UNIVERSITY OF TARTU
Faculty of Science and Technology
Institute of Computer Science
Data Science Curriculum

Fedor Stomakhin

Framework for Privacy-Preserving Synthesis of Textual Data

Master's Thesis (15 ECTS)

Supervisor(s): Sven Laur, DSc (Tech)
Liina Kamm, PhD

Tartu 2025

Framework for Privacy-Preserving Synthesis of Textual Data

Abstract: To safeguard patient privacy, sharing medical record data for research must adhere to various privacy regulations. To facilitate data sharing, various data protection techniques have been proposed, such as pseudonymization, anonymization and the use of synthetic data. The aim of synthetic data generation is, based on an original dataset, to produce a new dataset in a way that preserves the statistical relationships within the original data while not exposing any identifying or sensitive information about the data subjects therein.

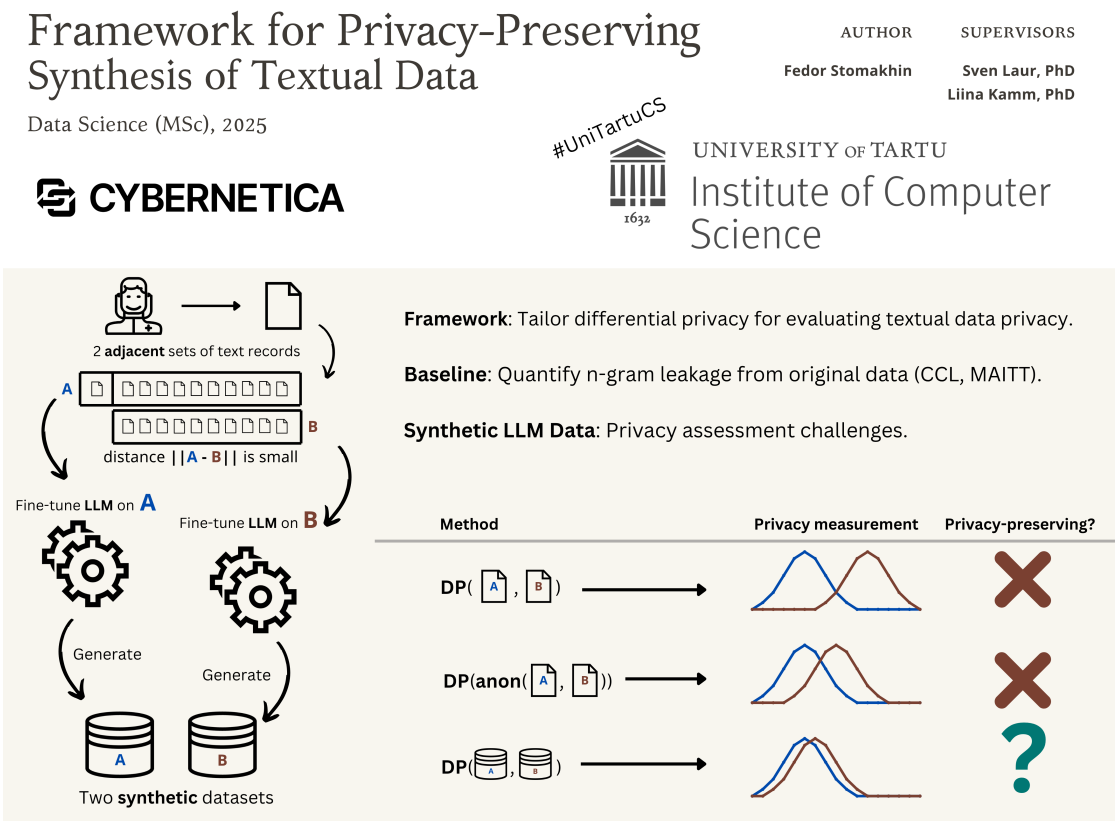
Synthetically generated data can still be insufficient from the point of view of privacy-preservation. For this purpose, approaches rooted in differential privacy (DP) have been proposed. DP typically relies on worst-case assumptions about attackers' knowledge, potentially leading to overly conservative measures. Applying DP principles to free-form text, such as medical epicrises, is complicated by their high dimensionality and complexity, as the same information can be conveyed in many different ways.

In this work, motivated by the challenges of sharing textual health data, we propose and apply a general framework for evaluating privacy risks in text generated by large language models (LLMs). Considering a journalist attack model, we adapt differential privacy principles, quantifying privacy loss (ε, δ) based on the outputs of specific attack functions rather than relying on worst-case assumptions of DP. We demonstrate the framework by establishing baseline privacy characteristics via direct n-gram sampling analysis on both medical and social media texts and by exploring membership inference signals using surprisal analysis on LLMs fine-tuned with social media texts. While assessing synthetic data from standard LLMs highlighted methodological challenges, the framework provides a methodology for evaluating the privacy properties of text generation models and their outputs, informing decisions on sharing such data for research purposes.

Keywords:
Synthetic data, differential privacy, electronic medical records, natural language processing, large language models.

CERCS: P160, P170, P176

Visual abstract:



Raamistik tekstiandmete privaatsust säilitavaks sünteesiks

Lühikokkuvõte: Patsientide privaatsuse kaitsmiseks peab terviseandmete jagamine teadusuuringuteks vastama erinevatele privaatsusregulatsioonidele. Selle protsessi hõlbustamiseks on välja pakutud mitmeid andmekaitsetehnikaid, nagu pseudonüümimine, anonüümimine ja sünteetiliste andmete kasutamine. Sünteetiliste andmete genereerimise eesmärk on algandmete põhjal luua uusi andmeid viisil, mis säilitab algsete andmete statistilised seosed, kuid ei avalda andmesubjekte isikustavat ega nende tundlikku teavet.

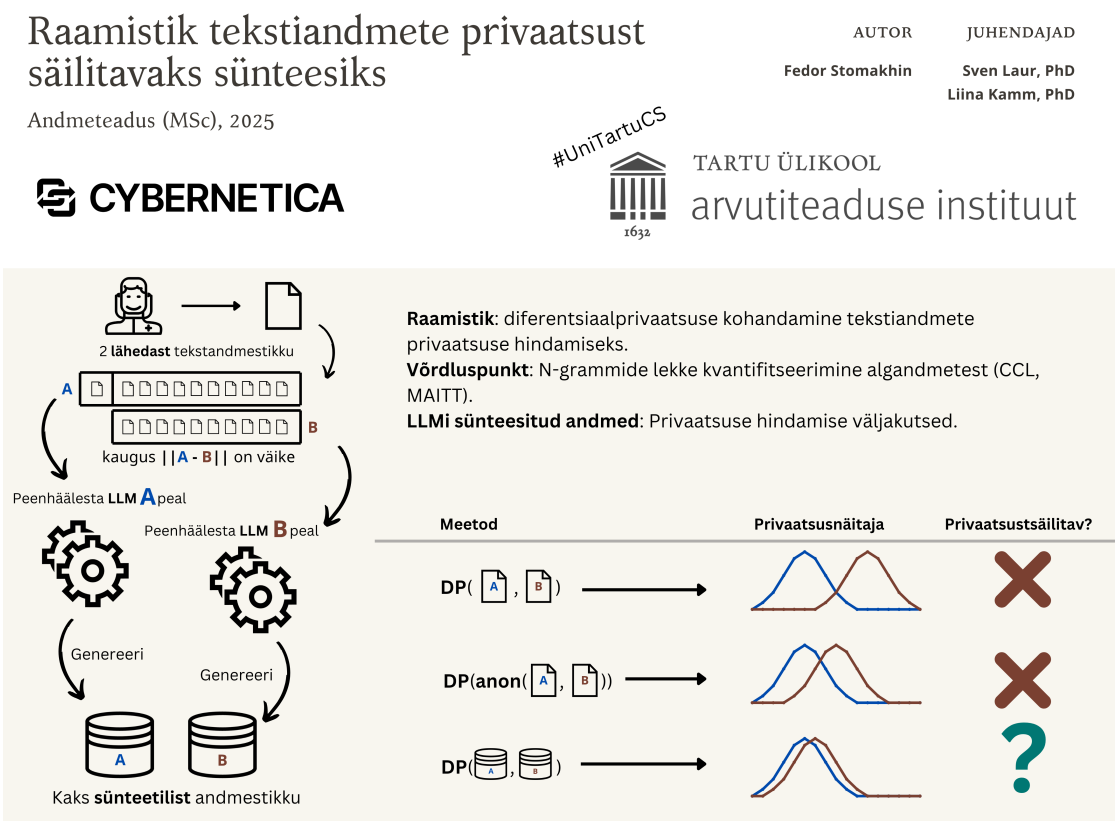
Andmesüntees võib siiski jääda privaatsuskaitse seisukohast ebapiisavaks. Seetõttu on välja pakutud diferentsiaalprivaatsusel (DP) põhinevaid lahendusi. DP lähtub reeglina ründaja teadmuse halvima juhu eeldustest, mis võib viia ülemäära konservatiivsete privaatsusmeetmeteni. Vabavormiliste tekstide, näiteks epikriiside kõrge dimensionaalsuse ja hõreduse tõttu on nendele DP põhimõtete rakendamine keeruline, kuna ühte ja sama teavet saab esitada erinevatel viisidel.

Lähtudes terviseandmetena esinevate tekstide jagamisega seotud väljakutsetest, pakume käesolevas töös välja ja rakendame üldist raamistikku suurte keelemudelite (LLM) genereeritud tekstide privaatsusriskide hindamiseks. Rakendame DP põhimõtteid ajakirjaniku ründemudeli korral, kvantifitseerides privaatsuskadu (ε, δ) spetsiifiliste ründefunktsioonide väljunditel, selle asemel et tugineda üldisematele DP halvima juhu eeldustele. Näitlikustame raamistikku defineerides privaatsuse baastasemed n-gramide otsevalimisel põhineva analüüsiga nii meditsiini- kui sotsiaalmeediatekstidel. Samuti uurime liikmelisuse järeldamise signaale peenhäälestatud LLM-ide puhul üllatuslikkusel põhineva analüüsiga sotsiaalmeediatekstidel peenhäälestatud LLM-idel. Ehkki LLM-ide sünteesitud andmete hindamine tõi esile metodoloogilised väljakutsed, pakub raamistik metoodika hindamaks tekstisünteesimudelite ja nende väljundite privaatsusomadusi, aidates teha otsuseid selliste andmete jagamise kohta teadustöö otstarbeks.

Võtmesõnad:
Sünteetilised andmed, diferentsiaalprivaatsus, meditsiiniandmed, loomuliku keele töötlus, suured keelemudelid.

CERCS: P160, P170, P176

Visuaalne abstrakt:



Contents

1 Introduction

The advent of data protection regulations, such as GDPR, has raised new challenges for the sharing of special categories of data, such as health data for research purposes. Data sharing enables larger sample sizes, which improves research quality, but also poses risks to patient privacy. These risks are usually mitigated with various privacy-enhancing technologies (PETs). An approach to sharing data in a way that does not violate the rights of data subjects could be the use of synthetic data [GGS23]. Synthetic data is data that has been generated in such a way as to retain the desirable properties and distributions of the original data, while not exposing the sensitive personal information contained therein. The utility of synthetic data – the ability to make meaningful use of it – depends on whether it retains these statistical properties. This utility must be balanced against the risk of inference of sensitive and personally identifiable information contained in the original data.

The scale of data required to train large language models (LLMs) makes it difficult to ensure that all data has been obtained and utilized ethically and legally. Recent legal cases highlight concerns surrounding data protection and copyright law in the era of LLMs. For instance, the New York Times (NYT) vs. OpenAI and Microsoft lawsuit [TMO23] centers on allegations that the companies illegally used the newspaper's content to train their AI models without obtaining proper licensing or consent. The initial evidence presented by the NYT includes GPT-4 outputs that match the copyrighted NYT content verbatim. Such outputs prompt considerations about what can be deduced regarding the presence of these strings in LLM training data.

Research into LLM scaling laws [AZL24] has shown that LLMs possess significant memorization capabilities, further raising privacy concerns. LLMs have been shown to sometimes generate verbatim segments from their training data [MSL+21, NCH+23, XFZ+23]. This phenomenon might be seen as evidence of overfitting, though such discussions are less relevant for intentionally overparameterized LLMs [RM22, SKR+23].

To mitigate the risk of LLMs leaking private information from their training data, pseudonymization of LLM outputs has been proposed. However, pseudonymization methods may not fully mask implicit information usable for inferring sensitive details [SVBV23], and stronger pseudonymization tends to degrade utility [EEA13]. Moreover, output pseudonymization is ineffective if the attacker has direct access to the model.

We must also differentiate between the generation of text segments highly similar to the training data and the actual leakage of private information. Generation of text segments present in the training data might simply be a result of the model's ability to generalize, rather than of memorization. This highlights the need for more robust quantitative metrics to assess the risk of personal information disclosure. Drawing inspiration from cryptography, where adversarial indistinguishability experiments are used to evaluate security, we approach the need for metrics from a differential privacy (DP) perspective, in a manner similar to counterfactual memorization[ZIL+23].

Traditional DP offers strong guarantees against worst-case attackers with potentially complete knowledge of the dataset, which can lead to overly conservative measures. Realistic attackers, particularly in the health data context, often have limited knowledge, such as information about a few individuals (a journalist attack model [AFM+22]) and general population statistics. Such attackers might attempt membership inference attacks, even with access only to synthetic data [GMCdM23] or by probing the model directly.

This thesis proposes and applies a framework for evaluating privacy risks under such assumptions. A key aspect of our approach is to reframe privacy guarantees by explicitly considering the attacker's knowledge and capabilities, moving beyond standard worst-case assumptions of DP. We adapt DP concepts by analyzing the logarithmic Bayes factor (LBF) computed on the outputs of specific, concrete attack functions, yielding quantifiable privacy loss (ε) and failure probability (δ) bounds relevant to those attacks. This framework is applied to: (1) Establish privacy baselines by analyzing direct n-gram sampling attacks on both social media (CCL) and medical text (MAITT) datasets. (2) Evaluate a surprisal-based membership inference attack targeting fine-tuned language models (on CCL). (3) Assess the privacy of LLM-generated synthetic data, highlighting significant methodological challenges related to evaluating stochastic generative processes.

The thesis is structured as follows: Section 2 covers preliminaries and related work. Section 3 details the proposed privacy metrics based on differential privacy and attack functions. Section 4 establishes baseline privacy bounds using a direct sampling attack on the datasets. Section 5 presents the analysis of LLM-generated synthetic data and discusses the associated evaluation challenges. Section 6 evaluates a surprisal-based membership inference attack. Section 7 interprets the findings and limitations, and Section 8 concludes the work. Various AI-based tools, such as Claude and Google AI Studio were used to organize thoughts and edit this work.

2 Preliminaries and related work

Sharing datasets for research while safeguarding the privacy of individuals is a persistent challenge. The difficulty lies in the elusive nature of privacy itself. What constitutes sensitive or personally identifiable information can be context-dependent and often only becomes apparent after a successful attack, as illustrated by the AOL search log release in 2006 [AOL06]. The company, aiming to contribute to research, published anonymized search logs of over 650,000 users. However, cross-referencing the allegedly anonymized data with phonebook listings enabled the identification [BZ06] of specific individuals and revealed personal details [MMO22] about their lives, highlighting the limitations of simple de-identification techniques.

Initial attempts to mitigate privacy risks often relied on legal contracts and agreements between data providers and researchers, restricting data usage and imposing confidentiality obligations. Although effective in limited settings, this approach quickly breaks down as the number of actors involved increases. The infamous case of Edward Snowden leaking classified information from the National Security Agency (NSA) [Gre13] exemplifies this vulnerability. As a contractor with high-level security clearance, Snowden exposed thousands of classified documents revealing global surveillance programs, demonstrating how even the most secure organizations can be compromised by insiders. Even with strict legal frameworks and internal controls, determined individuals can bypass safeguards, underscoring the need for more robust, mathematically grounded approaches to privacy protection.

To address this need, researchers have turned to privacy-enhancing technologies [SRTP+21, MPP+23, SY24] (PETs), aiming to quantify privacy risks and provide rigorous guarantees against potential attacks. These technologies encompass a wide spectrum of methods, from pseudonymization and data perturbation techniques to more sophisticated cryptographic protocols and privacy-preserving machine learning algorithms. This thesis explores the potential of differential privacy to quantitatively assess and mitigate privacy risks in the context of synthetic text data generated by LLMs.

2.1 De-identification and anonymization

De-identification is an umbrella term for various techniques that alter the base data in such a way as to separate a sensitive record from the data subject it pertains to. The simplest de-identification approach is pseudonymization [Pom94, PR04], where uniquely identifying information is replaced with another unique placeholder value. However, as the placeholder values can be re-linked to identifying values then this might be insufficient from the point of view of data protection for certain use-cases.

A more rigorous approach is k-anonymity [Swe02], which ensures that each record in a dataset is indistinguishable from at least $k-1$ other records with respect to attributes that indirectly identify a person (quasi-identifiers). While k-anonymity provides some

protection against identity disclosure, it may still be vulnerable to attribute disclosure attacks. For example, if all k records sharing the same quasi-identifiers also share the same sensitive attribute (e.g., a medical diagnosis), an attacker can still infer sensitive information about an individual even without uniquely singling them out.

To address this limitation, ℓ-diversity [MKGV07] extends k-anonymity by requiring that each group of records with the same quasi-identifiers have at least ℓ unique values for sensitive attributes. This provides stronger protection against attribute disclosure attacks, though it may still be vulnerable to other types of attacks.

In this case the identifying information might be replaced with less granular, non-unique placeholder values or removed altogether, however, the unique combinations of other data points still leave the possibility for re-identification of data subjects [Swe02, MHVB13, MRSP15, SVBV23, YRC23]. De-identification techniques usually degrade the utility of the data, and the stronger the privacy guarantees provided by a technique are, the more useful information usually tends to be lost [EEA13, IKL$^+$24].

2.2 Differential privacy

Differential privacy (DP), introduced by Dwork et al. [Dwo06], emerged as a response to the limitations of traditional de-identification techniques. Unlike approaches that focus on removing or obscuring identifying information, DP provides a mathematical framework for quantifying and bounding privacy loss. They key insight of DP is that it ensures that the presence or absence of any individual record cannot significantly affect the output of an analysis. Formally, a randomized algorithm M provides (ε, δ)-differential privacy for all datasets $\mathcal{X}$ and $\mathcal{Y}$ differing in at most one record, and for all possible outputs S, if:

$$\Pr\left[M(\mathcal{X}) \in S\right] \leq \exp\left(\varepsilon\right) \cdot \Pr\left[M(\mathcal{Y}) \in S\right] + \delta \ ,$$

where ε represents the privacy budget and δ allows for a small probability of privacy breach. Higher values of ε generally result in better utility but weaker privacy guarantees, while stricter privacy requirements (lower ε) often lead to noisier, less useful outputs.

2.3 Machine learning and neural networks

Neural networks are computational models inspired by biological neural systems that learn to perform tasks through pattern recognition in data. At their core, they consist of interconnected nodes (neurons) organized in layers, where each connection has an associated weight that is adjusted during training. The input layer receives raw data, which is then processed through one or more hidden layers, before producing output through the final layer. Each neuron computes a weighted sum of its inputs, applies a non-linear activation function (such as ReLU or sigmoid), and passes the result to the next layer. During training, the network adjusts its weights through backpropagation,

minimizing a loss function that measures the difference between predicted and actual outputs. This process allows neural networks to learn complex, non-linear relationships in data, making them useful tools for tasks ranging from image recognition to natural language processing. The specific architecture of a neural network, including the number and size of layers, types of connections, and choice of activation functions, is typically tailored to the particular problem being solved [GBC16].

2.4 Generative models and synthetic data

Synthetic data provides another way to share medical data without compromising patient privacy [EEH19, EE23, GGS23]. By using a model to generate a synthetic dataset that retains the statistical relationships of the base data, the utility of the base data is retained while seemingly preserving privacy. However, a new question is raised – whether the occurrence of certain data points from the base dataset in the synthetic output data is evidence of their memorization by the model or exposure to them during training. It has been shown that membership inference attacks against are possible for some types of synthetic datasets [GMCdM23]. Moreover, research on the evaluation of privacy metrics of synthetic data has generally focused on synthetic tabular data [YDvdS20], not synthetic free-form texts.

In LLMs, temperature and the sampling parameters are used to govern the stochasticity of the output generation process, directly impacting the distributional properties of the generated text. Temperature controls the spread of the probability distribution of the model's output tokens, with higher temperatures making the distribution more uniform. Higher temperature values promote exploration and less predictable outputs, while lower values yield more deterministic results. Top-k and top-n sampling restrict the model's word selection to the top k or top n% most probable words, respectively. Deterministic output is achieved with top-k sampling when k = 1, forcing the model to always select the single most likely word. This eliminates randomness but also potentially reduces creativity and diversity.

These parameters have direct implications for privacy. Higher temperature and less restrictive sampling methods introduce more randomness, potentially masking sensitive information embedded in the model's parameters. Conversely, deterministic output could inadvertently reveal private details if the model has memorized sensitive patterns from its training data.

2.5 Direct memorization of data

In classifiers, memorization occurs when a classifier can reproduce all class labels on the training data, resulting in zero training error. A model class is susceptible to memorization if it is complex enough to represent all possible binary labelings on some set of training data. This susceptibility is closely linked to the concept of Vapnik-Chervonenkis (VC)

dimension. The VC dimension measures the cardinality of the largest set of points that a model class can always classify correctly, regardless of the particular binary labels of these points. This ability to classify any labeling of a set of points is known as shattering the set.

The concept of memorization in classifiers does not directly apply to generative models, because generative models generate new data samples rather than merely assigning labels to existing ones. In other words, classifiers model the conditional probability of a label given a data point, whereas generative models aim to capture the entire data distribution.

Memorization, roughly defined as learning that requires only little generalization [HSB+23] (in turn a vaguely defined notion) can lead to verbatim reproduction of text segments from the training corpus, potentially revealing personally identifiable information (PII) or copyrighted material. Research has been done on methods to quantify and mitigate memorization.

The memorization problem in generative models, in particular LLMs can be illustrated using the example of New York Times vs. OpenAI [TMO23], where OpenAI was accused of illegally using the newspaper's content to train their AI models. The additional complexity of this case stems from the possible use of specific prompts used by the New York Times to extract the data, which narrow the focus to particular regions of the data distribution. This targeted querying can increase the risk of reproducing training data verbatim. This court case as well as the wider adoption of large language models have put the memorization capabilities of LLMs in the spotlight, both from the point of view of model capability and privacy of the training data.

LLM memorization is influenced by model architecture, training regime, generation parameters, and extraction techniques [SVT25]. Some architectures may be more prone to memorization than others. Increased repetition of sequences during training leads to higher memorization [ZXCS23], while diverse and unique training datasets mitigate this. Techniques like goldfish loss [HWJ+24], which drop tokens during training, could reduce memorization. Higher temperature values during generation decrease the likelihood of verbatim reproduction of memorized sequences [CTW+21].

In a black-box access model, it is not possible to infer the model's knowledge by looking at the weights. In a white-box access model, the extraction of information from a model via its weights is also nontrivial. Therefore, metrics of extractability and discoverability [CTW+21, NCH+23] have been developed – the extent to which some data can be extracted from or discovered inside the model, usually via querying the model itself.

From the point of view of model capability, researchers have tried to quantify the relationship between a model's ability to memorize its training data and its architecture, training regime and training data content. In one study [AZL24], the capacity of an LLM to memorize its training data depending on training regime and model architecture was

evaluated as a ratio of the training data information content to the model's parameter count. The models were trained on synthetic text data with known information content for 100 or 1000 epochs and then evaluated for the extractability of their training data by prompting the resulting pre-trained models. The models, mostly regardless of architecture saturated at about 2 bits per parameter memorized after 1000 epochs, or 1 bit per parameter after 100 epochs.

A study by Carlini et al. [CTW+21] demonstrated a decaying temperature approach to extract memorized content from LLMs. The strategy involved starting with a high temperature to encourage exploration and potentially land on a memorized sequence, then decreasing the temperature to fixate on that sequence. From this we can intuit that a consistently low temperature would result in predictable, potentially memorized outputs, while a consistently high temperature generates diverse outputs, unlikely to reveal specific memorized segments. A decaying temperature strategy, however, offers a balance between exploration and fixation, first spreading out and then potentially generating diverse memorized examples.

The development of prompting techniques for evaluating memorization is a separate line of research. These techniques range from prompting the model with prefixes obtained from known training data to more sophisticated adversarial attacks. These prompting techniques are often evaluated using a metric of extractability or discoverability.

Prefixes, in the context of LLMs are the initial sequences of tokens that are given as input to the model before the main instruction. However, there are also output prefixing techniques, where the beginning of the model's output is pre-filled by the user to steer its subsequent response in some direction. Prefix matching is a technique for detecting memorization in LLMs by making use of the overlap between the input prefixes and the training data. Exact and partial prefix matching aim to identify instances in which the model reproduces memorized sequences verbatim or with heightened probability when prompted with matching or partially matching prefixes [NCH+23, WMW+24]. Prefix-based sampling more generally explores the model's tendency to fall into memorized patterns by analyzing the diversity of outputs generated from the same prefix under different sampling conditions.

Adversarial prefixes exploit the model's sensitivity to input perturbations by systematically developing prefixes that maximize the likelihood of triggering the synthesis of desired content. This can involve gradient-based optimization or search algorithms to identify prefixes that steer the model towards desired (including memorized) outputs. Adversarial prefixing is a technique frequently used for LLM jailbreaking [ZZA+23], in which a model is steered to give unintended or unauthorized responses. Similar techniques exist using suffixes [ZWC+23].

Extending the notion of extractability, a notion of counterfactual memorization [ZIL+23] has been proposed, which helps to evaluate the memorization properties of a model architecture by training two models on training datasets differing only by the

extraction target. Then, the difference in memorization and extractability metrics of the two models could provide a counterfactual answer to whether the model architecture and training regime result in privacy leakage or not.

