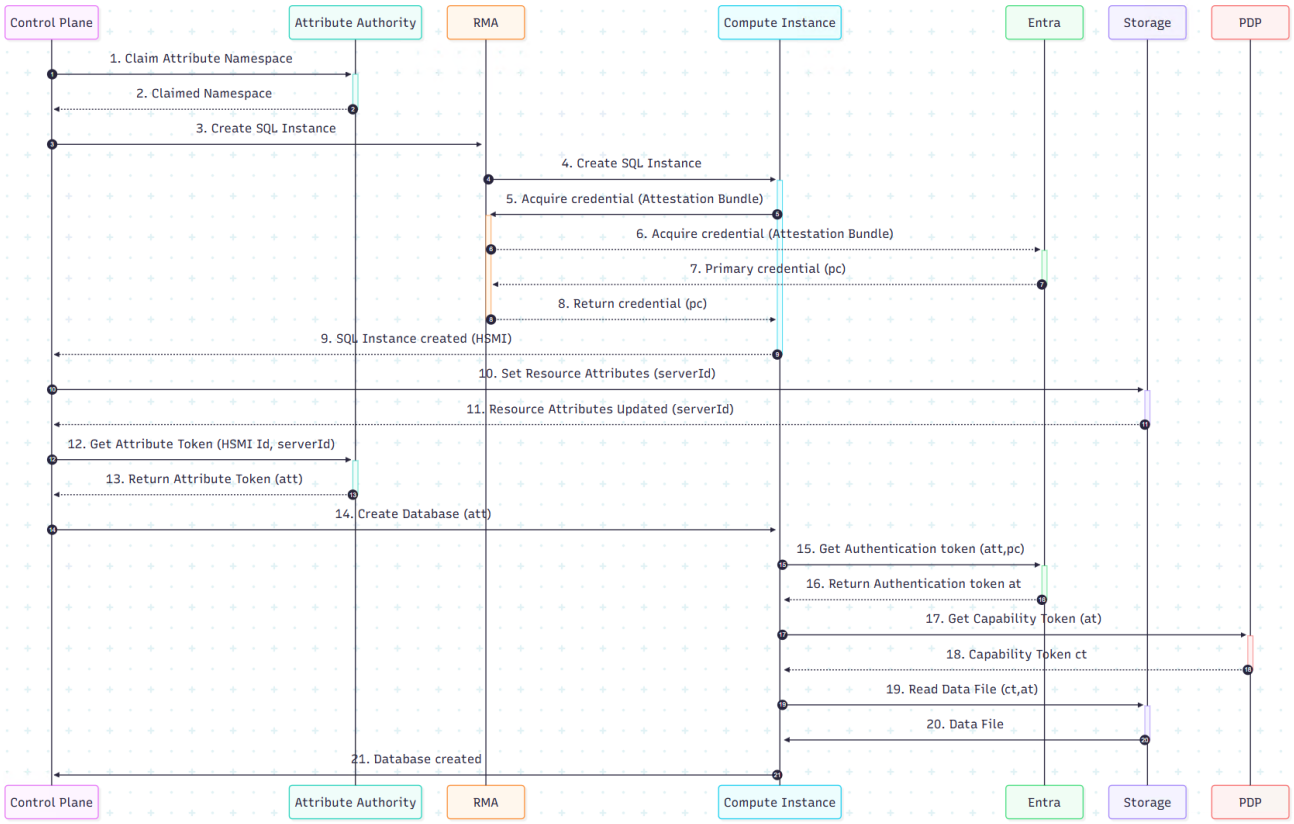


Figure 1: Example End-to-End Flow

precursor for SQL instances to access storage resources based on attributes. An example of such a role assignment, further described in Section 6 is—

```
{
   "Id": "b69889ff-3281-4ffa-8f75-6d7bd3be6616",
   "RoleDefinitionId": "ba92f5b42d11453da403e96b0029c9fe",
   "PrincipalId": "00000000-0000-0000-0000-000000000000",
   "Scope": "/subscriptions/f984cbdd-9e7e-4b97-9744-5c5d9295e332",
   "Condition": "@Principal[Microsoft.AttributeAttestation/AttributeNamespaces/SqlEus/attributes/
       readAccessGroups] ForAnyOfAnyValues:StringEqualsIgnoreCase SplitString{@Resource[Microsoft.
       Storage/storageAccounts/blobServices/containers/metadata:readAccessGroups]}"
}
```

Upon initiation of database creation, the SQL Control Plane instructs the Resource Management Authority (RMA), Service Fabric, to create a new SQL instance, marking eligibility for Hyperscale Managed Identity (HSMI). The RMA delegates provisioning to the compute instance, which then acquires a credential for the newly created HSMI—cryptographically bound to the resource instance. We explain credential issuance in Section 4. After acquiring the credentials, the compute instance returns an HSMI identifier back to SQL Control Plane—

```
spiffe://<internal-prefix>/s/10ef5b45-a7e5-4f96-9d11-90e8b5e06a87/rg/test-eus-rg/sf/test-eus-cluster
    /7af6ddcc-8407-427d-ac61-5a47a0ea8e00/SqlApplicationType/SqlApplicationName
```

In the example above, the identifier encodes the subscription '10ef5b45-a7e5-4f96-9d11-90e8b5e06a87', the resource group 'test-eus-rg' and cluster name 'test-eus-cluster', followed by a unique identifier of the cluster i.e. '7af6ddcc-8407-427d-ac61-5a47a0ea8e00'. These values represent the service fabric cluster where this SQL instance is being hosted. The next two values 'SqlApplicationType' and 'SqlApplicationName' encode the SQL instance and represent the sqlservr.exe process running on the

compute instance. Subsequent to instance provisioning, the SQL Control Plane designates appropriate storage containers for the instance and marks their metadata with the 'serverId' attribute, thus enabling access for all instances within the same logical grouping. As an example let's consider a new SQL database 'testdb' being provisioned in a server 'testsvr', and SQL control plane identifies 'testsvr' with a GUID 'b3f2c9d4-8a7e-4f1a-9d3b-7e6c2a1f5e8d'. SQL Control Plane will update designate a storage account (e.g. 'myaccount') and container (e.g. 'mycontainer') and update the container's metadata—

```
PUT https://mystorageaccount.blob.core.windows.net/mycontainer?comp=metadata
x-ms-meta-readAccessGroups: b3f2c9d4-8a7e-4f1a-9d3b-7e6c2a1f5e8d
```

The next operational step involves obtaining an encrypted attribute token from the Attribute Authority within the established namespace, embedding the same serverId. An example of such an attribute token is—

```
{
   "sub": "<internal-prefix>/s/10ef5b45-a7e5-4f96-9d11-90e8b5e06a87/rg/test-eus-rg/sf/test-eus-
       cluster/7af6ddcc-8407-427d-ac61-5a47a0ea8e00/SqlApplicationType/SqlApplicationName",
   "xms_attr": {
      "Microsoft.AttributeAttestation/AttributeNamespaces/SqlEus/Attributes/readAccessGroups": "
          b3f2c9d4-8a7e-4f1a-9d3b-7e6c2a1f5e8d",
   },
   ...
}
```

This token is securely stored in a key-value repository accessible by the SQL instance. During the database bootstrapping phase, the SQL instance utilizes its primary credential and the attribute token to request an authentication token from Microsoft Entra. We describe this authentication token flow with the primary credential and attributes in Section 5. The SQL instance subsequently calls the Authorization Service and Policy Decision Point (PDP) to retrieve a capability token containing precomputed authorization decisions. We describe the capability token based authorization model further in Section 4. Armed with authentication and authorization tokens, the SQL instance is authorized to interact with storage to retrieve or update data files as required. Note that for illustration we use the example of 'Read' permissions, and the same principle can be applied to 'Write' permissions.

4 Primary Credential Issuance

The credentialing process begins when a resource owner initiates the creation of a resource and designates it as eligible for HSMI identity. At this stage, the RMA assigns an HSMI identifier to the resource and records it within its internal registry. Credential issuance then produces artifacts—such as JWTs [12], X.509 certificates [13], and SPIFFE SVIDs [10]—that are cryptographically tied to a specific attested resource instance. Attestation serves as a key security primitive in the credential issuance lifecycle, acting as the mechanism by which the integrity of a compute instance is verified. During startup, the virtual Trusted Platform Module, or vTPM, records cryptographic measurements of the firmware, bootloader, operating system kernel, and critical system configuration into Platform Configuration Registers. These measurements are digitally signed using a hardware-backed root of trust, producing attestation evidence that is then submitted to the Microsoft Azure Attestation service. The attestation service maintains minimal policy state, which defines the expected secure configuration of the virtual machine, including legitimate software versions, enabled code integrity checks, and secure boot settings. The service verifies that the measurements conform to this policy and, if so, issues a signed attestation token.

As seen in Figure 2, upon resource creation, the RMA, assigns the compute resource a compute unit ID and an access ID. In step 2, a per-boot proof-of-possession (PoP) key is generated within protected memory. In step 4, an attestation token is retrieved from a trusted attestation provider by providing an attestation evidence. This attestation token is cryptographically bound to the per-boot key. This

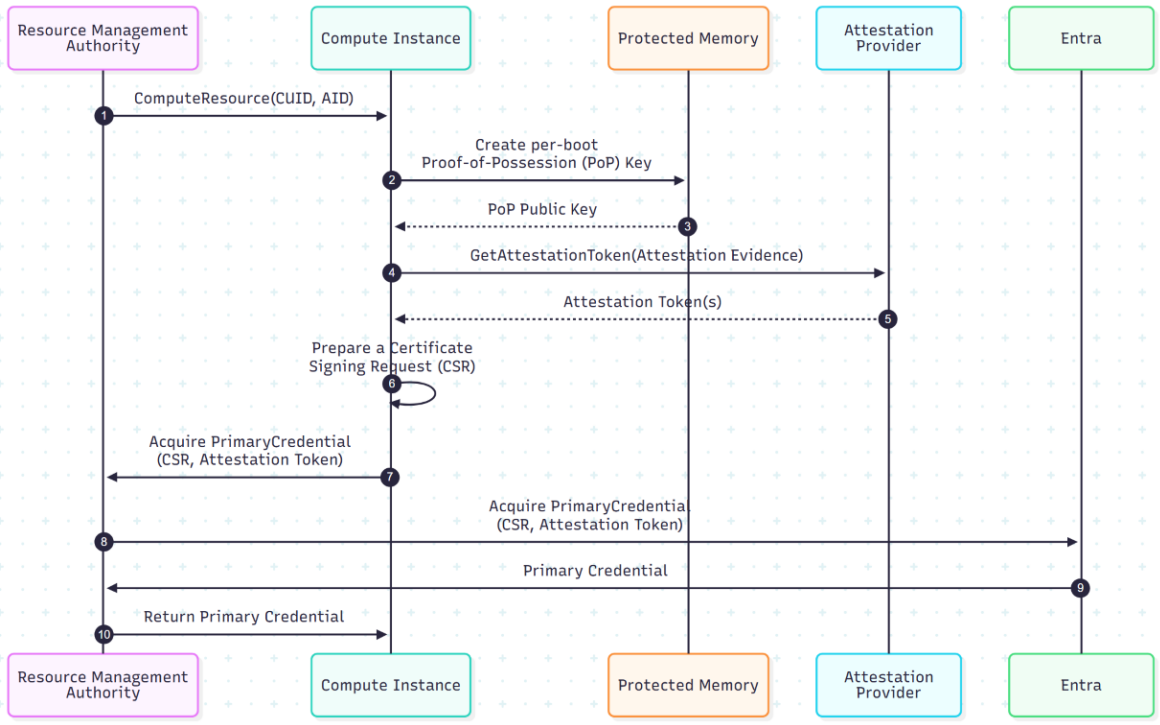


Figure 2: Credential Issuance Flow

token, along with the PoP public key, is returned to the Compute Instance in step 5. Reboots require re-attestation. Steps 2 and 4 occur within a protected execution environment (e.g., a secure enclave) during the resource's bootstrap phase. In step 8, the RMA verifies that the attestation token and the compute unit identifier (e.g., VMID or equivalent) correspond to a trusted resource in its registry and then requests a primary credential from the credential authority on behalf of the resource. This request includes the attestation bundle (comprising tokens from various attestation providers), the device binding key, a Certificate Signing Request (CSR) the resource creation timestamp, and two identifiers: (1) the HSMI ID and (2) a platform-specific resource ID. Both identifiers are included to maintain compatibility with existing access control policies. If either identifier changes, the RMA must issue a new assertion and request a new primary credential. Authorization assignments tied to the old identifier are migrated to the new one using a three-step process: (1) create new, (2) inherit from old, and (3) deprecate old. Upon successful validation of the RMA's assertion, the attestation tokens, and the namespace ownership of the HSMI, the identity authority issues a primary credential (e.g., JWT [12], X.509 [13], SPIFFE SVID [10]) that is cryptographically bound to the resource. This credential includes the finalized HSMI ID and the resource creation timestamp. In step 10, the primary credential is returned to the resource, which can then use it along with the proof of possession (PoP) of the binding key—to request access tokens from the credential authority.

5 Authentication

Once a resource has obtained its primary credential, it can request access tokens from the credential authority to authenticate to other services. These access tokens must be bound to the compute instance to ensure highest level of security is maintained. In Figure 3, Step 1, the compute instance uses the

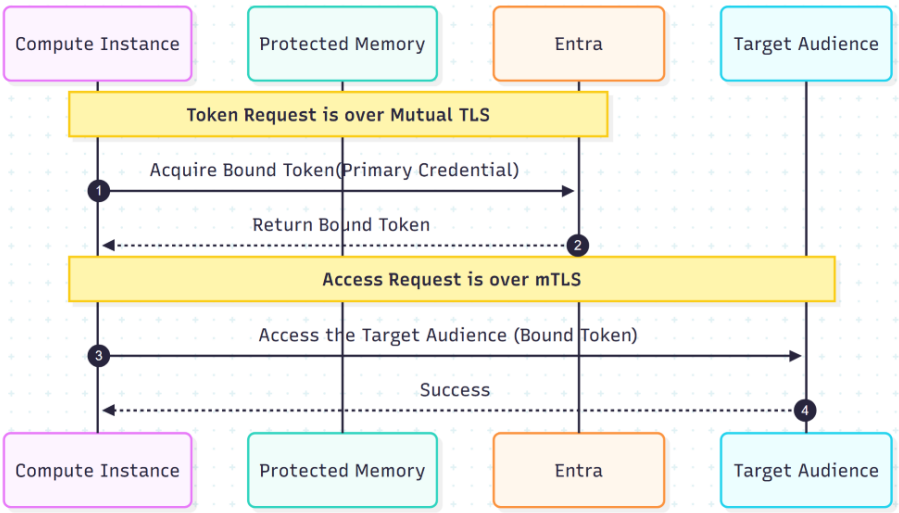


Figure 3: Token Issuance and Usage Flow

bound credential obtained in the previous step for acquiring a bound authentication token. Microsoft Entra token issuance service validates the credential and Proof-of-Possession of the private key used to obtain the primary credential. Once all the necessary validations are done, Microsoft Entra returns a bound token to the compute instance. In step 3, the compute instance uses this bound token to access the target audience i.e. Azure Storage.

As described in Section 2.11, the SQL control plane requests a signed attribute token from the Attribute Authority for a given HSMI and its associated attributes. The Attribute Authority validates that the SQL control plane possesses appropriate access rights, subsequently generating and returning a cryptographically signed attribute token containing the HSMI ID and requested attributes. This token is stored by the SQL instance for use in subsequent authentication requests. In the authentication token request to Microsoft Entra, the SQL instance provides the signed attribute token; Microsoft Entra verifies the signature and federates the attributes as claims in the resultant authentication token. This mechanism securely binds principal attributes to the authentication context. HMSIs can obtain attribute tokens from many attribute authorities and present them to Entra ID in a single authentication request. The resulting authentication token contains all attributes, segregated by their respective attribute authority.

In the example below we illustrate an authentication token obtained for the example HSMI described in Section 3. The claims 'cnf' and 'xms_tbflags' are used for token binding, and the 'xms_attr' claim encodes the principal attributes within the authentication token. We explain the 'xms_acb' claim in Section 6.1.

```
{
   "sub": "<internal-prefix>/s/10ef5b45-a7e5-4f96-9d11-90e8b5e06a87/rg/test-eus-rg/sf/test-eus-
       cluster/7af6ddcc-8407-427d-ac61-5a47a0ea8e00/SqlApplicationType/SqlApplicationName",
   "cnf": {
      "x5t#S256": "<hash_of_the_credential>"
   },
   "xms_tbflags":"1",
   "xms_attr": {
      "<encoded_attribute_authority_identifier>":
      {
         "Microsoft.AttributeAttestation/AttributeNamespaces/SqlEus/Attributes/readAccessGroups": "
             b3f2c9d4-8a7e-4f1a-9d3b-7e6c2a1f5e8d",
      }
   },
```

```
    "xms_acb": "<ACB_value_to_bind_to_auth_token>",
    ...
}
```

6 Authorization

We leverage Capability tokens that encapsulate pre-evaluated access rights in a signed, portable format, issued by authorization systems and authenticated by a trusted authority (e.g., Microsoft Entra). The token is carried by the accessing identity and presented to resource providers as part of access requests. Resource providers validate the token and authorize requests according to the client's request and the access rights delineated within the token. This approach eliminates the need for runtime policy fetching and evaluation, thereby facilitating strong caching affinity—a feature that remains challenging in distributed systems. This model delegates policy evaluation to the token issuer, allowing the resource to validate and enforce permissions without runtime policy loading and checks.

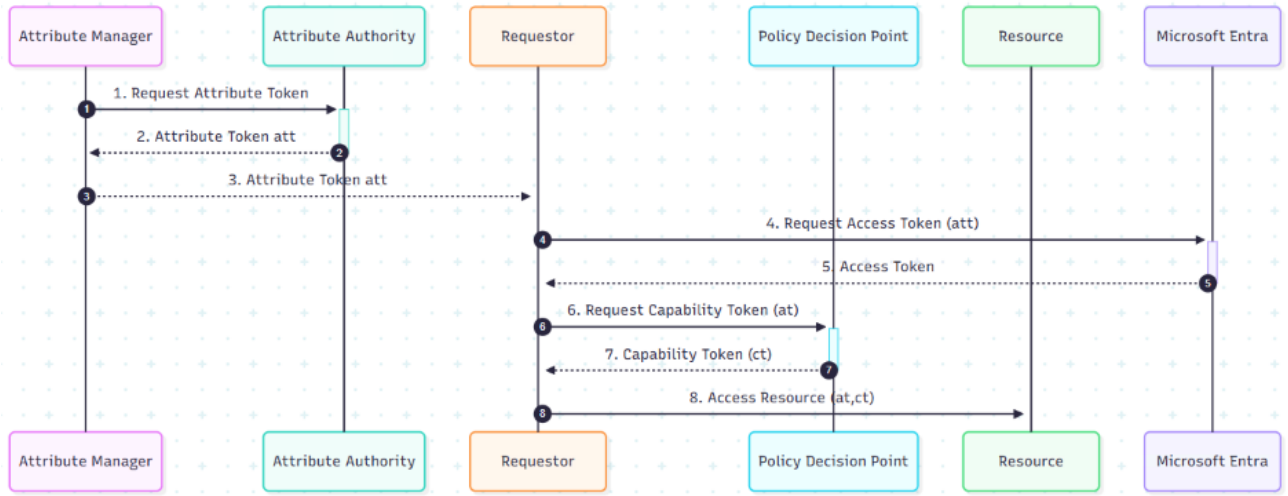


Figure 4: Combined Authorization Flow

Partial evaluation is a concept discussed in [11]. Resource attributes, along with certain environment attributes, may not be accessible at the time of capability token issuance. Consequently, the authorization service may be unable to fully evaluate all ABAC policies during issuance if such policies rely on unavailable attributes. To address these cases, the authorization service provides support for partial evaluation of ABAC policies where feasible, while retaining any remaining policies for deferred evaluation upon subsequent token requests. These partially evaluated policies are stored within the capability token. The resource owner bears responsibility for furnishing the relevant attributes and conducting the secondary evaluation of the deferred policies from the token, thereby ensuring that all necessary attributes are present and verified prior to granting access.

In the proposed ABAC model, a single role assignment, for each permission, added at a pre-defined scope encodes the policy such that only the identities possessing principal attributes from a specific namespace are permitted access to resources tagged with matching resource attributes. In the example below we illustrate the role assignment for the scenario described in Section 3, the 'Blob Container Read' role, identified by 'ba92f5b42d11453da403e96b0029c9fe' is granted to all principals i.e. represented by the universal identifier '00000000-0000-0000-0000-000000000000'. This assignment is scoped within the subscription 'f984cbdd-9e7e-4b97-9744-5c5d9295e332', along with an ABAC condition. The ABAC

condition ensures that HSMIs possessing a Principal attribute value for the key 'readAccessGroups', under the 'SqlEus' namespace, owned by SQL Control Plane, are authorized to read Storage containers provided that the Resource Attribute i.e. container metadata value for the same key matches.

```
{
   "Id": "b69889ff-3281-4ffa-8f75-6d7bd3be6616",
   "RoleDefinitionId": "ba92f5b42d11453da403e96b0029c9fe",
   "PrincipalId": "00000000-0000-0000-0000-000000000000",
   "Scope": "/subscriptions/f984cbdd-9e7e-4b97-9744-5c5d9295e332",
   "Condition": "@Principal[Microsoft.AttributeAttestation/AttributeNamespaces/SqlEus/attributes/
       readAccessGroups] ForAnyOfAnyValues:StringEqualsIgnoreCase SplitString{@Resource[Microsoft.
       Storage/storageAccounts/blobServices/containers/metadata:readAccessGroups]}"
}
```

Using the authentication token with federated attributes, the SQL instance requests a capability token from the authorization service for specific target storage resources. The authorization service performs authorization evaluation using PDP-R, considering both the principal attributes and target resource permissions. Upon successful evaluation, the authorization service returns a signed capability token containing pre-evaluated access decisions for the requested resource-action pairs. The claims in the capability token include—the identity performing the action (e.g. HSMI identifier), the targeted resource scope (e.g. Storage Account Container), and the permissions specifying which actions such e.g. Read or Write are authorized. An illustrative example of a capability token for the scenario described in Section 3 is provided below, where 'sub' claim identifies the HSMI, and 'xms_authz' claim encodes precomputed decisions for the 'Blob Container Read' permissions in the subscription scope 'a7c23d70-ffbd-4e6e-eee2-fc18f9518db2'. The claim 'ac' refers to 'access check' and the claim 'c' refers to 'condition'.

```
{
   "sub": "<internal-prefix>/s/10ef5b45-a7e5-4f96-9d11-90e8b5e06a87/rg/test-eus-rg/sf/test-eus-
       cluster/7af6ddcc-8407-427d-ac61-5a47a0ea8e00/SqlApplicationType/SqlApplicationName",
   "xms_acb": "<ACB_value_to_bind_to_auth_token>",
   "xms_authz": {
      "ac":{
         "/subscriptions/a7c23d70-ffbd-4e6e-eee2-fc18f9518db2":{
            "Microsoft.Storage/storageAccounts/blobServices/containers/blobs/read":{
               "ac":[{
                  "c":"@Principal[Microsoft.AttributeAttestation/AttributeNamespaces/SqlEus/
                      attributes/readAccessGroups] ForAnyOfAnyValues:StringEqualsIgnoreCase
                      SplitString{@Resource[Microsoft.Storage/storageAccounts/blobServices/containers
                      /metadata:readAccessGroups]}",
                  }]
   ...
}
```

The SQL Instance presents both the authentication token and capability token when accessing Azure Storage resources. The storage service validates both tokens and performs authorization based on the pre-evaluated permissions in the capability token. All tokens are cryptographically bound. The xms_acb claim ensures that all tokens are bound to the same authentication context. The capability token includes the xms_acb claim from the original authentication token, ensuring that leaked capability tokens cannot be used by unauthorized identities. This claim is further described in Section 6.1

6.1 Authorization Token Binding for Context Integrity

In distributed identity and access management systems, authorization token binding is essential for maintaining the integrity and contextual consistency of security tokens. This mechanism mitigates token mix-up attacks, where tokens intended for one context are erroneously or maliciously reused elsewhere. Token binding associates each token with a unique representation of its authorization context, ensuring validity strictly within the intended scope. Authorization and attribute tokens encapsulate claims about

a principal and its context, including client identity, IP address, cryptographic keys, identity type, roles, and external attributes. Without effective binding, tokens are vulnerable to substitution, resulting in potential unauthorized access. The Authorization Context Binding (ACB) claim i.e. 'xms_acb' addresses this by uniquely identifying the token's context. The ACB is computed as a cryptographic digest (e.g., SHA-256) over a normalized, serialized dictionary of context claims, ensuring consistency despite application-specific configurations. The ACB is embedded in authentication tokens and propagated to attribute and authorization tokens. Resources validate incoming requests by confirming matched ACB values across authentication and authorization tokens; mismatched or absent ACB values result in access denial. This process enforces context-specific token usage and supports independent refresh operations for authentication and authorization tokens. This mechanism enhances security by maintaining contextual integrity, enabling flexible token lifecycles, and providing strong guarantees against token misuse. The ACB claim is designed to be unique per principal and context, stable across token expirations, consistent for different audiences, and applicable across authentication, attribute, and authorization tokens, thereby facilitating secure token chaining.

7 Performance

Large data services such as Azure SQL are highly sensitive to I/O performance, especially during database recovery, failover, and cold start operations, where increased I/O latency can impact system reliability. The design described above involves calling multiple services. A straightforward implementation that calls all components every single time would significantly increase the latency of the service-to-service call. However, while multiple components are logically involved, the hot path does not call most of them. In the SQL-Storage example used above, the calls to the various components result in authentication and authorization tokens that are available to the SQL resource. Once the tokens are fetched, the data path involves SQL presenting the tokens to Storage, which redeems them efficiently. Current performance targets set the 99th percentile (P99) hot I/O path latency at 1 millisecond, which is comparable to the SAS and MSI methods described in Section 1; in particular, the improved security and scalability of HSMI does not carry a performance price. Experimental results for the described scenario, without token binding, show a P99 latency that is close to our target. Ongoing architectural improvements aim to reach the sub-millisecond P99 target.

The tokens, once fetched, have a Time-To-Live (TTL) in the order of hours. In order to avoid latencies when tokens expire, they are refreshed in the background at half their TTL, ensuring valid tokens are always available and minimizing latency spikes during critical operations. Capability tokens also prefetch most authorization decisions, enabling Azure Storage to process remaining authorization logic with minimal delay.

8 Related Work

Our work builds upon a rich body of research and industry practices in capability-based authorization and identity management. Several prior efforts have explored these concepts in depth. For instance, Li et al. extend OAuth to support enforcement of permission sequences and contextual constraints in distributed systems [3]. Alphabet's Fuchsia operating system adopts a capability-based model to govern inter-component permissions [4], and as of August 2021, it has been deployed across all Nest Hub devices [5]. Amazon Web Services offers a related mechanism through its session token model [6], though it is tailored to specific services like S3. In contrast, our approach generalizes the concept to support broader, cross-service scenarios. Similarly, the shift toward managed identities as a replacement for secrets and service accounts is gaining momentum across all major cloud providers [8], [7]. This

trend includes the adoption of hierarchical identity frameworks such as SPIFFE [10], which simplify permission management and delegation workflows [9]. What sets our work apart is the integration of these ideas into a unified model that treats attestation as a first-class primitive. By addressing the scalability and operational challenges of identity and authorization in cloud-native environments, our solution delivers a robust, extensible foundation for secure access control in modern distributed systems.

9 Conclusion

HSMIs and capability-based authorization tokens provide a scalable, secure, and flexible foundation for access control in cloud-native environments. By decoupling identity from directories and leveraging attestation and token binding, HSMIs address the limitations of traditional models and enable new scenarios in distributed systems.The capability-based authorization model, augmented with attribute tokens and namespacing, offers a scalable, flexible, and performant alternative to RBAC for modern distributed systems. By supporting decentralized, low-latency access control, this model is particularly well-suited for S2S scenarios. As enterprises adopt dynamic, federated architectures, this approach provides a strong foundation for secure and efficient authorization.

References

[1] I. Alagenchev et al., "Managed Identity Client Credential Release API Spec," Microsoft Internal Document, 2025.

[2] I. Alagenchev et al., "End to End S2S Token Binding Design," Microsoft Internal Document, 2025.

[3] A. S. Li, R. Safavi-Naini, and P. W. L. Fong, "A Capability-based Distributed Authorization System to Enforce Context-aware Permission Sequences," in *Proc. of the 27th ACM Symposium on Access Control Models and Technologies (SACMAT '22)*, New York, NY, USA: ACM, 2022, pp. 13–24. DOI: 10.1145/3532105.3535014.

[4] Fuchsia Project, "Capabilities - Fuchsia Component Framework," *Fuchsia.dev*, [Online]. Available: `https://fuchsia.dev/fuchsia-src/concepts/components/v2/capabilities`. [Accessed: Sep. 2, 2025].

[5] Wikipedia contributors, "Fuchsia (operating system)," *Wikipedia, The Free Encyclopedia*, [Online]. Available: `https://en.wikipedia.org/wiki/Fuchsia_(operating_system)`. [Accessed: Sep. 2, 2025].

[6] Amazon Web Services, "GetSessionToken - AWS Security Token Service," *AWS Documentation*, [Online]. Available: `https://docs.aws.amazon.com/STS/latest/APIReference/API_GetSessionToken.html`. [Accessed: Sep. 2, 2025].

[7] Google Cloud, "Configure managed workload identity authentication," *Google Cloud IAM Documentation*, [Online]. Available: `https://cloud.google.com/iam/docs/create-managed-workload-identities?authuser=1`. [Accessed: Sep. 2, 2025].

[8] Amazon Web Services, "IAM Roles - AWS Identity and Access Management," *AWS Documentation*, [Online]. Available: `https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html`. [Accessed: Sep. 2, 2025].

[9] SPIFFE Project, "SPIRE," *spiffe.io*, [Online]. Available: `https://spiffe.io/spire/`. [Accessed: Sep. 2, 2025].

[10] SPIFFE Project, "SPIFFE – Secure Production Identity Framework for Everyone," *spiffe.io*, [Online]. Available: `https://spiffe.io`. [Accessed: Sep. 2, 2025].

[11] S. Ramaswamy, T. Slepnev, and P. Allen, "Guide to Secure Web Services," *NIST Special Publication*

800-95, Aug. 2007. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-95.pdf.

[12] Michael B. Jones and John Bradley and Nat Sakimura, "JSON Web Token (JWT)," *Request for Comments 7519*, May. 2015. [Online]. Available: https://www.rfc-editor.org/info/rfc7519.

[13] International Telecommunication Union, "Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks (X.509)," *Recommendation X.509*, Oct. 2019. [Online]. Available: https://www.itu.int/rec/T-REC-X.509/en.

[14] Microsoft, "What is managed identities for Azure resources?," *Microsoft Documentation*, [Online]. Available: https://learn.microsoft.com/en-us/entra/identity/managed-identities-azure-resources/overview.

[15] Microsoft, "What is Azure attribute-based access control (Azure ABAC)?," *Microsoft Documentation*, [Online]. Available: https://learn.microsoft.com/en-us/azure/role-based-access-control/conditions-overview.

[16] Microsoft, "Trusted Execution Environment (TEE)," *Microsoft Documentation*, [Online]. Available: https://learn.microsoft.com/en-us/azure/confidential-computing/trusted-execution-environment.

[17] Microsoft, "Grant limited access to Azure Storage resources using shared access signatures (SAS)," *Microsoft Documentation*, [Online]. Available: https://learn.microsoft.com/en-us/azure/storage/common/storage-sas-overview.

[18] Microsoft, "What is Microsoft Entra?," *Microsoft Documentation*, [Online]. Available: https://learn.microsoft.com/en-us/entra/fundamentals/what-is-entra.

[19] Microsoft, "What is Azure SQL?," *Microsoft Documentation*, [Online]. Available: https://learn.microsoft.com/en-us/azure/azure-sql/azure-sql-iaas-vs-paas-what-is-overview?view=azuresql.

[20] Microsoft, "Introduction to Azure Storage," *Microsoft Documentation*, [Online]. Available: https://learn.microsoft.com/en-us/azure/storage/common/storage-introduction.

[21] Microsoft, "Virtual machines in Azure," *Microsoft Documentation*, [Online]. Available: https://learn.microsoft.com/en-us/azure/virtual-machines/overview.

[22] Microsoft, "Service Fabric terminology overview," *Microsoft Documentation*, [Online]. Available: https://learn.microsoft.com/en-us/azure/service-fabric/service-fabric-technical-overview.

Optimal Group Privacy for DP-SGD

Saeed Mahloujifar* Alexandre Sablayrolles† Graham Cormode‡ Somesh Jha§

Abstract

One challenging problem with differentially private machine learning is *privacy accounting*. After years of research, the community has successfully established tight privacy accounting methods for differentially private stochastic gradient descent (DP-SGD). Despite these advances, tight bounds for *group privacy* still remain elusive. Group privacy is an essential aspect of differential privacy that enables many applications. In this work, we develop tight bounds on group privacy for DP-SGD. In this work, we develop tight bounds on group privacy for DP-SGD. Our analysis uses a novel technique to show "dominating pairs of distributions" explicitly tailored for the case of group privacy. Our experiments show that our bounds are significantly better than previously known bounds in certain regimes. Surprisingly, we find that group privacy is significantly affected by sub-sampling. Two sets of hyper-parameters (sampling rate and noise) with the exact same (ϵ, δ) parameters can have significantly different group privacy curves.

1 Introduction

The Differentially Private Stochastic Gradient Descent (DP-SGD) algorithm [1, 25] is the leading method for training machine learning models and conducting a variety of optimization tasks with privacy guarantees. A critical facet that enables DP-SGD for privacy is the notion of (tight) privacy accounting. Privacy accounting addresses a fundamental question: What is the extent of privacy degradation when a differential privacy-protected task is performed repetitively? Since the inception of differential privacy, this question has been studied via "composition theorems" [8, 14]. Subsequent work has focused on tighter composition for privacy, specifically within the framework of DP-SGD. Building upon the foundation laid by Abadi et al. [1], furthered by the introduction of Rényi differential privacy (RDP) by Mironov [20], and enhanced by recent research on the precise analysis of Gaussian differential privacy [7, 11, 29, 32], we can now compute the privacy assurances of DP-SGD with high precision. This analysis has allowed acceptable utility levels for various tasks while simultaneously offering substantial privacy guarantees. Despite these advances, the domain of group privacy has not experienced equivalent progress. Group privacy poses a different question: how does the privacy guarantee deteriorate when the concern is about the impact on a *group* of examples, rather than individual ones? Intriguingly, we still lack better group privacy bounds than the rudimentary black-box bounds that were first introduced with differential privacy [8]. Thus, we ask: Can DP-SGD attain group privacy bounds superior to the black-box bounds for any DP mechanism?

The question of group privacy is significant in various analytical contexts. For instance, group privacy can prove beneficial in scenarios such as federated learning, where a user might contribute multiple data points and we want a "user-level" privacy boundary [13]. Similarly, there might be situations where the

*FAIR at Meta, Corresponding author, saeedm@meta.com
†Mistral AI, work done at Meta
‡Meta
§University of Wisconsin-Madison

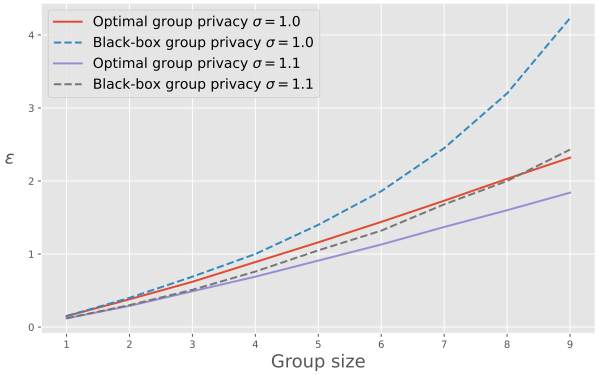


Figure 1: Group composition for 10 steps of DP-SGD with noise multiplier 1.0, sampling rate 0.01, and δ of 10^{-3}. The privacy curve obtained by our bounds is much smoother. We use a large value for δ as calculating the black box group privacy bound for small values of δ is computationally infeasible (unlike our bound).

same or very similar examples are repeated within collected datasets [15]. As demonstrated in recent studies on reconstruction attacks, these repeated points could be considerably more susceptible to privacy breaches [5]. Group privacy bounds enable us to understand the degree of increased vulnerability when examples are replicated. Another practical application of group privacy bounds arises in the context of auditing differential privacy [12, 23, 26]. This process involves injecting a number of examples into the training routine of a machine learning model using DP-SGD, with the aim of calculating a lower limit on privacy and comparing it against the (potentially loose) guaranteed bound. Attaining tighter constraints on group privacy thus helps narrow the gap between the guaranteed privacy levels and the lower limit, facilitating more precise auditing. Last, group privacy bounds contribute to robustness against poisoning attacks. It is well-known that DP affords some protection against manipulations in the training dataset [18]. Enhanced group privacy bounds serve to improve these protections. Hence, optimizing group privacy is not just a theoretical exercise but has profound implications for applications in ML and privacy.

Our contributions: In this work, we provide tight group privacy bounds for DP-SGD. Our bounds are based on our new "domination" theorem that shows the worst possible pair of distributions that might occur while running DP-SGD on two k-neighboring databases. We consider this "domination" result as our main technical contribution. Using the knowledge of worst-case distributions, we give a Monte-Carlo approach to estimate the differential privacy bounds. Our experiments show that our bounds can significantly outperform the previous (black-box) group privacy bounds. We also find a surprising relation between sampling rate and group privacy: in a nutshell, as the sub-sampling rate becomes smaller, the group privacy improves.

2 Preliminaries

We first define a notion of proximity for group privacy.

Definition 2.1 (k-neighboring) *A pair of datasets (D, D') are k-neighboring iff either (D', D) are k-neighboring or $|D \setminus D'| \leq k$ and $D' \setminus D = \emptyset$. We use $D \approx_k D'$ to denote that D and D' are k-neighboring. Note this is symmetric, i.e., $D \approx_k D' \iff D' \approx_k D$.*

Definition 2.2: A mechanism M is (ε, δ, k)-DP if for all k-neighboring datasets D and D' we have

$$\forall S; \qquad \Pr[M(D) \in S] \leq e^{\varepsilon} \Pr[M(D')] + \delta.$$

We also use a more fine-grained notion of privacy, f-DP.

Definition 2.3 (f-DP [7]) *A mechanism M is (f, k)-DP if for all datasets D, D' s.t. $D \approx_k D'$:*

$$\forall S; \qquad \Pr[M(D) \in S] \leq 1 - f(\Pr[M(D') \in S]).$$

Now we state the basic group privacy introduced by Dwork et al. [8]. We give a slightly improved version of the bound, with an almost identical proof to that of [8].

Theorem 2.1 (Black-box group privacy [8]) *If a mechanism is $(\epsilon, \delta, 1)$-DP, then it is also $(k\epsilon, k \cdot \frac{e^{k\epsilon}-1}{e^{\epsilon}-1} \cdot \delta, k)$-DP.*

Proof: We prove this by induction. For $k = 1$, the statement is trivial. Now assume the statement is correct for $k - 1$. Assume $D' = D \cup \{x_1, \ldots, x_k\}$. Let $D'' = D \cup \{x_1\}$. Now since (D, D'') are 1-neighboring and (D', D'') are $(k-1)$-neighboring. Therefore, by the fact that M is (ϵ, δ)-DP we have

$$\Pr[M(D) \in S] \leq e^{\epsilon} \Pr[M(D'') \in S] + \delta.$$

Also, by the induction hypothesis we have

$$\Pr[M(D'') \in S] \leq e^{(k-1)\epsilon} \cdot \Pr[M(D') \in S] + \frac{e^{(k-1)\epsilon} - 1}{e^{\epsilon} - 1}\delta.$$

Combining these two inequalities, we have

$$\Pr[M(D) \in S] \leq e^{(k-1)\epsilon+\epsilon} \Pr[M(D') \in S] + \left(\frac{e^{k\epsilon} - e^{\epsilon}}{e^{\epsilon} - 1} + 1\right)\delta = e^{k\epsilon} \Pr[M(D') \in S] + \frac{e^{k\epsilon} - 1}{e^{\epsilon} - 1}\delta.$$

And, this finishes the proof.

This group-privacy bound is tight when employed as a black-box. In other words, there exists a mechanism M that is (ϵ, δ)-DP which enjoys the exact same group privacy bound as stated in the theorem.

We also note that there is a previously-known black-box group privacy bound for f-DP.

Theorem 2.2 (Black-box group privacy for f-DP [7]) *If a mechanism M is $(f, 1)$-DP, then for all $k \in \mathcal{N}$ it is also (f_k, k)-DP where*

$$f_k(x) = 1 - (1 - f)^k(x).$$

One would expect that the group privacy based on f-DP to be much tighter than the black-box variant for DP. This is simply because knowing that a mechanism is f-DP contains much more information than knowing a mechanism is (ϵ, δ)-DP. However, as we will see in the later sections, this bound is also sub-optimal when sub-sampling is employed. We reiterate that this bound is tight if the only information that we have is the knowledge that the mechanism is f-DP. But when dealing with specific algorithms (e.g., DP-SGD), we can use more information about the inner dynamics of the mechanism to achieve better bounds. In this work, we try to improve these black-box bounds for a specific class of mechanisms: the adaptive composition of multiple sub-sampled Gaussian mechanisms.

DP-SGD and Composition of sub-sampled Gaussian mechanisms. A Gaussian mechanism is simply used to privately calculate the average of a function h over a dataset. By releasing the noisy average $\frac{1}{|D|}\sum_{x \in D} h(x) + \mathcal{N}(0, \sigma^2)$, one would guarantee that the reported average satisfies differential

Algorithm 1 DP-SGD

Require: Private dataset D, Loss function ℓ, Sampling rate p, number of steps n, noise multiplier σ, clipping threshold c

Initialize a model θ.
for $i \leftarrow [1, \ldots, m]$ **do**
  Sample a random batch $B \subseteq D$ using Poisson sampling with probability of p.
  $g = \frac{1}{p \cdot |D|} \sum_{x \in B} \frac{\Delta\ell(\theta, x)}{\max\left(c, \|\Delta\ell(\theta, x)\|\right)}$
  $\tilde{g} = g + c \cdot \mathcal{N}(0, \sigma^2)$
  $\theta = \textbf{update}(\theta, \tilde{g})$
end for
Output θ.

privacy so long as the function h has a bounded range. To further amplify privacy, we can sample a random batch $B \subset D$ using Poisson sampling and report the noisy average (noisy average should be calculated without using the actual size of the batch; see Algorithm 1) over the batch. Then, one can compose many of these mechanisms in an adaptive way and preserve differential privacy, thanks to composition theorems. DP-SGD (Algorithm 1) [1, 25] is the most tangible instantiation of this category of mechanisms and is used for many privacy preserving applications where we need to perform optimization.

Our goal is to analyze the group privacy for DP-SGD. We need several definitions for the analysis. The first is the notion of weighted total variation distance.

Definition 2.4 (Weighted total variation distance) *The weighted variation distance between two distributions X and Y with densities μ and ν for a weight $a > 0$ is:*

$$\mathbf{TV}_a(X, Y) = \int |\mu(x) - a \cdot \nu(x)| dx.$$

We note that this notion is closely related to that of the Hockey-stick divergence [24] and trade-off functions [7]. Hockey-stick divergence is the same integration with a difference that the integral is only taken over the positive values. We prefer weighted TVD because it is more convenient to avoid conditioning the integration. The next claim shows the relevance of the weighted total variation distance in the DP context.

Proposition 2.2: Let (X, Y) be a pair of random variables. Then for all S and $\epsilon > 0$ we have,

$$\Pr[X \in S] \leq e^{\epsilon} \Pr[Y \in S] + \tfrac{1}{2}(\mathbf{TV}_{e^{\epsilon}}(X, Y) + 1 - e^{\epsilon}).$$

Proof: Let S be an arbitrary set. Let μ and ν be the pdf of X, Y respectively. Let $G_\epsilon = \{x; \mu(x) - e^{\epsilon}\nu(x) \geq 0.\}$ and $\bar{G}_\epsilon = \{x; \mu(x) - e^{\epsilon}\nu(x) < 0.\}$

$$\Pr[X \in S] - e^{\epsilon} \Pr[Y \in S] = \int_S \mu(x) - e^{\epsilon}\nu(x) \leq \int_{G_\epsilon} \mu(x) - e^{\epsilon}\nu(x).$$

Let $\delta = \int_{G_\epsilon} \mu(x) - e^{\epsilon}\nu(x)$, we have $\int_{\bar{G}_\epsilon} \mu(x) - e^{\epsilon}\nu(x) = 1 - e^{\epsilon} - \delta$. We also have $\mathbf{TV}_a(X, Y) = \int_{G_\epsilon} \mu(x) - e^{\epsilon}\nu(x) - \int_{\bar{G}_\epsilon} \mu(x) - e^{\epsilon}\nu(x)$. Therefore $\mathbf{TV}_a(X, Y) = 2\delta - 1 + e^{\epsilon}$. Therefore, $\delta = \frac{\mathbf{TV}_a(X,Y) + 1 - e^{\epsilon}}{2}$.

Note that the above proposition is only stated in one direction. However, differential privacy requires the upper bound to hold in both directions. Due to an interesting property of weighted total variation distance, we can also bound the reverse direction without changing the order of distributions in $\mathbf{TV}(X, Y)$.

Corollary 2.2.1: Let (X, Y) be a pair of random variables. Then for all S and $\epsilon > 0$ we have,

$$\Pr[Y \in S] \leq e^{\epsilon} \Pr[X \in S] + \tfrac{1}{2}(e^{\epsilon}\mathbf{TV}_{e^{-\epsilon}}(X, Y) + 1 - e^{\epsilon}).$$

Proof: Observe that $\mathbf{TV}_a(Y, X) = a\mathbf{TV}_{\frac{1}{a}}(X, Y)$. Now we use Proposition 2.2 with X and Y swapped, and apply this observation to finish the proof.

In Section 4, where we explain how to calculate the optimal bounds, we will see why we are interested in preserving the order of pairs. We next define *dominating pairs* of distributions, specific to the case of group-privacy. This notion enables us to use a pair of distributions for privacy accounting and removes the complexity of the choice of dataset.

Definition 2.5 (k-Dominating pair of distributions) *A pair of distributions (X, Y)* dominates *a mechanism M if for any pair of k-neighboring datasets $D \approx_k D'$ with $|D| < |D'|$ and any $a > 0$ we have*

$$\mathbf{TV}_a(X, Y) \geq \mathbf{TV}_a(M(D), M(D')).$$

We say that (X, Y) tightly dominate *M if there are k-neighboring datasets (D, D') such that $M(D) \equiv X$ and $M(D') \equiv Y$.*

Note that domination is defined in an asymmetric way. That is, we fix the order of datasets so that D has fewer data points than D'. The following proposition shows the usefulness of dominating pairs for privacy analysis of a mechanism. This proposition directly follows by applying Proposition 2.2 and Corollary 2.2.1.

Proposition 2.2: A mechanism that is k dominated by (X, Y) is (ϵ, δ, k)-DP for

$$\delta = \tfrac{1}{2}(\max\left(\mathbf{TV}_{e^{\epsilon}}(X, Y), e^{\epsilon}\mathbf{TV}_{e^{-\epsilon}}(X, Y)\right) + 1 - e^{\epsilon})$$

Finally, we state the following lemma that shows we can obtain a dominating pair of distributions for the composition of multiple mechanisms. This has been proved for the case of $k = 1$ in previous work [7, 19, 32]. Here we omit the proof as it is exactly the same.

Lemma 2.2: If a series of mechanisms $M_1, \ldots, M_n$ are k-dominated by pairs of distributions (X_1, Y_1), $\ldots$, (X_n, Y_n) then the adaptive composition of M_i's is k-dominated by

$$(X_1 \times \cdots \times X_n, Y_1 \times \cdots \times Y_n)$$

where $\times$ is the product operation between distributions.

3 Optimal group privacy bounds

Next, we demonstrate a tightly k-dominating pair of distribution for a single step of DP-SGD. Note that, by Lemma 2.2, this will give us a tight dominating pair for multiple steps of DP-SGD as well. We first define two notions that abstract two properties of Gaussian mechanism which we need to prove our result.

Definition 3.1 (Compatible distributions) *We call a triplet of distributions (X, Y, Z) with densities ν_X, ν_Y, and ν_Z* compatible *iff there exists an increasing and continuous transition function g, with $g(0) = 0$ and $\lim_{t\to\infty} g(t) = \infty$ such that $\frac{\nu_Y(x)}{\nu_X(x)} \geq r$ if and only if we have $\frac{\nu_Z(x)}{\nu_X(x)} \geq g(r)$.*

Definition 3.2 (System of nice distributions) *Let $\mathcal{Y} = \{\mathcal{Y}_1, \ldots, \mathcal{Y}_k\}$ be a collection of sets of distributions and let X be a distribution. $(X, \mathcal{Y})$ form a system of nice distributions if the following conditions hold:*

1. *For each $\mathcal{Y}_i$ there exists $Y_i^* \in \mathcal{Y}_i$ such that*
$$\forall a > 0, \forall Y_i \in \mathcal{Y}_i; \mathbf{TV}_a(X, Y_i) \leq \mathbf{TV}_a(X, Y_i^*).$$
2. *for all $i < j \in [k]$ the triplet (X, Y_i^*, Y_j^*) is compatible.*

We now state a lemma that shows when we can get a dominating pair for a mixture of multiple mechanisms. For ease of readability, proofs for most claims in this Section are deferred to Appendix A.