Finally, ways to alter the training regime of LLMs to memorize less and generalize more have been proposed, such as goldfish loss [HWJ+24], which selectively ignores the loss for certain contextually selected tokens. This approach is conceptually related to regularization and dropout, both of which aim to prevent overfitting by introducing noise or constraints during training. While dropout randomly drops neurons during training, goldfish loss randomly excludes the contribution of individual tokens from the summary loss calculation, forcing the model to rely less on memorizing specific sequences and encouraging generalization. In other words, dropout operates in neuron space and the goldfish loss operates in token space. Like L_2 regularization, which adds a penalty equal to the square of the magnitude of coefficients to the loss function, goldfish loss indirectly penalizes complex models, forcing the model to consider a broader range of possibilities. However, while L_2 regularization results in a smoother decision boundary (a more continuous and less jagged hypersurface that separates different classes in the feature space), it is unclear if goldfish loss causes a smoother loss surface (a landscape of the loss function with fewer sharp local minima). On the one hand, the random exclusion of tokens might prevent the development of sharp local minima related to these tokens. On the other hand, the stochasticity introduced by the random exclusion of tokens might make the model's optimization trajectory more erratic.

Besides analyzing the privacy properties of existing models or mechanisms, techniques exist to train models with inherent DP guarantees. Differentially private stochastic gradient descent (DP-SGD) [ACG+16] modifies the training process via gradient clipping and noise addition, yielding model parameters that satisfy a cumulative (ε, δ)-DP guarantee calculated over the training steps. The relationship between the privacy of these parameters and the privacy of data subsequently generated from them is discussed in Section 3.9. Alternative approaches like private prediction [ABK+24] aim to provide DP guarantees for generation process (inference) itself.

In [BCH22], the authors investigate the potential for reconstructing training data from neural networks, focusing on a powerful informed adversary who knows all training data points except one, mirroring the implicit threat model of differential privacy. The authors demonstrate the feasibility of reconstructing the remaining data point in this scenario, both theoretically for convex models and experimentally for neural networks. For convex models like logistic regression, they derive closed-form reconstruction attacks. For more complex models like neural networks, they introduce reconstructor networks (RecoNNs), trained by the adversary to map model parameters to the target data point. Notably, this assumes white-box access to the model.

They showcase the effectiveness of their attack on image classifiers, highlighting the capacity of these models to store information enabling high-fidelity reconstruction. The

study further analyzes factors impacting attack success, including model architecture, training hyperparameters, and adversary knowledge. They find that training models with differential privacy (DP-SGD) using a sufficiently large privacy budget epsilon can effectively mitigate these attacks. The paper concludes that standard ML models can memorize enough information to permit reconstruction under strong adversarial knowledge.

3 Privacy metrics for synthetic data

The processing and sharing of sensitive data involves inherent privacy risks, particularly when applying machine learning to generate synthetic versions of this data. While synthetic data can theoretically preserve statistical properties without exposing individual records, we need rigorous ways to quantify the privacy guarantees it provides. Traditional privacy measures like k-anonymity were designed for direct data sharing, not for the complex information capture that occurs during model training and generation. A model trained on sensitive data might leak this information in subtle ways, from generating exact copies of training examples to preserving statistical patterns that enable inference of private attributes.

The processing of personally identifiable information (PII) carries significant privacy risks if the data or its derivatives are leaked. To address these risks, privacy-preserving approaches such as k-anonymity [Swe02] were introduced. k-anonymity requires that each unique combination of data features (columns) representing PII must include at least k data entries (rows), ensuring that no individual can be uniquely identified. However, this approach fails to protect against inference attacks when sensitive attributes lack diversity.

The 2006 AOL search history breach [AOL06] exposed the practical consequences of insufficient anonymization. AOL released "anonymized" search logs by replacing user IDs with random numbers, but journalists cross-referenced search patterns with public records to re-identify individuals, including a 62-year-old widow whose medical history and real estate interests were revealed [BZ06]. AOL had not implemented any privacy-preserving measures beyond pseudonymization. The breach demonstrated the failure of simple anonymization techniques and the risks of attribute homogeneity, leading to public backlash, the resignation of AOL's CTO, and the withdrawal of the dataset.

To address the limitations of k-anonymity, researchers introduced the notion of ℓ-diversity [MKGV07], which requires that within each group of records sharing the same quasi-identifiers, there must be at least ℓ distinct values for each sensitive attribute. By ensuring diversity in sensitive attributes, ℓ-diversity reduces the risk of inferring specific attribute values, even when k-anonymity is satisfied. However, ℓ-diversity is not foolproof: adversaries can still exploit background knowledge about global or group-specific attribute distributions to infer probabilistic information about individuals. Furthermore, stricter k-anonymity and ℓ-diversity requirements often degrade the utility of the data, as they necessitate more aggressive generalization or suppression of information.

Additionally, both k-anonymity and ℓ-diversity are designed for publishing microdata records and do not provide guidance for releasing aggregate statistics, such as group medians or averages. A different framework is needed to reason about the privacy implications of such summary statistics.

The limitations of k-anonymity and ℓ-diversity motivated the development of more robust privacy frameworks. Differential privacy (DP) [DMNS06] emerged as a gold

standard, providing mathematically rigorous guarantees against membership inference attacks. Unlike earlier approaches, DP does not rely on assumptions about attribute distributions or adversary background knowledge – instead, it quantifies privacy loss through a worst-case analysis of data influence. However, standard DP implementations often require adding substantial noise to achieve these guarantees, particularly when applied to complex data types like text. This noise injection can degrade data utility and may be unnecessarily conservative when the data generation process itself contains inherent randomness.

In this work, we address three limitations of conventional DP approaches:

1. We reformulate DP for settings where data generation already introduces randomness (e.g., through stochastic sampling in language models), enabling tighter privacy bounds without additional noise injection

2. We quantify the background knowledge required for effective attacks, moving beyond worst-case assumptions to provide practical privacy guarantees

3. We characterize how privacy guarantees strengthen when adversaries have limited information compared to theoretical worst-case scenarios

Our approach focuses on achieved privacy rather than prescribed noise levels. This proves particularly relevant for synthetic text generation, where the sampling process naturally introduces uncertainty that can provide privacy benefits. Rather than following the conventional DP paradigm of adding noise to guarantee privacy, we develop methods to quantify the privacy already afforded by existing system randomness. This avoids the utility degradation typically associated with DP while still providing formal guarantees against membership inference attacks.

Traditional differential privacy evaluates security by considering the attacker's probability of success over all possible pairs of adjacent datasets. This model assumes perfect adversarial knowledge, including complete understanding of the dataset and its distribution. While this provides objective guarantees, it may be overly conservative in practice. These limitations motivate our focus on explicit adversary modeling.

Our key insight is to reframe privacy guarantees around the attacker's knowledge. We ask: what information must an attacker possess to perform optimal attacks? What privacy guarantees can we provide when the attacker's knowledge is practically limited? To ground this, we examine membership inference attacks, where an adversary tries to determine whether specific records were used to generate the output.

Synthetic data generation introduces three natural obfuscation layers between training data and outputs. First, the model training process itself involves randomization through initialization and optimization. Second, the generation process uses random sampling to produce outputs, controlled by parameters like temperature. Finally, even the original data may contain inherent noise or uncertainty. These layers of randomness make it

difficult for an adversary to make definitive claims about the presence or absence of specific records in the original training data based on synthetic outputs.

3.1 Membership inference attacks

When k-anonymity and ℓ-diversity are insufficient, we can turn to differential privacy (DP). DP provides a formal mathematical framework for bounding how much information an adversary can gain about any individual by observing the output of a computation on a dataset. For instance, if we have a medical study of HIV patients, DP ensures that even if an adversary knows everything about all other participants, observing the study's outputs will not significantly increase their confidence about whether a specific person participated or what their HIV status might be. This bound on information gain protects against membership inference and many other types of attacks.

Consider a scenario where HIV prevalence in a medical study group is twice that of the general population. If an adversary can definitively determine whether someone participated in the study, they can immediately update their probability estimate of that person having HIV – even without directly accessing any health records. This illustrates how indistinguishability of membership maps to semantic security properties – if datasets with and without an individual are indistinguishable, we also protect the sensitive attributes that could be inferred from membership.

While analyzing full semantic security implications is out of scope for this work, we formalize our HIV example and similar scenarios through the framework of differential privacy. Specifically, we focus on two types of attacks: membership inference (MIA), where an adversary tries to determine if a target individual's record was used to generate the output, and attribute inference, where they attempt to learn protected attributes. The indistinguishability of DP bounds MIA success directly [TSBP22] and provides guarantees against learning additional information beyond population statistics.

Let us consider first a very simplistic scenario where there is a universe $\mathcal{U}$ of possible elements (e.g., individuals or records). A dataset, such as $\mathcal{X}$ or $\mathcal{U}$ is then a multiset of elements drawn from $\mathcal{U}$. Let $\mathfrak{D}$ denote the set of all such possible datasets that can serve as valid inputs to a mechanism. Let there be a randomized data aggregation routine (mechanism) $M(\cdot)$ that takes in the entire dataset and provides some output, for example estimates some averages. Let

$$\mathsf{Range}(M) = \{o \mid o \leftarrow M(\mathcal{X}), \mathcal{X} \in \mathfrak{D}\}$$

denote the set of all possible outputs that can be produced by the mechanism $M(\cdot)$.

In our context, we are interested in generative machine learning models. In such a case, we can split our mechanism into a training algorithm

$$\mathsf{Train} : \mathfrak{D} \to \Theta$$

that learns model parameters from a parameter space Θ, and a generation algorithm

$$\mathsf{Generate} : \Theta \times \Omega \to \mathcal{O}_{\text{element}}$$

which takes the learned parameters θ and an element ω from a random noise space Ω, and produces a single synthetic data element o. The space $\mathcal{O}_{\text{element}}$ is the set of all possible individual synthetic elements the generator can produce.

Data synthesis can be viewed as a particular data aggregation mechanism M implemented using this model pair. First, we train on input data $\mathcal{X}$ to obtain parameters

$$\theta \leftarrow \mathsf{Train}(\mathcal{X})$$

Then, data elements are generated using these parameters and random noise $\omega \in \Omega$:

$$o \leftarrow \mathsf{Generate}(\theta, \omega)$$

Following Kerckhoffs' principle, we must assume an adversary has complete knowledge of the Train and Generate algorithms, treating only the trained parameters θ as private. This principle emphasizes that security should not rely on obscurity – only the secrecy of the key (in this case, θ) ensures protection. This helps us analyze realistic threats – an adversary would likely know what model architecture and training process we used, but gathering information about model weights requires significant effort and access.

This setting raises two distinct privacy concerns: we must consider both the differential privacy of Train producing θ, and potential privacy leakage from synthetic outputs $o_1, \ldots, o_n \leftarrow \mathsf{Generate}(\theta, \omega)$. The leakage through syntetic outputs becomes critical as n increases. With more synthetic samples, an adversary gains additional information to approximate $\mathsf{Generate}(\theta, \cdot)$ and thus learn something about θ.

However, even if an attacker sees many synthetic samples, they might not recover the exact parameters θ. This is because many different parameter sets can produce identical outputs – for example, in a neural network, swapping two neurons and their connections can result in the same function. Still, if an attacker can find any parameter set that perfectly mimics the behavior of θ, the privacy guarantees ultimately depend on how private the training process itself was.

Given that the adversary has complete knowledge of the data generation mechanism $M(\cdot)$, as well as full access to both possible input datasets $\mathcal{X}$ and $\mathcal{Y}$, they can use Bayes' formula to update their beliefs about which dataset was used for training. Let o be a synthetic output observed by the attacker. For these fully known datasets $\mathcal{X}$ and $\mathcal{Y}$, the attacker can compute posterior probabilities:

$$\Pr[\mathcal{X}|o] = \frac{\Pr[M(\mathcal{X}) = o]}{\Pr[o]} \cdot \Pr[\mathcal{X}]$$
$$\Pr[\mathcal{Y}|o] = \frac{\Pr[M(\mathcal{Y}) = o]}{\Pr[o]} \cdot \Pr[\mathcal{Y}]$$

where $\Pr[\mathcal{X}]$ and $\Pr[\mathcal{Y}]$ represent the adversary's prior beliefs about which dataset was used for creating the output. The attacker's prior beliefs are subjective, they depend on prior background knowledge. For instance, in the context of genetic data, an attacker might have background knowledge about any of the following, which could be useful in membership inference:

- the likelihood of specific genetic variants in certain populations, paired with the target's demographic profile;
- if the target has not visited a doctor in a long time, they are less likely to be in medical research datasets;
- the target's public genealogical records or family medical history;
- the geographic distribution of genetic donors.

Such information helps attackers form concrete probability estimates about dataset membership before seeing any outputs.

This setup follows standard statistical analysis of hypothesis testing. If $\Pr[\mathcal{X}|o] \approx \Pr[\mathcal{X}]$ and $\Pr[\mathcal{Y}|o] \approx \Pr[\mathcal{Y}]$ then observing the synthetic output o provides little information to the attacker, as it fails to significantly alter their prior beliefs.

To reduce technical complexity, we consider the ratio of posterior beliefs where the denominator $\Pr[o]$ is factored out. Let $\beta(x)$ denote the posterior odds and α the prior odds. Let $BF(o)$ denote the Bayes factor. Then:

$$\underbrace{\frac{\Pr[\mathcal{X}|o]}{\Pr[\mathcal{Y}|o]}}_{\beta(x)} = \underbrace{\frac{\Pr[M(\mathcal{X}) = o]}{\Pr[M(\mathcal{Y}) = o]}}_{BF(o)} \cdot \underbrace{\frac{\Pr[\mathcal{X}]}{\Pr[\mathcal{Y}]}}_{\alpha} \quad . \tag{1}$$

The ratio of prior beliefs α is subjective, depending on the attacker's background knowledge. However, the Bayes factor $BF(o)$ is determined entirely by the mechanism $M(\cdot)$ and is thus objective.

3.2 Logarithmic Bayes factor

Since $\mathcal{X}$ and $\mathcal{Y}$ are equivalent datasets, there are two ways to define the posterior odds ratio between them. The logarithmic Bayes factor (LBF) quantifies the information gain about which dataset was used to produce the output o. Formally:

$$LBF_{\mathcal{X},\mathcal{Y}}(o) = \ln\left(\frac{\Pr[M(\mathcal{X}) = o]}{\Pr[M(\mathcal{Y}) = o]}\right)$$
$$LBF_{\mathcal{Y},\mathcal{X}}(o) = \ln\left(\frac{\Pr[M(\mathcal{Y}) = o]}{\Pr[M(\mathcal{X}) = o]}\right).$$

When not explicitly specified, we use $LBF(o)$ to denote $LBF_{\mathcal{X},\mathcal{Y}}(o)$. A positive $LBF_{\mathcal{X},\mathcal{Y}}(o)$ indicates increased confidence that the output came from dataset $\mathcal{X}$ rather than $\mathcal{Y}$. For instance, observing an element unique to $\mathcal{Y}$ would yield a positive $LBF_{\mathcal{Y},\mathcal{X}}(o)$ and a negative $LBF_{\mathcal{X},\mathcal{Y}}(o)$. From Equation (1), we can see that $LBF(o)$ quantifies how much an adversary can update their beliefs after observing a specific output o. If we wish to limit the adversary's ability to distinguish between $\mathcal{X}$ and $\mathcal{Y}$ (for instance, to hide whether a specific individual's records were used in training) with some bound ε, we must ensure $LBF(o) \leq \varepsilon$ for all possible outputs.

Theorem 1 (Symmetric LBF property)**.** *The logarithmic Bayes factor exhibits a skew-symmetric property such that:*

$$LBF_{\mathcal{X},\mathcal{Y}}(o) = -LBF_{\mathcal{Y},\mathcal{X}}(o) \ .$$

Proof. The skew-symmetry follows directly from the logarithmic identity:

$$\ln\left(\frac{\Pr\left[M(\mathcal{X}) = o\right]}{\Pr\left[M(\mathcal{Y}) = o\right]}\right) = -\ln\left(\frac{\Pr\left[M(\mathcal{Y}) = o\right]}{\Pr\left[M(\mathcal{X}) = o\right]}\right)$$

□

This property demonstrates that the information-theoretic perspective of the logarithmic Bayes factor remains invariant under dataset order exchange.

While we have considered individual outputs o, in practice outputs might be partially censored – for instance, when personal identifiers are removed. In such cases, we can only determine that the output belongs to some set $\mathcal{O} \subseteq \mathsf{Range}(M)$. For such sets, we extend this definition:

$$LBF_{\mathcal{X},\mathcal{Y}}(\mathcal{O}) = \ln\left(\frac{\Pr\left[M(\mathcal{X}) \in \mathcal{O}\right]}{\Pr\left[M(\mathcal{Y}) \in \mathcal{O}\right]}\right).$$

For sets of outputs, their LBF values cannot exceed the most extreme LBF values of individual elements in those sets, as shown in the following lemma.

Lemma 2 (Logarithmic Bayes factor summation constraint)**.** *Let $M(\cdot)$ be a mechanism operating on datasets $\mathcal{X}$ and $\mathcal{Y}$. For any $\mathcal{O} \subseteq \mathsf{Range}(M)$:*

$$\min_{o\in\mathcal{O}} LBF_{\mathcal{X},\mathcal{Y}}(o) \leq LBF_{\mathcal{X},\mathcal{Y}}(\mathcal{O}) \leq \max_{o\in\mathcal{O}} LBF_{\mathcal{X},\mathcal{Y}}(o) \ .$$

Proof. First, note that outputs where both probabilities are zero do not affect $LBF_{\mathcal{X},\mathcal{Y}}(\mathcal{O})$ as they contribute nothing to either sum. By definition:

$$LBF_{\mathcal{X},\mathcal{Y}}(\mathcal{O}) = \ln\left(\frac{\sum_{o\in\mathcal{O}}\Pr\left[M(\mathcal{X}) = o\right]}{\sum_{o\in\mathcal{O}}\Pr\left[M(\mathcal{Y}) = o\right]}\right) \ .$$

Let $d = \min_{o \in \mathcal{O}} \frac{\Pr[M(\mathcal{X})=o]}{\Pr[M(\mathcal{Y})=o]}$ and $c = \max_{o \in \mathcal{O}} \frac{\Pr[M(\mathcal{X})=o]}{\Pr[M(\mathcal{Y})=o]}$.

We know that for sequences of non-negative real numbers $(a_i)_{i=1}^n$ and $(b_i)_{i=1}^n$, if $\forall i \leq n$, $d \leq \frac{a_i}{b_i} \leq c$, then

$$d \leq \frac{a_1 + ... + a_n}{b_1 + ... + b_n} \leq c \ .$$

Applying this inequality to our probability ratios, since by construction each ratio lies between d and c:

$$d \leq \frac{\sum_{o \in \mathcal{O}} \Pr\left[M(\mathcal{X}) = o\right]}{\sum_{o \in \mathcal{O}} \Pr\left[M(\mathcal{Y}) = o\right]} \leq c \ .$$

Since logarithm is strictly monotonic and preserves ordering, we have $\ln(d) = \ln(\min(\cdot)) = \min(\ln(\cdot))$ and similarly for $\max$:

$$\min_{o \in \mathcal{O}} LBF_{\mathcal{X},\mathcal{Y}}(o) \leq LBF_{\mathcal{X},\mathcal{Y}}(\mathcal{O}) \leq \max_{o \in \mathcal{O}} LBF_{\mathcal{X},\mathcal{Y}}(o) \ .$$

For outputs where $\Pr\left[M(\mathcal{Y}) = o\right] = 0$ but $\Pr\left[M(\mathcal{X}) = o\right] \neq 0$, we have $LBF_{\mathcal{X},\mathcal{Y}}(o) \to \infty$, making the upper bound trivially true. The lower bound still holds since we either have all infinite values (making both sides infinite) or at least one finite value in the set we are taking the minimum of. □

3.3 Dataset proximity metric

In practice, it might be impossible to bound the LBF for two reasons. Firstly, the datasets $\mathcal{X}$ and $\mathcal{Y}$ might be too different in their content. Secondly, the mechanism might fail with a small probability and produce a revealing outcome o_*. In this section, we introduce proximity metrics that quantify dataset differences. The second challenge is explored in Section 3.4, where we examine how to handle mechanism failures.

The standard definition of differential privacy [DR14] quantifies the difference between databases in terms of different data points. Dwork and Roth used the ℓ_1 norm, which corresponds to the size of the symmetric difference of multisets over over the set of all possible elements $\mathcal{U}$, to measure this distance. Let $\mathcal{X}$ and $\mathcal{Y}$ be multisets. Then we can define count vectors x and y, where x_u and y_u denote the multiplicity of element $u \in \mathcal{U}$. For two multisets $\mathcal{X}$ and $\mathcal{Y}$, the cardinality of their symmetric difference is defined as

$$\|X \triangle Y\|_e = \sum_{u \in \mathcal{U}} |x_u - y_u| \ .$$

Example. Consider two multisets $\mathcal{X} = \{1, 2, 2\}$ and $\mathcal{Y} = \{2, 3\}$. Then $x = [1, 2, 0]$ is the count vector for $\mathcal{X}$ and $y = [0, 1, 1]$ is the count vector for $\mathcal{Y}$. Then the ℓ_1 distance is:

$$\begin{aligned}\|\mathcal{X} \triangle \mathcal{Y}\|_e &= |1 - 0| + |2 - 1| + |0 - 1| \\ &= 1 + 1 + 1 \\ &= 3\end{aligned}$$

This can be interpreted as the number of differing records between databases, where each term represents the absolute difference in the count of each unique element.

Definition 1 (Data Sources)**.** *Let S be a set of sources. For each source $s \in S$, let $\mathcal{D}_s$ be a multiset of elements generated by that source. A dataset $\mathcal{X}$ generated by these sources is defined as:*

$$\mathcal{X} = \bigcup_{s \in S} \mathcal{D}_s \ ,$$

where $\cup$ denotes multiset union preserving multiplicities.

Example. Consider a Twitter dataset where sources are users and $\mathcal{D}_s$ is the multiset of tweets made by user s. If an element appears n times in $\mathcal{D}_{s_1}$ and m times in $\mathcal{D}_{s_2}$, it appears $n + m$ times in their union.

Definition 2 (Source Set Symmetric Difference)**.** *For a set of possible sources S, let S_x denote the set of sources that generate x and S_y denote the set of sources that generate y. Then we can define the source-level difference of multisets X and Y as*

$$\|X \triangle Y\|_s = \|S_x \triangle S_y\|_e \ .$$

Example. Consider a dataset with sources $S = \{s_1, s_2, s_3, s_4\}$. Let users s_1 and s_2 have identical tweet sets $D_{s_1} = \{1, 2\}$ and $D_{s_2} = \{1, 2\}$, respectively. Let users s_3 and s_4 have different tweet sets $D_{s_3} = \{1, 1\}$ and $D_{s_4} = \{2, 2\}$.

Let $\mathcal{X} = D_{s_1} \cup D_{s_2}$ be generated by sources $\{s_1, s_2\}$ and $\mathcal{Y} = D_{s_3} \cup D_{s_4}$ be generated by sources $\{s_3, s_4\}$. Then:

$$\begin{aligned}S_x &= \{s_1, s_2\} \\ S_y &= \{s_3, s_4\} \\ \mathcal{X} &= \{1, 1, 2, 2\} \\ \mathcal{Y} &= \{1, 1, 2, 2\} \\ \|X \triangle Y\|_e &= 0 \\ \|X \triangle Y\|_s &= 4\end{aligned}$$

This example demonstrates that two datasets can be identical in terms of their elements ($\|X \triangle Y\|_e = 0$) while being completely different in terms of their sources ($\|X \triangle Y\|_s = 4$). This distinction between element-wise and source-wise differences leads us to define two forms of adjacency between datasets.

Definition 3 (Data Adjacency)**.** *Given two subsets of sources $S_1, S_2 \subseteq S$ generating datasets $\mathcal{X} = \bigcup_{i \in S_1} D_i$ and $\mathcal{Y} = \bigcup_{i \in S_2} D_i$, the datasets are considered element-wise k-adjacent if:*

$$\|\mathcal{X} \triangle \mathcal{Y}\|_e \leq k \ .$$

The same datasets are considered source-wise k-adjacent if:

$$\|\mathcal{X} \triangle \mathcal{Y}\|_s \leq k \ .$$

3.4 Mechanism failures

In order to define differential privacy, Dwork and Roth [DR14] introduce partial ordering between probability distributions.

Definition 4. *Let $\mathcal{X}$ and $\mathcal{Y}$ be multisets. Then the distribution $M(\mathcal{X})$ is (ε, δ)-dominated by $M(\mathcal{Y})$, if for any set of possible outcomes $\mathcal{O} \subseteq \mathsf{Range}(M)$:*

$$\Pr[M(\mathcal{X}) \in \mathcal{O}] \leq \exp(\varepsilon) \cdot \Pr[M(\mathcal{Y}) \in \mathcal{O}] + \delta \ .$$

It is difficult to interpret this definition, therefore we use a slightly more stringent version of dominance, which has a simpler interpretation.