Lemma 3.0: Let $\mathcal{Y} = \{\mathcal{Y}_1, \ldots, \mathcal{Y}_k\}$ be a collection of sets of distributions. Assume that $(X, \mathcal{Y})$ form a system of nice distributions. Let $p_1, \ldots, p_k \in [0, 1]$ with $p_1 + \ldots, p_k = 1$. Let $Y = p_1 \cdot Y_1 + \cdots + p_k \cdot Y_k$ be an arbitrary mixture of distributions with $Y_i \in \mathcal{Y}_i$. Let $Y^* = p_1 \cdot Y_1^* + \cdots + p_k \cdot Y_k^*$. We have
$$\mathbf{TV}_a(X, Y) \leq \mathbf{TV}_a(X, Y^*).$$

This Lemma, which is the key lemma for proving our result, helps us to reduce the complexity owing to the mixture distribution. By knowing the worst-case in each set of distributions, we can identify the worst-case for the mixture as well. Now we turn our attention to the specific case of the Gaussian mechanism and show how to use Lemma 3.0 to obtain the dominating pair. We first show an intuitive result that $\mathbf{TV}_a$ between pairs of isotropic Gaussians is an increasing function of the distance between them.

Proposition 3.0: Let $X \equiv \mathcal{N}(u_1, \sigma^2 \cdot I_d)$ and $Y \equiv \mathcal{N}(u_2, \sigma^2 \cdot I_d)$. Then, for any $a \in \mathbb{R}^+$, $\mathbf{TV}_a(X, Y)$ is only a function of $\|u_1 - u_2\|_2$ and σ^2. Moreover this function is monotonically increasing with respect to $\|u_1 - u_2\|_2$. That is, for any $a \geq \|u_1 - u_2\|_2$ we have

$$\mathbf{TV}_a(X, Y) \leq \mathbf{TV}_a(\mathcal{N}(0, \sigma^2), \mathcal{N}(a, \sigma^2)).$$

Now, we show that a triplet of isotropic Gaussians with collinear means are compatible.

Proposition 3.0: Let $\mu \in R^d$, $c \in R^+$ $X = \mathcal{N}(0^d, \sigma^2 \cdot I_d)$, $Y = (\mu, \sigma^2 \cdot I_d)$ and $Z = (c \cdot \mu, \sigma^2 \cdot I_d)$. Then (X, Y, Z) are compatible.

Finally, we show that collections of sets of isotropic Gaussians, where each set is restricted to have a mean within a ball, form a system of nice distributions.

Proposition 3.0: Let $X = \mathcal{N}(0^d, \sigma^2 \cdot I_d)$ be isotropic Gaussian centered at zero. Also, for $j \in [k]$ let $\mathcal{Y}j = \{\mathcal{N}(\mu, \sigma^2 \cdot I_d); \|\mu\| \leq r_j\}$ for some $r_j \in \mathbb{R}^+$ and let $\mathcal{Y} = \{\mathcal{Y}_1, \ldots, \mathcal{Y}_k\}$. Then $(X, \mathcal{Y})$ forms a nice system of distributions.

Finally, we put things together to prove the following:

Theorem 3.1 (Main result) *Let M be one step of DP-SGD with sub-sampling rate p and clipping threshold 1 and noise σ. Then M is k-dominated by (X, Y) where*
$$X = \mathcal{N}(0, \sigma^2) \quad \text{and} \quad Y = \mathcal{N}(B(k, p), \sigma^2)$$
where $B(k, p)$ is the binomial distribution.

Before proving this theorem we state the following corollary that shows how this extends to multiple steps.

Corollary 3.1.1: DP-SGD with T-steps, noise multiplier σ and sub-sampling rate p is (ϵ, δ, k)-DP, for an arbitrary $\epsilon \in [0,1]$, $k \in [N]$ with

$$\delta = \frac{1}{2} \max \Big(\mathbf{TV}_{e^\epsilon}(X, Y), e^\epsilon \mathbf{TV}_{e^{-\epsilon}}(X, Y) \Big) + 1 - e^\epsilon,$$
$$\text{and} \quad X = \mathcal{N}(0^T, \sigma^2) \quad \text{and} \quad Y = \mathcal{N}(B(k,p)^T, \sigma^2).$$

Proof: From Theorem 3.1 we know that $\mathcal{N}(0, \sigma^2), \mathcal{N}(B(k,p), \sigma^2)$ form a dominating pair for a single step of DP-SGD with aforementioned hyperparameters. Then using Lemma 2.2 we obtain that the pair (X, Y), for X and Y as stated in the theorem, form a k-dominating pair for all T steps of DP-SGD. Therefore, using Proposition 2.2 we finish the proof.

Proof:[Proof of Theorem 3.1] Let us fix D and $D' = D \cup \{x_1, \dots, x_k\}$. Assume we fix the randomness of sub-sampling on all points in D (e.g., assume all of examples in D are sampled). Then, conditioned on this sampling s, the distribution $M(D)|s$ is a Gaussian centered at some μ_0. On the other hand, $M(D')|s$ is a mixture of Gaussians where the center is determined by the choice of sub-sampling on $\{x_1, \dots, x_k\}$. So we can characterize $M(D')$ as a mixture of Gaussians with probability weights $p_{s'}$, and means $\mu_0 + \mu_{s'}$, and with standard deviations σ^2. That is, $M(D') = \sum_{s' \in \{0,1\}^k} p_{s'} \cdot \mathcal{N}(\mu_0 + \mu_{s'}, \sigma^2)$ (note that here the outer sum operator denotes the mixture of distributions). We are interested in upper-bounding $\mathbf{TV}_a(M(D)|s, M(D')|s)$. Since μ_0 is present in the mean of both $M(D)$ and $M(D')$ we can ignore it and upper bound $\mathbf{TV}_a(\mathcal{N}(0, \sigma^2), \sum_{s'} p_{s'} \mathcal{N}(\mu_{s'}, \sigma^2)$. We know that for all s', the norm $\|\mu_{s'}\|$ is bounded by $|s'|_1$ because the clipping threshold is 1. In other words, $\mathcal{N}(\mu_{s'}, \sigma^2) \in \mathcal{Y}_{|s'|_1}$ where $\mathcal{Y}_{|s'|_1}$ is defined as in Proposition 3.0. Now, using Proposition 3.0, we know that the $(X, \mathcal{Y})$ form a system of nice distributions. Therefore, by Lemma 3.0, and the fact that for each $\mathcal{Y}_i$, the distribution $Y_i^* = \mathcal{N}(i, \sigma^2)$ incurs the greatest $\mathbf{TV}_a(X, Y_i^*)$ (according to Proposition 3.0), we have

$$\mathbf{TV}_a(M(D)|s, M(D')|s) \leq \mathbf{TV}_a(\mathcal{N}(0, \sigma^2), \sum p_{s'} \mathcal{N}(|s'|_1, \sigma^2)).$$

Now observe that $\sum p_{s'} \mathcal{N}(|s'|_1, \sigma^2)$ is the same distribution as $\mathcal{N}(B(k,p), \sigma^2)$. This concludes the proof for a fixed choice of sub-sampling s. Finally, note that fixing the sub-sampling is without loss of generality because we have $\mathbf{TV}_a(p_1 X_1 + p_2 X_2, p_1 Y_1 + p_2 Y_2) \leq p_1 \mathbf{TV}_a(X_1, Y_1) + p_2 \mathbf{TV}_a(X_2, Y_2)$ for any X_1, X_2, Y_1 and Y_2.

Remark 1 (Tightness of our bound) *When we say our bound is tight, we mean that there are instantiations of DP-SGD that will exactly incur the same privacy loss as our theorem predicts. Namely, if one runs the auditing attacks to verify DP [23] for these instantiations, the difference between the empirical lower bounds and theoretical upper bounds should be negligible. We also note that the bound of Theorem 3.1 is only tight in the setting where we release all the intermediate steps of DP-SGD. We do not make any claims about the tightness of our bound when we only release the final model (i.e., the weighted sum of all the intermediate gradients) and leave this as an open question. In fact, to the best of our knowledge, it is not understood if the best existing analysis of DP-SGD (PRV accounting and MC accounting [11, 29]) achieve tight bounds even for groups of size 1, when we only release the final model.*

Comparison with group privacy through f-DP. The seminal work of Dong et al. [7] defines the notion of f-DP and its special case, GDP. A mechanism M is f-DP if for all neighboring datasets the trade-off function between $M(D)$ and $M(D')$ is greater than f on all points. This notion contains more information than DP (or RDP) as it embeds the entire privacy curve. The authors propose a simple group privacy bound; a mechanism that is f-DP, will be $1 - (1-f)^k$-DP for groups of size k, where $(1-f)^k$ denotes the k-fold composition of the function $1 - f$. The authors rightfully claim that this group privacy bound cannot be generally improved because it is tight for the pure Gaussian mechanism. However, there

are three main issues with using these group privacy bounds for the sub-sampled Gaussian mechanisms: (1) The bound is not necessarily tight for the case of sub-sampling. (2) Calculating the bounds needs estimation of the entire trade-off function (for extremely small values) which is computationally inefficient. (3) The estimation error of the trade-off function grows exponentially with the number of compositions. On the contrary, our domination result avoids all this issues and enables us to calculate tight group privacy bounds for DP-SGD. Although calculating the f-DP based group privacy bound is infeasible, we can still analytically show that our bound is better. The following proposition formalizes this statement.

Proposition 3.1 (f-DP group privacy) *Let f be the optimal trade-off function for T-steps of DP-SGD with noise multiplier $\sigma > 0$ and sampling rate $p < 1$. Then (ϵ, δ) parameters obtained by group privacy for groups of size $k > 1$ through black-box f-DP group privacy bounds of Theorem 2.2 is strictly worse than that of Theorem 3.1.*

But the crux of the proof lies in the fact that the trade-off function is always convex and the black-box bound will perform operations that involve $f(p \cdot X_1 + (1-p) \cdot X_2)$, while the white box bounds will leverage the knowledge of sub-sampling and achieves $p \cdot f(X_1) + (1-p) \cdot f(X_2)$. This effect will compound over multiple iterations and as long as the sub-sampling rate is below 1, and the group size is larger than one, our bound will be strictly better.

4 Calculating the bound using Monte Carlo approximation

In this section we describe our algorithm for calculating the bound of Corollary 3.1.1. Note that the bound described there does not have a closed form and we need to approximate it. Previous work has explored various ways to calculate these type of bound using Monte Carlo approximation [19, 29] and numerical methods [9, 11, 32]. However, we still need to devise a new method for calculating our bound because previous methods are not general enough to cover the calculation of our bounds out of the box. In this work, we focus on the Monte-Carlo methods for calculating our bound.

Recall that the bound of Corollary 3.1.1 shows how to calculate δ at a given ϵ and the formula involves calculating $\mathbf{TV}_a(X, Y)$ for a pair of distributions X, Y. The procedure for calculating this weighted total variation distance uses two key observations. This first observation is that the formula for calculating the weighted total variation distance can be converted into an expectation form as follows:

$$\begin{aligned}
\mathbf{TV}_a(X, Y) &= \int |\nu_X(x) - a\nu_Y(x)| dx \\
&= 2 \int \max\Big(\nu_X(x) - a\nu_Y(x), 0\Big) dx + a - 1 \\
&= 2\mathbb{E}_{x\sim X}\Big[\max\big(1 - a\frac{\nu_Y(x)}{\nu_X(x)}, 0\big)\Big] + a - 1
\end{aligned}$$

Our second observation is that we can efficiently sample points from X and we can also calculate the ratio between ν_Y and ν_X at any given point x. This is simply by calculating the ratio at each coordinate using the binomial weights and then multiplying all the ratios in different coordinates. Hence, we can use a simple Monte-Carlo approach to approximate this quantity. Algorithm 1 shows our procedure for Monte-Carlo approximation of the δ for a given ϵ.

In a nutshell, the algorithm samples m points $x_1, \ldots, x_m$ from X. Then it calculates the ratio $r_i = e^{\epsilon} \cdot \nu_Y(x_i)/\nu_X(x_i)$ for all x_i. We can do this in n steps by calculating the ratio for each dimension and then multiplying them. Calculating the ratio for a dimension takes time $O(k)$ because the distribution for each dimension is a mixture of k distributions. Note that the algorithm would calculate both

$\mathbf{TV}_{e^\epsilon}(X, Y)$ and $\mathbf{TV}_{e^{-\epsilon}}(X, Y)$ at the same run. This is because of Corollary 3.1.1 that requires both of these quantities to calculate the δ. The running time of this algorithm is $O(knm)$ and the accuracy of approximation δ improves with the number of samples. The following Proposition shows the dependence between the accuracy and number of samples.

Algorithm 2 Compute δ group privacy

Require: Sampling rate p, group size k, number of compositions n, noise multiplier σ, privacy parameter ϵ, number of samples for mean estimation m

function GAUSSIAN_PDF(x, σ)
  return $e^{-\frac{x^2}{2\sigma^2}}$
end function

function BINOM_MIXTURE_PDF$((x, k, \sigma, p))$
  $p \leftarrow 0$
  for $j \leftarrow 0$ to k **do**
    $p_j \leftarrow$ GAUSSIAN_PDF$(x - j, \sigma)$
    $p \leftarrow p + p_j \times$ BINOM_COEFFICIENT(j, k, p)
  end for
  return p
end function

$\delta_1 \leftarrow 0$
$\delta_2 \leftarrow 1 - e^\epsilon$
for $i \leftarrow 1$ to m **do**
  $r_1 \leftarrow e^\epsilon$
  $r_2 \leftarrow e^{-\epsilon}$
  for $j \leftarrow 1$ to n **do**
    $x \sim \mathcal{N}(0, \sigma^2)$
    $\nu \leftarrow$ BINOM_MIXTURE_PDF(x, k, σ, p)
    $\mu \leftarrow$ GAUSSIAN_PDF(x, σ)
    $r_1 \leftarrow r_1 \times \frac{\nu}{\mu}$
    $r_2 \leftarrow r_2 \times \frac{\nu}{\mu}$
  end for
  $\delta_1 \leftarrow \delta_1 + \frac{\max(1-r_1,0)}{m}$

  $\delta_2 \leftarrow \delta_2 + e^\epsilon \cdot \frac{\max(1-r_2,0)}{m}$
end for
output $\max(\delta_1, \delta_2)$

Proposition 4.0: Let M be the composition of n sub-sampled gaussian mechanisms with sampling rate p and noise multiplier σ. Let δ be the output of Algorithm 2 ran on these parameters, with m samples at a given ϵ. Then the mechanism M is $(\epsilon, \delta + \gamma, k)$-DP, with probability at least $e^{1-2e^{-2m\gamma^2}}$, where the probability is taken over the randomness of Algorithm 2.

Proof: Note that Algorithm 2 is essentially finding the mean of the random variable $\max\left(1 - a\frac{\nu_Y(x)}{\nu_X(x)}, 0\right)$ with m samples. This random variable is always between 0 and 1. Using a Chernoff-Hoefding bound, we conclude that the mean estimation has error more than $\gamma e^{-\epsilon}$ with probability at most $p_1 = 2e^{-2me^{-2\epsilon}\gamma^2}$.

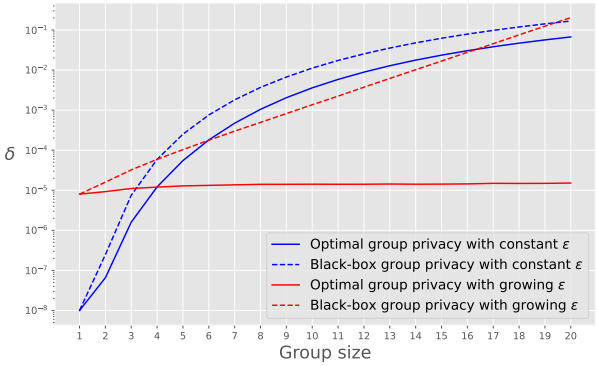


Figure 2: Group composition for 20 steps of DP-SGD with noise multiplier 1.0, sampling rate 0.01. In two of experiments we fix $\epsilon = 2.0$ and calculate the δ. In the other two experiments we grow ϵ linearly and set it to (group size) $\times$ 0.5. Surprisingly, in these experiments δ does not increase much, despite the exponential dependence of δ on the group size in the black box bound.

Therefore, the error of calculating δ_2 is at most γ with probability at least $1-p_1$. The error of calculating δ_1 is also at most γ with probability at least $1 - 2e^{-2m\gamma^2}$. Therefore, the error of $\max(\delta_1, \delta)$ is at most γ with probability at least $1 - 2e^{-2me^{-2\epsilon}\gamma^2} - 2e^{-2m\gamma^2}$. We remark that it is possible to approximate δ with more advanced Monte-Carlo techniques run with fewer samples, similar to what is done in [29]. However, for the purpose of this work, simple Monte-Carlo suffices. One might wonder if we can perform numerical accounting instead of Monte-Carlo. We currently believe that this would be possible for group privacy but it requires a non-black-box view of our proof. We leave this as an interesting open question.

5 Experiments

In this section we describe our experimental setup. We use a range of hyperparameters and group sizes to calculate the group privacy bound. We calculate the group privacy using Algorithm 2. In all experiments, we set m (number of trials) large enough so that with probability at least .99 the error of estimate is at most 1%. Since Algorithm 2 is designed to calculate δ at a given ϵ, we sometimes need to perform a search over ϵ that would give us the desired δ. For the experiments that require calculating ϵ at a given δ, we perform binary search to find the right ϵ.

Comparing with the black-box bound: First we compare our bound with the black-box bound of Theorem 2.1. In this experiment, we fix δ and aim at achieving $(\epsilon, \delta = 10^{-3}, k)$ group privacy for different groups sizes k. We use 10 steps of sub-sampled Gaussian mechanism with sampling probability $p = 0.01$ and noise multiplier $\sigma = 1.0$. Figure 1 shows that our bound can significantly outperform the black-box bound. The growth in ϵ with group size is much closer to linear than what the black-box bound predicts.

Note that for the black-box group privacy, the δ term grows exponentially with the group size. This means we need to calculate the ϵ for a very small value of δ' to be able to get $\delta < 10^{-5}$ after applying the group privacy bound. That is why in Figure 1, we chose $\delta = 10^{-3}$. In contrast, for our bounds there is no such issue and we can calculate ϵ for small values of δ. To illustrate this, we perform another experiment where we fix ϵ and show the growth of δ with ϵ. For this experiment, we use 20 steps of Gaussian mechanism with noise multiplier $\sigma = 2.0$ and sampling rate $p = 0.01$.

The results in Figure 2 (Blue curves) shows that the δ term grows really fast when we grow the group size, both for our bound and the black-box bound. To show the significance of the improvement

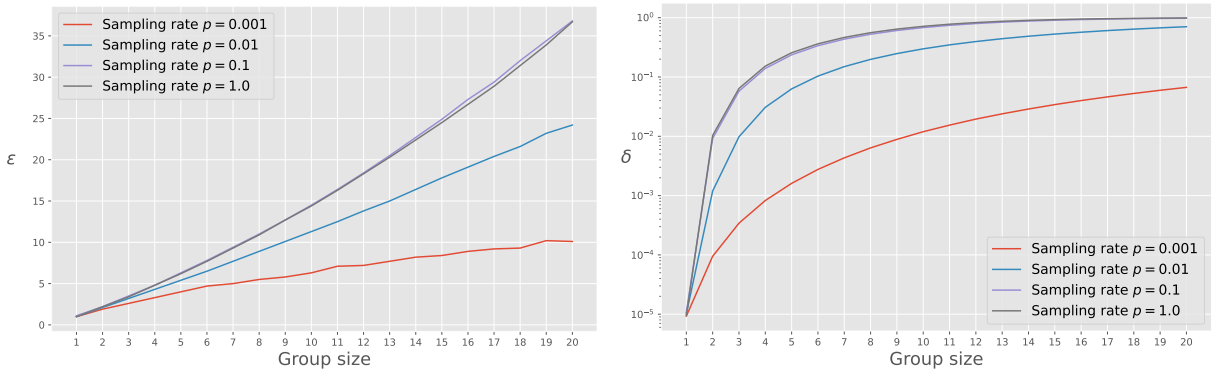


Figure 3: We change sampling rate and corresponding σ to ensure $(1.0, 10^{-5})$-DP when applied for 100 steps. The group privacy is much more graceful for smaller sampling.

in δ, we plot another curve, where we grow ϵ together with group in a linear way. Specifically, we set $\epsilon = k \times 0.5$ where k is the group size. The red curves in Figure 1 shows the comparison of our bound with the black box in this setting. As expected, in the black-box setting the δ still grows exponentially with linearly growing ϵ but the δ calculated using our bound is almost constant.

This experiment suggests a nice approximation of group privacy for sub-sampled Gaussian mechanism with very small sub-sampling rates. In particular, the ϵ seem to grow almost linearly with the group size while keeping the δ constant. Although this does not constitute a privacy guarantee, it is still a good rule of thumb.

Role of sub-sampling rate and step size on group privacy: To better understand the role of sub-sampling on group privacy, we perform experiments by simultaneously varying the sub-sampling rate and noise multiplier so that the (ϵ, δ) terms remain constant $(1.0, 10^{-5})$ (for groups of size 1). Then we calculate the optimal group privacy for each of these settings and compare the curves. Figures 3 show the resulting ϵ and δ terms respectively. In these experiments, we fix the number of steps to 100 and vary the sub-sampling rate to be one of $[0.001, 0.01, 0.1, 1.0]$. Then we find the σ that will satisfy $(1.0, 10^{-5}, 1)$-DP for each sub-sampling value. We then grow the group size for each sub-sampling rate and report the δ at a fixed ϵ in the top figure. We also report ϵ at a fixed δ in the bottom figure.

Our results show that sub-sampling can significantly change the group privacy profile. Lower sub-sampling rates lead to a more graceful degradation of privacy due to group size. We find the effect of sub-sampling on group privacy quite surprising. Note that the black-box group privacy would predict the exact same privacy curve in all scenarios. This finding might suggest an argument for using smaller batch sizes when doing private optimization. Although previous work [6, 27] suggest that larger batch size is better for the trade-off between accuracy and privacy, that dynamic might change if one is interested in the privacy for larger groups.

We also provide further experiments in Appendix B to demonstrate the role of sub-sampling rate at larger scale. We vary the sub-sampling rate and number of steps and observe that the role of sub-sampling diminishes as the number of steps increases. We believe this is mainly because of the behavior of the dominating pairs of distribution for sub-sampled Gaussian mechanism in the limit. We know that these dominating pairs behave similar to the dominating pairs of Gaussian distributions as the number of steps increase [28].

6 Applications and Implications

In this section, we present some of the application and implications of our bounds. Further exploration of these applications is left for future work.

Unit of Privacy: A challenging limitation of DP is that it requires a unit of privacy. The notion of neighboring dataset determines the smallest unit that we would want to protect privacy for. The work of Brown et al. [4] identifies this as one of the main challenges in employing DP for training language models. They question whether we should use words, sentences, paragraphs, documents, or even users as units of privacy. A simple solution to this issue would be to use the smallest unit of privacy imaginable and then apply group privacy to obtain the privacy bounds for larger units. However, the black-box group privacy bounds extremely degrade the privacy parameters, leading to meaningless privacy guarantees for larger privacy units. Our work can change this view and help with selecting small units of privacy, e.g., words or sentences.

Our group privacy bounds are also helpful for calculating user-level privacy [10, 16, 30], in settings that data is collected from multiple users, in a potentially heterogeneous way. In fact, using our general framework of Lemma 3.0, one can obtain even tighter bounds for groups/users that have certain properties. For example, if the gradients in a group are all orthogonal (e.g., they are examples from different classes in a logistic regression setting), $\sqrt{B(k,p)}$ would replace $B(k,p)$ in Theorem 3.1.

Robustness and DP: A large body of work has focused on the study of connections between robustness and DP [17, 21, 31]. Using DP-SGD for training a machine learning model would prevent a training-time attacker (a.k.a. poisoning attack) from changing the behavior of the trained model significantly. The reason relies on the fact that DP would limit the influence of each individual example on the output model. Hence, an adversary who can change a small fraction of the training data will not be able to change the distribution of the trained model more than a certain amount, determined by DP parameters. The (certified) robustness of the final model is determined by applying group privacy bounds, for the groups sizes that are equal to the number of points the adversary can add to the training set. Our improved group privacy bounds can improve the provable consequences of using DP-SGD for certified robustness.

Privacy auditing: A challenge with differential privacy is that verifying its correct implementation is difficult. Recent work has focused on this issue of "privacy auditing" by leveraging attacks that would fail when differential privacy is correctly deployed [12, 22, 26]. Specifically, one would run a membership inference attack to assert the correct implementation of DP; if membership inference succeeds with more than certain probability, then the implementation must be incorrect. We believe our optimal group privacy bounds can add more options for privacy auditing. Instead of individual membership inference attacks, we can focus on group membership inference attacks to verify the correctness of DP implementation. For example, in the context of generative models, it is observed that repetition of single point in the training set can significantly increase its chances of getting regurgitated [5]. In light of our group privacy bounds, an adversary that can distinguish between a model that is trained with 10 copies of a single sample from another model that is not trained on that specific sample, can be used to audit privacy tightly.

Fairness, accuracy, and privacy: Finally, we believe our group privacy bounds can explain some of the observations made about the accuracy and fairness of predictions in private models [2, 3]. Our optimal bounds on group privacy would imply that small groups will have a smaller effect on the models behavior than what was previously thought. This could lead to models that are unfair to small sub-populations. This effect can be exacerbated with certain hyperparameters (e.g. subsampling rate) that will make group privacy stronger. This shows that the choice of hyperparameters should not be

only influenced by the accuracy-privacy trade-off. There could be different hyperparameters that lead to exact same privacy and accuracy while showing different fairness of the model.

References

[1] M. Abadi, A. Chu, I. Goodfellow, B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. In *CCS*, 2016.

[2] E. Bagdasaryan, O. Poursaeed, and V. Shmatikov. Differential privacy has disparate impact on model accuracy. *Advances in neural information processing systems*, 32, 2019.

[3] L. Berrada, S. De, J. H. Shen, J. Hayes, R. Stanforth, D. Stutz, P. Kohli, S. L. Smith, and B. Balle. Unlocking accuracy and fairness in differentially private image classification. *arXiv preprint arXiv:2308.10888*, 2023.

[4] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[5] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramer, B. Balle, D. Ippolito, and E. Wallace. Extracting training data from diffusion models. *arXiv preprint arXiv:2301.13188*, 2023.

[6] S. De, L. Berrada, J. Hayes, S. L. Smith, and B. Balle. Unlocking high-accuracy differentially private image classification through scale. *arXiv preprint arXiv:2204.13650*, 2022.

[7] J. Dong, A. Roth, and W. J. Su. Gaussian differential privacy. *arXiv preprint arXiv:1905.02383*, 2019.

[8] C. Dwork, G. N. Rothblum, and S. Vadhan. Boosting and differential privacy. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 51–60. IEEE, 2010.

[9] B. Ghazi, P. Kamath, R. Kumar, and P. Manurangsi. Faster privacy accounting via evolving discretization. In *International Conference on Machine Learning*, pages 7470–7483. PMLR, 2022.

[10] B. Ghazi, P. Kamath, R. Kumar, P. Manurangsi, R. Meka, and C. Zhang. User-level differential privacy with few examples per user. *arXiv preprint arXiv:2309.12500*, 2023.

[11] S. Gopi, Y. T. Lee, and L. Wutschitz. Numerical composition of differential privacy. *Advances in Neural Information Processing Systems*, 34:11631–11642, 2021.

[12] M. Jagielski, J. Ullman, and A. Oprea. Auditing differentially private machine learning: How private is private SGD? *arXiv preprint arXiv:2006.07709*, 2020.

[13] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[14] P. Kairouz, S. Oh, and P. Viswanath. The composition theorem for differential privacy. In *International conference on machine learning*, pages 1376–1385. PMLR, 2015.

[15] N. Kandpal, E. Wallace, and C. Raffel. Deduplicating training data mitigates privacy risks in language models. In *International Conference on Machine Learning*, pages 10697–10707. PMLR, 2022.

[16] D. Levy, Z. Sun, K. Amin, S. Kale, A. Kulesza, M. Mohri, and A. T. Suresh. Learning with user-level privacy. *Advances in Neural Information Processing Systems*, 34:12466–12479, 2021.

[17] S. Liu, A. C. Cullen, P. Montague, S. M. Erfani, and B. I. Rubinstein. Enhancing the antidote: improved pointwise certifications against poisoning attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8861–8869, 2023.

[18] Y. Ma, X. Zhu, and J. Hsu. Data poisoning against differentially-private learners: Attacks and defenses. *arXiv preprint arXiv:1903.09860*, 2019.

[19] S. Mahloujifar, A. Sablayrolles, G. Cormode, and S. Jha. Optimal membership inference bounds for adaptive composition of sampled gaussian mechanisms. *arXiv preprint arXiv:2204.06106*, 2022.

[20] I. Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275. IEEE, 2017.

[21] M. Naseri, J. Hayes, and E. De Cristofaro. Local and central differential privacy for robustness and privacy in federated learning. *arXiv preprint arXiv:2009.03561*, 2020.

[22] M. Nasr, J. Hayes, T. Steinke, B. Balle, F. Tramèr, M. Jagielski, N. Carlini, and A. Terzis. Tight auditing of differentially private machine learning. *arXiv preprint arXiv:2302.07956*, 2023.

[23] M. Nasr, S. Songi, A. Thakurta, N. Papemoti, and N. Carlin. Adversary instantiation: Lower bounds for differentially private machine learning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 866–882. IEEE, 2021.

[24] I. Sason and S. Verdú. f-divergence inequalities. *IEEE Transactions on Information Theory*, 62(11):5973–6006, 2016.

[25] S. Song, K. Chaudhuri, and A. D. Sarwate. Stochastic gradient descent with differentially private updates. In *2013 IEEE global conference on signal and information processing*, pages 245–248. IEEE, 2013.

[26] T. Steinke, M. Nasr, and M. Jagielski. Privacy auditing with one (1) training run. *arXiv preprint arXiv:2305.08846*, 2023.

[27] X. Tang, A. Panda, V. Sehwag, and P. Mittal. Differentially private image classification by learning priors from random processes. *arXiv preprint arXiv:2306.06076*, 2023.

[28] H. Wang, S. Gao, H. Zhang, M. Shen, and W. J. Su. Analytical composition of differential privacy via the edgeworth accountant. *arXiv preprint arXiv:2206.04236*, 2022.

[29] J. T. Wang, S. Mahloujifar, T. Wu, R. Jia, and P. Mittal. A randomized approach for tight privacy accounting. *arXiv e-prints*, pages arXiv–2304, 2023.

[30] K. Wei, J. Li, M. Ding, C. Ma, H. Su, B. Zhang, and H. V. Poor. User-level privacy-preserving federated learning: Analysis and performance optimization. *IEEE Transactions on Mobile Computing*, 21(9):3388–3401, 2021.

[31] C. Xie, Y. Long, P.-Y. Chen, and B. Li. Uncovering the connection between differential privacy and certified robustness of federated learning against poisoning attacks. *arXiv preprint arXiv:2209.04030*, 2022.

[32] Y. Zhu, J. Dong, and Y.-X. Wang. Optimal accounting of differential privacy via characteristic function. In *International Conference on Artificial Intelligence and Statistics*, pages 4782–4817. PMLR, 2022.

A Deferred proofs

A.1 Proof of Lemma 3.0

Proof: We know that (X, Y_1^*, Y_i^*) are compatible for all $i \in [k]$. Let g_i be the corresponding transition function (see Definition 3.1). Note that g_1 is the identity function. Let t be constants in $[0, 1]$ such that the following holds:

$$\sum_{i\in[k]} \frac{ap_i}{g_i(ap_1/t)} = 1.$$

Note that this t exists because $f(t) = \sum_{i\in[k]} \frac{ap_i}{g_i(ap_1/t)}$ is an increasing and continuous function in t with $\lim_{t\to\infty} = \infty$ and $\lim_{t\to 0} f(t) = 0$. Therefore, there should exist a value of t that makes $f(t) = 1$. Now define $t_i = \frac{ap_i}{g_i(ap_1/t)}$. We are going to use these values to break up the integration. We have,

$$\begin{aligned}
\mathbf{TV}_a(X, Y) &= \int |\nu_X(x) - a\nu_Y(x)|dx \\
&= \int |\nu_X(x) - a\big(\sum_{i\in[k]} p_i \cdot \nu_{Y_i}(x)\big)|dx \\
&= \int |(t_1 + \cdots + t_k) \cdot \nu_X(x) - a\big(\sum_{i\in[k]} p_i \cdot \nu_{Y_i}(x)\big)|dx \\
&\leq \int \Big(\sum_{i\in[k]} |t_i \cdot \nu_X(x) - ap_i \cdot \nu_{Y_i}(x)|\Big)dx \\
&= \sum_{i\in[k]} \int |t_i \cdot \nu_X(x) - ap_i \cdot \nu_{Y_i}(x)|dx \\
&= \sum_{i\in[k]} t_i \cdot \mathbf{TV}_{\frac{ap_i}{t_i}}(X, Y_i) \\
&= \sum_{i\in[k]} t_i \cdot \mathbf{TV}_{g_i(\frac{ap_1}{t})}(X, Y_i) \\
&\leq \sum_{i\in[k]} t_i \mathbf{TV}_{g_i(\frac{ap_1}{t})}(X, Y_i^*)
\end{aligned}$$

Now we have multiple integrations on the absolute values and we need to move back to a single integration. This is where the reason behind the choice of t_i becomes clear. Let

$$s_i(x) = \text{sign}(\nu_X(x) - g_i(ap_1/t)\nu_{Y_i^*}(x)).$$

Based on the definition of g_i in Definition 3.1 we have

$$\forall i \in [k], \forall x; s_i(x) = s_1(x).$$

Using this, based on the fact that $t_i > 0$ we can conclude that

$$\forall j, \forall x; \text{sign}\Big(\sum_{i\in[k]} t_i\big(\nu_X(x) - g_i(ap_1/t)\nu_{Y_i^*}(x)\big)\Big) = s_j(x). \tag{1}$$

Now continuing our calculation of $\mathbf{TV}_a$ we have

$$\begin{aligned}
\mathbf{TV}_a(X,Y) &\le \sum_{i\in[k]} t_i \cdot \mathbf{TV}_{g_i(\frac{ap_1}{t})}(X, Y_i^*) \\
&= \sum_{i\in[k]} \int t_i |\nu_X(x) - g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)| dx \\
&= \int \Big(\sum_{i\in[k]} t_i |\nu_X(x) - g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)| \Big) dx \\
&= \int \Big(\sum_{i\in[k]} t_i \cdot s_i(x) \cdot \big(\nu_X(x) - g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)\big) \Big) dx \\
&= \int \operatorname{sign}\Big(\sum_{i\in[k]} t_i \cdot \big(\nu_X(x) - g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)\big) \Big) \cdot \Big(\sum_{i\in[k]} t_i \cdot \big(\nu_X(x) - g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)\big) \Big) dx \\
&= \int \Big| \sum_{i\in[k]} t_i \cdot \big(\nu_X(x) - g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)\big) \Big| dx \\
&= \int \Big| \nu_X(x) - \sum_{i\in[k]} t_i \cdot g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)\big) \Big| dx \\
&= \int \Big| \nu_X(x) - \sum_{i\in[k]} ap_i \cdot \nu_{Y_i^*}(x)\big) \Big| dx \\
&= \mathbf{TV}_a(X, p_1 \cdot Y_1^* + \cdots + p_k \cdot Y_k^*).
\end{aligned}$$

And this finishes the proof.

A.2 Proof of Proposition 3.0

Proof: The first part follows by the symmetry of isotropic Gaussian. For the second part (monotonicity) we use the definition of $\mathbf{TV}_a$. Without loss of generality we can assume $a \in [0,1]$ as otherwise we can work with $\mathbf{TV}_a(P,Q)/a = \mathbf{TV}_{1/a}(Q,P)$. Let $r = \|u_1 - u_2\|_2$. We can show that the derivative of the integral is always positive. In the following calculations, we use c_1, c_2, c_3 and c_4 to denote positive constants that are independent of r.

First note that $x^* = \frac{r^2 - 2\sigma^2 \ln(a)}{2r}$ is a middle point where $e^{-\frac{x^2}{2\sigma^2}} - ae^{-\frac{(x-r)^2}{2\sigma^2}}$ goes from positive to negative as x increases. By our assumption that $a \in [0,1]$, we have that $x^* > 0$. Recalling that $\operatorname{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z \exp(-t^2) dt$, and that $\operatorname{erf}(\infty) = 1$ so that (by symmetry) $\frac{2}{\sqrt{\pi}} \int_{-\infty}^0 \exp(-t^2) dt = 1$, we can

write

$$\mathbf{TV}_a(P,Q) = c_1\left(\int_{-\infty}^{\infty}\left|e^{-\frac{x^2}{2\sigma^2}} - ae^{-\frac{(x-r)^2}{2\sigma^2}}\right|dx\right)$$

By breaking the integral in an intermediate point we have

$$= c_1\left(\int_{-\infty}^{x^*} e^{-\frac{x^2}{2\sigma^2}} - ae^{-\frac{(x-r)^2}{2\sigma^2}} + \int_{x^*}^{\infty} ae^{-\frac{(x-r)^2}{2\sigma^2}} - e^{-\frac{x^2}{2\sigma^2}}\right)$$

By replacing the integrals with the CDF of Gaussian distribution we have

$$\begin{aligned}
&= c_1\left(1+\operatorname{erf}\left(x^*/\sqrt{2}\sigma\right) - a\operatorname{erf}((x^*-r)/\sqrt{2}\sigma)\right)\\
&\quad + \left(a(1-\operatorname{erf}\left((x^*-r)/\sqrt{2}\sigma\right) + (1-\operatorname{erf}\left(x^*/\sqrt{2}\sigma\right)\right)\\
&= c_2\left(\operatorname{erf}\left(\frac{r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right) + 1 - a\operatorname{erf}\left(\frac{-r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right) - a\right).
\end{aligned}$$

Now, let $f_1(r) = \operatorname{erf}\left(\frac{r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)$ and $f_2(r) = -a\operatorname{erf}\left(\frac{-r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)$. Taking the derivative with respect to r we have

$$\frac{\partial f_1}{\partial r} = c_3\left(\frac{1}{2\sqrt{2}\sigma} + \frac{\ln(a)\sigma}{2\sqrt{2}r^2}\right)e^{-\left(\frac{r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)^2}$$

$$\frac{\partial f_2}{\partial r} = c_3 a\left(\frac{1}{2\sqrt{2}\sigma} - \frac{\ln(a)\sigma}{2\sqrt{2}r^2}\right)e^{-\left(\frac{-r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)^2}$$

Now note that we have $e^{-\left(\frac{r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)^2} = a^{1/2}\cdot e^{-\left(\frac{-r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)^2}$. Therefore, we have

$$c_4\frac{\partial \mathbf{TV}_a}{\partial r} = e^{-\left(\frac{-r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)^2}\cdot\left(\frac{1+\sqrt{a}}{2\sqrt{2}\sigma} + \frac{\ln(a)\left(\sqrt{a}-1\right)\sigma}{2\sqrt{2}r^2}\right).$$

Now since $a\in[0,1]$, we have $\ln(a)\le 0$ and $\sqrt{a}-1<0$, which means the term $\frac{1+\sqrt{a}}{2\sqrt{2}\sigma} + \frac{\ln(a)(\sqrt{a}-1)\sigma}{2\sqrt{2}r^2}$ is positive. This implies that the whole gradient is positive.

A.3 Proof of Proposition 3.0

Proof: We have

$$\nu_Y(x)/\nu_X(x) = e^{\frac{\|x\|^2-\|x-\mu\|^2}{2\sigma^2}} \quad \text{and} \quad \nu_Z(x)/\nu_X(x) = e^{\frac{\|x\|^2-\|x-c\cdot\mu\|^2}{2\sigma^2}}.$$

For all x we have

$$\begin{aligned}
\|x-c\cdot\mu\|^2 - \|x\|^2 = \sum_{i=1}^{d} c^2\mu_i^2 - 2c\mu_i\cdot x_i &= \Big(\sum_{i=1}^{d} c^2 - c\mu_i^2\Big) + c\Big(\sum \mu_i^2 - 2\mu_i\cdot x_i\Big)\\
&= (c^2-c)\|\mu\| + c(\|x-\mu\|^2 - \|x\|^2).
\end{aligned}$$

Therefore, if $\frac{\nu_Y(x)}{\nu_x(x)} > r$ then we have $\|x\|^2 - \|x-\mu\|^2 > 2\sigma^2\cdot\ln(r)$, which implies $\|x\|^2 - \|x-\mu\|^2 > 2c\cdot\sigma^2\cdot\ln(r) + (c-c^2)\|\mu\|$, which in turn implies $\frac{\nu_z(x)}{\nu(x)} > e^{c\cdot\ln(r)+(c-c^2)\|\mu\|^2}$. Now observe that $g(r) = e^{c\cdot\ln(r)+(c-c^2)\|\mu\|^2}$ is an increasing and continuous function of r. Also observe that all the steps we took are reversible, therefore we have $\nu_Y(x)/\nu_X(x) > r$ if and only if $\nu_Z(x)/\nu_X(x) > g(r)$.

A.4 Proof of Propostion 3.0

Proof: Let $\mu \in R^d$ be an arbitrary unit vector with $\|\mu\| = 1$. By the symmetry of Gaussians, we know that for any constant $c \in \mathbb{R}^+$ we have $\mathbf{TV}_a(\mathcal{N}(0^d, \sigma^2 \cdot I_d)$, Let $\mu \in R^d$ be an arbitrary unit vector with $\|\mu\| = 1$. Again by the symmetry of Gaussians, we know that for any constant $c \in \mathbb{R}^+$ we have $\mathbf{TV}_a(\mathcal{N}(0^d, \sigma^2 \cdot I_d)$, Let $\mu \in R^d$ be an arbitrary unit vector with $\|\mu\| = 1$. Let us define $Y_j^* = \mathcal{N}(r_j \cdot \mu, \sigma^2 \cdot I_d)$. By Lemma 3.0 we know that for all $Y_j \in \mathcal{Y}_j$ we have

$$\mathbf{TV}_a(X, Y_j) \leq \mathbf{TV}_a(X, Y_j^*).$$

On the other hand, by Proposition 3.0 we know that for all $j, j' \in [k]$, the triplet $(X, Y_j^*, Y_{j'}^*)$ are compatible. Hence, $(X, \mathcal{Y})$ form a nice system of distributions.

A.5 Proof of Proposition 3.1

Proof: Let $f_{\sigma,p}$ be the trade-off function associated with a single step of the sub-sampled Gaussian mechanism with noise σ. When $p = 1.0$ we simply write f_σ. We also define $\bar{f}(x) = 1 - f(x)$. We have,

$$\bar{f}_{p,\sigma}(\alpha) = (1-p) \cdot \alpha + p \cdot (\bar{f}_\sigma(\alpha))$$

Now consider applying this function twice, we have

$$\begin{aligned}\bar{f}_{p,\sigma}(\bar{f}_{p,\sigma}(\alpha)) &= (1-p)\bar{f}_{\sigma,p}(\alpha) + p\bar{f}_\sigma(\bar{f}_{p,\sigma}(\alpha)) \\ &= (1-p)^2(\alpha) + p(1-p)\bar{f}_\sigma(\alpha) + p\bar{f}_\sigma\big((1-p)\alpha + p\bar{f}_\sigma(\alpha)\big).\end{aligned}$$

We know that trade-off functions are convex, so using Jensen's inequality we have,

$$\begin{aligned}\bar{f}_{p,\sigma}(\bar{f}_{p,\sigma}(\alpha)) &\geq (1-p)^2(\alpha) + 2p(1-p)\bar{f}_\sigma(\alpha) + (1-p)^2\bar{f}_\sigma\big(\bar{f}_\sigma(\alpha)\big) &(2)\\ &= (1-p)^2\alpha + 2p(1-p)\bar{f}_\sigma(\alpha) + p^2\bar{f}_{\sigma/2}(\alpha) &(3)\end{aligned}$$

Now let $\mu \equiv \mathcal{N}(0, \sigma)$ and $\nu \equiv (1-p)^2\mathcal{N}(0, \sigma) + 2p(1-p)\mathcal{N}(1, \sigma) + p^2\mathcal{N}(2, \sigma)$. The trade-off function between μ and ν is equal to

$$1 - T(\mu, \nu)(\alpha) = (1-p)^2\alpha + 2p(1-p) \cdot f_\sigma(\alpha) + p^2 f_{\sigma/2}(\alpha),$$

which is equal to the right hand side of Equation 2. Using a simple induction, we can show that for all k, defining $\mu = \mathcal{N}(0, \sigma)$ and $\nu = \mathcal{N}(B(k, p), \sigma)$, we can show that the trade-off function between μ and ν is always dominated by the function $f_{\sigma,p}^k$. Also note that this domination is strict as long as $k > 1$ and $0 < p < 1$. This shows that for a single step of DP-SGD, our group privacy bound is strictly better than what is entailed by applying the trade-off function recursively. For more than one step, we can simply use Lemma 2.2 and show that our bound is strictly better for many steps of DP-SGD as well.

B Extra experiments

B.1 Growth of group privacy with step size

In this section, we demonstrate the growth of group privacy parameters with the step size. In each plot, we fix the sampling rate and noise parameters and calculate the group privacy at various step sizes. In general, we observe that the growth of group privacy is faster in the smaller iterations, but it becomes slower as the number of steps further increase.

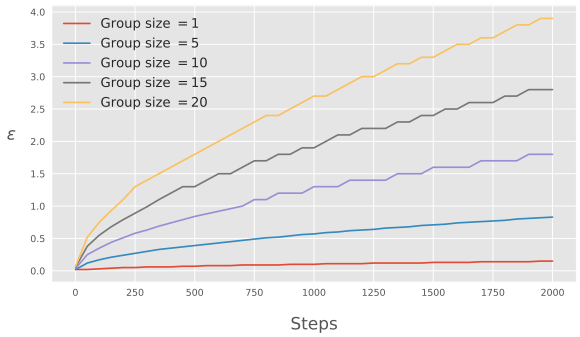


Figure 1: Noise Multiplier=10.00, Sampling rate=0.01.

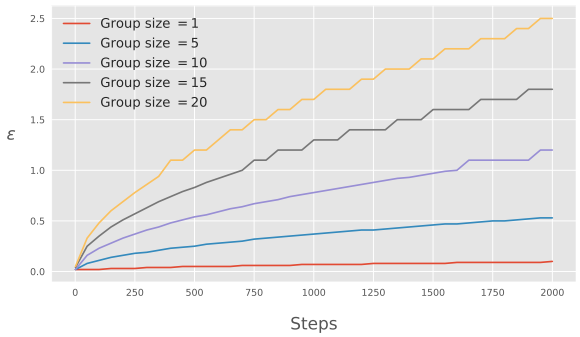


Figure 2: Noise Multiplier=15.00, Sampling rate=0.01.

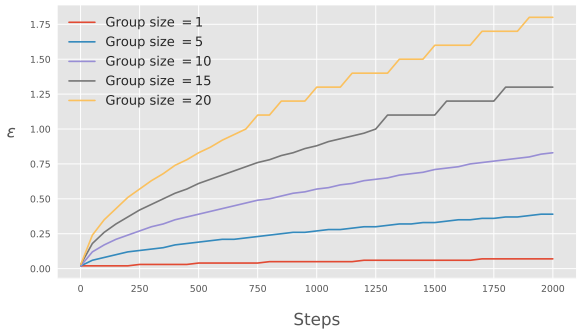


Figure 3: Noise Multiplier=20.00, Sampling rate=0.01.

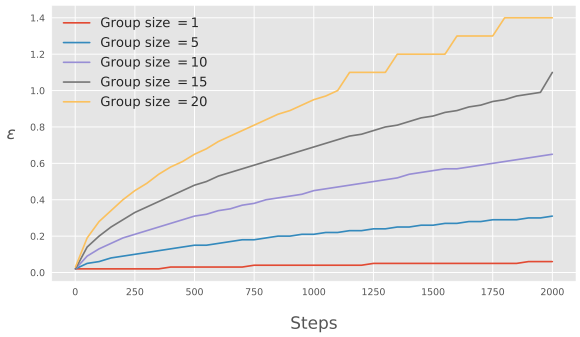


Figure 4: Noise Multiplier=25.00, Sampling rate=0.01.

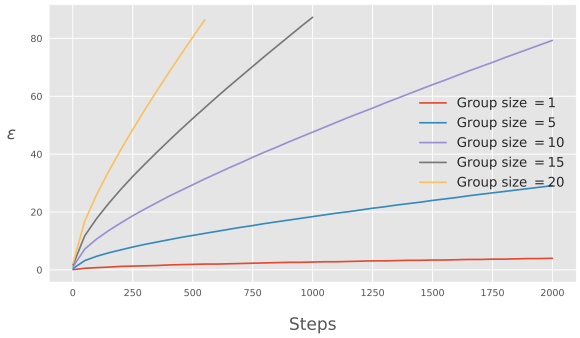


Figure 5: Noise Multiplier=5.00, Sampling rate=0.10.