Definition 5. *Let $\mathcal{X}$ and $\mathcal{Y}$ be multisets and $o \in \mathcal{O}$ be any output of the mechanism M. The distribution $M(\mathcal{X})$ is ε-dominated by $M(\mathcal{Y})$ with failure probability δ, if*

$$\Pr[o \leftarrow M(\mathcal{X}) : LBF_{\mathcal{X},\mathcal{Y}}(o) > \varepsilon] \leq \delta \ . \tag{2}$$

We are usually interested in the smallest possible failure probability δ in the definition. Given a fixed target ε and datasets $\mathcal{X}$ and $\mathcal{Y}$, let $\mathcal{F}_\mathcal{X}$ and $\mathcal{F}_\mathcal{Y}$ be sets of failing outcomes:

$$\mathcal{F}_\mathcal{X} = \{o \in \mathsf{Range}(M) : LBF_{\mathcal{X},\mathcal{Y}}(o) > \varepsilon\}$$
$$\mathcal{F}_\mathcal{Y} = \{o \in \mathsf{Range}(M) : LBF_{\mathcal{Y},\mathcal{X}}(o) > \varepsilon\} \ .$$

We first fix a privacy target ε and then compute δ as the minimal probability of outputs that exceed this target threshold. For a privacy target ε, $LBF_{\mathcal{X},\mathcal{Y}}$ and $LBF_{\mathcal{Y},\mathcal{X}}$ exceed the threshold ε with at most probability δ in both directions:

$$\Pr[M(\mathcal{X}) \in \mathcal{F}_\mathcal{X}] \leq \delta$$
$$\Pr[M(\mathcal{Y}) \in \mathcal{F}_\mathcal{Y}] \leq \delta \ .$$

These δ bounds correspond to Definition 5 failure probability – first for $M(\mathcal{X})$ being dominated by $M(\mathcal{Y})$, then for $M(\mathcal{Y})$ being dominated by $M(\mathcal{X})$. When both of these dominance relations hold with the same δ, this naturally leads us to strict (ε, δ)-differential privacy. We can now reach a correspondence with the original DP definition.

Lemma 3. *Let M be a randomized mechanism, ε be the privacy threshold, and $\mathcal{X}, \mathcal{Y}$ be datasets. If the distribution $M(\mathcal{X})$ is ε-dominated by $M(\mathcal{Y})$ with failure probability δ, then $M(\mathcal{X})$ is (ε, δ)-dominated by $M(\mathcal{Y})$.*

Proof. Let $\mathcal{O} \subseteq \mathsf{Range}(M)$ be an arbitrary set of possible outputs and $\mathcal{F}_{\mathcal{X}}$ be the failure set. We split $\mathcal{O}$ into two disjoint sets: $\mathcal{O}_1 = \mathcal{O} \cap \mathcal{F}_{\mathcal{X}}$ containing outputs where $LBF > \varepsilon$, and $\mathcal{O}_2 = \mathcal{O} \setminus \mathcal{F}_{\mathcal{X}}$ containing outputs where $LBF \leq \varepsilon$. By the law of total probability:

$$\Pr\left[M(\mathcal{X}) \in \mathcal{O}\right] = \Pr\left[M(\mathcal{X}) \in \mathcal{O}_1\right] + \Pr\left[M(\mathcal{X}) \in \mathcal{O}_2\right] \ .$$

Since $\mathcal{O}_1$ is a subset of the failure set and using our assumption about failure probability:

$$\Pr\left[M(\mathcal{X}) \in \mathcal{O}_1\right] \leq \Pr\left[M(\mathcal{X}) \in \mathcal{F}_{\mathcal{X}}\right] \leq \delta \ .$$

For outputs in $\mathcal{O}_2$, by definition of LBF:

$$\Pr\left[M(\mathcal{X}) \in \mathcal{O}_2\right] \leq \exp(\varepsilon) \cdot \Pr\left[M(\mathcal{X}) \in \mathcal{O}_2\right]$$

Combining these bounds:

$$\begin{aligned}\Pr\left[M(\mathcal{X}) \in \mathcal{O}\right] &= \Pr\left[M(\mathcal{X}) \in \mathcal{O}_1\right] + \Pr\left[M(\mathcal{X}) \in \mathcal{O}_2\right] \\ &\leq \delta + \exp(\varepsilon) \cdot \Pr\left[M(\mathcal{X}) \in \mathcal{O}_2\right] \\ &\leq \delta + \exp(\varepsilon) \cdot \Pr\left[M(\mathcal{X}) \in \mathcal{O}\right]\end{aligned}$$

This corresponds to the definition of (ε, δ)-domination. □

Note that the reverse implication does not hold due to the loosening substitutions performed in the last inequality. The following lemma shows that ε-domination with failure probability δ is more a more stringent notion than (ε, δ)-domination.

Lemma 4. *Let M be a randomized mechanism and $\varepsilon > 0$. If $M(\mathcal{X})$ is (ε, δ)-dominated by $M(\mathcal{Y})$ where δ is the minimal such value, then the probability of LBF failures must be at least δ. Formally, there exists $\delta_2 \geq \delta$ such that:*

$$\Pr\left[M(\mathcal{X}) \in \mathcal{F}_{\mathcal{X}}\right] \leq \delta_2$$

where

$$\mathcal{F}_{\mathcal{X}} = \{o \in \mathsf{Range}(M) : LBF_{\mathcal{X},\mathcal{Y}}(o) > \varepsilon\} \ .$$

Proof. Let $\delta_2 = \Pr[M(\mathcal{X}) \in \mathcal{F}_\mathcal{X}]$ be the probability of LBF failures. Assume towards contradiction that $\delta_2 < \delta$. By Lemma 3, if $\Pr[M(\mathcal{X}) \in \mathcal{F}_\mathcal{X}] \leq \delta_2$, then $M(\mathcal{X})$ is (ε, δ_2)-dominated by $M(\mathcal{Y})$.

However, this contradicts our assumption that δ was minimal for (ε, δ)-domination, since we found $\delta_2 < \delta$ that also satisfies the domination property.

Therefore, our assumption must be false and we have $\delta_2 \geq \delta$, completing the proof. □

This result can be intuitively understood by observing that in the standard definition, the additive term δ is effectively scaled by $1/\Pr[M(\mathcal{Y}) \in \mathcal{O}]$ when comparing probability ratios directly. Since $\Pr[M(\mathcal{Y}) \in \mathcal{O}] \leq 1$, this scaling can make violations appear smaller in the standard definition than when measured directly in our strict definition. Specifically, for outputs where $\Pr[M(\mathcal{Y}) = o]$ is very small, even minor absolute differences between $\Pr[M(\mathcal{X}) = o]$ and $\exp(\varepsilon)\Pr[M(\mathcal{Y}) = o]$ can lead to large LBF violations, requiring a larger δ_2 to account for them.

3.5 Recipe for estimating differential privacy

Dwork et al. [Dwo06] introduce differential privacy, a framework for quantifying privacy guarantees. Note that if we fix concrete datasets $\mathcal{X}$ and $\mathcal{Y}$, we can evaluate their privacy properties directly. When making general statements about differential privacy for only a fixed mechanism, we require these properties to hold for all adjacent datasets.

Definition 6. *Let M be a randomized mechanism. We say that M is (ε, δ)-differentially private if for all adjacent datasets $\mathcal{X}$, $\mathcal{Y}$ and for all sets of outputs $\mathcal{O} \subseteq \mathsf{Range}(M)$:*

$$\Pr[M(\mathcal{X}) \in \mathcal{O}] \leq \exp(\varepsilon)\Pr[M(\mathcal{Y}) \in \mathcal{O}] + \delta$$
$$\Pr[M(\mathcal{Y}) \in \mathcal{O}] \leq \exp(\varepsilon)\Pr[M(\mathcal{X}) \in \mathcal{O}] + \delta$$

When $\delta = 0$, the mechanism M satisfies ε-differential privacy (ε-DP), a stricter form of privacy where the additive term δ is eliminated. In this case, the parameter ε specifies the maximum allowable multiplicative difference in the probabilities of observing any output $\mathcal{O}$ when the mechanism is applied to adjacent datasets. This effectively serves as a privacy budget: each application of the mechanism consumes a portion of this budget, and as more queries are made, the total privacy budget consumed increases.

Definition 7 (Strict (ε, δ)-Differential Privacy)**.** *Let M be a randomized mechanism and let $\mathcal{X}$, $\mathcal{Y}$ be adjacent datasets. Define failure sets:*

$$\mathcal{F}_\mathcal{X} = \{o \in \mathsf{Range}(M) : LBF_{\mathcal{X},\mathcal{Y}}(o) > \varepsilon\}$$
$$\mathcal{F}_\mathcal{Y} = \{o \in \mathsf{Range}(M) : LBF_{\mathcal{Y},\mathcal{X}}(o) > \varepsilon\} \quad .$$

We say that M satisfies strict (ε, δ)-differential privacy if for all adjacent datasets $\mathcal{X}$, $\mathcal{Y}$:

$$\Pr\left[M(\mathcal{X}) \in \mathcal{F}_{\mathcal{X}}\right] \leq \delta$$
$$\Pr\left[M(\mathcal{Y}) \in \mathcal{F}_{\mathcal{Y}}\right] \leq \delta \ .$$

Theorem 5 (Computing δ from ε)**.** *For fixed adjacent datasets $\mathcal{X}$, $\mathcal{Y}$ and fixed $\varepsilon > 0$, let:*

$$\mathcal{F}_{\mathcal{X}} = \{o \in \mathsf{Range}(M) : LBF_{\mathcal{X},\mathcal{Y}}(o) > \varepsilon\}$$
$$\mathcal{F}_{\mathcal{Y}} = \{o \in \mathsf{Range}(M) : LBF_{\mathcal{Y},\mathcal{X}}(o) > \varepsilon\} \ .$$

Then the minimal δ such that $M(\mathcal{X})$ is ε-dominated by $M(\mathcal{Y})$ with failure probability δ ***and*** *$M(\mathcal{Y})$ is ε-dominated by $M(\mathcal{X})$ with failure probability δ is:*

$$\delta = \max(\Pr\left[M(\mathcal{X}) \in \mathcal{F}_{\mathcal{X}}\right], \Pr\left[M(\mathcal{Y}) \in \mathcal{F}_{\mathcal{Y}}\right]) \ .$$

Proof. Let $\delta_1 = \Pr\left[M(\mathcal{X}) \in \mathcal{F}_{\mathcal{X}}\right]$ and $\delta_2 = \Pr\left[M(\mathcal{Y}) \in \mathcal{F}_{\mathcal{Y}}\right]$.

According to Definition 5: $M(\mathcal{X})$ is ε-dominated by $M(\mathcal{Y})$ with failure probability δ_x if

$$\Pr\left[o \leftarrow M(\mathcal{X}) : LBF_{\mathcal{X},\mathcal{Y}}(o) > \varepsilon\right] \leq \delta_x \ .$$

The minimal such δ_x is achieved when $\delta_x = \Pr\left[M(\mathcal{X}) \in \mathcal{F}_{\mathcal{X}}\right] = \delta_1$.

$M(\mathcal{Y})$ is ε-dominated by $M(\mathcal{X})$ with failure probability δ_y if

$$\Pr\left[o \leftarrow M(\mathcal{Y}) : LBF_{\mathcal{Y},\mathcal{X}}(o) > \varepsilon\right] \leq \delta_y \ .$$

The minimal such δ_y is achieved when $\delta_y = \Pr\left[M(\mathcal{Y}) \in \mathcal{F}_{\mathcal{Y}}\right] = \delta_2$.

We require a single δ such that both conditions hold:

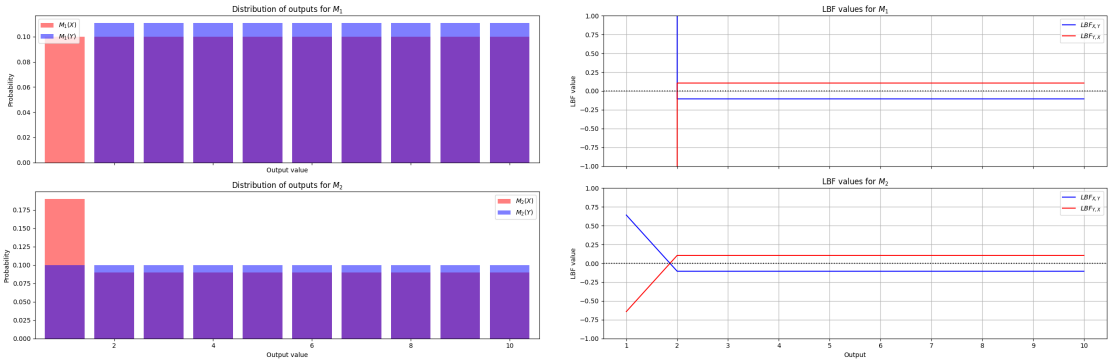
$$\Pr\left[M(\mathcal{X}) \in \mathcal{F}_{\mathcal{X}}\right] \leq \delta$$
$$\Pr\left[M(\mathcal{Y}) \in \mathcal{F}_{\mathcal{Y}}\right] \leq \delta$$

This means we need $\delta_1 \leq \delta$ and $\delta_2 \leq \delta$. For δ to be minimal while satisfying both inequalities, it must be the smallest value that is greater than or equal to both δ_1 and δ_2. Therefore, the minimal $\delta = \max(\delta_1, \delta_2) = \max(\Pr\left[M(\mathcal{X}) \in \mathcal{F}_{\mathcal{X}}\right], \Pr\left[M(\mathcal{Y}) \in \mathcal{F}_{\mathcal{Y}}\right])$. □

In practice, we can compute (ε, δ) pairs efficiently using a two-pointer approach. First, we compute the probabilities $\Pr\left[M(\mathcal{X}) = o\right]$ and $\Pr\left[M(\mathcal{Y}) = o\right]$ for every $o \in \mathsf{Range}(M)$ over $\mathcal{X}$ and $\mathcal{Y}$ – note that this requires only summary statistics of some function over $\mathcal{X}$ and $\mathcal{Y}$ as separate inputs. From these probabilities, we calculate $LBF_{\mathcal{X},\mathcal{Y}}$ and $LBF_{\mathcal{Y},\mathcal{X}}$ values. Then we sort both $LBF_{\mathcal{X},\mathcal{Y}}$ and $LBF_{\mathcal{Y},\mathcal{X}}$ arrays in descending order,

then rearrange their corresponding probabilities (P_x and P_y) using the same permutation as their respective LBF arrays. This sorting means that for any threshold ε, all outputs with $LBF > \varepsilon$ appear consecutively at the start of the arrays. Thus, the failure sets $\mathcal{F}_{\mathcal{X}}$ and $\mathcal{F}_{\mathcal{Y}}$ can be computed by accumulating probabilities from the start of each sorted array until we reach outputs with $LBF \leq \varepsilon$. By the Theorem 5, δ for that ε is the maximum of these accumulated probabilities.

Starting from infinity and sweeping through unique LBF values as thresholds gives us a sequence of (ε, δ) pairs. Each ε represents a privacy threshold and its corresponding δ represents the probability of exceeding it.



(a) Output distributions for M_1 and M_2.

(b) Log-Bayes factors (LBFs) for M_1 and M_2.

Figure 1. Comparison of output distributions and LBFs for mechanisms M_1 and M_2. (a) Output distributions for mechanisms M_1 and M_2 over datasets $\mathcal{X}$ (red) and $\mathcal{Y}$ (blue). Top: M_1 shows zero probability of generating 1 from $\mathcal{Y}$, while maintaining uniform distribution over other elements. Bottom: M_2 can generate 1 from both datasets, with a fixed probability $\frac{1}{10}$ plus the probability from uniform sampling. (b) Corresponding LBF values for outputs of M_1 and M_2

Basic sampling mechanism example (M1). Let $\mathcal{X} = \{1, \ldots, 10\}$ and $\mathcal{Y} = \{2, \ldots, 100\}$ be sets of positive integers. Let $M_1(\mathcal{X})$ return a random sample element from $\mathcal{X}$ and $M_1(\mathcal{Y})$ from $\mathcal{Y}$, correspondingly. As shown in Figure 1a (top), in the case of $o = 1$, it is impossible to obtain it from $M_1(\mathcal{Y})$, as $1 \notin \mathcal{Y}$ and thus

$$LBF_{\mathcal{X},\mathcal{Y}}(1) = \ln\left(\frac{\Pr\left[M_1(\mathcal{X}) = 1\right]}{\Pr\left[M_1(\mathcal{Y}) = 1\right]}\right) = \ln\left(\frac{1/10}{0}\right) = \infty \ .$$

Therefore, no finite ε exists such that $\Pr\left[M_1(\mathcal{X}) = 1\right] \leq \exp(\varepsilon) \Pr\left[M_1(\mathcal{Y}) = 1\right]$. Since it was sufficient to show that no bound holds for at least one example, this mechanism is not ε-DP for any ε. If we loosen the $\delta = 0$ restriction, we are allowed to exceed the ε

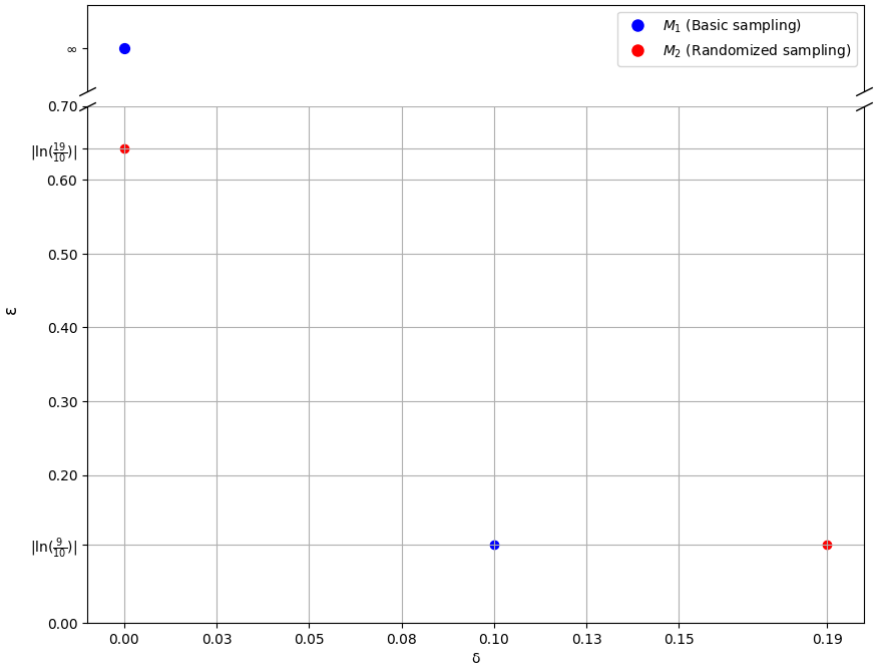


Figure 2. Privacy guarantees for mechanisms M_1 (blue) and M_2 (red) expressed as (ε, δ) pairs. M_1 requires infinite ε to achieve $\delta = 0$, while M_2 achieves finite ε values for $\delta = 0$.

threshold with probability δ. Note that for $s \neq 1$ we can express

$$LBF_{\mathcal{X},\mathcal{Y}}(s) = \ln\left(\frac{\Pr\left[M_1(\mathcal{X}) = s\right]}{\Pr\left[M_1(\mathcal{Y}) = s\right]}\right) = \ln\left(\frac{1/10}{1/9}\right) = \ln\frac{9}{10} \quad .$$

The probability of $M_1(\mathcal{X})$ outputting 1, which always breaks any finite ε threshold, is $\frac{1}{10}$, as the mechanism of M_1 returns a random element of its input. In all other cases, LBF never exceeds $\left|\ln\frac{9}{10}\right|$. In this setting, such a mechanism is either ($\varepsilon = \left|\ln\frac{9}{10}\right|$, $\delta = \frac{1}{10}$)-DP or simply ∞-DP, implicitly with $\delta = 0$. Note that in the case of infinite ε, we can simply say that the mechanism M_1 is in this case not differentially private as long as $\delta = 0$.

Randomized sampling mechanism example (M2). Let $\mathcal{X} = \{1, \ldots, 10\}$ and $\mathcal{Y} = \{2, \ldots, 10\}$ be sets of positive integers. Let $M_2(\mathcal{X})$ return 1 with probability $\frac{1}{10}$ and a random sample element from $\mathcal{X}$ with probability $\frac{9}{10}$ (and similarly for $\mathcal{Y}$). As demonstrated in Figure 1a (bottom), in this setting, it is indeed possible for $M_2(\mathcal{Y})$ to output 1:

$$\frac{Pr[M_2(\mathcal{X}) = 1]}{Pr[M_2(\mathcal{Y}) = 1]} = \frac{\left(\frac{1}{10}\right) \cdot 1 + \left(\frac{9}{10}\right) \cdot \left(\frac{1}{10}\right)}{\left(\frac{1}{10}\right) \cdot 1 + \left(\frac{9}{10}\right) \cdot \left(\frac{0}{9}\right)} = \frac{\frac{1}{10} + \frac{9}{100}}{\frac{1}{10}} = \frac{19}{10}$$

For all other possible outputs, such as 2:

$$\frac{Pr[M_2(\mathcal{X}) = 2]}{Pr[M_2(\mathcal{Y}) = 2]} = \frac{\left(\frac{1}{10}\right) \cdot 0 + \left(\frac{9}{10}\right) \cdot \left(\frac{1}{10}\right)}{\left(\frac{1}{10}\right) \cdot 0 + \left(\frac{9}{10}\right) \cdot \left(\frac{1}{9}\right)} = \frac{\frac{9}{100}}{\frac{1}{10}} = \frac{9}{10}$$

We can therefore say that in this setting, M_2 is $\left|\ln \frac{19}{10}\right|$-DP, implicitly with $\delta = 0$. M_2 is also ($\varepsilon = \left|\ln \frac{9}{10}\right|$, $\delta = \frac{19}{100}$)-DP, as the ε threshold can be exceeded with probability $\frac{19}{100}$ for $\varepsilon = \left|\ln \frac{9}{10}\right|$ when M_2 is sampling from $\mathcal{X}$, which is the worst case for this value.

3.6 Limitations of differential privacy

The differential privacy bounds discussed so far assume perfect adversarial knowledge. An adversary with complete information about datasets $\mathcal{X}$ and $\mathcal{Y}$ can compute the LBF of any observation and achieve the maximal posterior odds gain of $\exp(\varepsilon)$. However, this faces two practical limitations. First, privacy breaches might only be possible for specific outputs $\mathcal{O}$ that occur rarely – this is handled by the (ε, δ)-profile. The second limitation is more severe: computing LBF requires complete knowledge of both datasets $\mathcal{X}$ and $\mathcal{Y}$, which itself would constitute a more significant privacy breach than success in a membership inference attack.

The computational challenge of requiring complete datasets can be reduced by considering sufficient statistics. For a mechanism M, let $SS_M(\mathcal{X})$ denote a sufficient statistic that captures all information needed to determine the behavior of M on dataset $\mathcal{X}$. For instance, if M outputs single tokens, then $SS_M(\mathcal{X})$ could be the probability distribution over tokens that M outputs when run on $\mathcal{X}$. Importantly, knowing the complete dataset $\mathcal{X}$ provides no additional information about the outputs of M beyond what $SS_M(\mathcal{X})$ tells us.

This means our adversary does not need complete knowledge of $\mathcal{X}$ and $\mathcal{Y}$ – they only need $SS_M(\mathcal{X})$ and $SS_M(\mathcal{Y})$. While this is still a strong knowledge assumption, it is significantly more tractable than requiring full datasets.

Example. Consider a medical research setting where $\mathcal{X}$ is a dataset containing detailed health records of patients receiving an experimental HIV treatment. Suppose we have a basic mechanism M that, given a dataset, uniformly samples and outputs a single random word from all words in patients' records. For this simple random word sampling mechanism, a sufficient statistic $SS_M(\mathcal{X})$ would be the word frequency distribution. An adversary trying to determine if a specific target patient is receiving this treatment would need:

- $SS_M(\mathcal{X})$: The word frequency distribution over all patient records
- $SS_M(\mathcal{Y})$: The word frequency distribution over the dataset with the target patient's record removed

Obtaining either distribution would constitute a massive privacy breach – the adversary would need to know the medical details of every patient in the study just to construct these statistics.

Even having access to aggregate word frequencies could reveal sensitive information, as certain medical terms or treatment descriptions might be strongly correlated with HIV status. For our word sampling mechanism, the normalized word frequency distribution is exactly the sufficient statistic. Notably, an adversary who knows both $SS_M(\mathcal{X})$ and $SS_M(\mathcal{Y})$ can compute their difference to infer the target patient's contribution directly. In the case of word count frequencies, if $\mathcal{X}$ contains all patients and $\mathcal{Y}$ excludes our target patient, then $SS_M(\mathcal{X}) - SS_M(\mathcal{Y})$ (appropriately scaled) directly reveals the word distribution in the target patient's records. This demonstrates why even summary statistics themselves constitute a serious privacy breach.

To address this limitation, we need to consider more realistic scenarios where attackers have limited knowledge. Instead of knowing exact datasets $\mathcal{X}$ and $\mathcal{Y}$, an attacker typically has some background knowledge Bg that suggests which datasets are plausible candidates for $\mathcal{X}$ and $\mathcal{Y}$.

Let I denote either element-wise or source-wise inclusion in the dataset – that is, either the inclusion of a specific record or all records from a particular source (as defined in Section 3.3). For a specific output o, we are interested in how much more likely the adversary, having observed this output, would consider the underlying dataset to include the record or the source rather than not.

Let $\beta(o)$ represent the posterior odds ratio after observing output o:

$$\beta(o) = \frac{\Pr[I|o, Bg]}{\Pr[\neg I|o, Bg]} \ .$$

Using Bayes' theorem and following the same structure as Equation (1):

$$\beta(o) = \underbrace{\frac{\Pr[o|I, Bg]}{\Pr[o|\neg I, Bg]}}_{BF(o)} \cdot \underbrace{\frac{\Pr[I|Bg]}{\Pr[\neg I|Bg]}}_{\alpha} \ .$$

Here α represents the adversary's prior beliefs and is independent of the observed output o. The Bayes factor $BF(o)$ quantifies how the observation updates these beliefs, analogous to the ratio of mechanism probabilities in Equation (1).

First, we express $BF(o)$ in terms of all possible datasets that satisfy the inclusion/exclusion condition:

$$\Pr[o|I, Bg] = \sum_{\mathcal{X}} \Pr[o, \mathcal{X}|I, Bg] = \sum_{\mathcal{X}} \Pr[o|\mathcal{X}, I, Bg] \cdot \Pr[\mathcal{X}|I, Bg] \quad .$$

Under complete knowledge of dataset $\mathcal{X}$, both Bg and I become redundant, as the dataset itself tells us definitively if the target patient is included:

$$\Pr[o|\mathcal{X}, I, Bg] = \Pr[o|\mathcal{X}] = \Pr[M(\mathcal{X}) = o] \quad .$$

This gives us the extended form of $BF_{\mathcal{X},\mathcal{Y}}(o)$:

$$BF_{\mathcal{X},\mathcal{Y}}(o) = \frac{\Pr[o|I, Bg]}{\Pr[o|\neg I, Bg]} = \frac{\sum_{\mathcal{X}} \Pr[M(\mathcal{X}) = o] \cdot \Pr[\mathcal{X}|I, Bg]}{\sum_{\mathcal{Y}} \Pr[M(\mathcal{Y}) = o] \cdot \Pr[\mathcal{Y}|\neg I, Bg]} \quad .$$

Taking the logarithm yields the subjective $LBF_{\mathcal{X},\mathcal{Y}}$:

$$LBF_{\mathcal{X},\mathcal{Y}}(o) = \ln\left(\frac{\sum_{\mathcal{X}} \Pr[M(\mathcal{X}) = o] \cdot \Pr[\mathcal{X}|I, Bg]}{\sum_{\mathcal{Y}} \Pr[M(\mathcal{Y}) = o] \cdot \Pr[\mathcal{Y}|\neg I, Bg]}\right) \quad .$$

This subjective LBF quantifies how an adversary with background knowledge Bg updates their belief about target inclusion after seeing output o. The terms $\Pr[\mathcal{X}|I, Bg]$ and $\Pr[\mathcal{Y}|\neg I, Bg]$ represent the adversary's prior beliefs about possible datasets with and without the target and therefore remain constant.

This formulation faces two problems. First, computing this form of LBF is infeasible as it requires summing over all possible datasets. Second, an adversary's probability distributions over datasets ($\Pr[\mathcal{X}|I, Bg]$ and $\Pr[\mathcal{Y}|\neg I, Bg]$) may not reflect reality.

Example. Consider our HIV treatment dataset. An adversary might incorrectly believe that the presence of a standard antiretroviral drug (e.g., tenofovir) indicates participation in the experimental treatment group. If this drug appears in the output, the adversary would compute a high LBF value, incorrectly concluding strong evidence of inclusion, when in reality this drug is commonly prescribed in standard HIV care regardless of experimental treatment participation. Conversely, the same adversary might dismiss the appearance of a rare HIV genetic resistance test result as irrelevant noise, when this test is actually performed exclusively on experimental treatment participants. They would compute a low LBF for this observation, failing to recognize its significance as evidence of inclusion. In both cases, the adversary's incorrect beliefs lead to miscalibrated probability assignments ($\Pr[\mathcal{X}|I, Bg]$ and $\Pr[\mathcal{Y}|\neg I, Bg]$) and consequently unreliable

inference, reaching false conclusions about membership in the first case and missing important signals in the second.

These limitations imply that while differential privacy bounds provide strong theoretical guarantees under perfect knowledge assumptions, we need additional tools to reason about privacy in practical scenarios where adversaries have limited and potentially incorrect information.

An important insight is that working with sufficient statistics often simplifies modeling the attacker's knowledge. Consider our word sampling mechanism: instead of needing distributions over complete datasets, we only need to reason about unigram frequencies. If our background knowledge assumes texts are sampled from standard language distributions, we can model expected word frequencies using Zipfian distributions typical of natural language [Pia14]. This gives us a tractable analytical form for background knowledge that is much simpler than reasoning about full dataset distributions.

3.7 Attack functions as a way to compress the output

Given the limitations of standard differential privacy, we can take a more practical approach by considering specific known membership inference attacks (MIAs) and quantifying their severity in terms comparable to differential privacy. In many real-world scenarios, we already have well-established techniques for performing membership inference against machine learning models. Rather than considering the theoretical worst-case adversary with perfect knowledge, we can analyze how effective these practical attacks are by formalizing them as attack functions, which are specific algorithms that attempt to infer membership based on observed outputs. This allows us to measure their privacy impact using the same mathematical framework as differential privacy while focusing on realistic threat models.

First, we need to specify what the attack functions compute and how they relate to the membership inference problem. Then we can examine what background knowledge they require and how this knowledge affects their success rate. This reformulation achieves three key properties:

1. It quantifies what background knowledge an attacker needs, letting us bound their posterior odds change objectively;

2. As the attacker's knowledge approaches perfect information, the bounds converge to standard differential privacy;

3. For any practical attack strategy, we can evaluate both its effectiveness and the minimum background knowledge required for success.

The main insight is that while we cannot compute exact LBF values with imperfect knowledge, we can model privacy guarantees through concrete attack functions.

Consider a classifier function $\mathsf{Score} : \mathfrak{D} \to \mathcal{O}$ that attempts to extract meaningful information about dataset membership, where $\mathcal{O}$ can be any output space. For instance, $\mathcal{O}$ could be $\mathbb{R}$ for likelihood scores, or $\{0, 1\}$ for binary decisions. Note that $\mathsf{Score} \circ M$ is itself a mechanism. Since this is just post-processing the output of M, the new mechanism has DP guarantees at least as strong as M, moreover, they might be stronger, as we reduce the output of M onto a single score, losing information in the process. After composition, we can treat it as a standalone mechanism, forgetting our access to M and analyzing only the final outputs. Instead of optimizing typical ML metrics like precision or recall, we analyze how much this composed mechanism can leak about the input dataset. This lets us reason about differential privacy with respect to specific attack functions rather than all possible outputs of M. The attack function acts as a lens, focusing the privacy analysis on the specific information leakage relevant to that particular adversarial strategy. This generality motivates Definitions 8 and 9 and leads to Theorem 6 for computing actual privacy bounds.

Definition 8 (Strict (ε, δ)-Differential Privacy against a single attack)**.** *Let M be a randomized mechanism and A be an attack function. Let $\mathcal{X}$ and $\mathcal{Y}$ be adjacent datasets. Define failure sets:*

$$\mathcal{F}_{\mathcal{X}} = \left\{o \in \mathsf{Range}(A) : LBF^{A}_{\mathcal{X},\mathcal{Y}}(o) > \varepsilon\right\}$$
$$\mathcal{F}_{\mathcal{Y}} = \left\{o \in \mathsf{Range}(A) : LBF^{A}_{\mathcal{Y},\mathcal{X}}(o) > \varepsilon\right\}$$

where for each $o \in \mathsf{Range}(A)$:

$$LBF^{A}_{\mathcal{X},\mathcal{Y}}(o) = \ln\left(\frac{\Pr\left[A(M(\mathcal{X})) = o\right]}{\Pr\left[A(M(\mathcal{Y})) = o\right]}\right) \quad .$$

We say that A satisfies (ε, δ)-differential privacy if for all adjacent datasets $\mathcal{X}$, $\mathcal{Y}$:

$$\Pr\left[A(M(\mathcal{X})) \in \mathcal{F}_{\mathcal{X}}\right] \leq \delta$$
$$\Pr\left[A(M(\mathcal{Y})) \in \mathcal{F}_{\mathcal{Y}}\right] \leq \delta \quad .$$

Definition 8 addresses static adversaries who must commit to their attack function before observing any outputs from the mechanism M. This models situations where the attack is decided upon ahead of time, rather than being tailored to specific outputs. We formalize this notion as differential privacy against static adversaries.

Definition 9 (Strict (ε, δ)-Differential Privacy against a static adversary, for an attack class)**.** *Let M be a randomized mechanism and $\mathfrak{A}$ be a class of attack functions. Let $\mathcal{X}$ and $\mathcal{Y}$ be adjacent datasets. For each $A \in \mathfrak{A}$, define failure sets:*

$$\mathcal{F}^{A}_{\mathcal{X}} = \left\{o \in \mathsf{Range}(A) : LBF^{A}_{\mathcal{X},\mathcal{Y}}(o) > \varepsilon\right\}$$
$$\mathcal{F}^{A}_{\mathcal{Y}} = \left\{o \in \mathsf{Range}(A) : LBF^{A}_{\mathcal{Y},\mathcal{X}}(o) > \varepsilon\right\}$$

where for each $o \in \mathsf{Range}(A)$*:*

$$LBF^{A}_{\mathcal{X},\mathcal{Y}}(o) = \ln\left(\frac{\Pr\left[A(M(\mathcal{X})) = o\right]}{\Pr\left[A(M(\mathcal{Y})) = o\right]}\right)$$

We say that M satisfies (ε, δ)*-differential privacy against* $\mathfrak{A}$ *if for all adjacent datasets* $\mathcal{X}$*,* $\mathcal{Y}$ *and for all* $A \in \mathfrak{A}$*:*

$$\Pr\left[A(M(\mathcal{X})) \in \mathcal{F}^{A}_{\mathcal{X}}\right] \leq \delta$$
$$\Pr\left[A(M(\mathcal{Y})) \in \mathcal{F}^{A}_{\mathcal{Y}}\right] \leq \delta$$

The definition above addresses the case where an adversary must commit to a specific attack function from class $\mathfrak{A}$ before observing any outputs. However, in many practical scenarios, adversaries can adapt their attack strategy after seeing mechanism outputs. This stronger adversarial model warrants its own privacy definition.

Definition 10 (Strict (ε, δ)-Differential Privacy against adaptive adversaries, for an attack class)**.** *Let M be a randomized mechanism and* $\mathfrak{A}$ *be a class of attack functions. Let* $\mathcal{X}$ *and* $\mathcal{Y}$ *be adjacent datasets. We say that M satisfies* (ε, δ)*-differential privacy against adaptive adversaries from* $\mathfrak{A}$ *if for all adjacent datasets* $\mathcal{X}$*,* $\mathcal{Y}$ *and for all* $o \in \mathsf{Range}(M)$*:*

$$\Pr\left[\exists A \in \mathfrak{A} : A(o) = 1 \wedge LBF^{A}_{\mathcal{X},\mathcal{Y}}(o) > \varepsilon\right] \leq \delta$$
$$\Pr\left[\exists A \in \mathfrak{A} : A(o) = 1 \wedge LBF^{A}_{\mathcal{Y},\mathcal{X}}(o) > \varepsilon\right] \leq \delta$$

In this adaptive model, the adversary first observes output o from mechanism M, then selects the attack function $A \in \mathfrak{A}$ that maximizes their information gain. This represents a stronger adversary who can tailor their attack method to each specific output. Note that for each output o, the preimage $A^{-1}(1) \subseteq \mathsf{Range}(M)$ is a subset of the mechanism's range containing o. It is essentially a failure set specifically chosen by the adaptive adversary to contain o while maximizing information gain.

Theorem 6 (Computing δ for Attack-Based DP)**.** *For fixed datasets* $\mathcal{X}$*,* $\mathcal{Y}$*, attack function A, and fixed* $\varepsilon > 0$*, let* $\mathcal{F}^{A}_{\mathcal{X}}$ *and* $\mathcal{F}^{A}_{\mathcal{X}}$ *be defined as above. Then the minimal* δ *such that* M *satisfies strict* (ε, δ)*-differential privacy against* A *is:*

$$\delta = \max(\Pr\left[A(M(\mathcal{X})) \in \mathcal{F}^{A}_{\mathcal{X}}\right], \Pr\left[A(M(\mathcal{Y})) \in \mathcal{F}^{A}_{\mathcal{Y}}\right]) \ .$$

Proof. Let $\delta_1 = \Pr\left[A(M(\mathcal{X})) \in \mathcal{F}^{A}_{\mathcal{X}}\right]$ and $\delta_2 = \Pr\left[A(M(\mathcal{Y})) \in \mathcal{F}^{A}_{\mathcal{Y}}\right]$.

According to Definition 8, for M to satisfy strict (ε, δ)-differential privacy against A, both of the following conditions must hold:

$$\Pr\left[A(M(\mathcal{X})) \in \mathcal{F}^{A}_{\mathcal{X}}\right] \leq \delta \tag{1}$$
$$\Pr\left[A(M(\mathcal{Y})) \in \mathcal{F}^{A}_{\mathcal{Y}}\right] \leq \delta \tag{2}$$

Substituting the definitions of δ_1 and δ_2: Condition (1) requires $\delta_1 \leq \delta$. Condition (2) requires $\delta_2 \leq \delta$.

For δ to be the minimal value satisfying both inequalities, it must be the smallest value that is greater than or equal to both δ_1 and δ_2. Therefore, the minimal $\delta = \max(\delta_1, \delta_2) = \max(\Pr\left[A(M(\mathcal{X})) \in \mathcal{F}_{\mathcal{X}}^A\right], \Pr\left[A(M(\mathcal{Y})) \in \mathcal{F}_{\mathcal{Y}}^A\right])$. □

The generalization from differential privacy against a single attack function to an entire attack class models realistic threat scenarios where we must defend against not just a single attack strategy, but an entire class of potential attacks. For example, in authorship attribution, we might need to protect against all possible n-gram frequency analyzers, not just one specific implementation.

It must be noted that the plausibility of (ε, δ) quantification depends on what background knowledge we assume available to attackers. This becomes tractable when either sufficient statistics are believably available or when the information content required to compute them is small enough to be realistically obtainable.

Definition 11 (Sufficient Statistics for Privacy Mechanisms)**.** *Let M be a randomized mechanism operating on dataset $\mathcal{X}$. A function SS_M mapping datasets to summary information is a sufficient statistic for M if:*

$$\Pr\left[M(\mathcal{X}) = o | SS_M(\mathcal{X}), \mathcal{X}\right] = \Pr\left[M(\mathcal{X}) = o | SS_M(\mathcal{X})\right]$$

That is, conditioning on the sufficient statistic makes the output distribution independent of the original dataset, meaning that knowing the complete dataset provides no additional information about the outputs of M outputs beyond what $SS_M(\mathcal{X})$ tells us.

Since ε represents the maximum over all plausible score functions, this assumes the adversary knows both which function is optimal and its corresponding sufficient statistics. For a restricted class of functions, we can view this as multiple attackers using different functions, where we must maintain security against all of them. The plausibility stems from our beliefs about what information adversaries can realistically access.

The framework becomes less abstract when we consider specific attacks. Take a surprisal-based attack that outputs the negative log probability of observing certain patterns. Here, $\mathsf{Range}(A) = \mathbb{R}$, meaning we no longer deal with probabilities but with real-valued scores. However, this complicates privacy analysis by introducing an infinite space of sufficient statistics – we have gone from the space of outputs of M to $\mathbb{R}$, which is even less tractable. When the output space is $\mathbb{R}$, the sufficient statistic becomes a function, making the adversary's core assumption – that they can know the complete statistical description of the output – impossible to satisfy. Rather, they must work with an approximation of this function.

This motivates us to consider a simpler but practical special case: attack functions with binary outputs that make concrete decisions about membership. This restriction to the space $\{0, 1\}$ leads us to examine probing attacks.

3.8 Probing attacks and limiting behaviour

The simplest attacks in text generation exploit statistical patterns in language use – if a person's writing contains rare words or distinctive phrases, finding these patterns in synthetic text suggests their data was used in training. This intuition is grounded in basic probability: rare linguistic features are less likely to appear by chance, indicating training data membership.

Let the target element be a specific data point p, which could represent an individual, a document, or any distinguishable unit in the dataset. Let us consider adversaries who try to determine if p is present in the dataset. The adversaries express their belief through a binary decision: one indicates presence in $\mathcal{X}$, zero indicates absence (presence in $\mathcal{Y}$). Such an adversary employs a classifier function that takes a mechanism M output as its input and returns their binary guess. We focus on the subclass of attack functions where the output range is restricted to {0,1}.

Example. Consider two simple binary classifier attacks on text data. First, let $A_w(o)$ be 1 if word w appears in output o, and 0 otherwise. This attack function directly tests for specific patterns the target p commonly uses. A more sophisticated version assigns a style score $s(o)$ to outputs based on n-gram frequencies or other stylometric features, then thresholds this score: $A_t(o) = 1$ if $s(o) \geq t$ and 0 otherwise. To illustrate the attack's effectiveness, one could construct a Receiver Operating Characteristic (ROC) curve by varying the threshold t and plotting the true positive rate against the false positive rate when distinguishing between outputs from $\mathcal{X}$ and $\mathcal{Y}$.

Note that such an attack function might operate on sufficient statistics rather than complete datasets. For example, a writing style classifier A might compute its score based on n-gram frequencies, word distributions, or other stylometric features that form sufficient statistics for authorship detection. Specifically for A_w, $SS_{M,A_w}(\mathcal{X})$ is a frequency set of all the possible outputs of M over $\mathcal{X}$: $\{(w, \text{freq}_{\mathcal{X}}(w)) : w \in \mathsf{Range}(M)\}$.

We have defined A_w as an indicator attack function for a specific word. To explore privacy guarantees, we can extend this to multiple words. The key question is: if we have sufficient information for each word, how do static and adaptive definitions converge to the same objective privacy measure?

An optimal adversary would try to minimize their uncertainty about whether element p belongs to the dataset. Without explicitly modeling the adversary's background knowledge, they would look for an output o that maximizes:

$$\underset{o}{\text{argmax}} \frac{\Pr\left[M(D) = o | p \in D\right]}{\Pr\left[M(D) = o | p \notin D\right]} \quad . \tag{3}$$

When we consider the adversary's background knowledge Bg, this becomes more complex as they must account for their beliefs about possible datasets (or, equivalently, their sufficient statistics):

$$\frac{\sum_{\mathcal{X}} \Pr\left[M(\mathcal{X}) = o\right] \cdot \Pr\left[\mathcal{X}|I, Bg\right]}{\sum_{\mathcal{Y}} \Pr\left[M(\mathcal{Y}) = o\right] \cdot \Pr\left[\mathcal{Y}|\neg I, Bg\right]} \quad . \tag{4}$$

For such an optimal o, we can construct a simple attack function:

$$A_o(x) = \begin{cases} 1, & x = o \\ 0, & \text{otherwise} \end{cases} . \tag{5}$$

As terms containing background knowledge are constant,

$$\exists A_o \in \mathfrak{A} : \frac{\Pr\left[A_o(M(\mathcal{X})) = 1\right]}{\Pr\left[A_o(M(\mathcal{Y})) = 1\right]} \iff \frac{\Pr\left[M(\mathcal{X}) = o\right]}{\Pr\left[M(\mathcal{Y}) = o\right]} \quad . \tag{6}$$

When the adversary's background knowledge equals complete knowledge of the datasets $\mathcal{X}$ and $\mathcal{Y}$ or the summary statistics $SS_M(\mathcal{X})$ and $SS_M(\mathcal{Y})$, in other words, when $\mathfrak{A}$ is not constrained, this formulation becomes equivalent to Definition 6. This is because with complete knowledge, $\Pr\left[\mathcal{X}|I, Bg\right] = 1$ for the true dataset and 0 for all others, eliminating the sums in our expressions (similarly for true and false summary statistics). The equivalence for $(\varepsilon, \delta) - DP$ follows similarly.

In practice, constraining $\mathfrak{A}$ to a specific class of binary attack functions allows us to tailor privacy metrics to realistic threats. This means that the attacker might not always have an optimal o to construct A_o^p. When working with a constrained subclass of $\mathfrak{A}$, we would like to ensure that the composition of attack functions from this subclass with our mechanism satisfies differential privacy – that is, for attack functions in this subclass, the composition $A \circ M$ is ε-differentially private (and similarly for (ε, δ)-DP):

$$\ln \frac{\Pr\left[A(M(\mathcal{X})) = 1\right]}{\Pr\left[A(M(\mathcal{Y})) = 1\right]} \leq \varepsilon \tag{7}$$

$$\ln \frac{\Pr\left[A(M(\mathcal{Y})) = 1\right]}{\Pr\left[A(M(\mathcal{X})) = 1\right]} \leq \varepsilon \quad . \tag{8}$$

When working with a class of binary attack functions (Range(A) = $\{0, 1\}$), we can apply Definition 9 directly. Note that for binary attacks, each failure set $\mathcal{F}^A$ can only be $\emptyset$, $\{0\}$, $\{1\}$, or $\{0, 1\}$. The case $\mathcal{F}^A = \{0, 1\}$ indicates catastrophic failure where both possible attack outputs exceed the privacy threshold. Thereby, the original differential privacy definition is specialized to the binary outputs of attack functions. While the original definition considered arbitrary subsets of the mechanism's range, and

Definition 9 extended it to an arbitrary output space, here we only need to consider subsets of $\{0, 1\}$.

This binary specialization simplifies the analysis, but fundamental computational challenges remain. With a binary subset of $\mathfrak{A}$, computing exact values for ε or δ would still require summing over all possible datasets or summary statistics weighted by background knowledge. While constraining $\mathfrak{A}$ to specific attack strategies (like threshold-based classifiers) makes the problem more concrete, it does not resolve the computational complexity or the issue of potentially incorrect beliefs encoded in the attack functions. Nonetheless, by focusing on implementable attack functions rather than theoretical bounds, we can attempt to empirically evaluate concrete privacy risks in realistic scenarios where adversaries have limited capabilities and potentially incorrect assumptions about the data.

3.9 Data synthesis with guaranteed privacy

The primary appeal of differential privacy lies in its ability to provide provable privacy guarantees for mechanisms by construction, rather than measuring privacy loss after the fact (as we have done). Such mechanisms eliminate the need to reason about possible summary statistics, background knowledge, or input datasets. For example, the exponential mechanism provides ε-DP by design through carefully calibrated noise addition.

Beyond general theoretical mechanisms, specific techniques have been developed to integrate differential privacy directly into machine learning workflows, aiming to produce models or outputs with built-in guarantees.

One established approach is training the model itself using DP-SGD [ACG$^+$16]. This method modifies the standard training procedure by clipping per-example gradients and adding noise before parameter updates. The resulting model parameters θ satisfy a cumulative (ε, δ)-DP guarantee with respect to the training data $\mathcal{X}$, typically calculated using techniques like the moments accountant [ACG$^+$16] that compose the privacy loss over all training iterations.

An alternative paradigm, particularly relevant when using large pre-trained models where private retraining is impractical, is private prediction at inference time [ABK$^+$24]. This approach focuses on ensuring the generated output sequence is differentially private with respect to sensitive inputs provided at inference (e.g., prompts containing private data). Techniques involve processing sensitive inputs in batches, aggregating and clipping model logits, and using sampling methods interpretable as the exponential mechanism. This method provides guarantees directly on the synthetic output, bypassing the complexities of privacy transfer from model parameters.

Both DP-SGD (providing guarantees on model parameters) and private prediction (providing guarantees on generated outputs) are approaches to achieving differential

privacy in synthetic data generation and both involve different implementation complexities and trade-offs compared to the post-hoc analysis framework used earlier in this thesis. These methods shift the privacy burden from post-hoc analysis to the design of the generation process itself. Although important alternative strategies, they are beyond the scope of the empirical evaluation carried out in this thesis.

Beyond these specific implementations, the core probabilistic sampling mechanism used within standard LLM inference itself is similar to the exponential mechanism, particularly when considering the role of the temperature parameter. Let us model $M(\mathcal{X})$ as a random query that combines the training of an LLM on a training dataset $\mathcal{X}$ and the subsequent generation of a single token based on some prompt, by means of composition described above.

In the exponential mechanism, the quality function q assigns utility scores to each possible output, with higher scores indicating more desirable outputs given the input data. Instead of deterministically selecting the highest-scoring output, the mechanism smooths the output distribution based on these scores – responses closer to the highest-utility output are still more likely, but not guaranteed. The weight assigned to an output r from the range R is:

$$w(r) = \exp\left(\frac{\varepsilon \cdot q(\mathcal{X}, r)}{2\Delta q}\right) \tag{9}$$

where Δq is the sensitivity of q – the maximum change in score when changing a single element in the input. The probability of selecting r is then:

$$\Pr[M(\mathcal{X}) = r] = \frac{w(r)}{\sum_{r' \in R} w(r')} \quad . \tag{10}$$

We see that as ε approaches 0, the output distribution approaches uniform over $\mathsf{Range}(M)$, providing maximum privacy but minimum utility. As ε approaches infinity, the mechanism becomes deterministic, always selecting the highest-scoring output.

Similarly, in LLMs, the softmax function with temperature scaling in the output layer assigns weights to each token in the vocabulary:

$$w(z_i) = \exp(\frac{z_i}{T}) \tag{11}$$

where z_i are the logits and T is the temperature parameter. The probability of selecting token i is then:

$$\Pr[y_i] = \frac{w(z_i)}{\sum_j w(z_j)} \quad . \tag{12}$$

If we consider the logits z_i as analogous to the scoring function $q(\mathcal{X}, r)$, and the temperature parameter T as inversely related to $\varepsilon/(2\Delta q)$, we can posit a threshold for T to impose DP-constraints on the generation of a single LLM token:

$$T = \frac{2\Delta q}{\varepsilon} \ . \tag{13}$$

This relationship suggests that higher temperature T might correspond to stronger privacy (lower effective ε), while lower T moves towards deterministic output, weakening privacy guarantees. However, formalizing this analogy into an (ε, δ)-DP guarantee faces significant hurdles. Firstly, defining and computing the necessary sensitivity term, Δq, is problematic. If interpreted as the sensitivity of the model's logits to changes in the training data, calculating this for complex LLMs is generally considered intractable. Secondly, the analogy implicitly assumes token-level independence, which is violated by the autoregressive nature of LLM generation. Therefore, while temperature offers an intuitive control over output randomness, achieving rigorous (ε, δ)-DP for the generated sequence requires explicit mechanisms for sensitivity analysis and bounding (such as logit clipping with respect to input prompts, as used in [ABK$^+$24]) and careful composition accounting over the generated tokens, techniques employed by dedicated private prediction methodologies. The temperature analogy remains a useful conceptual parallel but lacks the necessary components for providing formal privacy guarantees on its own.