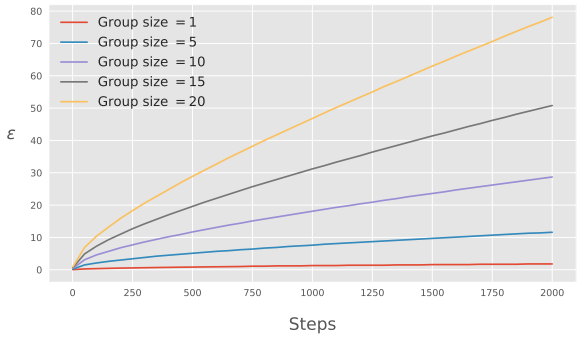


Figure 6: Noise Multiplier=10.00, Sampling rate=0.10.

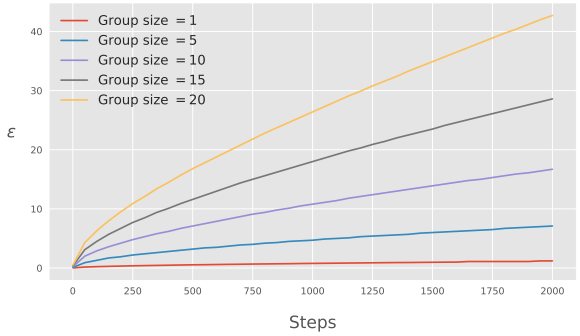


Figure 7: Noise Multiplier=15.00, Sampling rate=0.10.

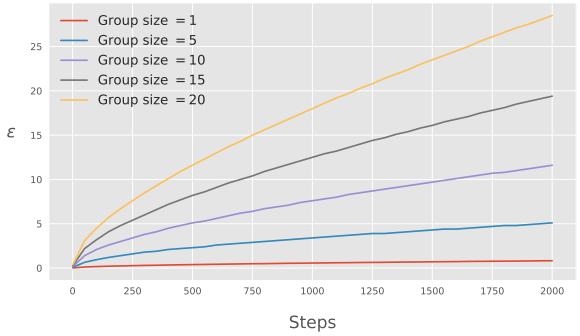


Figure 8: Noise Multiplier=20.00, Sampling rate=0.10.

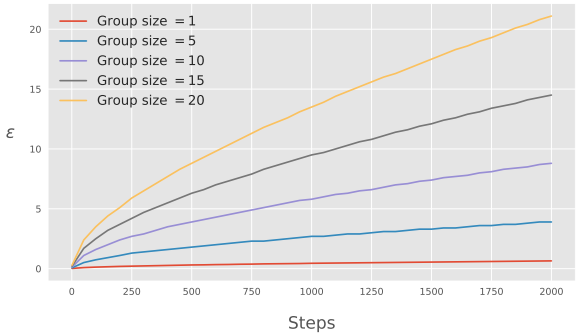


Figure 9: Noise Multiplier=25.00, Sampling rate=0.10.

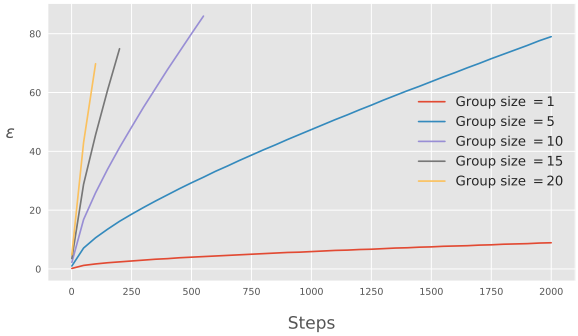


Figure 10: Noise Multiplier=5.00, Sampling rate=0.20.

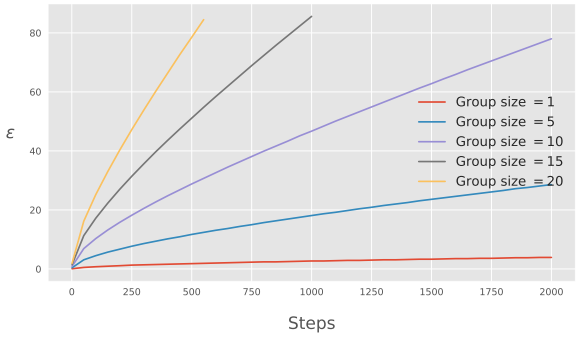


Figure 11: Noise Multiplier=10.00, Sampling rate=0.20.

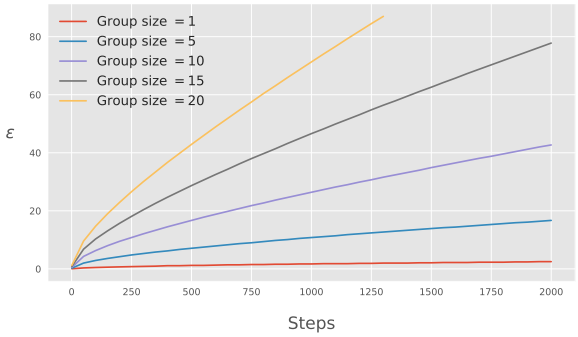


Figure 12: Noise Multiplier=15.00, Sampling rate=0.20.

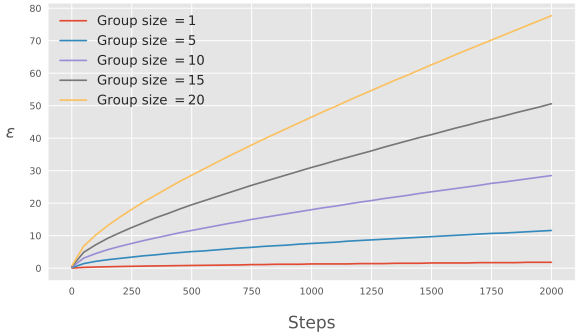


Figure 13: Noise Multiplier=20.00, Sampling rate=0.20.

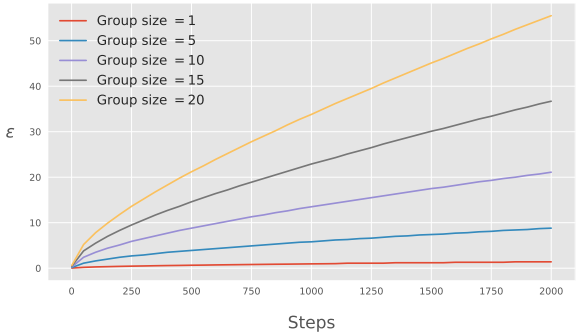


Figure 14: Noise Multiplier=25.00, Sampling rate=0.20.

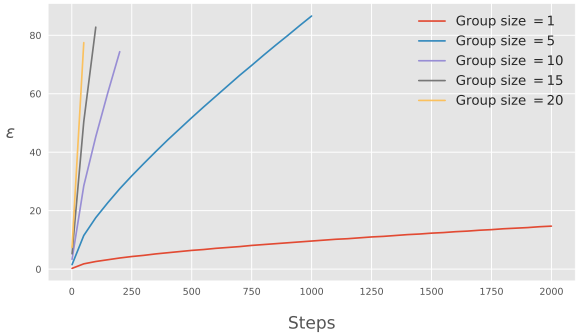


Figure 15: Noise Multiplier=5.00, Sampling rate=0.30.

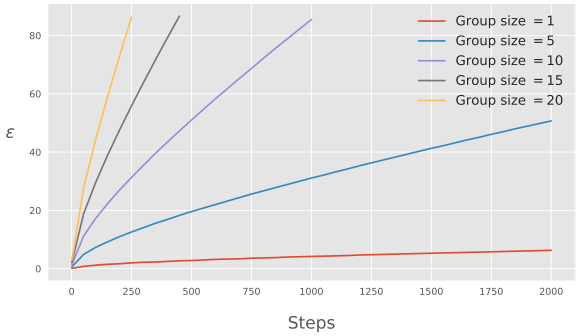


Figure 16: Noise Multiplier=10.00, Sampling rate=0.30.

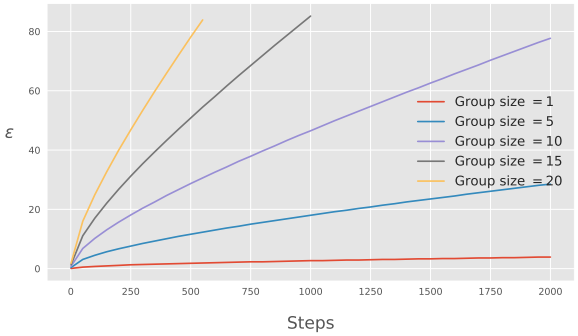


Figure 17: Noise Multiplier=15.00, Sampling rate=0.30.

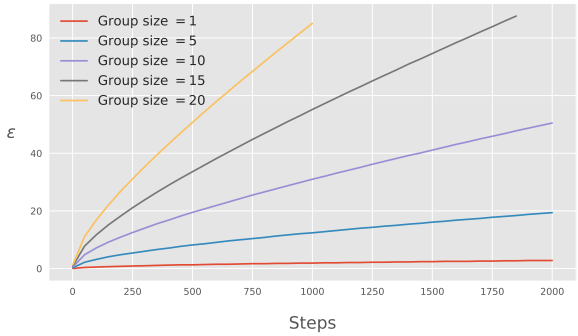


Figure 18: Noise Multiplier=20.00, Sampling rate=0.30.

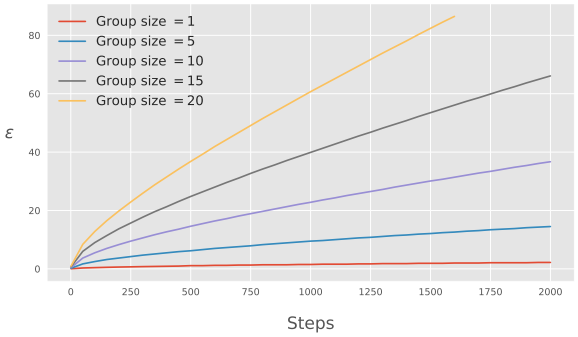


Figure 19: Noise Multiplier=25.00, Sampling rate=0.30.

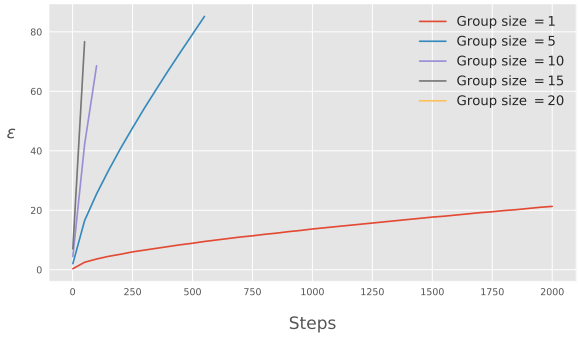


Figure 20: Noise Multiplier=5.00, Sampling rate=0.40.

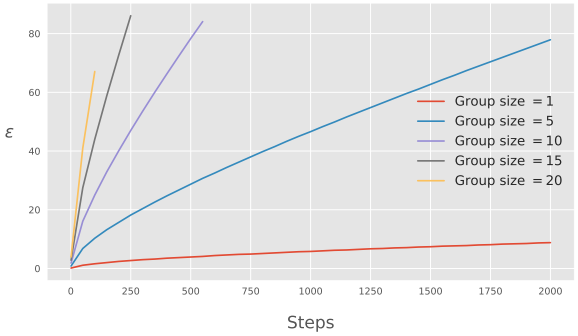


Figure 21: Noise Multiplier=10.00, Sampling rate=0.40.

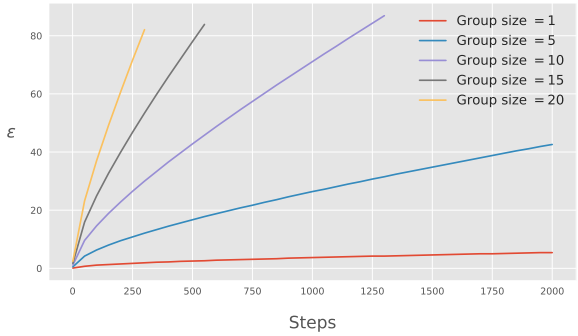


Figure 22: Noise Multiplier=15.00, Sampling rate=0.40.

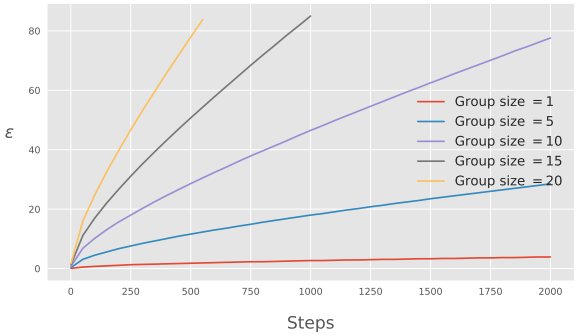


Figure 23: Noise Multiplier=20.00, Sampling rate=0.40.

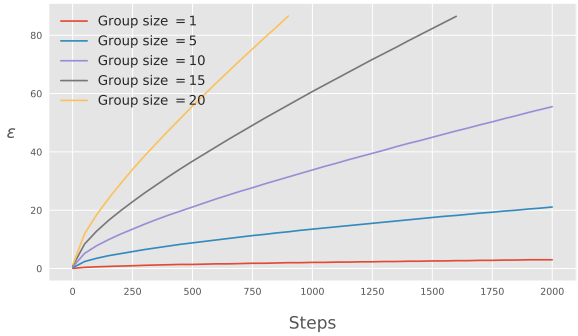


Figure 24: Noise Multiplier=25.00, Sampling rate=0.40.

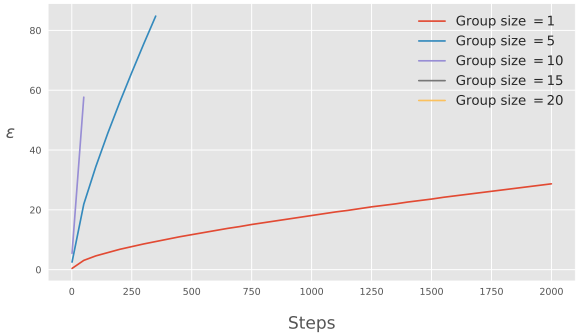


Figure 25: Noise Multiplier=5.00, Sampling rate=0.50.

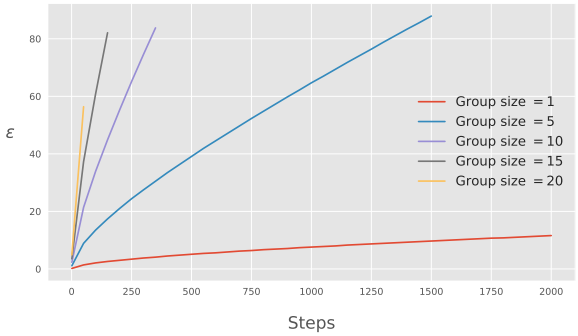


Figure 26: Noise Multiplier=10.00, Sampling rate=0.50.

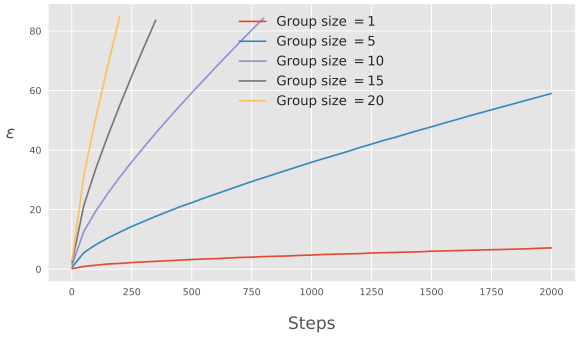


Figure 27: Noise Multiplier=15.00, Sampling rate=0.50.

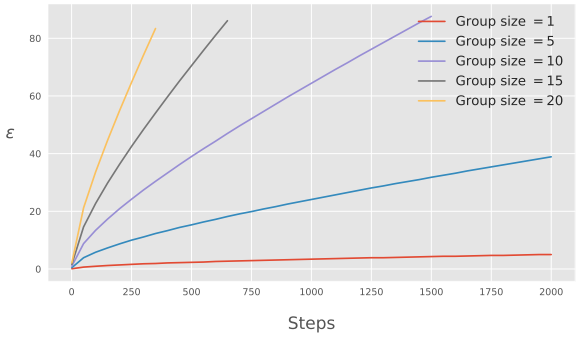


Figure 28: Noise Multiplier=20.00, Sampling rate=0.50.

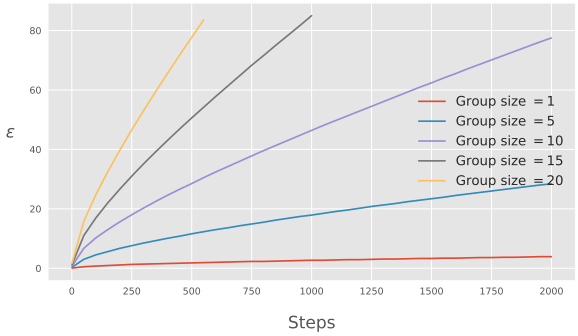


Figure 29: Noise Multiplier=25.00, Sampling rate=0.50.

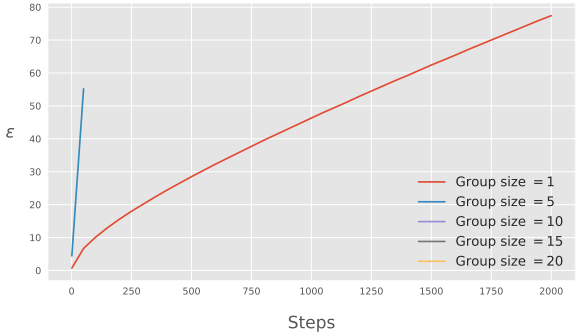


Figure 30: Noise Multiplier=5.00, Sampling rate=1.00.

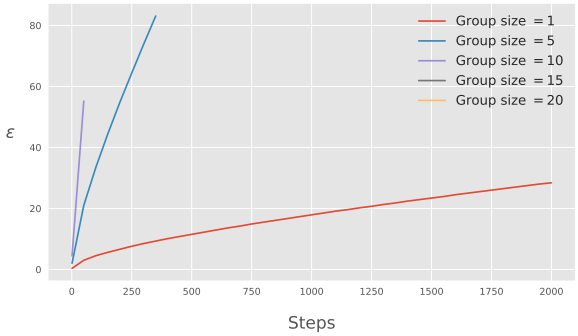


Figure 31: Noise Multiplier=10.00, Sampling rate=1.00.

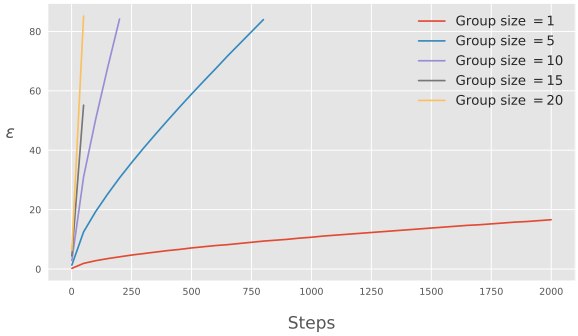


Figure 32: Noise Multiplier=15.00, Sampling rate=1.00.

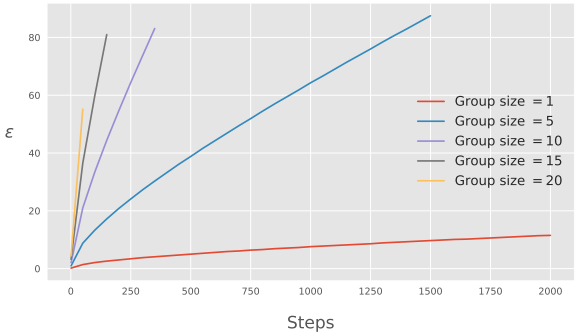


Figure 33: Noise Multiplier=20.00, Sampling rate=1.00.

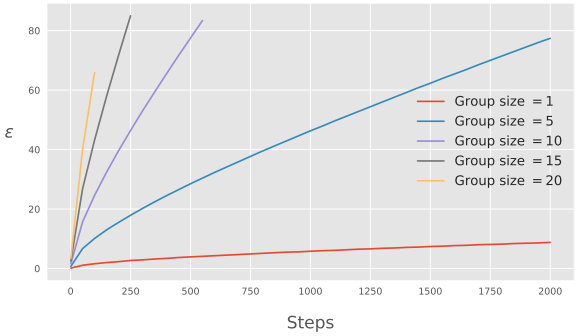


Figure 34: Noise Multiplier=25.00, Sampling rate=1.00.

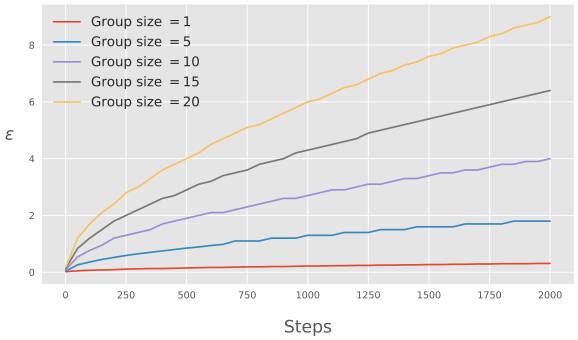


Figure 35: Noise Multiplier=5.00, Sampling rate=0.01.

B.2 Effect of sub-sampling rate on group privacy at various step sizes

In this section, we demonstrate the effect of sub-sampling rate on the group privacy. In the following plots, we set the step size to a fixed value T. We also fix a list of of sampling rates $\{p_i\}_{i\in[10]}$. For each sampling rate, we carefully select a σ_i so that the privacy cost of performing T steps with sampling rate p_i at noise σ_i is exactly the same. Then we calculate the group privacy for all these settings and compare them. Our initial experiments suggested that decreasing the sampling rate will improve group privacy. We ablate this with various step sizes. We try to perform this ablation by plotting figures for a various step sizes. We observe that the effect of sub-sampling rate on group privacy is much more pronounced at smaller step sizes.

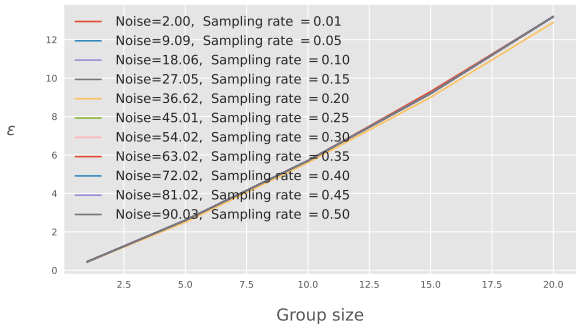


Figure 36: Steps=500.

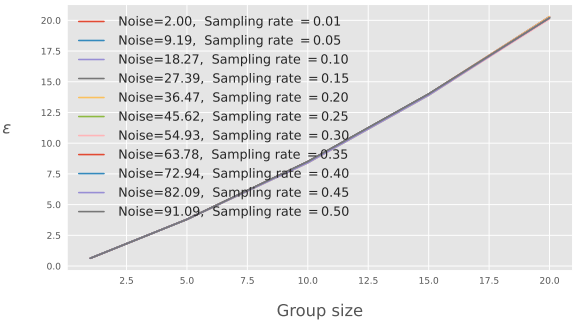


Figure 37: Steps=1000.

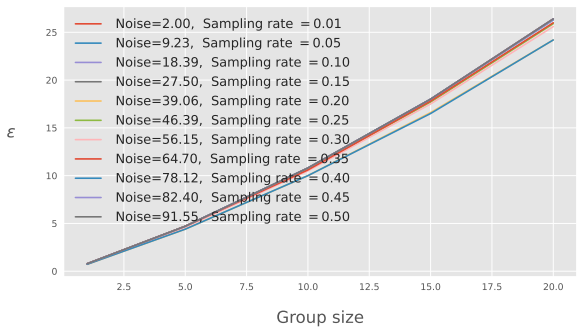


Figure 38: Steps=1500.

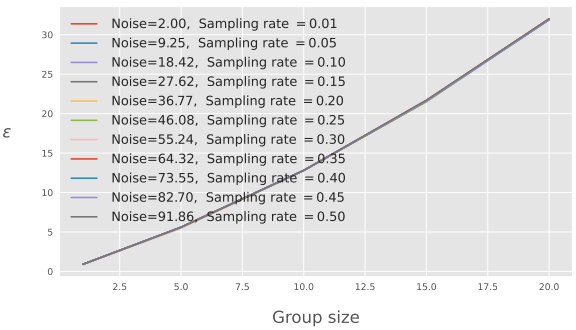


Figure 39: Steps=2000.

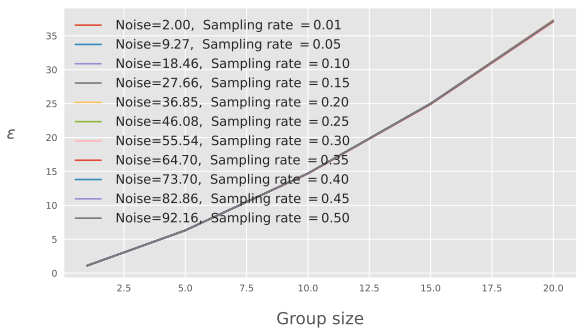


Figure 40: Steps=2500.

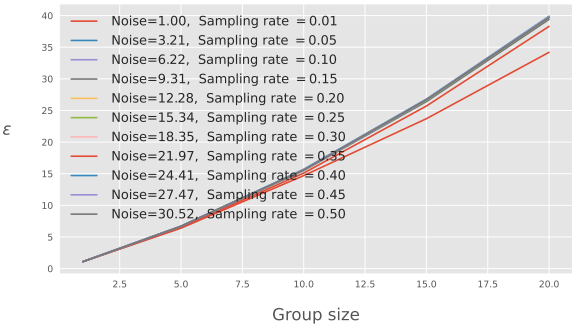


Figure 41: Steps=300.

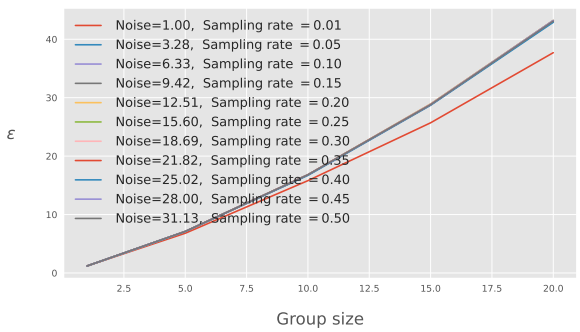


Figure 42: Steps=350.

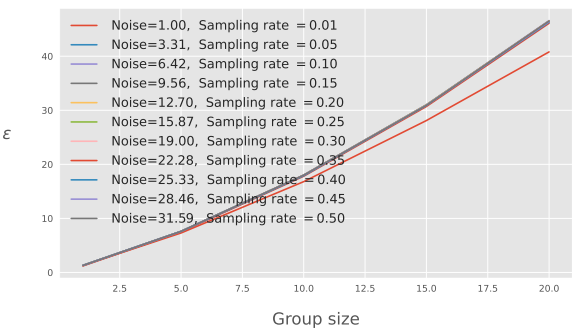


Figure 43: Steps=400.

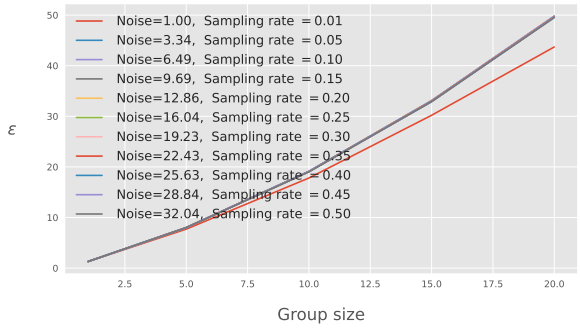


Figure 44: Steps=450.

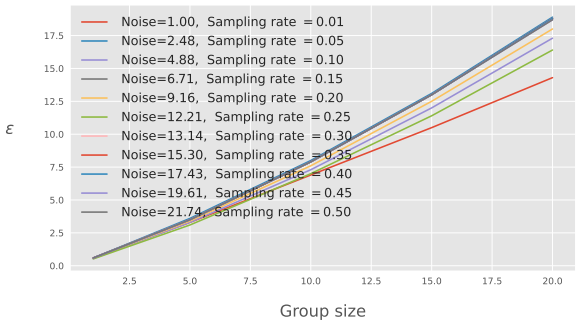


Figure 45: Steps=50.

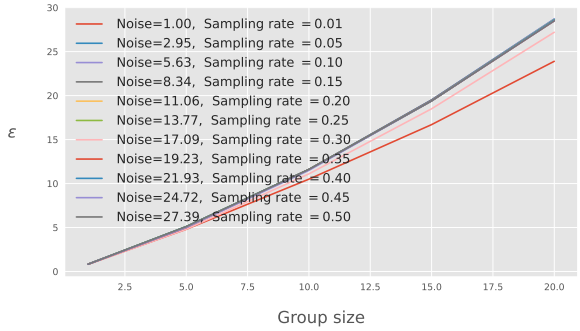


Figure 47: Steps=150.

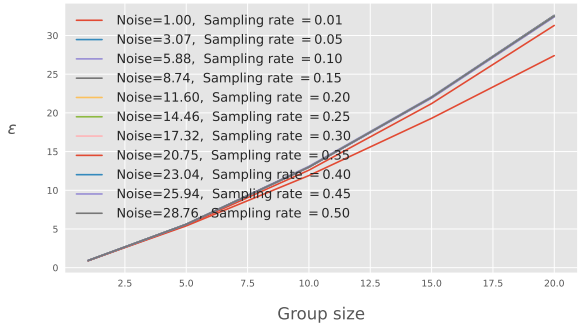


Figure 48: Steps=200.

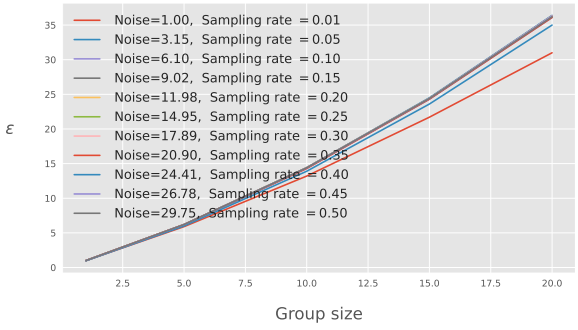


Figure 49: Steps=250.

Rethinking Benchmarks for Differentially Private Image Classification*

Sabrina Mokhtari
University of Waterloo
s4mokhtari@uwaterloo.ca

Sara Kodeiri
University of Waterloo
skodeiri@uwaterloo.ca

Shubhankar Mohapatra
University of Waterloo
s3mohapa@uwaterloo.ca

Florian Tramèr
ETH Zürich
florian.tramer@inf.ethz.ch

Gautam Kamath
University of Waterloo
g@csail.mit.edu

Abstract

We revisit benchmarks for differentially private image classification. We suggest a comprehensive set of benchmarks, allowing researchers to evaluate techniques for differentially private machine learning in a variety of settings, including with and without additional data, in convex settings, and on a variety of qualitatively different datasets. We further test established techniques on these benchmarks in order to see which ideas remain effective in different settings. Finally, we create a publicly available leader board for the community to track progress in differentially private machine learning.

1 Introduction

Machine learning (ML) models have been repeatedly demonstrated to leak sensitive information pertaining to their training data. These issues manifest through a number of different types of attacks, including membership inference [31, 56], model inversion [26], and even training data extraction [13, 14, 57]. This can be problematic if the training data contains privacy-sensitive information belonging to people. To alleviate such concerns, a popular solution is differential privacy (DP) [23]. DP is a rigorous notion of individual data privacy, which can be used to mask the presence or absence of any single training data point when observing a trained model. In particular, training a model with DP provably prevents all the aforementioned attacks.

The past decade has seen significant effort and success in training ML models with DP, including image classifiers [1, 18, 48, 62], large language models [2, 44, 69], and other generative models [6, 8, 12, 20, 29, 67]. However, in a recent position paper, Tramèr, Kamath, and Carlini critique a number of trends in DP ML [63]. Most pertinent to our work, they question whether benchmarks used in DP ML are truly measuring progress in the field, specifically in the context of DP image classification, which will be our focus. The most common benchmark datasets used in DP image classification include MNIST [43],

*Authors SMok, SK, and SMoh have equal contributions and are listed alphabetically in order of *first* name. Authors FT and GK are listed in reverse alphabetical order.

CIFAR-10 [41], and ImageNet [19]. While significant progress has been made on each, TKC question whether this progress generalizes to privacy-sensitive settings where DP may be deployed. For example, CIFAR-10 and ImageNet are both composed primarily of natural images of everyday objects. While these datasets indeed have some privacy concerns [9], it is less clear whether they resemble domains where DP is of high practical concern, such as, e.g., medical images. Since, informally speaking, medical images appear to qualitatively differ from those in the aforementioned datasets, it is unclear whether techniques previously established to be effective remain so in these settings. This question is even more pronounced when models are pre-trained on public data (i.e., supplementary data which is not subject to any privacy constraints), a popular trend in private ML. In such settings, the chosen "public" datasets are often visually similar to the private ones – as a representative example, [18] treat ImageNet as public and privately fine-tune on CIFAR-10. On the other hand, for domains such as medical images, private images may be specialized and ill-represented in public pre-training datasets. Finally, further muddying the waters is the fact that results on these benchmark datasets are often reported for incomparable settings, in particular, with vastly differing public pre-training datasets. Overall, these issues make it difficult to isolate which ideas and techniques are truly effective in privacy-critical settings.

Our contributions are as follows:

- We propose standardized benchmark datasets and evaluation settings to measure progress in DP image classification, with a particular focus on privacy-sensitive domains;
- We release a public leaderboard for DP ML, for the community to track improvements on these benchmarks;
- We evaluate previously established techniques for DP image classification across a variety of settings to see which are and are not broadly effective.

2 Preliminaries

We recall the celebrated notion of differential privacy.

Definition 1 ([22, 23]) *An algorithm $M : \mathcal{X}^n \to \mathcal{Y}$ is (ε, δ)-differentially private if, for all neighboring datasets (i.e., datasets that differ in exactly one entry) X and X' and all events $S \subseteq \mathcal{Y}$, we have that* $\Pr[M(X) \in S] \leq e^{\varepsilon} \Pr[M(X') \in S] + \delta$.

DP is a quantitative definition of individual data privacy. The privacy cost is measured by the parameters (ϵ, δ), also called the privacy budget. Smaller values of ϵ correspond to stricter privacy guarantees, and it is standard in the literature to set $\delta \ll \frac{1}{n}$, where n is the size of the database. Complex DP algorithms can be built from the basic algorithms following two important properties of differential privacy: 1) Post-processing states that for any function g defined over the output of the mechanism $\mathcal{M}$, if $\mathcal{M}$ satisfies (ϵ, δ)-DP, so does $g(\mathcal{M})$; 2) Basic composition states that if for each $i \in [k]$, mechanism $\mathcal{M}_i$ satisfies (ϵ_i, δ_i)-DP, then a mechanism sequentially applying $\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_k$ satisfies $(\sum_{i=1}^k \epsilon_i, \sum_{i=1}^k \delta_i)$-DP.

Given a function $f : \mathcal{D} \to \mathbb{R}^d$, the *Gaussian mechanism* adds noise drawn from a normal distribution $\mathcal{N}(0, S_f^2 \sigma^2)$ to each dimension of the output, where S_f is the ℓ_2-sensitivity of f, defined as $S_f = \max_{D,D' \text{differ in a row}} \|f(D) - f(D')\|_2$. For $\epsilon \in (0, 1)$, if $\sigma \geq \sqrt{2 \ln(1.25/\delta)}/\epsilon$, then the Gaussian mechanism satisfies (ϵ, δ)-DP.

We focus on training ML models subject to DP, which (due to its post-processing property) allows the trained model to be publicly released without further privacy concerns. The most popular method for DP training of ML models is differentially private stochastic gradient descent (DPSGD) [1, 5, 58]. In contrast to non-private SGD where batches are sliced from the training dataset, DPSGD at each

iteration works by sampling "lots" from the training with probability L/n, where L is the (expected) lot size and n is the total data size. A set of queries are computed over those samples. These queries include gradient computation, updates to batch normalization, or accuracy metric calculations. As there is no a priori bound on these query outputs, the sensitivity S_f is set by clipping the maximum ℓ_2 norm of the gradient to a user-defined parameter C. The gradient of each point is then noised and published. All DP optimizers follow the same framework in which they take steps on the computed noisy gradient as in its non-private counterpart. The privacy cost of the whole training procedure is calculated using privacy accounting techniques. We discuss the specifics of DPSGD for our experiments in Section 3.

3 Benchmark design

In this section, we report our specific prescriptions for benchmarks, including datasets, parameters, and best practices, in a variety of settings, in order to standardize and (ideally) propel progress in DP image classification in privacy-critical settings. We note that we (intentionally) do not introduce any new datasets, and instead appeal to existing ones. This is because using established datasets allows for easier comparisons between the private and non-private setting, and introducing an entirely new dataset would serve no benefit for our setting.

Datasets We prescribe using the following two medical image datasets (which have been commonly used in other areas of machine learning) as benchmarks for DP ML: a) CheXpert [33], a chest X-ray dataset; and b) EyePACS [25], a diabetic retinopathy dataset. These datasets are primarily chosen due their privacy-critical domain. We hope that progress on these benchmarks would align with progress (i.e., increased utility) on truly private tasks in such settings. Secondarily, we choose these datasets due to diversity in their sizes, balance of classes, and in the case of CheXpert, for inclusion of a multilabel dataset. Further description of these datasets and justification of these choices appears in Section 4. In addition, we recommend continuing to use CIFAR-10 [41] and ImageNet [19] as benchmarks for training DP ML models *from scratch*, without any pretraining data. Indeed, keeping the caveats of [63] in mind, the popularity of these datasets still allows for direct comparison of accuracy on these tasks, and thus to track "how far behind" DP ML is behind the non-private setting.

Public datasets One of the most successful ways to improve the utility of DP ML has been pre-training the model on "public" data (i.e., data free of any privacy constraints). As discussed by [63], the size and nature of the pre-training data can dramatically affect the downstream utility of a privately fine-tuned model. Therefore, for fair comparison between different techniques, we prescribe tracking progress with the following datasets treated as public: a) no public data, for the "purest" measure of progress in DP ML; b) ImageNet-1K, perhaps the most commonly used large image classification dataset c) LAION-2B, due to it being the pre-training data for OpenCLIP's ViT-G/14 (representing the common use-case of privately fine-tuning a pre-trained CLIP model), and d) "anything goes." To elaborate on the last of these, we use "anything goes" to refer to the case when public pre-training data is unrestricted (barring data-leakage-like considerations where the private dataset contaminates the public one): it may include large-scale Internet datasets, additional domain-specific data, etc. As mentioned before, results in this category may not be directly comparable with each other. Nonetheless, they serve as a measure of absolute progress on a benchmark.

Privacy parameters It is not clear how to compare results on DP image classification at varying levels of the privacy parameters ε and δ. For example, is 90% accuracy at $\varepsilon = 1$ better or worse than 95% at $\varepsilon = 2$? We propose fixing the value of ε to be $1, 3, 5$ and 8 to facilitate direct comparisons between

results. This set of ε covers both high and low privacy regimes across the range usually considered in DP ML. We additionally propose fixing δ to to be the largest power of 10 that is at most the inverse of the training set size (consistent with previous parameter settings), though in many parameter regimes, δ can be dramatically increased or decreased with minor effect on the value of ε.

Privacy accounting. Every DP algorithm is associated with a proof of privacy, which provides an upper bound on the value of ε and δ. For DPSGD, this is generally automated using "privacy accountants," which take as input various hyperparameters and δ, and outputs the value of ε. Over time, improved accounting methods have given increasingly tight analyses, culminating in "exact" privacy accounting techniques [1, 28, 40, 45, 46]. However, as highlighted by some recent works [17, 42, 51], simply using a tighter accountant may give the illusion of an improved result, even if the training procedure is identical. Therefore, we recommend that the privacy accounting method (or, if not using DPSGD, the specific proof followed) is reported in order to keep track of such discrepancies (ideally, all future DPSGD works ought to use exact privacy accountants).

Applicable techniques. The most popular algorithm for DP ML is DPSGD [1, 5, 58], in part due to its flexibility: it can be used to privately train any differentiable model, even non-convex ones. Other methods, such as objective perturbation [15, 34, 38, 54], are usually applicable only to convex models. Consequently, in addition to several non-convex settings, we suggest some standardized convex settings so that a wider variety of methods may be compared and evaluated. We recommend linear probe (i.e., logistic regression) on a) Wide ResNet-28-10 pre-trained on ImageNet-1K;[1] and b) OpenCLIP's ViT-G/14 pre-trained on LAION-2B.

"Anything goes" zero-shot Parallel to the literature on DP ML, the general ML community has studied the challenging "zero-shot" setting, in which goal is to correctly classify a test image without seeing a single image in its training set. Naturally, this requires large-scale public pre-training to achieve acceptable results. In terms of DP, this corresponds to $\varepsilon = 0$ but with "anything goes" pre-training (described above). We suggest tracking the current SOTA for such settings, as a) they serve as an important measure of absolute progress on benchmarks; and b) it is otherwise easy to report a DP result with "anything goes" public data and $\varepsilon > 0$ as SOTA, despite being already dominated by existing zero-shot results.

Overall, we remind that our community's goal ought *not* be to get the highest numbers on these specific datasets, but instead to improve our techniques and understanding of DP image classification for settings that may generalize to those used in practice. We thus focus on a breadth of settings to hopefully cover a range of conditions in which DP classifiers may be deployed. Even if a model can achieve high utility on a benchmark in the "anything goes" zero-shot setting, this does not mean the problem is necessarily "solved." For instance, due to legal, ethical, computational, or safety reasons, depending on the specific setting, it may not be possible to use large, uncurated public datasets for pre-training in a real-world deployment. Therefore, we consider all settings outlined above to be of potential practical or technical interest, and do not identify any of them as "canonical" or more important than another.

3.1 Leaderboard

Tracking progress on benchmark datasets via leaderboards is an established practice in (non-private) ML.[2] This is not yet the case for DP ML: a broad and up-to-date knowledge of the literature is required

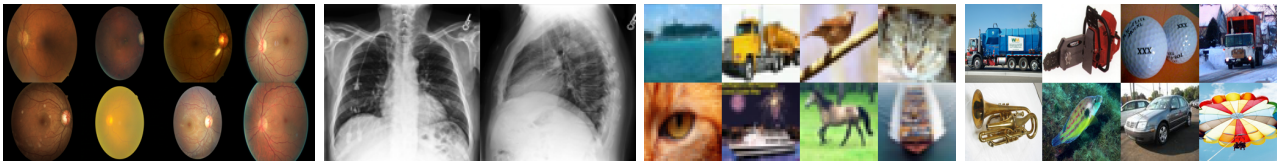
[1] Inspired by [18]. While they release their weights in JAX, we release comparable PyTorch weights with the code https://github.com/mshubhankar/DP-Benchmarks.

[2] See, e.g., `https://paperswithcode.com/sota`

to keep track of the latest results, making entering the field especially challenging and intimidating for newcomers. As one of our contributions, we alleviate this issue by creating and maintaining a leaderboard for DP ML.[3]

Due to the particulars of the DP setting, it is unnatural to simply incorporate results into an existing leaderboard for the non-private setting. Specifically, beyond just the specific dataset, a leaderboard for DP ML would need to track many of the considerations already discussed, including the privacy parameters (ε, δ), which privacy accountant was used, and which public datasets were used. Another difference from the non-private setting is the issue of *correctness*. For a proposed algorithm, the DP guarantees must be mathematically *proven*, and a claimed result could be false if there is a bug in the proof. This is in addition to existing concerns from the non-private setting on whether results are independently reproducible or not. However, since it is notoriously easy to have bugs in a proof of DP, we incorporate a *verification* system to our leaderboard. By default, all results are unverified when added. However, anyone is able to submit a pull request to our GitHub to verify that they reproduced the result, and believe correctness of the privacy proof (if applicable).

At present, our leaderboard focuses exclusively on DP image classification (as does this paper), though it may be extended to other problems (e.g., DP natural language understanding or generation).



(a) EyePACS (b) CheXpert (c) CIFAR-10 (d) ImageNet

Figure 1: EyePACS and CheXpert qualitatively look different than common benchmark datasets such as CIFAR-10 and ImageNet.

4 Datasets and Architectures

Here, we describe the relevant datasets and architectures, which are later explored in the experiment section.[4]

4.1 Overview of datasets

CheXpert The CheXpert dataset [33] has 224,316 chest X-ray images of size 390×320 from 64,540 patients. Images may have multiple labels, where the possible labels correspond to five pathology classes: 'Cardiomegaly', 'Edema', 'Consolidation', 'Atelectasis', and 'Pleural Effusion'. In our work, following prior state-of-the-art training, we re-scale all images to size 224×224 and augment the dataset using random affine transformations [70, 71].

EyePACS Kaggle EyePACS [25] contains retinal images of diverse populations with various degrees of diabetic retinopathy (DR). Each image is classified into one of five classes depending on the severity of the disease. The classification task is diagnostic of DR, as measured on a scale from 0 (no DR) to 4 (severe DR). The training set consists of 35,126 and the test set contains 53,576 color eye fundus images.

[3] Our leaderboard is available at https://private-machinelearning.github.io/

[4] Any omitted hyperparameter or architectural details appear in the code repository https://github.com/mshubhankar/DP-Benchmarks

To speak to these particular dataset selections: as mentioned before, we chose medical images to address a privacy-critical setting where DP may be deployed. Within this area, chest X-rays and fundus images are two of the most common domains, so we chose one of the most popular datasets from each of these domains. Additionally, we took guidance from [53], which also focuses on medical image classification, and studies CheXpert and a Google-proprietary DR dataset. While there are several public fundus photography datasets, most of them are very small (< 100 images) and thus not settings we would expect DP to function well: EyePACS is the most popular one of an acceptable size.

4.2 Overview of architectures and techniques

ScatterNets ScatterNets [47] (SN) are convolutional neural networks (CNNs) that utilize pre-defined wavelets for their architecture and filters. In other words, the features are "hand-crafted" rather than learned from data, and thus use neither public nor private data. Tramèr and Boneh [62] employ this architecture for DP image classification, using DPSGD to train either linear or convolutional layers acting on these features, and demonstrate compelling results on MNIST and CIFAR-10, particularly for small values of ε. We exclusively use ScatterNets without any public data.

Wide-ResNets The Wide-ResNet [72] (WRN) is a variant of the ResNet[30] that reduces issues of vanishing and exploding gradients by making the model wider instead of deeper. De et al. [18] use them to reach DP SOTA in multiple settings on CIFAR-10. They consider both DP training from scratch, and DP fine-tuning after being (publicly) pre-trained on ImageNet-1K (downsampled to 32×32, which we call IN-32 [16]).[5] To allow direct comparison, we emulate their setting as much as possible, e.g., using weight standardization [10], group normalization, and their choices of hyperparameters for pre-training. We use both without any public data, and pre-trained on ImageNet-1K.

Additionally, Tang et al. [60] utilize WRN-16-4 to achieve DP SOTA performance on CIFAR-10, when no extra public data is used for pretraining. They leverage image priors generated by random processes [3] instead of starting from random initialization, outperforming [62] and [18] when they only train from scratch. Moreover, they achieve SOTA performance using only a linear probe, making for a direct comparison to the linear ScatterNet method of [62]. We adopt the same architecture and replicate their settings to the greatest extent possible, incorporating techniques such as augmentation multiplicity and normalization. Tang et al. [60] build on the approach of De et al. [18] by using the third-to-last layer of the network, which has a dimension of 4096. We adopt a similar strategy but reduce the dimensionality to 2048. This adjustment is necessary due to the larger image sizes in our datasets (CheXpert and EyePACS with 224×224 images) compared to CIFAR-10 (32×32 images) and resource constraints.

CLIP-based models CLIP [52] is a popular contrastive learning pre-training technique, which allows one to jointly train a language and image encoder. CLIP has been observed to enable robust zero-shot image classification when pre-training on very large Internet datasets. We use two ViT [21] models pre-trained using CLIP: OpenAI's ViT-B/16 (pre-trained on the proprietary WebImageText (WIT) dataset) and OpenCLIP's ViT-G/14 model (pre-trained on LAION-2B [55]).[6] Besides pre-training data, these models differ in their size (12 and 48 layers, respectively) and patch size (16 and 14, respectively). For zero-shot experiments we use these models as-is, for DP fine-tuning experiments, we use only the image encoder as a feature extractor, and on top of that, apply either a linear layer (i.e., logistic

[5] They use WRN-16-4 and WRN-40-4 for from-scratch experiments and WRN-28-10 for fine-tuning experiments. For simplicity, we use WRN-28-10 in all our experiments.