4 Attacks against direct n-gram sampling

Building on the attack-based differential privacy framework established in Section 3, we first empirically evaluate privacy bounds for a baseline mechanism: direct sampling of n-grams. We focus on the simple case where outputs are generated by sampling individual words (unigrams) or short sequences (n-grams) according to their frequencies directly from an original dataset. This mechanism, while basic, provides insights into fundamental privacy limits inherent in the data's statistical properties, which any text generator preserving frequency distributions must contend with.

4.1 General framework

The core mechanism studied here samples n-grams according to their frequencies in the input dataset. Given a universe of sources S, we construct pairs of adjacent datasets $\mathcal{X}$ and $\mathcal{Y}$ by excluding a subset of sources $U \subseteq S$:

$$\mathcal{X} = \bigcup_{s \in S} \mathcal{D}_s \qquad \text{and} \qquad \mathcal{Y} = \bigcup_{s \in S \setminus U} \mathcal{D}_s \ .$$

The privacy analysis then compares the output distributions of n-grams when sampling directly from $\mathcal{X}$ and from $\mathcal{Y}$.

For the direct sampling mechanism operating on $\mathcal{X}$ or $\mathcal{Y}$, the empirical n-gram frequencies serve as sufficient statistics. Assuming an attacker has knowledge of these frequency distributions allows for the objective computation of the logarithmic Bayes factor ($LBF_{\mathcal{X},\mathcal{Y}}$) as defined in Section 3. We therefore precompute these frequency distributions for the objective baseline analysis.

Note that these baseline privacy measurements are influenced by source randomness. Evaluating worst-case privacy requires examining all possible source subsets U. Since this is intractable, we sample random subsets for fixed sizes $|U|$ and analyze their typical behavior. This approach provides insights into practical risk but does not guarantee bounds against the absolute worst-case source exclusion. Systematically identifying the worst-case subset U for n-gram sampling remains computationally challenging and is outside the scope of this work.

The primary goal of this section is to establish objective (ε, δ)-differential privacy bounds for simple n-gram sampling, providing a baseline against which more complex generation mechanisms can be compared later.

The analysis primarily focuses on the privacy implications of observing a single n-gram. If an adversary observes k n-grams, $o_1, \ldots, o_k$, sampled independently, the total logarithmic Bayes factor (LBF) can be approximated by summing individual LBFs: $LBF_{\mathcal{X},\mathcal{Y}}(o_1, \ldots, o_k) \approx \sum_{j=1}^{k} LBF_{\mathcal{X},\mathcal{Y}}(o_j)$. If each observation satisfies $|LBF_{\mathcal{X},\mathcal{Y}}(o_j)| \leq \varepsilon_{\text{single}}$ (outside a failure event of probability δ_{single}), then k such observations could lead to a total $|LBF_{\mathcal{X},\mathcal{Y}}(\text{sequence})| \lesssim k \cdot \varepsilon_{\text{single}}$. This basic composition

implies that maintaining a target overall privacy loss $\varepsilon_{\text{total}}$ for k n-grams requires a per-n-gram budget $\varepsilon_{\text{single}} \approx \varepsilon_{\text{total}}/k$. Consequently, the (ε, δ)-curve for publishing k words would shift towards higher ε values (potentially by a factor of k) for a given δ, or require a significantly higher δ (e.g., up to $k \cdot \delta_{\text{single}}$ via union bound) to maintain the same ε. This linear summation is a basic heuristic; advanced composition offers tighter bounds but is beyond this section's scope. The single n-gram results presented subsequently form the basis for such compositional analysis.

4.2 Datasets

Evaluating privacy guarantees with a focus on membership inference requires datasets with clear source attribution and representative text distributions. The privacy loss incurred as a result of an attack depends on the particular features of a dataset. For example, if a dataset has few unique elements or repetitive data, a privacy attack is less likely to be successful compared to a dataset that is known to have many unique elements. We use two datasets.

The Cheng-Caverlee-Lee (CCL) Twitter dataset [CCL10] (hereafter $\mathcal{D}_{\text{CCL}}$) serves as training data, containing 9,000,659 tweets from 21,022 users collected between September 2009 and January 2010. We chose this dataset as it provides public data for reproducible research, clear source attribution via user IDs, natural language distributions, and diverse writing styles across topics.

We also evaluate privacy guarantees on the MAITT [OTM+23] dataset (hereafter $\mathcal{D}_{\text{MAITT}}$), an internal medical text dataset containing epicrisis documents from Estonian hospitals. This represents a practical use case where privacy preservation is critical. The dataset is source-attributed to individual patients (`pat_id`), allowing source-wise privacy analysis. For the direct sampling analysis presented in Section 4.4, the entirety of the available word count data derived from this dataset was used, essentially acting as a summary statistic.

The underlying electronic health records (EHRs) are structured documents comprising multiple sections (e.g., anamnesis, objective status, diagnosis, treatment plan). However, for the purpose of the word-count analysis performed here, this structure was simplified: all textual content associated with a patient was effectively treated as a single collection of words attributed to that patient source. The specific table used aggregated word counts per patient across all their associated text fields.

For both the `MAITT` dataset and the subset of the CCL dataset, we construct pairs $(\mathcal{X}, \mathcal{Y})$ that differ by excluding specific sources from $\mathcal{Y}$, with exclusion set sizes $|U| \in \{1, 10, 100, 1000\}$. This range covers scenarios from individual to group privacy. For later evaluation of attacks on synthetic data, we also generate a single pair of synthetic datasets with $|U| = 100$ based on the subset of the CCL dataset.

Table 1 summarizes some statistics for the data subsets used. Due to computational constraints, only the first 10% of the $\mathcal{D}_{\text{CCL}}$ dataset was used. The full $\mathcal{D}_{\text{MAITT}}$ word count

data was used for direct sampling analysis.

Table 1. Summary statistics for datasets used.

Statistic	$\mathcal{D}_{\text{CCL}}$ (10% subset)	$\mathcal{D}_{\text{MAITT}}$ (Word Counts)
Total word occurrences	∼119 million	∼66 million
Unique sources (Users/Patients)	∼106 thousand	∼146 thousand
Unique words (Vocabulary)	∼2.7 million	∼4.4 million
Avg. words per source	∼1,121	∼2,508
Avg. distinct words per source	∼378	∼1,233

4.3 Privacy measurement

To quantify the privacy loss associated with direct n-gram sampling, we employ the objective differential privacy framework (Section 3) and compute the (ε, δ) trade-off curve using the method from Theorem 5.

Consider the mechanism M that directly samples an item o (unigram or n-gram) from dataset $\mathcal{D} \in \{\mathcal{X}, \mathcal{Y}\}$. The probability is its relative frequency, $\Pr[M(\mathcal{D}) = o] = f_{\mathcal{D}}(o)$. These frequency distributions, $f_{\mathcal{X}}$ and $f_{\mathcal{Y}}$, are the sufficient statistics. For each possible output o, we compute the LBF:

$$\begin{aligned} LBF_{\mathcal{X},\mathcal{Y}}(o) &= \ln\left(f_{\mathcal{X}}(o)/f_{\mathcal{Y}}(o)\right) \\ LBF_{\mathcal{Y},\mathcal{X}}(o) &= \ln\left(f_{\mathcal{Y}}(o)/f_{\mathcal{X}}(o)\right) = -LBF_{\mathcal{X},\mathcal{Y}}(o) \quad . \end{aligned}$$

Infinite LBF occurs if o exists in one dataset but not the other ($f_{\mathcal{Y}}(o) = 0, f_{\mathcal{X}}(o) > 0$), implying $\delta > 0$ is required for any finite ε.

Given $\varepsilon > 0$, the failure sets are:

$$\begin{aligned} \mathcal{F}_{\mathcal{X}} &= \{o \in \mathsf{Range}(M) : LBF_{\mathcal{X},\mathcal{Y}}(o) > \varepsilon\} \\ \mathcal{F}_{\mathcal{Y}} &= \{o \in \mathsf{Range}(M) : LBF_{\mathcal{Y},\mathcal{X}}(o) > \varepsilon\} \quad . \end{aligned}$$

The minimal failure probability δ for this ε is:

$$\delta(\varepsilon) = \max\left(\sum_{o \in \mathcal{F}_{\mathcal{X}}} f_{\mathcal{X}}(o), \sum_{o \in \mathcal{F}_{\mathcal{Y}}} f_{\mathcal{Y}}(o)\right) \quad .$$

Evaluating this across ε traces the (ε, δ)-curve for the direct sampling mechanism. This applies identically to unigrams and n-grams using their respective frequencies.

4.4 Results and analysis of direct sampling

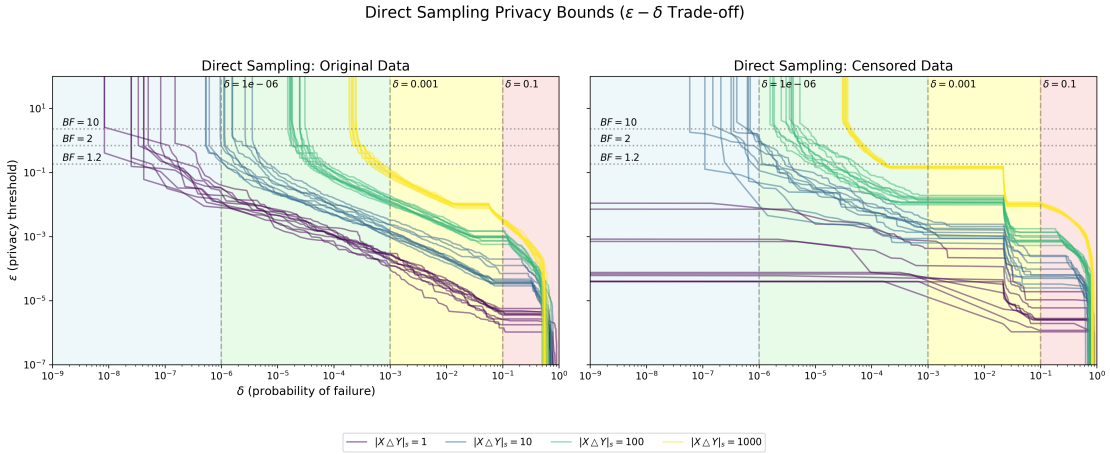
We sample words from datasets with source attribution. As evaluating privacy for all possible excluded source subsets U is computationally intractable, we analyze random subsets for various sizes $|U|$ to understand typical privacy behavior relevant for practical risk assessment. Our experiments reveal several findings about failure probabilities when sampling words directly from datasets with source attribution.

1. Failure set behavior and dataset comparison. The trade-off between privacy loss ε and failure probability δ is visualized in Figure 3, with vertical lines indicating critical δ thresholds and horizontal lines marking significant jumps in posterior probabilities (Bayes factors). For the CCL dataset (Figure 3a, left panel – original data), the probability mass of failure sets $\mathcal{F}_{\mathcal{X}}$ and $\mathcal{F}_{\mathcal{Y}}$ grows with the number of excluded sources $|U|$. This trend is also directly illustrated in Figure 4, which shows how individual (ε, δ) points grow together with $|U|$. This is further explored in Figure 5, which demonstrates a power-law relationship between ε and $|U|$ at a fixed δ level. This is intuitive: the more sources are excluded when constructing $\mathcal{Y}$ from $\mathcal{X}$, the more the probability ratios of words tend to diverge, increasing the mass of the failure sets at any given threshold ε. This is particularly true for sources that provide rare words; for such words, the probability ratios change the most. In the worst case, if a source that is the unique provider of some word to the dataset is removed, this results in infinite $LBF_{\mathcal{X},\mathcal{Y}}$.

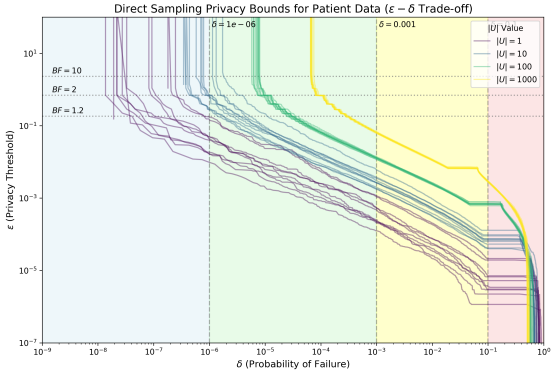
A question arises when interpreting these curves for practical privacy: can acceptable privacy (ε) be achieved with a tolerable failure probability (δ)? For the CCL dataset with $|U| = 1000$ excluded users, achieving a low failure probability such as $\delta = 10^{-6}$ is infeasible. At $\delta = 10^{-3}$, the corresponding $\varepsilon \approx \ln(1.2) \approx 0.18$, indicating that an adversary's posterior odds of correct membership inference can increase by a factor of 1.2 when observing a single sampled word. If multiple (e.g., $k = 10$) words are sampled, then under naive composition (simple summation of ε values), the cumulative privacy loss increases to approximately $\varepsilon \approx 1.8$, which allows for non-negligible inference risk. Achieving a tighter privacy guarantee, such as $\varepsilon = \ln(1.01) \approx 0.01$, would require accepting a substantially larger failure probability, e.g., $\delta \approx 10^{-1}$ for $|U| = 1000$.

The direct sampling analysis performed on the medical epicrises dataset ($\mathcal{D}_{\text{MAITT}}$) yields qualitatively similar (ε, δ) trade-off curves (Figure 3b). Larger exclusion sets $|U|$ lead to higher privacy loss (rightward shift of curves). This confirms that basic statistical properties relevant to differential privacy in direct sampling manifest similarly across different text types, from social media to medical records. The appropriate choice of (ε, δ) ultimately depends on the application's acceptable risk profile for the specific data type.

2. Effect of censoring (CCL dataset). In the censored CCL data (Figure 3a, right panel), words with count $|X| \leq 2$ were replaced with a `<CENSORED>` token. This



(a) CCL dataset: original (left) and censored (right).



(b) MAITT dataset.

Figure 3. Trade-off between privacy loss ε and failure probability δ for direct word sampling. (a) Compares original and censored CCL data. (b) Shows results for the MAITT dataset. Larger $|U|$ corresponds to higher overall ε. Vertical lines at $\delta = 10^{-6}$, $\delta = 10^{-3}$, $\delta = 10^{-1}$ indicate failure probability thresholds. Horizontal lines at $\varepsilon = \ln(10), \ln(2), \ln(1.2)$ mark posterior probability jumps.

replacement results in a noticeable shift of the curves to the left. Specifically, when $|U| = 1$, we observe that ε never reaches infinity, even with $\delta = 0$, since rare words (those with lower counts) are censored, preventing extreme divergence between $\mathcal{X}$ and $\mathcal{Y}$. At higher $|U|$, where fewer sources are excluded, the effect of censoring becomes less pronounced, and the curves resemble the original non-censored data. This illustrates that censoring helps mitigate extreme privacy loss in cases where rare words dominate the dataset, leading to a more stable trade-off between ε and δ.

For $|U| = 1$, censoring words with counts ≤ 2 prevents infinite ε at $\delta = 0$. For larger $|U|$ (e.g., $|U| = 1000$), it still reduces privacy risk but leaves non-trivial leakage:

$\varepsilon \approx \ln(1.2)$ at $\delta = 10^{-3}$. Simple count-based censoring alone does not guarantee strong privacy at tight benchmarks (e.g., $\varepsilon \leq \ln(1.01)$). While it dampens the largest LBF spikes, it does not eliminate distinguishability arising from frequency shifts in common words. For highly sensitive data (e.g., genetic), additional mechanisms may be required.

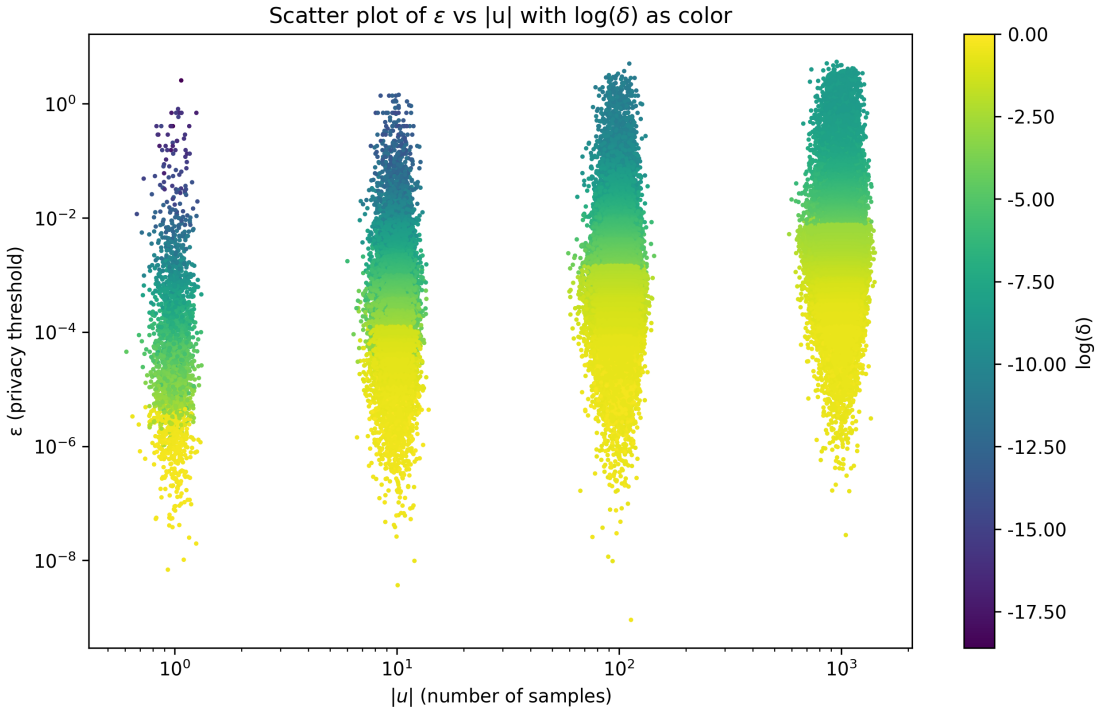


Figure 4. A scatterplot of ε (y-axis) and δ (color) for different $|U|$ when sampling words directly from the (non-synthetic) CCL dataset. As $|U|$ (x-axis) increases, so do both ε and δ, moreover, at the same ε threshold δ increases rapidly with $|U|$. Note that $|U|$ is categorical, with jitter applied to the x-axis for clarity.

3. Limiting behavior. We observe cases where $\mathcal{F}_{\mathcal{X}}$ contains words that appear in $\mathcal{X}$ but not in $\mathcal{Y}$ (or vice versa), giving $LBF \to \infty$. These words will always be in the failure set regardless of ε, creating a minimum $\delta > 0$ below which no finite ε privacy guarantee is possible. For example, in the CCL dataset analysis for $|U| = 100$ (Figure 3a, left panel), the curve shoots upwards around $\delta \approx 10^{-5}$. This means any attempt to guarantee privacy with $\varepsilon < \infty$ must accept a failure probability of at least 10^{-5}, because this fraction of the probability mass corresponds to words exclusively present in $\mathcal{X}$ or $\mathcal{Y}$. For $|U| = 1000$, this minimum δ increases to roughly 10^{-4}. This floor on δ represents the inherent leakage from unique identifiers (in this case, unique words tied to the excluded sources) and is visible in both CCL and MAITT datasets. Censoring (Figure 3a, right panel) mitigates

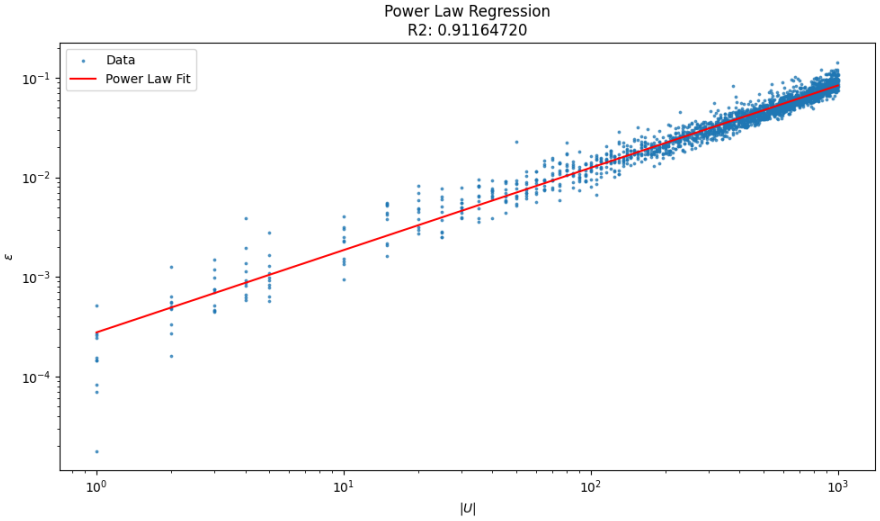


Figure 5. Behavior of ε at $\delta = 0.001$ as a function of $|U|$. The power law fit $\varepsilon = 0.00027764 \cdot |U|^{0.8267}$ ($R^2 = 0.9116$) reveals sublinear scaling. This improves upon the linear $O(|U|)$ dependence (implied by the triangle inequality for pure DP, [DR14]) but does not achieve the $O(\sqrt{|U|})$ bound of advanced composition [DRV10].

this by reducing the number of words unique to $\mathcal{X}$ or $\mathcal{Y}$. For $|U| = 1$ with censoring, infinite ε at $\delta = 0$ is prevented.

4. n-gram effects and sparsity. Figure 6 shows the empirical complementary cumulative distribution function (CCDF) of n-gram counts for a single representative source from the CCL dataset (comparing 1-grams and 3-grams) and the MAITT dataset (1-grams). This plot illustrates the sparsity of n-grams, particularly longer ones. For the sampled CCL user, 67.1% of their unigrams and a significantly higher 97.1% of their trigrams occur only once (i.e., are unique to that user within their own text corpus). For the sampled MAITT patient, 71.8% of their unigrams occur only once within their texts. While trigram data for MAITT is not shown, it is expected that trigrams would exhibit even greater sparsity than unigrams in the medical texts as well.

This high prevalence of unique n-grams, especially for trigrams, has direct privacy implications. If a source (e.g., a user or patient) contributes many n-grams that are unique globally across the entire dataset (or at least unique to a small set of sources including them), then the presence of such an n-gram in any output (e.g., a directly sampled snippet or synthetically generated text) can act as a strong identifier for that source. For example, if 97.1% of a user's trigrams are unique within their own text, and a substantial fraction of these are also rare or unique across the whole dataset, then sampling one such trigram provides strong evidence about its origin. Consider a scenario where 100 users (1% of a 10,000 user dataset like CCL) are excluded. If these users contribute a proportional share

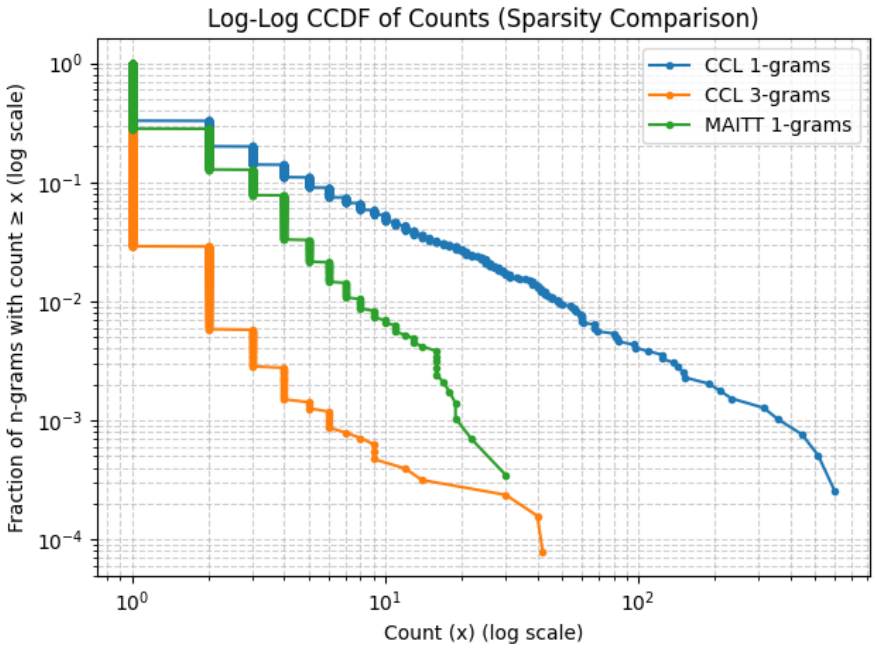


Figure 6. Log-log empirical CCDF of n-gram counts for a single source from CCL (1-grams and 3-grams) and MAITT (1-grams), illustrating n-gram sparsity. For the sampled CCL user, 67.1% of their unigrams and 97.1% of their trigrams occur only once within their own texts. For the sampled MAITT patient, 71.8% of their unigrams occur only once within their texts.

of trigrams, and a large fraction of these (e.g., approaching 90-95%) are globally unique to them as a group, then randomly sampling a trigram from the full dataset has a roughly $0.01 \times 0.95 \approx 0.95\%$ chance of perfectly identifying one of the excluded users, leading to an infinite LBF, and also setting a floor for the minimum failure probability δ.

Comparing unigram and trigram failure sets for the CCL dataset (Figure 7) reveals distinct scaling behaviors for privacy loss ε (at fixed $\delta = 0.01$) as a function of the number of excluded users $|U|$. Both show approximately linear scaling of $\log(\varepsilon)$ with $\log(|U|)$, consistent with the power-law relationship observed earlier (Figure 5), but trigrams exhibit a steeper slope.