[6] `https://github.com/mlfoundations/open_clip`

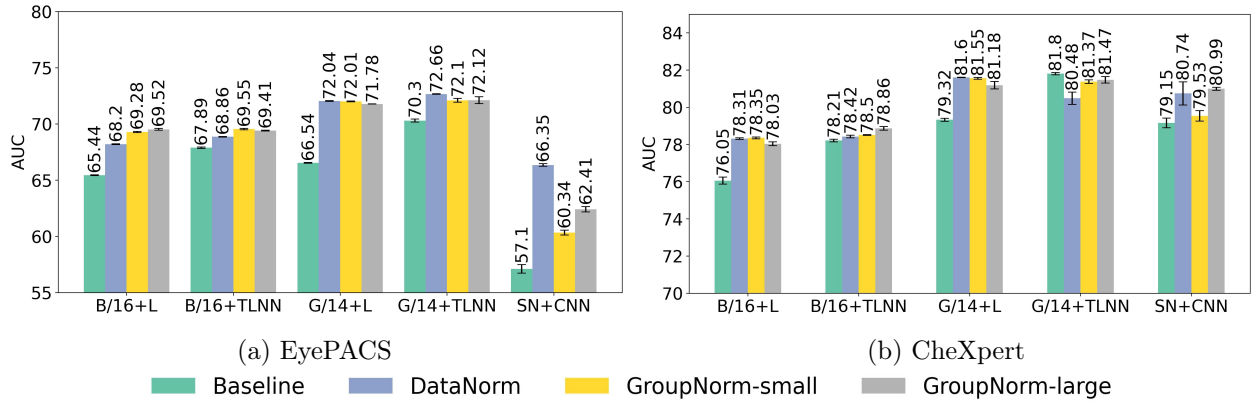


Figure 2: Normalization generally improves the final performance of all models. For CLIP-ViT models, GroupNorm small and large are groups of 8 and 16, respectively. For ScatterNets, GroupNorm small is 9 and large is 27. The choice of 27 over 81 is due to its superior performance. All experiments are done at $\varepsilon = 3$.

regression) or a two-layer neural network (TLNN, featuring tanh/tempered sigmoid activations [48]). We exclusively use CLIP-based models with their respective public pre-training datasets.

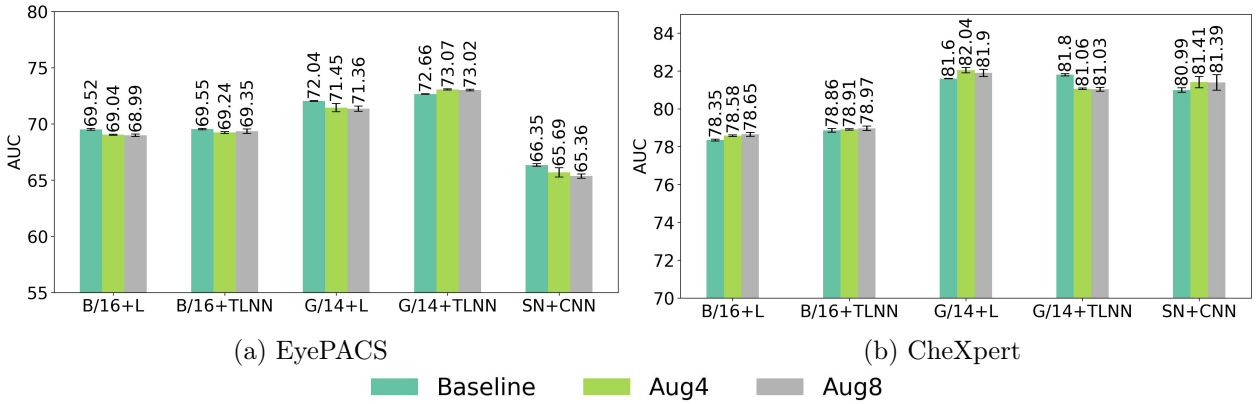
5 Experiments

Beyond proposing a variety of datasets and evaluation settings for benchmarking, we experimentally investigate techniques and the resulting utility obtained therein. Some of the key questions guiding our exporation: how many of the lessons learned in DP image classification on datasets like CIFAR-10 and ImageNet transfer to the privacy-critical setting of medical images? How much and when does public data help for such datasets, which may be ill-represented in public data? And, in absolute terms, how well can we do on these datasets with DP, in various evaluation settings?

After describing our experimental setup (Section 5.1), we revisit the efficacy of several ablations commonly employed in DP settings (Section 5.2). Finally, we make more broad conclusions about DP image classification based on our results (Section 5.3). Our code is included in the code repository.

5.1 Experimental Setup

We use PyTorch [49], and the Opacus library [68] for DP ML. We employed the Adam optimizer [39] across all experiments, both private and non-private, with a default learning rate of 0.001. We run our experiments at a variety of privacy levels ($\varepsilon \in [1, 3, 5, 8]$) with fixed delta values proportional to the inverse of the dataset size (10^{-6} for CheXpert and 10^{-5} for EyePACS), as we prescribed earlier. Batch size and total training epochs were fixed at 1024 and 20, respectively. A hyper-parameter search was performed to identify the optimal clipping norm within the range $[0.001, 0.01, 0.1, 1, 10]$. Following established metrics for all these datasets, we use AUC for CheXpert and EyePACS, and accuracy for CIFAR-10. We report mean and standard deviation over three independent runs. We used early stopping for non-private numbers due to overfitting, a phenomenon we did not observe for the DP setting due to its natural regularization properties [37].



(a) EyePACS (b) CheXpert

Figure 3: Augmentation multiplicity helps in general for CheXpert but not for EyePACS. We evaluate augmentation multiplicity by adding 4 and 8 augmentations of each image in the training data. All experiments are done at $\varepsilon = 3$.

5.2 Revisiting DP ablations

One of the most comprehensive ablation studies for DP image classification is by [18]. By using group normalization, large batches, weight standardization, augmentation multiplicity, and parameter averaging, they manage to raise CIFAR-10 accuracy on a validation set from 50.8% to an impressive 79.7%. We fix $\varepsilon = 3$ and, focusing on the CLIP ViTs and ScatterNets, run the exact same ordered sequence of ablations, without carrying forward the latest technique if it does not show improved utility. Broadly speaking, while [18]'s techniques proved highly effective for CIFAR-10, our results reveal mixed outcomes depending on various parameters.

Normalization Batch normalization [32] is not compatible with DPSGD because it combines information across a batch, making it impossible to bound the impact of a single image in the dataset. Instead, prior work has shown that variants including group normalization [66] and data normalization can be suitable replacements [7, 18, 24, 62].

Group normalization splits the channels of the hidden activations of an image into groups and normalizes the activations within each group. For the CLIP ViT models, with an input dimension of 512 and 1280 for the B/16 and G/14 respectively, we experiment with 8 and 16 groups. For Scatter features, with dimension (243, H/4, W/4) for RGB images and following [62], we use 9, 27, and 81 groups. Data normalization works on data channels by normalizing using the corresponding mean and variance across the training data. Normalizing in such a way, however, incurs a privacy cost as the per-channel means and variances must be privately estimated. We use Gaussian noise with ($\sigma = 8$) to estimate these means and variances for all runs, following [62].

In Figure 2 we show that normalization generally improves the final performance of all models, though the most effective normalization differs across architecture and dataset. Interestingly, the four experiments where data normalization was superior involved models with larger unclipped gradients. In these cases, the optimal clipping norm chosen during hyperparameter tuning was also the highest value (10). This suggests that data normalization can effectively manage large gradient magnitudes, especially when clipping underestimates the true gradient norms. Detailed results for our experiments are given in Table 1 and Table 2.

Table 1: Studying the impact of normalization for ScatterNet + CNN, normalization consistently improves performance. Data normalization tends to outperform group normalization for EyePACS and CIFAR-10, particularly due to the large gradients of their Scatter features.

Dataset	Model	Baseline	DataNorm	GroupNorm9	GroupNorm27	GroupNorm81
EyePACS (AUC)	SN + CNN	57.1 ± 0.38	$\mathbf{66.35} \pm 0.12$	60.34 ± 0.22	62.41 ± 0.24	63.68 ± 0.15
CheXpert (AUC)	SN + CNN	79.15 ± 0.26	80.74 ± 0.62	79.53 ± 0.28	$\mathbf{80.99} \pm 0.08$	80.72 ± 0.35
CIFAR-10 (Acc)	SN + CNN	55.18 ± 0.28	$\mathbf{68.29 \pm 0.17}$	65.97 ± 0.13	66.26 ± 0.11	66.45 ± 0.45

Table 2: Normalization impact for CLIP ViT models: Normalization generally improves performance, but it also depends on the architecture and dataset. Normalizations marked in red show a drop in performance compared to the baseline.

Dataset	Model	Baseline	DataNorm	GroupNorm8	GroupNorm16
EyePACS	B/16 + Linear	65.44 ± 0.04	68.2 ± 0.04	69.28 ± 0.06	$\mathbf{69.52 \pm 0.08}$
EyePACS	B/16 + TLNN	67.89 ± 0.06	68.86 ± 0.03	$\mathbf{69.55 \pm 0.06}$	69.41 ± 0.04
EyePACS	G/14 + Linear	66.54 ± 0.04	$\mathbf{72.04 \pm 0.03}$	72.01 ± 0.04	71.78 ± 0.01
EyePACS	G/14 + TLNN	70.30 ± 0.13	$\mathbf{72.66 \pm 0.03}$	72.1 ± 0.17	72.12 ± 0.31
EyePACS	G/14(CLIPA) + Linear	63.88 ± 0.08	$\mathbf{73.02 \pm 0.2}$	70.7 ± 0.06	70.62 ± 0.07
EyePACS	G/14(CLIPA) + TLNN	64.9 ± 0.2	$\mathbf{72.9 \pm 0.1}$	70.87 ± 0.25	70.8 ± 0.2
CheXpert	B/16 + Linear	76.05 ± 0.19	78.31 ± 0.04	$\mathbf{78.35 \pm 0.05}$	78.03 ± 0.1
CheXpert	B/16 + TLNN	78.21 ± 0.07	78.42 ± 0.07	78.5 ± 0.02	$\mathbf{78.86 \pm 0.1}$
CheXpert	G/14 + Linear	79.32 ± 0.08	$\mathbf{81.6 \pm 0.01}$	81.55 ± 0.05	81.18 ± 0.2
CheXpert	G/14 + TLNN	$\mathbf{81.80 \pm 0.06}$	80.48 ± 0.33	81.37 ± 0.1	81.47 ± 0.17
CheXpert	G/14(CLIPA) + Linear	72.39 ± 0.07	76.12 ± 0.4	$\mathbf{77.34 \pm 1.1}$	77.17 ± 0.5
CheXpert	G/14(CLIPA) + TLNN	77.65 ± 0.89	75.74 ± 1.1	77.38 ± 0.3	$\mathbf{77.43 \pm 0.7}$
CIFAR10	CLIP + Linear	99.64(93.91)	99.76(94.41)	99.75(94.43)	**99.75(94.57)**
CIFAR10	CLIP + TLNN	99.69(94.10)	99.74(94.15)	99.74(94.36)	**99.75(94.51)**

Larger Batch Size The impact of larger batch sizes in differentially private training has been observed both theoretically [4, 59] and empirically [2, 18]. In Table 3, scaling the batch size from 1024 to 4096 showed that CheXpert benefited in 80% of experiments, while EyePACS did not. This disparity is likely due to CheXpert having a training set six times larger than EyePACS, resulting in fewer model update steps for EyePACS and potential underfitting with a fixed number of epochs. We further observed that increasing the number of epochs showed a positive impact of larger batch sizes on EyePACS when using the ScatterNet model.

Weight Standardization We experiment with weight standardization (WS) on the Scatternet + CNN model as it applies to only convolution layers. From our results in Table 3, we observe that weight standardization does not help with EyePACS but helps with CheXpert and CIFAR-10. As alluded by prior work [10, 18], we also observe a positive correlation of group normalization with WS. However, due to a limited number of experiments, we do not have strong evidence either way.

Augmentation Multiplicity We apply a sequence of augmentations to our benchmark datasets: reflect padding, random cropping, and random horizontal flipping. While [18] recommend 16 augmentations per image, due to computational constraints with large datasets, we use 4 and 8 augmentations. As shown in Figure 3, contrary to [18]'s findings, augmentation multiplicity (augmult) does not consistently yield positive effects. Except for one experiment, (ViT-G/14+TLNN), augmentations generally benefit

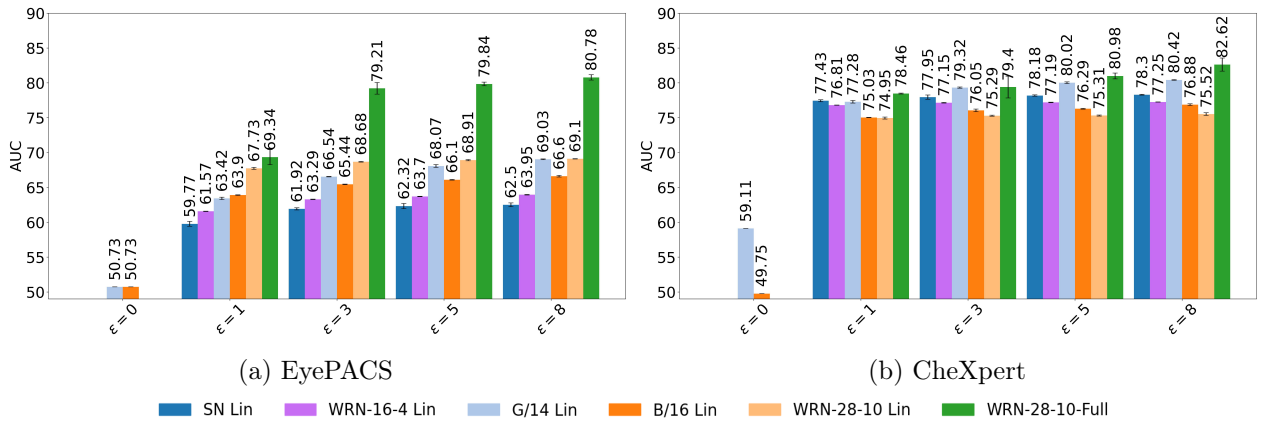


Figure 4: Pre-training datasets have different impacts: Wide-ResNet, pretrained on ImageNet, performs best on EyePACS, while ViT-G/14 with linear probe surpasses Wide-ResNet 28-10 linear probe across all ε values on CheXpert. Furthermore, ViT-G/14 achieves near-random performance on EyePACS in zero-shot settings but attains a non-trivial 59.11% AUC on CheXpert.

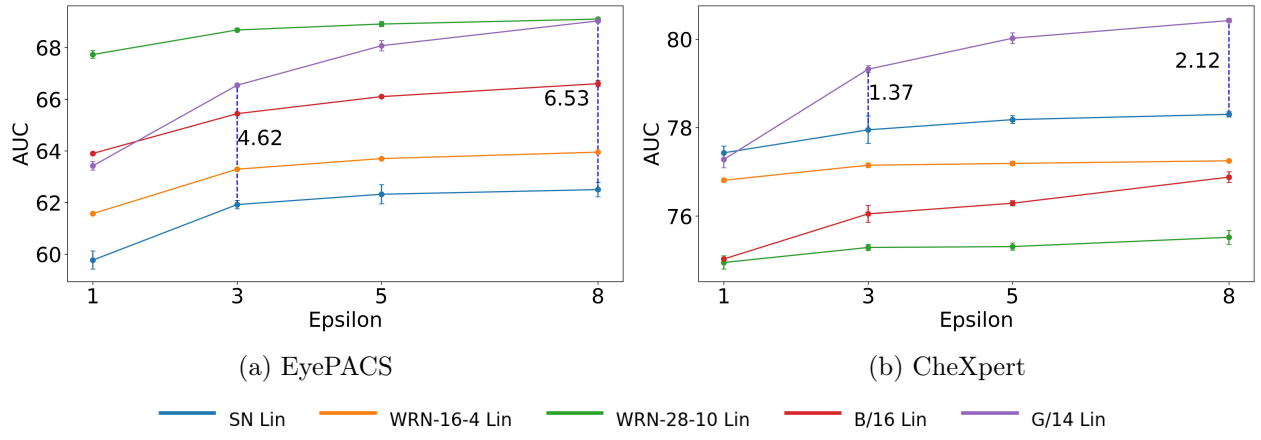


Figure 5: Pre-training public data is more beneficial with higher ε values. For CheXpert, ScatterNet performs better at smaller ε values, while pretrained models show marginal improvements at larger ε values. Similarly, for EyePACS, CLIP ViT-G/14 linear performs better as ε value increases.

Table 3: Studying all ablations together, we observe an almost consistent improvement in performance for CIFAR-10, whereas this pattern is not observed with the other datasets.

Dataset	Model	Best Normalization	+Larger Batch	+WS	+Best Augmult	+EMA
EyePACS	SN + CNN	66.35 ± 0.12	66.24 ± 0.23	65.81 ± 0.06	65.69 ± 0.42	$\mathbf{66.65 \pm 0.16}$
EyePACS	B/16 + L	$\mathbf{69.52 \pm 0.08}$	68.95 ± 0.24	-	69.04 ± 0.05	69.09 ± 0.12
EyePACS	B/16 + TLNN	$\mathbf{69.55 \pm 0.06}$	69.16 ± 0.28	-	69.24 ± 0.08	69.3 ± 0.21
EyePACS	G/14 + L	$\mathbf{72.04 \pm 0.03}$	71.59 ± 0.24	-	71.45 ± 0.37	71.29 ± 0.17
EyePACS	G/14 + TLNN	72.66 ± 0.03	$\mathbf{73.07 \pm 0.06}$	-	73.07 ± 0.05	73 ± 0.03
EyePACS	CLIPA + L	$\mathbf{73.02 \pm 0.2}$	68.5 ± 0.07	-	68.37 ± 0.05	70.94 ± 0
EyePACS	CLIPA + TLNN	72.09 ± 0.1	$\mathbf{72.17 \pm 0.1}$	-	72.04 ± 1.6	71.9 ± 0.07
CheXpert	SN + CNN	80.99 ± 0.08	81.54 ± 0.34	82.11 ± 0.29	81.41 ± 0.3	$\mathbf{82.24 \pm 0.22}$
CheXpert	B/16 + L	78.35 ± 0.05	78.42 ± 0.07	-	$\mathbf{78.65 \pm 0.1}$	$\mathbf{78.65 \pm 0.04}$
CheXpert	B/16 + TLNN	78.86 ± 0.1	78.86 ± 0.1	-	78.97 ± 0.12	$\mathbf{79.01 \pm 0.1}$
CheXpert	G/14 + L	81.6 ± 0.01	81.76 ± 0.05	-	$\mathbf{82.04 \pm 0.14}$	82.01 ± 0.05
CheXpert	G/14 + TLNN	81.8 ± 0.06	81.04 ± 0.16	-	81.06 ± 0.34	81.14 ± 0.26
CheXpert	CLIPA + L	77.34 ± 1.1	80.17 ± 0.08	-	$\mathbf{80.38 \pm 1.5}$	80.34 ± 0.15
CheXpert	CLIPA + TLNN	77.43 ± 0.7	80.4 ± 0.2	-	$\mathbf{80.75 \pm 0.18}$	80.52 ± 0.02
CIFAR10	B/16 + L	99.75(94.57)	99.74(94.49)	-	**99.77(94.76)**	99.76(94.67)
CIFAR10	B/16 + TLNN	99.75(94.51)	99.76(94.55)	-	**99.79(94.81)**	99.78(94.76)
CIFAR-10	SN + CNN	68.29 ± 0.17	66.47 ± 0.31	68.96 ± 0.26	$\mathbf{69.16 \pm 0.08}$	68.07 ± 0.24

CheXpert but not EyePACS. Future work may explore the effectiveness of dataset-specific augmentations, which could potentially yield more beneficial results. We show detailed experiment results in Table 4.

Parameter Averaging The final ablation that [18] suggests is the exponential moving average (EMA)[50] of all the parameters in the model. In Table 3, we notice that EMA occasionally improves performance, which contradicts the findings of [18] that it consistently enhances results across all experiments.

5.3 Experimental findings

We highlight some findings from our experimental results.

Different pre-training datasets offer varying degrees of improvement depending on the private data We compare representative models publicly pre-trained on a variety of datasets on both CheXpert and EyePACS. Results are displayed in Figure 4. For the case of no pre-training data, we choose ScatterNet+Linear, due to its consistently superior utility compared to Wide-ResNet trained from scratch, particularly for high privacy (i.e., low ε) settings.

On the other end of the spectrum, when we allow large-scale public pre-trarining, the CLIP ViT models provide a good indication of zero-shot performance (i.e., $\varepsilon = 0$).

When analyzing CheXpert, ViT-B/16 performs close to random in the zero-shot setting, whereas ViT-G/14 achieves an AUC of 59.11%, moderately better than random. Moving from ScatterNet+linear to Wide-ResNet+Linear, there is a noticeable decrease in AUC, yet ViT-G/14 consistently outperforms across various ε values, indicating that ViT-G/14 is a better fit for CheXpert. Notably, at $\varepsilon = 8$, Wide-ResNet with full fine-tuning exceeds the performance of ViT-G/14. However, considering that Wide-ResNet is fully fine-tuned while ViT-G/14 is not, this doesn't necessarily make Wide-ResNet better suited for CheXpert. Nevertheless, for smaller ε, it is clear that ViT-G/14 is the superior model with only linear fine-tuning.

Table 4: Studying the impact of augmentation multiplicity, we find that it consistently improves performance for CIFAR-10. However, looking at EyePACS and CheXpert, we observe inconsistent behavior, except that it generally seems to reduce performance with EyePACS. For the third column, we take the best result from Table 3 after best normalization, larger batch size, and weight standardization.

Dataset	Model	Norm + Larger BS + WS	+Augmult(4)	+Augmult(8)
EyePACS	SN + CNN	**66.35 ± 0.12**	65.69 ± 0.42	65.36 ± 0.19
EyePACS	B/16 + L	**69.52 ± 0.08**	69.04 ± 0.05	68.99 ± 0.11
EyePACS	B/16 + TLNN	**69.55 ± 0.06**	69.24 ± 0.08	69.35 ± 0.2
EyePACS	G/14 + L	72.04 ± 0.03	71.45 ± 0.37	71.36 ± 0.24
EyePACS	G/14 + TLNN	73.07 ± 0.05	**73.07 ± 0.05**	73.02 ± 0.07
EyePACS	G/14 (CLIPA)+ L	73.02 ± 0.2	68.37 ± 0.05	68.14 ± 1.2
EyePACS	G/14 (CLIPA)+ TLNN	72.9 ± 0.1	72.04 ± 1.6	71.98 ± 0.3
CheXpert	SN + CNN	82.11 ± 0.29	81.41 ± 0.3	81.39 ± 0.41
CheXpert	B/16 + L	78.42 ± 0.07	78.58 ± 0.04	**78.65 ± 0.1**
CheXpert	B/16 + TLNN	78.86 ± 0.1	78.91 ± 0.04	**78.97 ± 0.12**
CheXpert	G/14 + L	81.76 ± 0.05	**82.04 ± 0.14**	81.9 ± 0.19
CheXpert	G/14 + TLNN	81.8 ± 0.06	81.06 ± 0.04	81.03 ± 0.1
CheXpert	G/14 (CLIPA)+ L	77.34 ± 1.1	80.34 ± 0.2	**80.38 ± 1.5**
CheXpert	G/14 (CLIPA)+ TLNN	77.43 ± 0.7	80.5 ± 0.1	**80.75 ± 0.18**
CIFAR-10 (ACC)	SN + CNN	68.96 ± 0.26	69.07 ± 0.2	**69.16** ± 0.08
CIFAR-10 (ACC)	B/16 + L	99.75(94.57)	99.76(94.68)	**99.77(94.76)**
CIFAR-10 (ACC)	B/16 + TLNN	99.76(94.55)	99.78(94.75)	**99.79(94.81)**

Looking at Figure 4 for EyePACS, both CLIP ViT models show random performance in the zero-shot setting, indicating no improvement from pretraining. Conversely, Wide-ResNet linear exhibits a significant performance boost when transitioning from ScatterNet linear to Wide-ResNet linear, maintaining its superiority across all ε values. Although we notice that as we move toward less private regimes, the power of pre-trained ViT-G/14 becomes more evident, particularly from $\varepsilon = 1$ to $\varepsilon = 3$, approaching the performance of Wide-ResNet linear. However, there remains a substantial gap between fully fine-tuned Wide-ResNet and the other models, unlike CheXpert, suggesting that Wide-ResNet is better suited for EyePACS.

Public pre-training data helps more with higher ε values We compare feature generation methods in Figure 5 since, in all cases, there is a linear classifier on top of diverse feature extractors. On CheXpert, linear fine-tuning with ScatterNet shows the best performance at $\varepsilon = 1$. However, as ε increases, pretrained models, especially ViT-G/14, begin to outperform other methods significantly. While full fine-tuning of CLIP has not been explored, a direct comparison of features shows ViT-G/14's superiority when ε is sufficiently large. As ε value increases further, ViT-G/14's performance improves notably, highlighting its strong pretrained performance under less stringent privacy constraints.

When comparing the best performance on CheXpert across our proposed methods, ScatterNet achieves superior results compared to CLIP ViT models and Wide-ResNet on $\varepsilon = 3$, as shown in Figure 5. However, as ε values increase, pretrained models begin to perform better, and the performance gap between ScatterNet and the other models widens.

For EyePACS, we don't see the same pattern, likely because EyePACS is a much smaller dataset (about one-sixth the size) and Scatter features have high dimensionality, making it hard to balance this complexity with private training. We use ScatterNet linear as the baseline for the no pretraining regime and compare it to other architectures' linear fine-tuning for a fair comparison.

Table 5: Test AUC for EyePACS at different epsilons. Baselines include ScatterNet (SN), WideResNet (WRN) and CLIP models on datasets with different public data pre-training. The SOTA is due to [64].