Counter-intuitively, at $|U| = 1$, trigram attacks incur lower privacy loss (ε) than unigrams for this fixed $\delta = 0.01$. This occurs despite trigrams being sparser (Figure 6), which leads to more unique items with infinite LBF and thus a higher minimum achievable failure probability ($\delta_{min,tri} > \delta_{min,uni}$). The value ε is determined by the LBF threshold where the cumulative probability of the failure set reaches the target $\delta = 0.01$. The explanation lies in the distribution of LBF values for non-unique items. For unigrams, removing one source likely perturbs the frequencies of many common

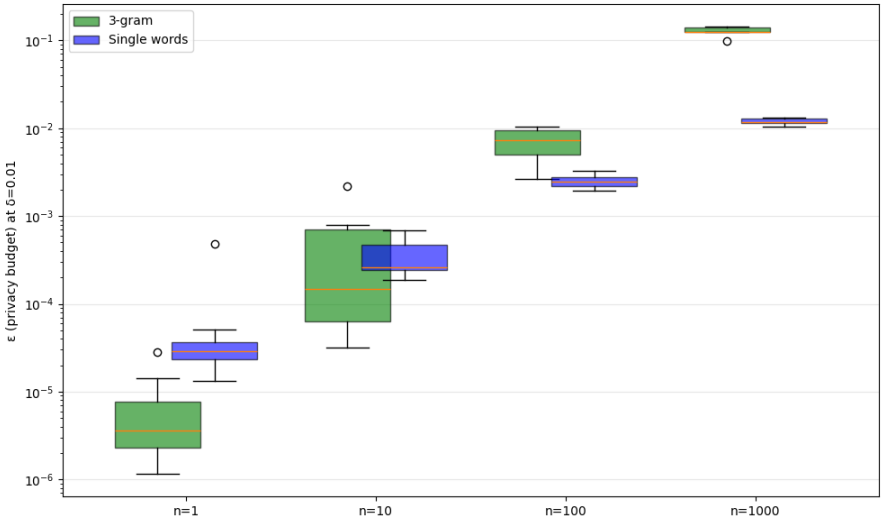


Figure 7. Privacy loss ε at fixed $\delta = 0.01$ (1% permitted failure probability) for different numbers of excluded users $|U|$, comparing unigram (blue) and trigram (green) attack functions when sampling directly from the CCL dataset. Each box represents 10 random trials. Note the logarithmic scale on the y-axis. Both attacks show approximately linear scaling of $\log(\varepsilon)$ with $\log(|U|)$, but trigrams exhibit steeper scaling, crossing over unigrams between $|U| = 10$ and $|U| = 100$ despite starting from a lower base value.

words slightly, creating a large pool of items with small-to-moderate LBFs. Reaching $\delta = 0.01$ might require accumulating probability mass down to a relatively high ε_{uni} threshold that includes many of these moderately distinguishing words. For trigrams, after accounting for the unique items contributing to $\delta_{min,tri}$, the remaining non-unique trigrams might include some with large finite LBFs (due to larger relative frequency changes upon source removal). However, if these high-LBF trigrams are themselves very infrequent, their contribution to the cumulative failure probability might be small. Reaching the target $\delta = 0.01$ could then necessitate including trigrams with only modest LBF values, resulting in a lower overall threshold $\varepsilon_{tri} < \varepsilon_{uni}$.

As $|U|$ increases, removing more sources drastically increases the LBF values for many more n-grams. The steeper scaling for trigrams shows they are more sensitive to this increased distinguishability, likely because their baseline frequencies are lower, causing their LBF values to increase more rapidly than those of more robust unigrams. This leads to the observed crossover between $|U| = 10$ and $|U| = 100$.

5. High-δ behavior. The $\varepsilon - \delta$ curves (Figure 3) show that ε approaches zero as δ increases beyond the probability mass of the most frequent words in either dataset. This occurs because once δ is high enough to cover all words with significantly different

frequencies between datasets, the remaining words must necessarily have similar frequencies, allowing for very small privacy losses ε. This reinforces the notion that privacy challenges in direct sampling are dominated by the tails of the frequency distribution.

6. High values of LBF correlate with low probability $P_{\mathcal{X}}$. The objective analysis assumes the adversary knows the complete frequency distributions $f_{\mathcal{X}}$ and $f_{\mathcal{Y}}$. This is unrealistic. A more practical adversary might only know or estimate frequencies for a limited subset of potentially distinguishing words. Understanding which words typically exhibit high $|LBF|$ values is thus crucial for assessing risks under such limited knowledge, as these words are candidates for ε-violators that an attacker might focus on. The relationship between $P_{\mathcal{X}}$ and LBF, explored below, informs which words are likely to be such violators. This aspect is further explored in the context of subjective privacy loss in Section 5.4 (Figure 13). To investigate which words contribute most to privacy violations in direct sampling, we examined the relationship between sampling probability $P_{\mathcal{X}}$ and privacy impact ($LBF_{\mathcal{X},\mathcal{Y}}$ and $LBF_{\mathcal{Y},\mathcal{X}}$).

Figure 8 shows LBF over the sampling probability for the CCL dataset, which exhibits a negative correlation between LBF magnitude and sampling probability $P_{\mathcal{X}}$. The visualization reveals a strong negative trend on a log-log scale: higher-frequency words consistently show lower absolute LBF values (lower privacy impact). The largest LBF magnitudes occur almost exclusively for words with very low sampling probabilities ($P_{\mathcal{X}} < 10^{-5}$). This supports the intuition that privacy violations stem primarily from rare words. The shift in the (ε, δ) trade-off observed in the censored CCL data (Figure 3a, right panel) further suggests that censoring rare words helps mitigate privacy risks by removing words with high LBF values.

Figure 9 shows the LBF over the sampling probability for the MAITT patient dataset, which also exhibits a similar negative correlation to the CCL data (Figure 8). In both Figure 8 and Figure 9, a notable feature for the $LBF_{\mathcal{Y},\mathcal{X}}$ values (red points) is the potential for a pronounced horizontal artifact. This occurs for words present in the dataset $\mathcal{Y}$ (i.e., after exclusion of sources U) whose individual counts are identical in $\mathcal{X}$ and $\mathcal{Y}$ (meaning they were not contributed by any source in U), but the total word count of dataset $\mathcal{Y}$ is smaller than that of $\mathcal{X}$. Consequently, their sampling probability increases by a constant factor. This results in a constant positive $LBF_{\mathcal{Y},\mathcal{X}}$ for all such words, creating the observed horizontal line if many such words exist.

These direct sampling results on patient data provide the relevant baseline for subsequent analyses involving synthetic medical text generation. When considering synthetic data generated by an LLM, the privacy guarantees depend on how accurately the model reproduces the original frequency distributions. If an LLM perfectly preserves unigram frequencies, the (ε, δ)-curve would match Figure 3. However, deviations introduced during generation will alter the LBF values (as discussed conceptually in Section 5.2).

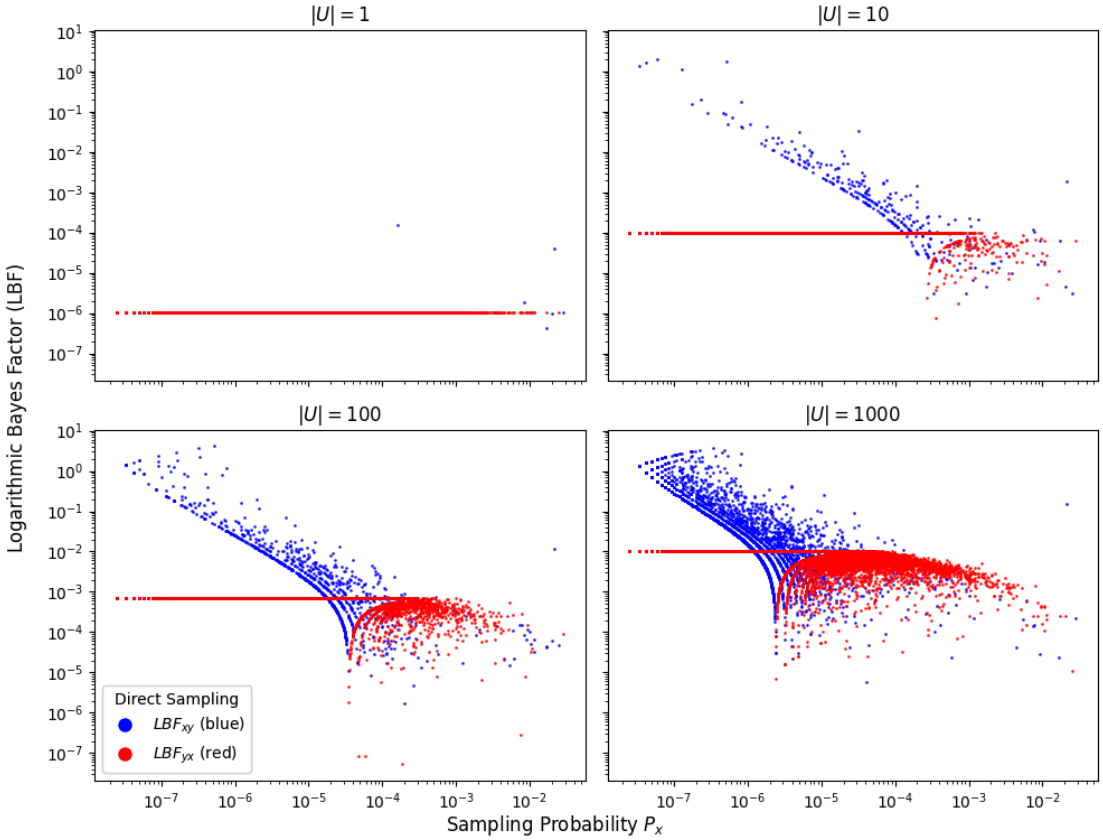


Figure 8. Relationship between sampling probability ($P_{\mathcal{X}}$) and logarithmic Bayes factor (LBF) for direct sampling across different exclusion set sizes $|U|$. Each point represents a unigram. Blue points show $LBF_{\mathcal{X},\mathcal{Y}}$ and red points show $LBF_{\mathcal{Y},\mathcal{X}}$. A negative relationship is observed on the log-log scale: higher probability words exhibit lower absolute LBF values. Notably, for $LBF_{\mathcal{Y},\mathcal{X}}$, the words in $\mathcal{Y}$ whose counts remain unaffected by the exclusion of U are scaled up by a constant factor as the total sum of counts decreases, causing a straight horizontal artifact.

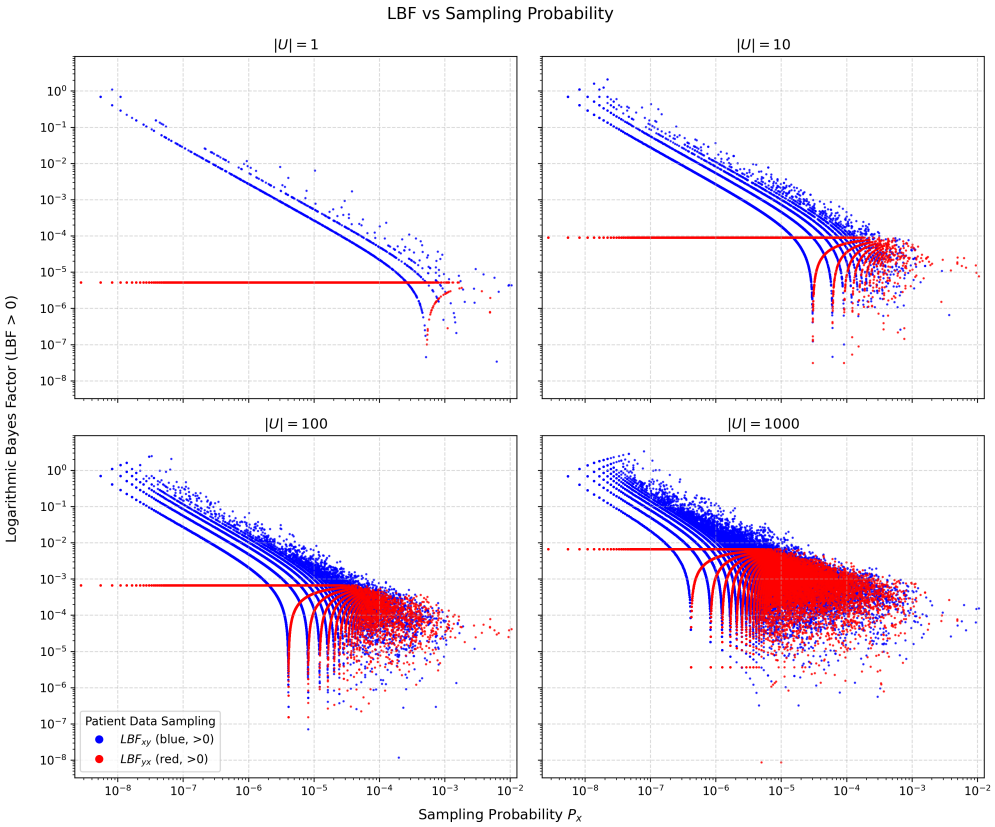


Figure 9. LBF vs Sampling Probability (P_x) for Patient Data ($\mathcal{D}_{\text{MAITT}}$) with Direct Sampling. Facets show different exclusion set sizes $|U|$. Only LBF > 0 is plotted due to the log scale on the y-axis. Note the pronounced horizontal artifact for LBF_{yx} (red).

5 Privacy analysis of LLM-generated synthetic data

Having established baseline privacy bounds for direct n-gram sampling in Section 4, we now turn to evaluating synthetic data generated by an LLM as a potential privacy-enhancing mechanism. The central question is whether the process of model training and stochastic generation obscures the distinguishing features present in the original data, offering better privacy protection than direct sampling.

5.1 General framework for synthetic data evaluation

To compare the baseline with generative approaches, we evaluate a synthetic data mechanism. For this, we fine-tune an LLM separately on adjacent datasets $\mathcal{X}$ and $\mathcal{Y}$ (defined as in Section 4, with $|U| = 100$ for this specific evaluation) to obtain models $M_{\mathcal{X}}$ and $M_{\mathcal{Y}}$. We then generate synthetic datasets $S_{\mathcal{X}}$ and $S_{\mathcal{Y}}$ from these models, matching the size of the original datasets. The privacy analysis compares the statistical properties (empirical unigram frequencies) of $S_{\mathcal{X}}$ versus $S_{\mathcal{Y}}$.

The sources of randomness relevant here include source randomness (affecting $\mathcal{X}, \mathcal{Y}$), training randomness (LLM initialization/optimization affecting $M_{\mathcal{X}}, M_{\mathcal{Y}}$), and generation randomness (stochastic sampling creating $S_{\mathcal{X}}, S_{\mathcal{Y}}$). Our evaluation uses single instances for $M_{\mathcal{X}}, M_{\mathcal{Y}}$ and $S_{\mathcal{X}}, S_{\mathcal{Y}}$. Analyzing multiple training runs and multiple generation runs for each pair $(\mathcal{X}, \mathcal{Y})$ was computationally infeasible given project constraints (time, resources). While providing only a snapshot, this single-instance analysis allows initial comparison but, as discussed later, limits the reliability of conclusions about the mechanism's inherent privacy.

The sufficient statistics for this analysis are the empirical unigram frequencies within the generated synthetic datasets $S_{\mathcal{X}}$ and $S_{\mathcal{Y}}$. Using empirical frequencies from single generated datasets $S_{\mathcal{X}}, S_{\mathcal{Y}}$ serves two purposes. First, it allows a direct comparison to the baseline direct sampling analysis using equivalent statistics. Second, it reflects the information an attacker might realistically obtain by observing a large sample of synthetic output. These empirical frequencies can be seen as noisy estimates of the true expected frequencies from the models $M_{\mathcal{X}}, M_{\mathcal{Y}}$. If we assume the generation process is stable, these empirical frequencies might approximate the mean frequencies.

When analyzing the privacy of the synthetic data itself, the relevant sufficient statistic SS for an attacker observing the synthetic outputs, given a sampling mechanism, is the empirical frequency distribution within the synthetic dataset, i.e., $f'_{S_{\mathcal{X}}}(w)$ and $f'_{S_{\mathcal{Y}}}(w)$. The attacker uses these observed synthetic frequencies (or probabilities derived from them) to compute their LBF and make inferences. The original frequencies $f_{\mathcal{X}}(w), f_{\mathcal{Y}}(w)$ are relevant for understanding how much the generator distorts the original data and how this distortion impacts privacy, but the direct basis for the privacy calculation in the synthetic scenario is formed by the frequency distributions of the synthetic data itself.

5.2 Synthetic sampling strategy and LBF

To assess whether generating synthetic data provides meaningful privacy benefits over direct data sharing, we now consider a scenario where the mechanism M involves an additional step. Instead of sampling directly from the original datasets $\mathcal{X}$ or $\mathcal{Y}$, we first train a generative model (e.g., an LLM) on $\mathcal{X}$ (resulting in model $M_{\mathcal{X}}$) and $\mathcal{Y}$ (resulting in model $M_{\mathcal{Y}}$). Then, we generate synthetic samples $S_{\mathcal{X}} \sim M_{\mathcal{X}}$ and $S_{\mathcal{Y}} \sim M_{\mathcal{Y}}$. The privacy analysis then compares the distributions of these synthetic outputs. The central question is whether the generative process itself, through its inherent randomness, smoothing effects, or potential failure to perfectly memorize, adds a layer of privacy protection beyond what is achieved by simply sampling from the original data.

If the generation process significantly alters the probability distributions compared to the original data, it might obscure the differences between $M(\mathcal{X})$ and $M(\mathcal{Y})$, potentially reducing privacy loss, though likely at the cost of data utility. Conversely, if the generator faithfully reproduces the original distributions, the privacy guarantees may not improve substantially, rendering the synthesis step potentially ineffective in terms of differential privacy improvement over direct sampling.

A key factor in this analysis is how well the generative model preserves the statistical properties of the original data, particularly the marginal word distributions. Since any effective text generator should aim to approximate these distributions, privacy violations observed in simple unigram sampling indicate limitations that synthetic data generation must also overcome. Let $f_{\mathcal{X}}(w)$ and $f_{\mathcal{Y}}(w)$ be the true frequencies (or probabilities) in the original datasets. A generator might produce synthetic data where the empirical frequency of word w is $f'_{S_{\mathcal{X}}}(w) = f_{\mathcal{X}}(w) + \delta_{\mathcal{X}}(w)$ when trained on $\mathcal{X}$, and $f'_{S_{\mathcal{Y}}}(w) = f_{\mathcal{Y}}(w) + \delta_{\mathcal{Y}}(w)$ when trained on $\mathcal{Y}$. The privacy loss for observing word w in the synthetic output is then determined by the ratio of these perturbed frequencies:

$$LBF_{\text{synth}}(w) = \ln\left(\frac{f'_{S_{\mathcal{X}}}(w)}{f'_{S_{\mathcal{Y}}}(w)}\right) = \ln\left(\frac{f_{\mathcal{X}}(w) + \delta_{\mathcal{X}}(w)}{f_{\mathcal{Y}}(w) + \delta_{\mathcal{Y}}(w)}\right) \quad .$$

Whether this synthetic LBF offers better privacy (i.e., is smaller in magnitude) than the LBF from direct sampling depends on the nature and magnitude of the deviations $\delta_{\mathcal{X}}(w)$ and $\delta_{\mathcal{Y}}(w)$ introduced by the generator.

We can approximate the change in LBF compared to direct sampling due to these perturbations using a first-order Taylor expansion around the original frequencies:

$$\begin{aligned}
\Delta LBF(w) &= LBF_{\text{synth}}(w) - LBF_{\text{direct}}(w) \\
&\approx \frac{\partial LBF_{\text{direct}}}{\partial f_{\mathcal{X}}}\delta_{\mathcal{X}}(w) + \frac{\partial LBF_{\text{direct}}}{\partial f_{\mathcal{Y}}}\delta_{\mathcal{Y}}(w) \\
&= \frac{\delta_{\mathcal{X}}(w)}{f_{\mathcal{X}}(w)} - \frac{\delta_{\mathcal{Y}}(w)}{f_{\mathcal{Y}}(w)} \quad .
\end{aligned}$$

This approximation highlights that the relative changes in frequency (δ/f) are crucial. For instance, consider a scenario where the perturbations are roughly proportional to the frequencies, e.g., $|\delta_{\mathcal{X}}(w)| \approx 0.1 \cdot |f_{\mathcal{X}}(w)|$ and $|\delta_{\mathcal{Y}}(w)| \approx 0.1 \cdot |f_{\mathcal{Y}}(w)|$. In this case, the change in LBF is approximately bounded by $|0.1 \pm 0.1| \leq 0.2$ (assuming worst-case opposite signs for the relative errors). This suggests that if the generator introduces errors that scale with frequency (e.g., a consistent 10% relative error), the impact on LBF might be relatively controlled across different words.

However, generative models might introduce errors that do not scale proportionally. It is plausible to assume that the generation process might add or subtract a small absolute number of occurrences for any given word, perhaps 1-2 instances, regardless of its original frequency. Consider a rare word w with $f_{\mathcal{X}}(w) = 5$ occurrences and $f_{\mathcal{Y}}(w) = 2$ occurrences. The direct sampling $LBF_{\text{direct}}(w) = \ln(5/2) \approx 0.92$. If the generator adds 1 occurrence when trained on $\mathcal{X}$ ($\delta_{\mathcal{X}} = 1$) and subtracts 1 when trained on $\mathcal{Y}$ ($\delta_{\mathcal{Y}} = -1$), the synthetic frequencies become $f'_{S_{\mathcal{X}}}(w) = 6$ and $f'_{S_{\mathcal{Y}}}(w) = 1$. The resulting $LBF_{\text{synth}}(w) = \ln(6/1) \approx 1.79$, a significant increase in privacy loss. Conversely, if $\delta_{\mathcal{X}} = -1$ and $\delta_{\mathcal{Y}} = 1$, then $f'_{S_{\mathcal{X}}}(w) = 4$, $f'_{S_{\mathcal{Y}}}(w) = 3$, and $LBF_{\text{synth}}(w) = \ln(4/3) \approx 0.29$, a decrease in privacy loss.

This demonstrates that for infrequent words, even small absolute discrepancies in counts introduced by the generator can dramatically alter the LBF, potentially worsening or improving privacy for specific outputs compared to direct sampling. If the synthetic generation process perfectly preserved the original word probability distributions from $\mathcal{X}$ and $\mathcal{Y}$ into $S_{\mathcal{X}}$ and $S_{\mathcal{Y}}$ respectively, then $LBF_{\text{synth}}(w)$ would equal $LBF_{\text{direct}}(w)$ for all words w. However, any deviation or noise introduced by the generator means $P(w|S_{\mathcal{X}}) \neq P(w|\mathcal{X})$ or $P(w|S_{\mathcal{Y}}) \neq P(w|\mathcal{Y})$. Such distortions can either amplify or dampen the original LBF. If the generator, for instance, makes a word w (that was more indicative of $\mathcal{X}$ than $\mathcal{Y}$) even more disproportionately likely in $S_{\mathcal{X}}$ compared to $S_{\mathcal{Y}}$, then $|LBF_{\text{synth}}(w)| > |LBF_{\text{direct}}(w)|$, exacerbating privacy loss for that word. Conversely, if the generator tends to smooth distributions, for example by censoring very rare words or by making their probabilities in $S_{\mathcal{X}}$ and $S_{\mathcal{Y}}$ more similar, then $|LBF_{\text{synth}}(w)|$ could decrease, improving privacy for those specific words, often at the cost of utility. The scatter plots in Figure 12 illustrate that such deviations from original probabilities occur. The net impact on LBF (panel (a)) depends on how these generator-induced changes affect the ratio $P(w|S_{\mathcal{X}})/P(w|S_{\mathcal{Y}})$ compared to the original $P(w|\mathcal{X})/P(w|\mathcal{Y})$. A detailed discussion of these observed deviations is in Section 5.4.

5.3 Privacy measurement for synthetic data

The subsequent analysis compares direct sampling results with synthetic data. For the synthetic data evaluation, both objective and subjective, we compute privacy loss based on the empirical unigram frequencies observed in single instances of generated synthetic datasets $S_{\mathcal{X}}$ and $S_{\mathcal{Y}}$ derived from models $M_{\mathcal{X}}$ and $M_{\mathcal{Y}}$ respectively, (as described in

Subsection 5.1). This approach allows for direct comparison but, as discussed at the end of Section 5.4, carries important limitations regarding the handling of generation randomness.

When evaluating the privacy of synthetic data generation, the mechanism M encompasses both training models ($M_{\mathcal{X}}$ on $\mathcal{X}$, $M_{\mathcal{Y}}$ on $\mathcal{Y}$) and generating synthetic datasets ($S_{\mathcal{X}} \sim M_{\mathcal{X}}$, $S_{\mathcal{Y}} \sim M_{\mathcal{Y}}$). The privacy analysis then concerns an adversary observing samples drawn from these synthetic datasets $S_{\mathcal{X}}$ and $S_{\mathcal{Y}}$.

The relevant statistics for this analysis are the empirical frequencies of outputs o (unigrams or n-grams) within the generated synthetic datasets, denoted $f'_{S_{\mathcal{X}}}(o)$ and $f'_{S_{\mathcal{Y}}}(o)$. From the perspective of an adversary observing samples from the synthetic datasets, these empirical frequency distributions serve as the necessary sufficient statistics.

The LBF is calculated based on these synthetic frequencies:

$$LBF^{\text{synth}}_{\mathcal{X},\mathcal{Y}}(o) = \ln\left(\frac{f'_{S_{\mathcal{X}}}(o)}{f'_{S_{\mathcal{Y}}}(o)}\right) \quad .$$

The failure sets $\mathcal{F}^{\text{synth}}_{\mathcal{X}}$ and $\mathcal{F}^{\text{synth}}_{\mathcal{Y}}$ are determined using LBF^{synth} and the chosen ε. The corresponding minimal failure probability $\delta^{\text{synth}}(\varepsilon)$ is computed using the probability mass within the synthetic distributions:

$$\delta^{\text{synth}}(\varepsilon) = \max\left(\sum_{o\in\mathcal{F}^{\text{synth}}_{\mathcal{X}}} f'_{S_{\mathcal{X}}}(o), \sum_{o\in\mathcal{F}^{\text{synth}}_{\mathcal{Y}}} f'_{S_{\mathcal{Y}}}(o)\right) \quad .$$

This yields the (ε, δ)-curve for the synthetic data generation mechanism. Comparing this curve to the one obtained from direct sampling reveals the privacy impact (beneficial or detrimental) introduced by the synthesis step. Computing the (ε, δ)-curve uses the efficient sorting and accumulation method described previously.