Public data	Model	Test AUC (%)				
		$\varepsilon = 1$	$\varepsilon = 3$	$\varepsilon = 5$	$\varepsilon = 8$	$\varepsilon = \infty$
None	SN + L	59.77 ± 0.35	61.92 ± 0.16	62.32 ± 0.37	62.5 ± 0.28	69.70 ± 0.11
None	SN + CNN	63.73 ± 0.11	66.36 ± 0.17	66.59 ± 0.43	67.37 ± 0.27	69.28 ± 0.20
None	WRN-16-4+L	61.57 ± 0.02	63.29 ± 0.04	63.7 ± 0.04	63.95 ± 0.04	67.74 ± 0.01
None	WRN (Scratch)	55.45 ± 0.18	56.53 ± 0.08	57.14 ± 0.34	57.65 ± 0.22	61.97 ± 0.09
IN-32	WRN + Linear	67.73 ± 0.15	68.68 ± 0.05	68.91 ± 0.10	69.10 ± 0.03	73.21 ± 0.09
IN-32	WRN (Full)	69.34 ± 1.09	79.21 ± 0.83	79.84 ± 0.27	80.78 ± 0.38	83.61 ± 0.03
WIT	B/16 + Linear	63.9 ± 0.04	65.44 ± 0.04	66.1 ± 0.05	66.6 ± 0.12	69.93 ± 0.01
WIT	B/16 + TLNN	65.12 ± 0.04	67.89 ± 0.06	69.22 ± 0.1	69.84 ± 0.02	70.54 ± 0.01
LAION	G/14 + Linear	63.42 ± 0.17	66.54 ± 0.04	68.07 ± 0.2	69.03 ± 0.06	69.88 ± 0.2
LAION	G/14 + TLNN	65.47 ± 0.02	70.30 ± 0.13	71.74 ± 0.3	72.3 ± 0.19	73.36 ± 0.15
DataComp1B	G/14(CLIPA) + Linear	63.42 ± 0.06	63.88 ± 0.08	63.8 ± 0.12	64.33 ± 0.32	70.87 ± 0.01
DataComp1B	G/14(CLIPA) + TLNN	64.41 ± 1.00	64.9 ± 0.20	65.07 ± 0.26	65.67 ± 0.08	75.42 ± 0.08
IN-1K	SOTA	-	-	-	-	95.1

As illustrated in Figure 5, increasing the ε value amplifies ViT-G/14's performance advantage over the ScatterNet baseline, widening the gap. However, we do not observe any significant changes in ViT-B/16 and Wide-ResNet linear. ViT-B/16 appears to perform poorly regardless of privacy settings. On the other hand, Wide-ResNet linear consistently maintains a significant gap between its linear model and ScatterNet. This can be explained by the fact that Wide-ResNet linear can already achieve high AUC in the $\varepsilon = 1$ case, leaving little room for improvement.

The fact that Wide-ResNet maintains its advantage from the start is not surprising, given that as discussed earlier, the pre-trained model seems to help with EyePACS the most. However, ViT-G/14's performance improves more as the ε value increases. The detailed numbers for this experiment are provided in Table 5 and Table 6.

Progress on CIFAR10 does not translate to progress on benchmark datasets Looking at Figure 4, we notice that ViT-G/14 achieves an astonishing 99.75% zero-shot accuracy on CIFAR-10. In stark contrast, the same model's zero-shot performance on CheXpert and EyePACS is significantly lower, with AUC scores of 59.11% and 50.73%, respectively—the latter essentially equating to random guessing. Additionally, Wide-ResNet achieves 94.7% accuracy on CIFAR-10 at $\varepsilon = 1$, yet only 78.52% and 71.00% AUC on CheXpert and EyePACS, respectively.

Upon reviewing the ablation experiments in section 5.2, it becomes evident that the techniques beneficial for CIFAR-10 do not necessarily yield similar advantages for EyePACS and CheXpert datasets. The patterns observed in CIFAR-10 did not replicate in these medical image datasets, and notably, performance on CheXpert and EyePACS showed inconsistency.

Additionally, we observe that incorporating synthetic data as demonstrated by Tang et al. [60], leads to SOTA performance on CIFAR-10 without pretraining. However, in our experiments, ScatterNet outperforms [60]'s approach on CheXpert, whereas on EyePACS, Tang et al. achieve better results.

6 Related Work

Several works have evaluated the privacy-utility tradeoffs for DPML algorithms [35, 36, 73]. Jayaraman et al. [36] explored the impact of various variants of DP for ML algorithms. They explored the privacy

Table 6: Test AUC for CheXpert at different epsilons. Baselines include ScatterNet (SN), WideResNet (WRN) and CLIP models on datasets with different public data pre-training. The SOTA is from [7](Private) and [70](Non-Private).

Public data	Model	Test AUC (%)				
		$\varepsilon = 1$	$\varepsilon = 3$	$\varepsilon = 5$	$\varepsilon = 8$	$\varepsilon = \infty$
None	SN + CNN	78.16 ± 0.22	79.15 ± 0.26	79.16 ± 0.18	79.68 ± 0.04	80.65 ± 0.12
None	SN + Linear	77.43 ± 0.15	77.95 ± 0.31	78.18 ± 0.09	78.30 ± 0.06	78.94 ± 0.13
None	WRN-16-4+L	76.81 ± 0.02	77.15 ± 0.05	77.19 ± 0.05	77.25 ± 0.01	77.49 ± 0.03
None	WRN (Scratch)	76.9 ± 0.02	77.8 ± 0.04	77.89 ± 0.05	78.68 ± 0.10	87.31 ± 0.07
IN-32	WRN + Linear	74.95 ± 0.15	75.29 ± 0.07	75.31 ± 0.08	75.52 ± 0.16	75.91 ± 0.08
IN-32	WRN (Full)	78.46 ± 0.07	79.40 ± 1.57	80.98 ± 0.42	82.62 ± 0.94	87.62 ± 0.09
WIT	B/16 + Linear	75.03 ± 0.03	76.05 ± 0.19	76.29 ± 0.06	76.88 ± 0.12	76.89 ± 0.01
WIT	B/16 + TLNN	77.28 ± 0.19	78.21 ± 0.07	78.33 ± 0.04	78.54 ± 0.06	78.56 ± 0.01
LAION	G/14 + Linear	77.28 ± 0.19	79.32 ± 0.08	80.02 ± 0.12	80.42 ± 0.05	80.48 ± 0.02
LAION	G/14 + TLNN	80.63 ± 0.4	81.80 ± 0.06	82.25 ± 0.02	$82.27 \pm 0.0.4$	82.28 ± 0.01
DataComp1B	G/14(CLIPA) + Linear	71.45 ± 0.27	72.39 ± 0.07	72.98 ± 0.38	72.37 ± 0.41	78.35 ± 1.3
DataComp1B	G/14(CLIPA) + TLNN	77.3 ± 0.60	77.65 ± 0.89	77.67 ± 0.25	77.51 ± 0.85	80.62 ± 0.06
IN-21K	SOTA	86.3	-	-	89.2	-
IN-1K	SOTA	-	-	-	-	93 [7]

leakage concerning the privacy parameter ϵ for the same algorithm. The work of Zhao et al. [73] and Jarin et al. [35] similarly study the privacy-utility tradeoffs for different DP ML algorithms and evaluate them against membership inference attacks. There have also been some attempts at benchmarking DP algorithms [27, 61, 65]. Tao et al. [61] and Gong et al. [27] benchmark different synthetic data generation algorithms for tabular data and image data respectively. The work of Wei et al. [65] is closest to our work, where they benchmark different DPML algorithms on standard ML datasets such as MNIST/CIFAR-10 and comment on the effects of improvements made in DPML literature. In our work, we take a different stance than them and propose a new benchmark based on privacy-critical medical datasets. Compared to their work, we also experimented with more established architectures based on various techniques, such as Scatternets and CLIP-based models.

7 Future Work

While our work focused on image classification, future research should explore benchmarks in other areas such as Natural Language Understanding and Generation. In addition, to ensure fair comparisons, future work could investigate the use of more advanced model architectures. For instance, experiments using the NFNet-F7 [11] model pre-trained on ImageNet-1K could be compared with our Wide-ResNet experiments.

Future research should also extend to a wider range of datasets, both within and beyond the medical domain. This exploration will help in understanding the generalizability of DP ML techniques and identifying domain-specific challenges.

The continued maintenance and updating of the leaderboard we have established will be crucial for tracking long-term progress in the field and identifying emerging trends or breakthroughs. This ongoing effort will provide valuable insights into the evolution of DP ML techniques over time.

8 Conclusion

We suggest a number of standardized settings for benchmarking DP image classification, particularly with a focus on privacy-critical domains such as medical images. We also provide a leaderboard to help track progress on image classification benchmarks. In our experimental investigation, we find that several of the techniques which have enjoyed great success for DP ML are *not* universally effective across datasets and architectures, and furthermore that progress on standard benchmarks like CIFAR-10 do *not* transfer to medical images. Of course, it is hard and rare to design universally effective techniques. Indeed, our experiments are for a limited number of datasets and a limited number of architectures, so it is impossible to make a conclusion broad enough to encompass the entire field of DP image classification. However, it is clear that present work leaves the door open for new ideas and techniques that push the envelope on private image classification in these settings.

References

[1] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM Conference on Computer and Communications Security*, CCS '16, pages 308–318, New York, NY, USA, 2016. ACM.

[2] R. Anil, B. Ghazi, V. Gupta, R. Kumar, and P. Manurangsi. Large-scale differentially private BERT. *arXiv preprint arXiv:2108.01624*, 2021.

[3] M. Baradad Jurjo, J. Wulff, T. Wang, P. Isola, and A. Torralba. Learning to see by looking at noise. In *Advances in Neural Information Processing Systems 34*, NeurIPS '21, pages 2556–2569. Curran Associates, Inc., 2021.

[4] R. Bassily, V. Feldman, C. Guzmán, and K. Talwar. Stability of stochastic gradient descent on nonsmooth convex losses. *Advances in Neural Information Processing Systems*, 33:4381–4391, 2020.

[5] R. Bassily, A. Smith, and A. Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *Proceedings of the 55th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '14, pages 464–473, Washington, DC, USA, 2014. IEEE Computer Society.

[6] B. K. Beaulieu-Jones, Z. S. Wu, C. Williams, R. Lee, S. P. Bhavnani, J. B. Byrd, and C. S. Greene. Privacy-preserving generative deep neural networks support clinical data sharing. *Circulation: Cardiovascular Quality and Outcomes*, 12(7):e005122, 2019.

[7] L. Berrada, S. De, J. H. Shen, J. Hayes, R. Stanforth, D. Stutz, P. Kohli, S. L. Smith, and B. Balle. Unlocking accuracy and fairness in differentially private image classification. *arXiv preprint arXiv:2308.10888*, 2023.

[8] A. Bie, G. Kamath, and G. Zhang. Private GANs, revisited. *Transactions on Machine Learning Research*, 2023.

[9] A. Birhane and V. U. Prabhu. Large image datasets: A pyrrhic win for computer vision? In *2021 IEEE Winter Conference on Applications of Computer Vision*, WACV '21, pages 1536–1546. IEEE, 2021.

[10] A. Brock, S. De, and S. L. Smith. Characterizing signal propagation to close the performance gap in unnormalized resnets. *CoRR*, abs/2101.08692, 2021.

[11] A. Brock, S. De, S. L. Smith, and K. Simonyan. High-performance large-scale image recognition without normalization, 2021.

[12] T. Cao, A. Bie, A. Vahdat, S. Fidler, and K. Kreis. Don't generate me: Training differentially private generative models with sinkhorn divergence. In *Advances in Neural Information Processing Systems 34*, NeurIPS '21, pages 12480–12492. Curran Associates, Inc., 2021.

[13] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramer, B. Balle, D. Ippolito, and E. Wallace. Extracting training data from diffusion models. In *32nd USENIX Security Symposium*, USENIX Security '23, pages 5253–5270. USENIX Association, 2023.

[14] N. Carlini, F. Tramèr, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. Brown, D. Song, U. Erlingsson, A. Oprea, and C. Raffel. Extracting training data from large language models. In *30th USENIX Security Symposium*, USENIX Security '21, pages 2633–2650. USENIX Association, 2021.

[15] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(29):1069–1109, 2011.

[16] P. Chrabaszcz, I. Loshchilov, and F. Hutter. A downsampled variant of imagenet as an alternative to the CIFAR datasets. *CoRR*, abs/1707.08819, 2017.

[17] L. Chua, B. Ghazi, P. Kamath, R. Kumar, P. Manurangsi, A. Sinha, and C. Zhang. How private are DP-SGD implementations? In *Proceedings of the 41st International Conference on Machine Learning*, ICML '24. JMLR, Inc., 2024.

[18] S. De, L. Berrada, J. Hayes, S. L. Smith, and B. Balle. Unlocking high-accuracy differentially private image classification through scale. *arXiv preprint arXiv:2204.13650*, 2022.

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR '09, pages 248–255, Washington, DC, USA, 2009. IEEE Computer Society.

[20] T. Dockhorn, T. Cao, A. Vahdat, and K. Kreis. Differentially private diffusion models. *Transactions on Machine Learning Research*, 2023.

[21] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*. OpenReview.net, 2021.

[22] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor. Our data, ourselves: Privacy via distributed noise generation. In *Proceedings of the 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, EUROCRYPT '06, pages 486–503, Berlin, Heidelberg, 2006. Springer.

[23] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the 3rd Conference on Theory of Cryptography*, TCC '06, pages 265–284, Berlin, Heidelberg, 2006. Springer.

[24] F. Dörmann, O. Frisk, L. N. Andersen, and C. F. Pedersen. Not all noise is accounted equally: How differentially private learning benefits from large sampling rates, 2021.

[25] Eyepacs. Diabetic retinopathy detection, 2015. data retrieved from Kaggle, https://www.kaggle.com/c/diabetic-retinopathy-detection.

[26] M. Fredrikson, S. Jha, and T. Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 2015 ACM Conference on Computer and Communications Security*, CCS '15, pages 1322–1333. ACM, 2015.

[27] C. Gong, K. Li, Z. Lin, and T. Wang. Dpimagebench: A unified benchmark for differentially private image synthesis, 2025.

[28] S. Gopi, Y. T. Lee, and L. Wutschitz. Numerical composition of differential privacy. In *Advances in Neural Information Processing Systems 34*, NeurIPS '21, pages 11631–11642. Curran Associates, Inc., 2021.

[29] F. Harder, M. Jalali, D. J. Sutherland, and M. Park. Pre-trained perceptual features improve differentially private image generation. *Transactions on Machine Learning Research*, 2023.

[30] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition, 2015.

[31] N. Homer, S. Szelinger, M. Redman, D. Duggan, W. Tembe, J. Muehling, J. V. Pearson, D. A. Stephan, S. F. Nelson, and D. W. Craig. Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays. *PLoS Genetics*, 4(8):1–9, 2008.

[32] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning*, ICML '15, pages 448–456. JMLR, Inc., 2015.

[33] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. Ball, K. Shpanskaya, J. Seekins, D. A. Mong, S. S. Halabi, J. K. Sandberg, R. Jones, D. B. Larson, C. P. Langlotz, B. N. Patel, M. P. Lungren, and A. Y. Ng. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*, AAAI '19, pages 590–597, 2019.

[34] R. Iyengar, J. P. Near, D. Song, O. Thakkar, A. Thakurta, and L. Wang. Towards practical differentially private convex optimization. In *Proceedings of the 40th IEEE Symposium on Security and Privacy*, SP '19, pages 299–316, Washington, DC, USA, 2019. IEEE Computer Society.

[35] I. Jarin and B. Eshete. Dp-util: comprehensive utility analysis of differential privacy in machine learning. In *Proceedings of the Twelfth ACM Conference on Data and Application Security and Privacy*, pages 41–52, 2022.

[36] B. Jayaraman and D. Evans. Evaluating differentially private machine learning in practice. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 1895–1912, 2019.

[37] C. Jung, K. Ligett, S. Neel, A. Roth, S. Sharifi-Malvajerdi, and M. Shenfeld. A new analysis of differential privacy's generalization guarantees. In *Proceedings of the 11th Conference on Innovations in Theoretical Computer Science*, ITCS '20, pages 31:1–31:17, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[38] D. Kifer, A. Smith, and A. Thakurta. Private convex empirical risk minimization and high-dimensional regression. In *Proceedings of the 25th Annual Conference on Learning Theory*, COLT '12, pages 25.1–25.40, 2012.

[39] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*, ICLR '15, 2015.

[40] A. Koskela, J. Jälkö, and A. Honkela. Computing tight differential privacy guarantees using FFT. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, AISTATS '20, pages 2560–2569. JMLR, Inc., 2020.

[41] A. Krizhevsky. Learning multiple layers of features from tiny images, 2009.

[42] C. J. Lebeda, M. Regehr, G. Kamath, and T. Steinke. Avoiding pitfalls for privacy accounting of subsampled mechanisms under composition. *arXiv preprint arXiv:2405.20769*, 2024.

[43] Y. LeCun, C. Cortes, and C. Burges. MNIST handwritten digit database, 2010.

[44] X. Li, F. Tramèr, P. Liang, and T. Hashimoto. Large language models can be strong differentially private learners. In *Proceedings of the 10th International Conference on Learning Representations*, ICLR '22, 2022.

[45] I. Mironov. Rényi differential privacy. In *Proceedings of the 30th IEEE Computer Security Foundations Symposium*, CSF '17, pages 263–275, Washington, DC, USA, 2017. IEEE Computer Society.

[46] I. Mironov, K. Talwar, and L. Zhang. Rényi differential privacy of the sampled gaussian mechanism. *arXiv preprint arXiv:1908.10530*, 2019.

[47] E. Oyallon and S. Mallat. Deep roto-translation scattering for object classification. *CoRR*, abs/1412.8659, 2014.

[48] N. Papernot, A. Thakurta, S. Song, S. Chien, and Ú. Erlingsson. Tempered sigmoid activations for deep learning with differential privacy. In *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence*, AAAI '21, pages 9312–9321, 2021.

[49] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, NeurIPS '19, pages 8026–8037. Curran Associates, Inc., 2019.

[50] B. Polyak and A. Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30:838–855, 07 1992.

[51] N. Ponomareva, H. Hazimeh, A. Kurakin, Z. Xu, C. Denison, H. B. McMahan, S. Vassilvitskii, S. Chien, and A. G. Thakurta. How to DP-fy ML: A practical guide to machine learning with differential privacy. *Journal of Artificial Intelligence Research*, 77:1113–1201, 2023.

[52] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[53] M. Raghu, C. Zhang, J. Kleinberg, and S. Bengio. Transfusion: Understanding transfer learning for medical imaging, 2019.

[54] R. Redberg, A. Koskela, and Y.-X. Wang. Improving the privacy and practicality of objective perturbation for differentially private linear learners. In *Advances in Neural Information Processing Systems 36*, NeurIPS '23, pages 13819–13853. Curran Associates, Inc., 2023.

[55] C. Schuhmann, R. Beaumont, R. Vencu, C. W. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, P. Schramowski, S. R. Kundurthy, K. Crowson, L. Schmidt, R. Kaczmarczyk, and J. Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

[56] R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership inference attacks against machine learning models. In *Proceedings of the 38th IEEE Symposium on Security and Privacy*, SP '17, pages 3–18, Washington, DC, USA, 2017. IEEE Computer Society.

[57] G. Somepalli, V. Singla, M. Goldblum, J. Geiping, and T. Goldstein. Diffusion art or digital forgery? investigating data replication in diffusion models. In *Proceedings of the 2023 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR '23, pages 6048–6058. IEEE Computer Society, 2023.

[58] S. Song, K. Chaudhuri, and A. D. Sarwate. Stochastic gradient descent with differentially private updates. In *Proceedings of the 2013 IEEE Global Conference on Signal and Information Processing*, GlobalSIP '13, pages 245–248, Washington, DC, USA, 2013. IEEE Computer Society.

[59] K. Talwar, A. Thakurta, and L. Zhang. Private empirical risk minimization beyond the worst case: The effect of the constraint set geometry. *arXiv preprint arXiv:1411.5417*, 2014.

[60] X. Tang, A. Panda, V. Sehwag, and P. Mittal. Differentially private image classification by learning priors from random processes, 2023.

[61] Y. Tao, R. McKenna, M. Hay, A. Machanavajjhala, and G. Miklau. Benchmarking differentially private synthetic data generation algorithms, 2022.

[62] F. Tramèr and D. Boneh. Differentially private learning needs better features (or much more data). In *Proceedings of the 9th International Conference on Learning Representations*, ICLR '21, 2021.

[63] F. Tramèr, G. Kamath, and N. Carlini. Position: Considerations for differentially private learning with large-scale public pretraining. In *Proceedings of the 41st International Conference on Machine Learning*, ICML '24. JMLR, Inc., 2024.

[64] M. Voets, K. Møllersen, and L. A. Bongo. Reproduction study using public data of: Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *PLOS ONE*, 14(6):e0217541, June 2019.

[65] C. Wei, M. Zhao, Z. Zhang, M. Chen, W. Meng, B. Liu, Y. Fan, and W. Chen. Dpmlbench: Holistic evaluation of differentially private machine learning. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 2621–2635, 2023.

[66] Y. Wu and K. He. Group normalization. *CoRR*, abs/1803.08494, 2018.

[67] L. Xie, K. Lin, S. Wang, F. Wang, and J. Zhou. Differentially private generative adversarial network. *arXiv preprint arXiv:1802.06739*, 2018.

[68] A. Yousefpour, I. Shilov, A. Sablayrolles, D. Testuggine, K. Prasad, M. Malek, J. Nguyen, S. Gosh, A. Bharadwaj, J. Zhao, G. Cormode, and I. Mironov. Opacus: User-friendly differential privacy library in PyTorch. *arXiv preprint arXiv:2109.12298*, 2021.

[69] D. Yu, S. Naik, A. Backurs, S. Gopi, H. A. Inan, G. Kamath, J. Kulkarni, Y. T. Lee, A. Manoel, L. Wutschitz, S. Yekhanin, and H. Zhang. Differentially private fine-tuning of language models. In *Proceedings of the 10th International Conference on Learning Representations*, ICLR '22, 2022.

[70] Z. Yuan, Y. Yan, M. Sonka, and T. Yang. Large-scale robust deep auc maximization: A new surrogate loss and empirical studies on medical image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3040–3049, 2021.

[71] Z. Yuan, D. Zhu, Z.-H. Qiu, G. Li, X. Wang, and T. Yang. Libauc: A deep learning library for x-risk optimization. In *29th SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023.

[72] S. Zagoruyko and N. Komodakis. Wide residual networks. *CoRR*, abs/1605.07146, 2016.

[73] B. Z. H. Zhao, M. A. Kaafar, and N. Kourtellis. Not one but many tradeoffs: Privacy vs. utility in differentially private machine learning. In *Proceedings of the 2020 ACM SIGSAC Conference on Cloud Computing Security Workshop*, pages 15–26, 2020.

It's FREE to join!

TCDE

tab.computer.org/tcde/

The Technical Committee on Data Engineering (TCDE) of the IEEE Computer Society is concerned with the role of data in the design, development, management and utilization of information systems.

- Data Management Systems and Modern Hardware/Software Platforms
- Data Models, Data Integration, Semantics and Data Quality
- Spatial, Temporal, Graph, Scientific, Statistical and Multimedia Databases
- Data Mining, Data Warehousing, and OLAP
- Big Data, Streams and Clouds
- Information Management, Distribution, Mobility, and the WWW
- Data Security, Privacy and Trust
- Performance, Experiments, and Analysis of Data Systems

The TCDE sponsors the International Conference on Data Engineering (ICDE). It publishes a quarterly newsletter, the Data Engineering Bulletin. If you are a member of the IEEE Computer Society, you may join the TCDE and receive copies of the Data Engineering Bulletin without cost. There are approximately 1000 members of the TCDE.

Join TCDE via Online or Fax

ONLINE: Follow the instructions on this page:

www.computer.org/portal/web/tandc/joinatc

FAX: Complete your details and fax this form to **+61-7-3365 3248**

Name _____

IEEE Member # _____

Mailing Address _____

_____

Country _____

Email _____

Phone _____

TCDE Mailing List

TCDE will occasionally email announcements, and other opportunities available for members. This mailing list will be used only for this purpose.

Membership Questions?

Xiaoyong Du
Key Laboratory of Data Engineering and Knowledge Engineering
Renmin University of China
Beijing 100872, China
duyong@ruc.edu.cn

TCDE Chair

Xiaofang Zhou
School of Information Technology and Electrical Engineering
The University of Queensland
Brisbane, QLD 4072, Australia
zxf@uq.edu.au

IEEE Computer Society
10662 Los Vaqueros Circle
Los Alamitos, CA 90720-1314

Non-profit Org.
U.S. Postage
PAID
Los Alamitos, CA
Permit 1398