5.4 Results and analysis of synthetic data

We compare the privacy properties of the synthetic data generated by LLM trained on the 10% of the CCL dataset and the same dataset excluding the data from $|U| = 100$ random users with the direct sampling baseline, noting the significant limitations of the evaluation of single-generation instances.

5.4.1 Objective privacy loss of synthetic data

The objective privacy loss is assessed by comparing empirical unigram frequency distributions from single generated instances $S_{\mathcal{X}}$ and $S_{\mathcal{Y}}$. Figures 10 and 11 visualize the

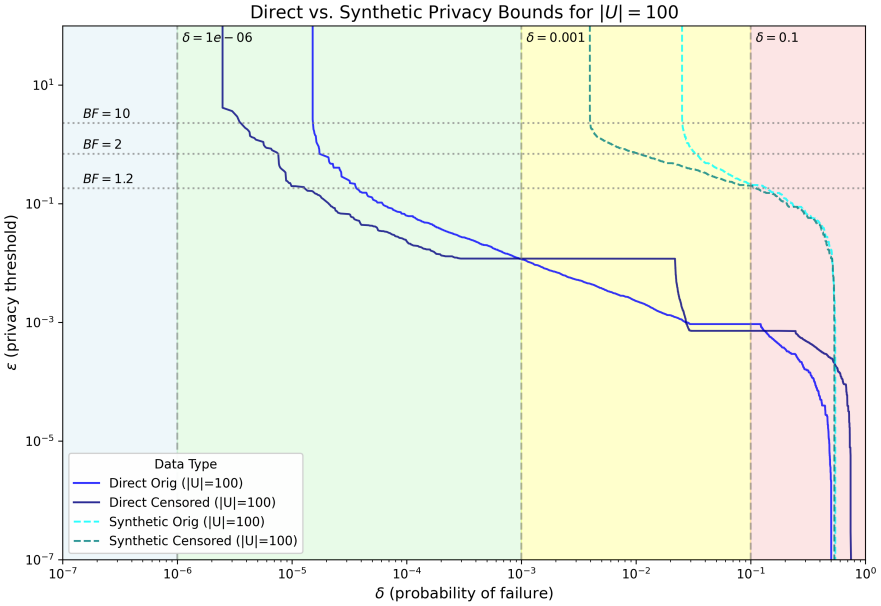


Figure 10. Conceptual comparison of objective (ε, δ) trade-offs for direct sampling vs. synthetic data (single instance) when $|U| = 100$. The synthetic curves (dashed lines) appear to show higher ε for a given δ compared to direct sampling.

resulting (ε, δ) trade-off and logarithmic Bayes factor (LBF) distributions relative to the direct sampling baseline.

The synthetic data results presented here are based on single generation instances and likely overestimate privacy loss due to ignoring generation randomness, a limitation discussed further in Section 7. The objective (ε, δ)-curves for synthetic data (conceptually represented for exclusion set sizes $|U| = 100$ in Figure 10) appear to lie above those for direct sampling, indicating higher privacy loss ε for a given failure probability δ, or a higher δ for a given ε. Furthermore, a direct comparison of LBF values for the $|U| = 100$ case (Figure 11) indicates that, for words with similar sampling probabilities $P_{\mathcal{X}}$, the LBF magnitudes calculated from the single synthetic instance often appear higher than those from direct sampling. While censoring rare words might still provide a relative privacy improvement for synthetic data (analogous to Figure 3a, right panel), the absolute privacy level measured in this single instance appears worse than direct sampling.

To further investigate the relationship between direct sampling and the synthetic data instance, Figure 12 provides word-level scatter plots for words common to both the original data subset (used for the direct sampling trial with $|U| = 100$) and the generated synthetic datasets $S_{\mathcal{X}}$ and $S_{\mathcal{Y}}$.

Panel (b) of Figure 12 compares the empirical probability of words $P_{\mathcal{X}}^{\text{direct}}$ in the original dataset $\mathcal{X}$ with their probability $P_{\mathcal{X}}^{\text{synth}}$ in the synthetic dataset $S_{\mathcal{X}}$. A positive

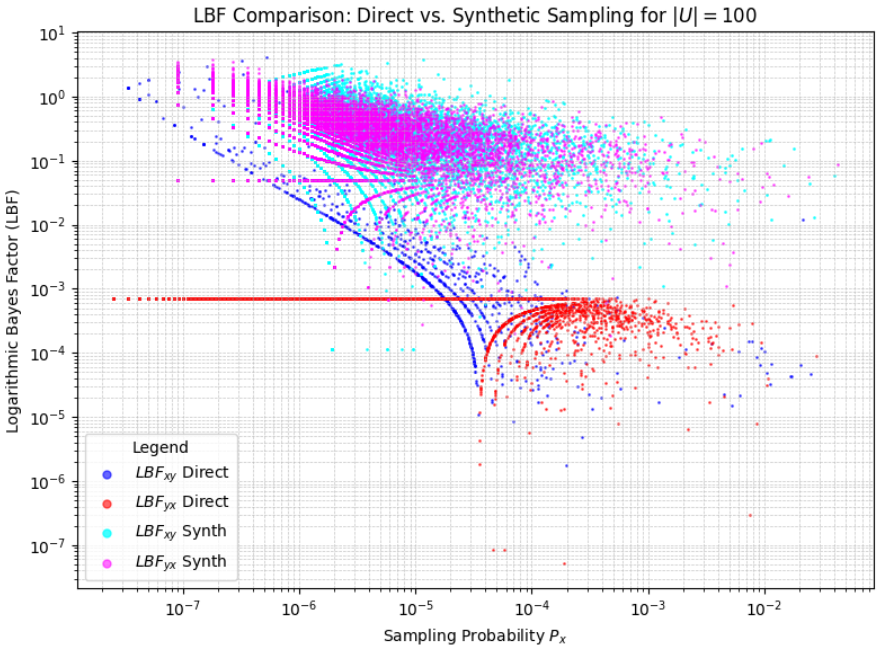
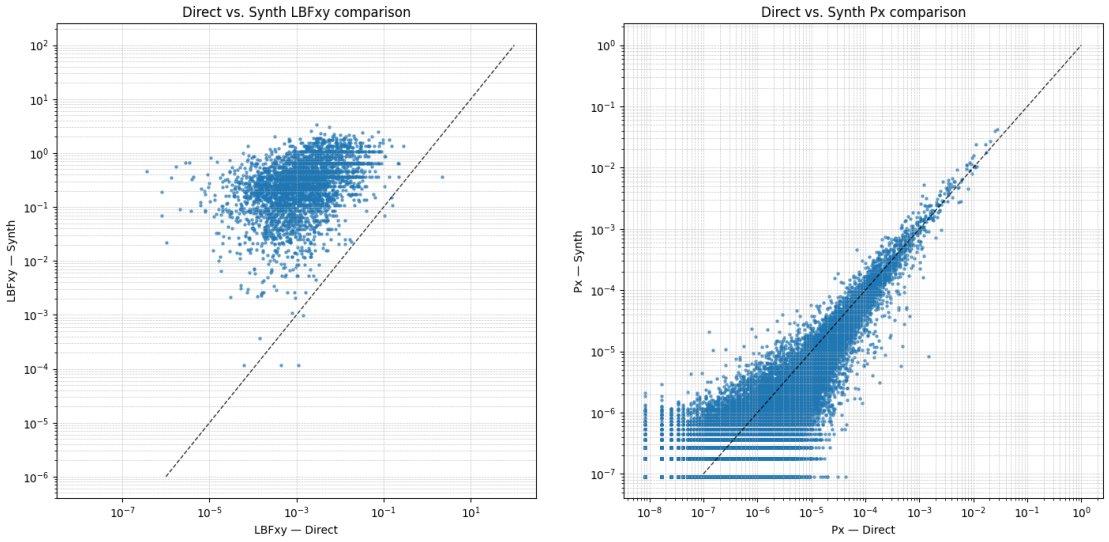


Figure 11. Comparison of LBF vs. $P_{\mathcal{X}}$ for direct sampling versus synthetic data generation when $|U| = 100$. Direct sampling LBFs are shown in blue ($LBF_{\mathcal{X},\mathcal{Y}}$) and red ($LBF_{\mathcal{Y},\mathcal{X}}$). Synthetic data LBFs are overlaid in cyan ($LBF^{\text{synth}}_{\mathcal{X},\mathcal{Y}}$) and magenta ($LBF^{\text{synth}}_{\mathcal{Y},\mathcal{X}}$). For many words with similar $P_{\mathcal{X}}$, the synthetic data appears to exhibit higher absolute LBF values in this single-instance evaluation.

correlation is evident, indicating that the LLM preserves general frequency trends from $\mathcal{X}$ in $S_{\mathcal{X}}$. However, there is considerable variance, with points scattered around the $y = x$ diagonal, showing that $P^{\text{synth}}_{\mathcal{X}}$ values are noisy approximations of $P^{\text{direct}}_{\mathcal{X}}$. The magnitude of the variance, in some cases resulting in a 10^{-3}-fold difference in either direction, seems to invalidate the assumption of a 10% change in marginal word distributions from Subsection 5.2. For infrequent words (low values on the x-axis), it appears points may tend to lie below the $y = x$ line, suggesting the model $M_{\mathcal{X}}$ might make these words even rarer in $S_{\mathcal{X}}$. Such deviations can be attributed not only to sampling randomness during generation but also to inherent model characteristics. For instance, language models may not represent all words according to their true frequencies from the training data; particularly, words that appeared rarely in training might be generated even less often by the model than their original low frequency would suggest, or their contextual usage might be learned less accurately [MGC+24].

Panel (a) of Figure 12 compares $LBF^{\text{direct}}_{\mathcal{X},\mathcal{Y}}$ with $LBF^{\text{synth}}_{\mathcal{X},\mathcal{Y}}$ for words where both values are positive. A substantial number of points lie significantly above the $y = x$ diagonal, indicating that for many words providing evidence for $\mathcal{X}$ over $\mathcal{Y}$, their distinguishing



(a) Direct vs. Synthetic LBF_{xy} Comparison

(b) Direct vs. Synthetic P_x Comparison

Figure 12. Word-level comparison of (a) $LBF_{\mathcal{X},\mathcal{Y}}$ and (b) $P_{\mathcal{X}}$ between direct sampling (x-axis) and the single synthetic data instance (y-axis) for $|U| = 100$. Both axes are on a logarithmic scale. The dashed line represents $y = x$. These plots only include words present in the vocabulary of both the original data subset (for the direct sampling trial) and the generated synthetic datasets due to the merging process for comparison.

power appears amplified in this synthetic instance ($LBF^{\text{synth}}_{\mathcal{X},\mathcal{Y}} > LBF^{\text{direct}}_{\mathcal{X},\mathcal{Y}}$). This observed amplification likely results from a several things. First, systematic bias in the LLM might contribute: if the model tends to undersample low-frequency words (as suggested by panel (b) for some rare words), this effect could be more pronounced for model $M_{\mathcal{Y}}$ trained on dataset $\mathcal{Y}$, where distinguishing words were already rarer. A more drastic reduction in the estimated $\hat{P}(w|S_{\mathcal{Y}})$ compared to $\hat{P}(w|S_{\mathcal{X}})$ would increase the ratio and amplify the LBF. Second, and potentially significantly, variance from estimating probabilities using finite samples $S_{\mathcal{X}}$ and $S_{\mathcal{Y}}$ introduces noise. Calculating the LBF using these noisy estimates ($\ln(\hat{P}(w|S_{\mathcal{X}})/\hat{P}(w|S_{\mathcal{Y}}))$) can easily produce values with larger magnitudes than the LBF calculated from the original, larger datasets, especially for low-frequency words where relative noise is higher. It is important to note that panel (a) only displays words where $LBF^{\text{synth}}_{\mathcal{X},\mathcal{Y}}$ remains positive; instances where sampling noise might have caused $\hat{P}(w|S_{\mathcal{Y}}) \geq \hat{P}(w|S_{\mathcal{X}})$ (resulting in $LBF^{\text{synth}}_{\mathcal{X},\mathcal{Y}} \leq 0$) are not shown, meaning the plot visualizes only part of the effect of noise. Thus, the apparent amplification of LBF likely stems from both potential bias of the model as well as the noise inherent in the single-instance evaluation method. The impact of vocabulary differences (words dropped or hallucinated) is also not captured here.

These word comparisons show how a single synthetic data realization can deviate

from the original distributions. Such deviations in a specific instance can lead to pessimistic privacy evaluations. However, this result must be interpreted with caution. The privacy calculation for the synthetic data was based on the empirical unigram frequencies derived from a single generated synthetic dataset instance for $M(\mathcal{X})$ and $M(\mathcal{Y})$. This methodology treats the variations inherent in the LLM data synthesis process (sampling noise) as fixed, known differences between the underlying distributions. It effectively assumes the attacker is capable of deterministically reproducing the synthetic dataset $S_{\mathcal{X}}$ (or its sufficient statistic) given knowledge of $\mathcal{X}$, and similarly for $S_{\mathcal{Y}}$.

This assumption likely leads to an overestimation of the true privacy loss attributable to the synthetic generation process. A real attacker faces uncertainty from the model's sampling randomness, which this evaluation method does not capture. Random fluctuations between the single instances of $S_{\mathcal{X}}$ and $S_{\mathcal{Y}}$ can artificially inflate the calculated LBF values.

It is plausible that the tendency observed in Figure 11 for some synthetic LBF magnitudes to exceed those from direct sampling could persist even if averaged over many generation runs. This would suggest the LLM might systematically change word frequency distributions in ways that make datasets $\mathcal{X}$ and $\mathcal{Y}$ (or their synthetic counterparts $S_{\mathcal{X}}$ and $S_{\mathcal{Y}}$) more distinguishable.

To better distinguish systematic effects of the LLM on LBF values from noise introduced by single-instance evaluation, a more robust experimental design would be necessary. Such a design should aim to average out stochasticity from both generation and training:

1. For a fixed pair of trained models $M_{\mathcal{X}}$ and $M_{\mathcal{Y}}$, generate multiple synthetic dataset instances ($S_{\mathcal{X},1}, \ldots, S_{\mathcal{X},N_{gen}}$ and $S_{\mathcal{Y},1}, \ldots, S_{\mathcal{Y},N_{gen}}$). The LBF analysis would then be based on frequencies averaged across these N_{gen} instances for each model, or by analyzing the distribution of LBFs obtained from pairwise $(S_{\mathcal{X},i}, S_{\mathcal{Y},j})$ comparisons. This would reveal if LBF amplification persists once sampling noise from generation is mitigated.

2. Repeat the above by retraining models $M_{\mathcal{X}}$ and $M_{\mathcal{Y}}$ multiple times (N_{train} runs) using different random seeds. This would help determine if any systematic LBF alteration is a consistent property of the fine-tuning process for this architecture and data, or specific to particular training outcomes.

Such an approach, while computationally intensive, is needed for a more definitive assessment of the inherent impact of an LLM on objective privacy metrics and is beyond the scope of this thesis.

5.4.2 Subjective privacy loss under limited knowledge

Standard DP analysis assumes an attacker knows the full output distributions or sufficient statistics. To model attackers with incomplete knowledge, we simulated scenarios where

the attacker focuses only on the N words exhibiting the highest absolute LBF values, for $N \in \{100, 1000, 5000\}$. Figure 13 compares three such scenarios when $|U| = 100$ sources were excluded:

- **Base:** Attacking the original data using the Top N words identified from the original data's LBFs.
- **Synth Top N from Synth:** Attacking the synthetic data using the Top N words identified from the synthetic data's LBFs.
- **Synth Top N from Base:** Attacking the synthetic data using the Top N words identified from the original data's LBFs.

Subsequently, δ levels of such words are computed for predefined ε thresholds on the attacked data. These scenarios represent different assumptions about the attacker's knowledge and ability to analyze either the original or the synthetic data distribution.

For $N = 100$ and $N = 1000$, the resulting failure probability δ is mostly unchanged across the tested ε range $[\ln(1.01), \ln(10)]$, indicating all selected words possess $|LBF| > \ln(10)$; the observed δ thus represents the total probability mass of these highly distinguishing words. A consistent relative ordering of these δ levels emerges: $\delta(\text{Synth Top N from Base}) < \delta(\text{Base}) < \delta(\text{Synth Top N from Synth})$. In the 'Synth Top N from Base' scenario, the final δ in the synthetic data is computed over the highest-δ words in the base data that exceed a certain ε level in there. However, if they do not exceed it in the synthetic data, their delta is not accounted for in the current setup. The lowest relative ordering of this scenario indicates the synthetic generator mitigates the impact of the original data's most distinguishing words. This is further supported by preliminary checks that indicated that the fine-tuned model reproduced only a very small fraction (approximately 0.1%) of trigrams (as opposed to unigrams analyzed here) that were unique to the specific user sets U excluded during training. The rest of the scenarios are less realistic but also less privacy-preserving.

This scenario represents a plausible attacker leveraging prior knowledge but unable to completely reproduce the synthetic generation process. Conversely, the highest δ occurs in the 'Synth Top N from Synth' setup, implying the generator introduces artifacts or amplifies noise, creating signals that appear highly distinguishing within the synthetic data but might not reflect true information leakage; this scenario could represent an attacker having total knowledge of model generation and sampling randomness. The 'Base' scenario provides the reference risk. For $N = 5000$, the δ points very slightly decrease as ε increases, but the relative ordering persists. These findings, however, rely on the single-instance evaluation and its inherent limitations.

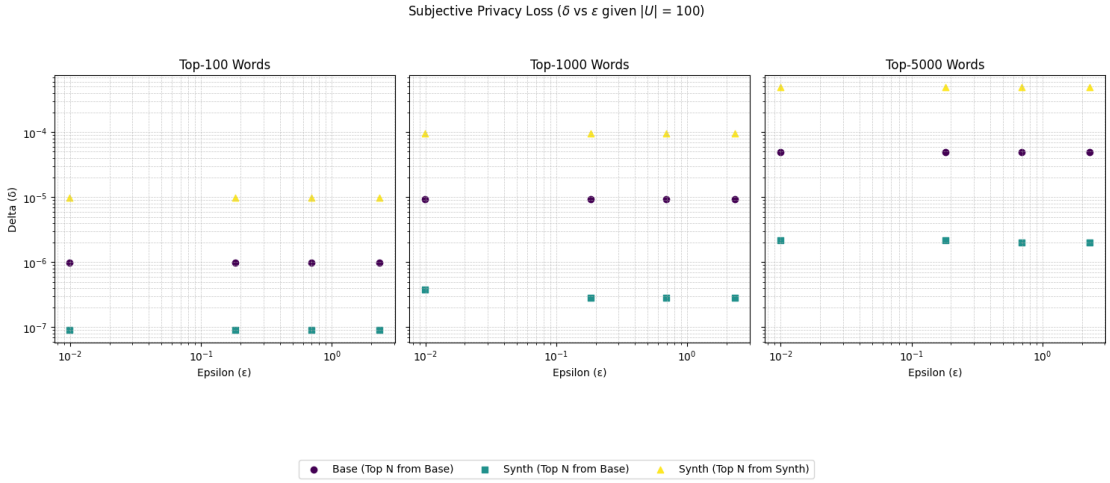


Figure 13. Subjective privacy loss trade-off (δ vs ε) for $|U| = 100$. Simulates an attacker with knowledge limited to the N most informative words in terms of LBF ($N \in \{100, 1000, 5000\}$). δ was computed for fixed ε thresholds at $\ln(10)$, $\ln(2)$, $\ln(1.2)$ and $\ln(1.01)$. Compares 'Base', 'Synth Top N from Synth', and 'Synth Top N from Base' scenarios. Failure probability δ rises with N. Note the logarithmic scales.

6 Surprisal attack

While Sections 4 and 5 examined privacy risks based on the distribution of generated outputs, publishing a fine-tuned language model itself poses a different risk: an attacker might probe the model's internal behaviour on known text samples to infer membership in the training data. This section evaluates such a risk using a surprisal attack. This attack differs from the sampling attack by focusing on the model's assigned probabilities for existing text, rather than the characteristics of novel generated text. The core assumption is that an attacker can query the target model to obtain probability-related information for chosen inputs. The limitation is that this attack directly assesses inference risk via model probing, not leakage through generated sequences.

A common technique for membership inference against models is to compare how surprised the model is by different inputs [CTW+21]. Surprisal, formally the negative logarithm of the probability assigned by the model to a sequence, quantifies how unexpected that sequence is according to the model's learned patterns; lower surprisal implies higher probability and a better fit. The intuition, though complex for LLMs capable of strong generalization, is that models might assign lower surprisal (higher probability) to inputs similar to those seen during training. Accessing the information needed to compute surprisal (i.e., token probabilities) does not necessarily require direct white-box access to model logits. This information can often be closely approximated from standard API outputs providing probabilities or confidence scores, or potentially extracted via specialized techniques [CPD+24]. We test whether surprisal values could potentially distinguish tweets from users whose data was excluded during training compared to those whose data was included.

6.1 Experiment setup

We fine-tune a GPT-2 implementation 11 times on the first 10% of our Twitter dataset $\mathcal{D}_{\text{CCL}}$. The implementation is based on Karpathy's nanoGPT [Kar22], a minimal, educational GPT-2 codebase. GPT-2 is a well-established transformer architecture, making it a relevant baseline for reproducibly studying LLM properties. One model M_{all} is trained on all available data from this subset $\mathcal{X}_{\text{all}}$. The other 10 models $M_1, \ldots, M_{10}$ are each trained on a dataset $\mathcal{Y}_i$ derived from $\mathcal{X}_{\text{all}}$ by excluding tweets from a different, randomly selected set U_i of 100 users, i.e. $|U_i| = 100$.

To establish a baseline for comparison, we also performed a control experiment. For each model M_i, instead of comparing included versus excluded tweets, we compared surprisal scores for tweets from two disjoint random subsets of users who were both included in the training of M_i. This helps verify that any observed differences in the main experiment are due to the included/excluded status, not just random variation between user groups or general model behaviour on different samples of seen data.

6.2 Surprisal calculation

For a sequence of tokens $x = (x_1, \ldots, x_n)$, the surprisal $S(x)$ is the sum of the negative log-likelihoods of the true tokens given their preceding context according to model M:

$$S(x) = -\sum_{t=1}^{n-1} \log P(x_{t+1}|x_1, \ldots, x_t; M) \quad .$$

Here, $P(x_{t+1}|x_1, \ldots, x_t; M)$ is the probability assigned by model M to the true next token x_{t+1} given the prefix $(x_1, \ldots, x_t)$. This probability is obtained from the model's output layer, typically by applying the softmax function to the logits (the raw scores before normalization) corresponding to the vocabulary tokens at that position.

Calculating surprisal thus requires querying model M to obtain these token probabilities for the given sequence x. This equals the total cross-entropy loss for predicting the sequence. While direct probability access is assumed here, it is also possible to obtain logits or logprobs covertly, by making use of techniques that utilize the ability to bias certain logits [CPD+24], which are not explored in this work.

6.3 Exploration

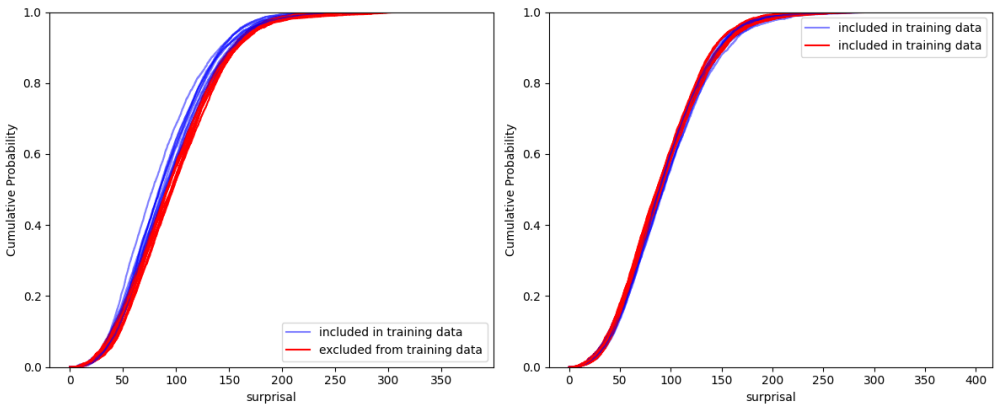


Figure 14. Empirical cumulative distribution functions (CDFs) of tweet surprisal values across 10 fine-tuned models. Left: Comparison between tweets from included users (blue) and excluded users (red) for each model. Right: Control comparison between two disjoint sets of included users for each model. The consistent rightward shift of the red curves (excluded) on the left indicates systematically higher surprisal for data not seen during training.

Figures 14 and 15 visualize the distributions of surprisal values per model. The empirical CDF plots (Figure 14, left) show a consistent rightward shift for the surprisal

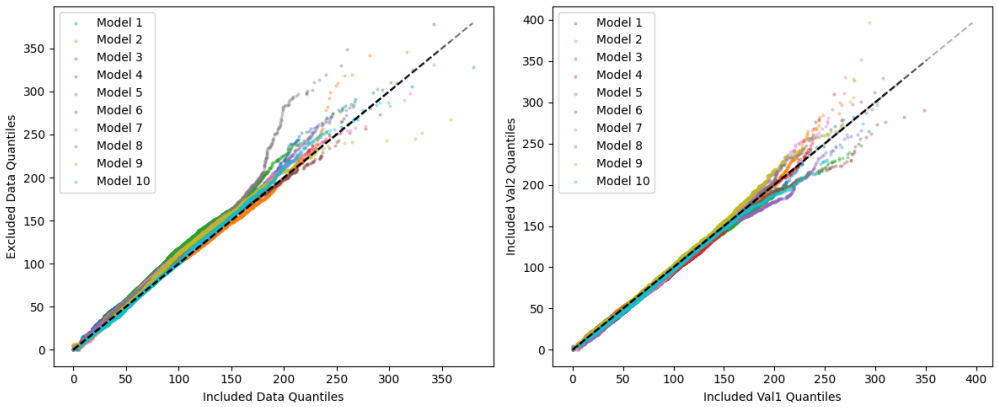


Figure 15. Quantile-quantile (Q-Q) plots comparing surprisal distributions across 10 fine-tuned models. Left: quantiles of excluded user tweet surprisals versus included user tweet surprisals. Right: control comparison between quantiles of two disjoint sets of included user tweet surprisals. The deviation above the identity line ($y = x$) on the left plot, especially at higher quantiles, confirms that excluded tweets tend to have higher surprisal values than included tweets at the same quantile rank. The control plot (right) shows points scattered symmetrically around the identity line.

distributions of excluded tweets (red) compared to included tweets (blue) across all models. Similarly, the Q-Q plots (Figure 15, left) show quantiles for excluded tweets consistently lying above the identity line. Both visualizations indicate systematically higher surprisal for data not seen during training, although the overlap appears significant. The control comparisons (right panels in both figures) show no such systematic differences between disjoint sets of included tweets.

This provides empirical evidence that the fine-tuned models assign detectably higher surprisal values, on average, to tweets from users whose data was excluded during training. However, the visual separation in the plots is modest, suggesting that while a statistical difference exists, its practical utility for high-confidence membership inference on individual tweets might be limited without further aggregation or analysis. Quantifying the actual privacy risk associated with this observed separability requires applying the LBF framework. To obtain concrete measurements of this signal's magnitude before proceeding to LBF-based privacy bounds, we performed analysis per model, calculating the difference in median surprisal and the area under the ROC curve (AUC). The results are presented in Table 2.

The results in Table 2 confirm the visual impressions. The median surprisal score for excluded tweets was consistently higher than for included tweets across all models (mean difference: 7.304). Furthermore, the AUC was consistently above 0.5 (mean AUC: 0.542), verifying that surprisal holds statistically significant, albeit generally weak, discriminative power for membership inference within each model instance. These

results are based on models fine-tuned for only one epoch; stronger separation (higher median differences and AUCs) might be expected with more extensive training.

Table 2. Per-model surprisal difference analysis. Median difference is the difference between the median of the excluded and included data surprisals. AUC reflects discriminability (0.5 = random guess). Positive median differences and AUC > 0.5 indicate a detectable membership signal.

Model index	median difference	AUC
1	12.222	0.573
2	1.471	0.515
3	13.705	0.580
4	8.218	0.543
5	1.948	0.519
6	6.431	0.523
7	11.723	0.566
8	6.711	0.545
9	9.318	0.546
10	1.296	0.509

The observed statistical difference, although modest, motivates evaluating its potential use in an attack. The potential for using this difference in a threshold-based attack can be further quantified by examining the threshold likelihood ratio (TLR) $LR(\tau)$, defined as a function of the surprisal threshold τ:

$$LR(\tau) = \frac{\Pr\left[S(x) > \tau | \text{excl}\right]}{\Pr\left[S(x) > \tau | \text{incl}\right]} \ .$$

Figure 16 plots this ratio for each model.

As seen in Figure 16, the threshold likelihood ratio $LR(\tau)$ often exhibits an increasing trend as the threshold τ increases. However, this behavior can be noisy, particularly at higher values of τ. As τ increases, the probabilities $\Pr\left[S(x) > \tau | \text{excl}\right]$ and $\Pr\left[S(x) > \tau | \text{incl}\right]$ are estimated from progressively fewer data points, leading to higher variance and potential unreliability in the $LR(\tau)$ estimate. Caution is thus warranted when interpreting $LR(\tau)$ values derived from sparse data at high thresholds. Despite this noise, the general trend suggests that higher surprisal values tend to be more indicative of excluded tweets.

For some models, the ratio eventually diverges towards infinity (marked by $*$ where calculable). This divergence occurs when the threshold τ surpasses the maximum surprisal observed for any included tweet (i.e., $\Pr\left[S(x) > \tau | \text{incl}\right] \to 0$), while at least one excluded tweet still has a surprisal value exceeding τ (i.e., $\Pr\left[S(x) > \tau | \text{excl}\right] > 0$).

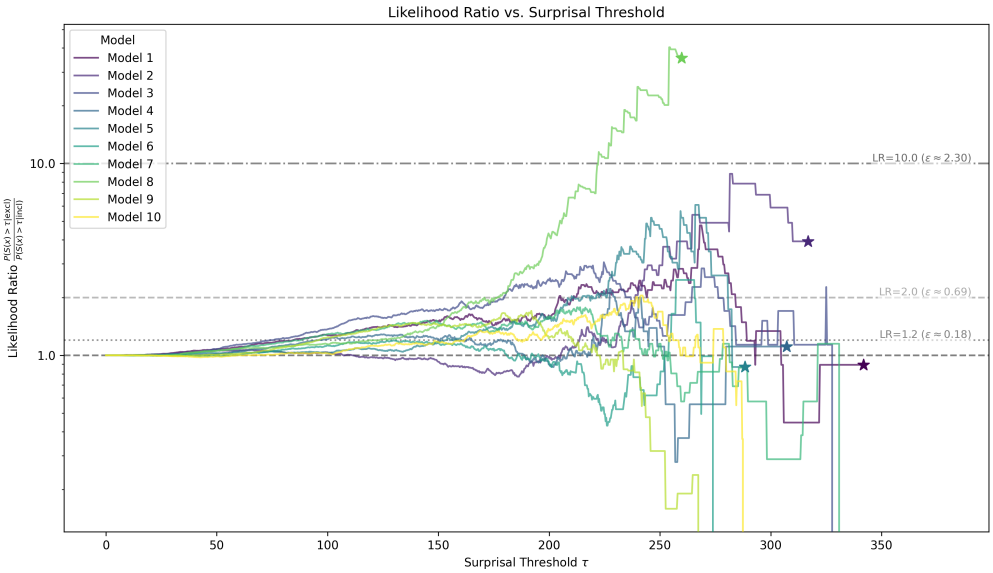


Figure 16. Threshold likelihood ratio versus surprisal threshold τ for each of the 10 models. The y-axis is on a logarithmic scale. A ratio greater than 1 indicates that exceeding the threshold is more likely for an excluded tweet than an included one. The general upward trend shows increasing distinguishing power at higher thresholds. Stars (*) mark the highest finite threshold for each model where the ratio could be calculated before the denominator $\Pr[S(x) > \tau|\text{incl}]$ becomes zero.

In simpler terms, for these models, the single most surprising tweet from the excluded set is more surprising than any tweet from the included set: $\max_{x \in T_{i,\text{excl}}} S(x) > \max_{x \in T_{i,\text{incl}}} S(x)$, where $T_{i,\text{excl}}$ and $T_{i,\text{incl}}$ represent the sets of all tweets from excluded and included users for model M_i, respectively. Conversely, for other models, if the maximum surprisal for included tweets exceeds that of excluded tweets (i.e., $\max_{x \in T_{i,\text{incl}}} S(x) > \max_{x \in T_{i,\text{excl}}} S(x)$), the ratio will instead drop towards zero at very high thresholds as $\Pr[S(x) > \tau|\text{excl}]$ becomes zero while $\Pr[S(x) > \tau|\text{incl}]$ remains positive. Figure 16 illustrates instances of both divergence behaviors.

While achieving extreme separation (very high or very low ratios) is possible at these high thresholds, this typically occurs when only a small fraction of tweets exceed the threshold. The plot demonstrates that substantial likelihood ratios (e.g., $LR(\tau) > 10$ or $LR(\tau) < 0.1$) are achievable across various models, motivating the formal analysis of a threshold attack.

6.4 Attack model

The exploration in Section 6.3 demonstrates that fine-tuned LLMs can assign detectably different surprisal scores to text fragments depending on whether similar fragments were present in their training data. While the observed difference was modest for individual tweets after limited fine-tuning (Table 2), this phenomenon points towards potential privacy risks associated with releasing or granting query access to such models.

A standard differential privacy analysis might aim to bound the leakage from the entire model training and release process. This would involve analyzing the probability distributions over possible trained models given adjacent datasets ($\mathcal{X}$ vs $\mathcal{Y} = \mathcal{X} \setminus U$) and is generally considered infeasible for complex models like LLMs.

However, the measured surprisal difference is directly relevant to a more specific, practical privacy scenario. Consider a situation where a model M, trained on a potentially sensitive dataset $\mathcal{X}$, has already had its weights published or made accessible via an API. If a user then considers publishing a new text fragment x' known to originate from a specific user group U, the public availability of M creates a new risk. An adversary could query M to calculate the surprisal $S(x'; M)$ of the fragment. Based on the principle observed in our exploration, if $S(x'; M)$ is low, the adversary might infer that group U was likely part of the original training set $\mathcal{X}$. Conversely, a high surprisal value would suggest U was likely not included. The public model M acts as background knowledge that enables membership inference about group U based on the surprisal of the newly released fragment x'.

An attacker could further employ a surprisal threshold, τ, which they might determine based on some background knowledge or by analyzing the model's typical behavior. If the calculated surprisal $S(x'; M)$ for the new fragment x' exceeds this threshold τ, the attacker infers that U was likely excluded from M's training data; otherwise, U is inferred to be included. This thresholding transforms the continuous surprisal score into a binary attack function, as discussed conceptually in Section 3.8, simplifying the decision-making process for the adversary. The effectiveness of such an attack depends on the separation between the surprisal distributions for included versus excluded data and the choice of threshold τ.

Quantifying the privacy risk of releasing fragment x' (from group U) using differential privacy, even when a model M (trained on $\mathcal{X}$) is already public, remains challenging. A formal analysis would require comparing the distribution of surprisal scores $S(x'; M)$ under two assumptions about M's training: first, assuming M was trained with U included ($M = M_{\mathcal{X}}$), and second, assuming M was trained with U excluded ($M = M_{\mathcal{Y}}$). Specifically, one needs to characterize $P(S(x'; M_{\mathcal{X}})|x' \in U)$ versus $P(S(x'; M_{\mathcal{Y}})|x' \in U)$. While this avoids reasoning about the probability of model parameters themselves, accurately characterizing these conditional surprisal distributions for $M_{\mathcal{X}}$ and its counterfactual $M_{\mathcal{Y}}$ across all possible fragments $x' \in U$ and training random seeds is computationally prohibitive. Therefore, deriving formal or empirically

approximating (ε, δ)-bounds for this scenario is beyond the scope of this work. Nonetheless, the empirical results confirm the existence of a measurable signal in surprisal scores that indicates a possible privacy risk in such scenarios.

7 Discussion

This thesis developed and applied a framework to evaluate privacy risks in synthetic textual data, motivated by challenges in sensitive domains like EHRs under a journalist attack model. Addressing concerns about potential leakage from LLMs, we adapted DP, using LBFs derived from specific attack function outputs to quantify privacy loss ε and failure probability δ. This approach focuses on realistic adversarial capabilities, unlike standard worst-case analysis of DP. Evaluations assumed either limited attacker knowledge (specific attacks) or, for baseline comparisons, full knowledge of relevant sufficient statistics (empirical n-gram frequencies).

The framework produced quantifiable privacy bounds. Analyzing direct n-gram sampling established baselines for both the CCL Twitter dataset (Figures 3a, 8) and the MAITT patient dataset (Figures 3b, 9). For both datasets, privacy loss increased with the number of differing sources $|U|$, confirming DP intuitions. The analysis highlighted the influence of rare n-grams via infinite LBF values (mitigated by censoring, Figure 3a right panel) and demonstrated distinct scaling behaviours for unigrams versus trigrams (Figure 7). The qualitative similarity of results across text types confirms the general applicability of these baseline phenomena.

The comparison between direct sampling and synthetic data from a fine-tuned LLM (GPT-2 based, on CCL data) yielded counter-intuitive objective results. The evaluation comparing empirical frequency distributions from single generated instances $S_{\mathcal{X}}$ and $S_{\mathcal{Y}}$ indicated worse objective privacy properties for the synthetic data than direct sampling (Figures 10, 11). This result, however, stems from evaluating privacy based on an implicit, potentially unrealistic assumption: that the attacker knows, or the privacy risk is defined by, the exact empirical distributions of these single, specific generated instances. This approach fails to account for the inherent stochasticity in the training and generation processes of the LLM, which introduce uncertainty for an attacker. Consequently, using single-instance frequencies likely overestimates the true privacy risk posed by the generative mechanism. This interpretation is supported by evidence that the LLM reproduced only a small fraction ($\approx 0.1\%$) of trigrams unique to specific excluded user sets.

Furthermore, the subjective analysis under limited attacker knowledge (Figure 13) showed that an attacker leveraging known base data vulnerabilities 'Synth Top N from Base' performed worse against synthetic data than against the base data itself. This suggests the generation process likely provides privacy benefits when evaluating based on more realistic attacker knowledge assumptions. To validate this, further experiments could be made with explicit modelling of an attacker's limited knowledge and access. The impact of the knowledge assumptions could then be contrasted with the inherent effect of the LLM in altering the marginal distribution of outputs. This could be further investigated through the experimental design outlined in Section 5.4 involving multiple generation runs and model retraining.

The exploration of surprisal scores on CCL data demonstrated the framework's applicability to model-probing attacks. The findings (Figures 14, 15; Table 2) confirmed that surprisal values can contain a membership inference signal, distinguishing between data seen and unseen by the model during fine-tuning. This highlights a potential privacy risk if models are public and can be queried for surprisal-like information, as discussed in Section 6, although a full differential privacy quantification of this specific risk scenario remains challenging.

Overall, the attack-based framework provides a method for quantifying privacy within specific scenarios. However, the empirical comparison between LLM synthesis and direct sampling yielded results that are likely misleading about the true privacy benefits of synthesis, due to the reliance on single generation instances for the evaluation. This approach effectively neglects crucial sources of randomness (training, sampling) that impact realistic attack success. Nonetheless, standard LLMs, lacking specific privacy-enhancing techniques during training (e.g., DP-SGD [ACG+16], goldfish loss [HWJ+24]) or inference (e.g., private prediction [ABK+24]), should not be assumed to provide strong objective DP guarantees without rigorous, correctly designed evaluation.

Several limitations persist. The primary one concerning LLM evaluation is the reliance on single generation instances for assessing objective privacy, which yields unreliable estimates of privacy loss by neglecting generation stochasticity. Compounding this, the objective DP framework generally assumes strong adversarial knowledge of output distributions or sufficient statistics. This assumption may be overly pessimistic compared to attackers with more realistic, partial knowledge, as suggested by the subjective analysis (scenario 'Synth Top N from Base'). The scope of attack functions quantitatively evaluated was limited to n-gram frequencies; while surprisal was explored as a signal, a full (ε, δ)-analysis of surprisal-based attacks was not included. Other attack types, such as semantic attacks or those by adaptive adversaries, were not considered.

Generalizability requires confirming findings, particularly from the synthetic data evaluation (which was performed only on CCL data), on the target EHR data ($\mathcal{D}_{\text{MAITT}}$). The surprisal signal exploration also needs to be replicated and potentially extended on $\mathcal{D}_{\text{MAITT}}$. The framework itself makes assumptions about attacker knowledge, which differ between the objective and subjective analyses. Furthermore, the choice of LLM architecture and its training and sampling hyperparameters can significantly influence results.

Future research must address the evaluation of synthetic data privacy by developing methods that properly account for generation stochasticity, perhaps by analyzing expected distributions or averaging results over multiple generation runs. Applying the direct sampling analysis framework comprehensively to the $\mathcal{D}_{\text{MAITT}}$ and further investigating the surprisal signal's strength and characteristics on this medical data are essential next steps. Exploring broader attack classes (e.g., attribute inference, semantic similarity attacks) and incorporating utility metrics alongside privacy bounds would provide a

more complete picture. Evaluating models trained with explicit privacy-enhancing techniques (such as DP-SGD [ACG+16] or goldfish loss [HWJ+24]) or those employing private inference methods within this LBF-based framework would also offer valuable comparisons.

8 Conclusion

Sharing sensitive textual data like electronic health records requires robust privacy measures. Synthetic data from LLMs is a potential avenue, but assessing its privacy properties remains challenging due to its textual nature. This thesis introduced and applied a framework to evaluate the privacy of synthetic text under a journalist attack model, adapting DP principles.

We proposed using LBF computed over the outputs of specific attack functions to quantify privacy loss ε and failure probability δ relative to concrete adversarial models. This framework was used to establish privacy bounds for direct n-gram sampling from both social media (CCL) and medical record (MAITT) datasets, serving as a baseline and highlighting the role of data characteristics, such as rare elements.

Our comparison of this baseline to LLM-generated synthetic data on CCL was hampered by evaluating privacy based on single generation instances. While the calculation was correct for the specific instances observed, this approach neglects the impact of training and generation randomness, likely leading to an overestimation of privacy risks for the synthetic data relative to the generative process itself. Separately, an exploration of surprisal scores on CCL data confirmed the presence of a membership inference signal, demonstrating the framework's potential applicability to model-probing threat vectors, though a full quantitative privacy analysis for this scenario was not completed.

The primary value of this work lies in the proposed attack-based evaluation framework, offering a way to quantify privacy for specific attacks by explicitly considering realistic attacker capabilities rather than relying solely on traditional worst-case assumptions of DP. However, the empirical results underscore critical challenges in assessing generative model privacy. Standard LLM synthesis, without dedicated privacy enhancements cannot be assumed to provide strong protection. On the other hand, evaluating LLM-generated synthetic data for privacy using protocols that neglect generation randomness likely results in an overestimation of privacy risk. Robust evaluation protocols are crucial. Future work should focus on correcting these evaluation methods, applying the framework comprehensively to real health data (e.g. MAITT, including surprisal and synthetic analyses), performing robust synthetic data evaluations, and assessing diverse attack functions and privacy-enhancing training and synthesis techniques as well as model architectures.

References

[ABK+24] Kareem Amin, Alex Bie, Weiwei Kong, Alexey Kurakin, Natalia Ponomareva, Umar Syed, Andreas Terzis, and Sergei Vassilvitskii. Private prediction for large-scale synthetic text generation, 2024.

[ACG+16] 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*, CCS'16. ACM, October 2016.

[AFM+22] Mário S. Alvim, Natasha Fernandes, Annabelle McIver, Carroll Morgan, and Gabriel H. Nunes. Flexible and scalable privacy assessment for very large datasets, with an application to official governmental microdata. *PoPETs*, 2022(4):378–399, 2022. arXiv:2204.13734 [cs.CR].

[AOL06] AOL. AOL Search Data, 2006. Accessed: 2024-10-01.

[AZL24] Zeyuan Allen-Zhu and Yuanzhi Li. Physics of Language Models: Part 3.3, Knowledge Capacity Scaling Laws, 2024.

[BCH22] Borja Balle, Giovanni Cherubin, and Jamie Hayes. Reconstructing Training Data with Informed Adversaries, 2022.

[BZ06] Michael Barbaro and Tom Zeller. A face is exposed for AOL searcher no. 4417749. *New York Times*, 08 2006.

[CCL10] Zhiyuan Cheng, James Caverlee, and Kyumin Lee. You are where you Tweet: A content-based approach to geo-locating Twitter users. In *International Conference on Information and Knowledge Management*, pages 759–768, 10 2010.

[CPD+24] Nicholas Carlini, Daniel Paleka, Krishnamurthy Dj Dvijotham, Thomas Steinke, Jonathan Hayase, A. Feder Cooper, Katherine Lee, Matthew Jagielski, Milad Nasr, Arthur Conmy, Itay Yona, Eric Wallace, David Rolnick, and Florian Tramèr. Stealing Part of a Production Language Model, 2024.

[CTW+21] Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting Training Data from Large Language Models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650. USENIX Association, August 2021.

[DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating Noise to Sensitivity in Private Data Analysis. In Shai Halevi and Tal Rabin, editors, *Theory of Cryptography*, pages 265–284, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[DR14] Cynthia Dwork and Aaron Roth. The Algorithmic Foundations of Differential Privacy. *Found. Trends Theor. Comput. Sci.*, 9(3–4):211–407, aug 2014.

[DRV10] Cynthia Dwork, Guy N. Rothblum, and Salil Vadhan. Boosting and Differential Privacy. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 51–60, 2010.

[Dwo06] Cynthia Dwork. Differential Privacy. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *Automata, Languages and Programming*, pages 1–12, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[EE23] Khaled El Emam. Status of synthetic data generation for structured health data. *JCO Clinical Cancer Informatics*, 7:e2300071, 2023.

[EEA13] Khaled El Emam and Luk Arbuckle. *Anonymizing Health Data: Case Studies and Methods to Get You Started*. O'Reilly Media, Inc., 2013.

[EEH19] Khaled El Emam and Richard Hoptroff. The synthetic data paradigm for using and sharing data. *Cutter Executive Update*, 19(6):1–12, 2019.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[GGS23] Aldren Gonzales, Guruprabha Guruswamy, and Scott Smith. Synthetic data in health care: A narrative review. *PLOS Digital Health*, 2:e0000082, 01 2023.

[GMCdM23] Florent Guépin, Matthieu Meeus, Ana-Maria Cretu, and Yves-Alexandre de Montjoye. Synthetic is all you need: removing the auxiliary data assumption for membership inference attacks against synthetic data, 2023.

[Gre13] Glenn Greenwald. NSA Collecting Phone Records of Millions of Verizon Customers Daily. *The Guardian*, 2013. Accessed: 2024-10-01.

[HSB+23] Valentin Hartmann, Anshuman Suri, Vincent Bindschaedler, David Evans, Shruti Tople, and Robert West. SoK: Memorization in General-Purpose Large Language Models, 2023.

[HWJ+24] Abhimanyu Hans, Yuxin Wen, Neel Jain, John Kirchenbauer, Hamid Kazemi, Prajwal Singhania, Siddharth Singh, Gowthami Somepalli, Jonas Geiping, Abhinav Bhatele, and Tom Goldstein. Be like a Goldfish, Don't Memorize! Mitigating Memorization in Generative LLMs, 2024.

[IKL+24] Eunyoung Im, Hyeoneui Kim, Hyungbok Lee, Xiaoqian Jiang, and Ju Han Kim. Exploring the tradeoff between data privacy and utility with a clinical data analysis use case. *BMC Medical Informatics and Decision Making*, 24(1):147, 2024.

[Kar22] Andrej Karpathy. NanoGPT. `https://github.com/karpathy/nanoGPT`, 2022.

[MGC+24] Richard Diehl Martinez, Zebulon Goriely, Andrew Caines, Paula Buttery, and Lisa Beinborn. Mitigating Frequency Bias and Anisotropy in Language Model Pre-Training with Syntactic Smoothing, 2024.

[MHVB13] Yves-Alexandre Montjoye, Cesar Hidalgo, Michel Verleysen, and Vincent Blondel. Unique in the Crowd: The Privacy Bounds of Human Mobility. *Scientific reports*, 3:1376, 03 2013.

[MKGV07] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. L-diversity: Privacy beyond k-anonymity. *ACM Trans. Knowl. Discov. Data*, 1(1):3–es, March 2007.

[MMO22] Sean MacAvaney, Craig Macdonald, and Iadh Ounis. Reproducing Personalised Session Search over the AOL Query Log, 2022.

[MPP+23] Fatemeh Mosaiyebzadeh, Seyedamin Pouriyeh, Reza M. Parizi, Quan Z. Sheng, Meng Han, Liang Zhao, Giovanna Sannino, and Daniel Macêdo Batista. Privacy-Enhancing Technologies in Federated Learning for the Internet of Healthcare Things: A Survey, 2023.

[MRSP15] Yves-Alexandre Montjoye, Laura Radaelli, Vivek Singh, and Alex Pentland. Unique in the shopping mall: On the reidentifiability of credit card metadata. *Science (New York, N.Y.)*, 347:536–9, 01 2015.

[MSL+21] R. Thomas McCoy, Paul Smolensky, Tal Linzen, Jianfeng Gao, and Asli Celikyilmaz. How much do language models copy from their training data? Evaluating linguistic novelty in text generation using RAVEN, 2021.

[NCH+23] Milad Nasr, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A. Feder Cooper, Daphne Ippolito, Christopher A. Choquette-Choo, Eric Wallace, Florian Tramèr, and Katherine Lee. Scalable extraction of training data from (production) language models, 2023.

[OTM+23] Marek Oja, Sirli Tamm, Kerli Mooses, Maarja Pajusalu, Harry-Anton Talvik, Anne Ott, Marianna Laht, Maria Malk, Marcus Lõo, Johannes Holm, Markus Haug, Hendrik Šuvalov, Dage Särg, Jaak Vilo, Sven Laur, Raivo Kolde, and Sulev Reisberg. Transforming Estonian health data to the Observational Medical Outcomes Partnership (OMOP) Common Data Model: lessons learned. *JAMIA Open*, 6(4):ooad100, 12 2023.

[Pia14] Steven T. Piantadosi. Zipf's word frequency law in natural language: a critical review and future directions. *Psychonomic Bulletin & Review*, 21(5):1112–1130, Oct 2014.

[Pom94] Klaus Pommerening. Medical Requirements for Data Protection. In Klaus Brunnstein and Eckart Raubold, editors, *Applications and Impacts, Information Processing '94, Volume 2, Proceedings of the IFIP 13th World Computer Congress, Hamburg, Germany, 28 August - 2 September, 1994*, volume A-52 of *IFIP Transactions*, pages 533–540. North-Holland, 1994.

[PR04] Klaus Pommerening and Michael Reng. Secondary use of the EHR via pseudonymisation. *Studies in Health Technology and Informatics*, 103:441–446, 2004.

[RM22] Jason W. Rocks and Pankaj Mehta. Memorizing without overfitting: Bias, variance, and interpolation in overparameterized models. *Physical Review Research*, 4(1), March 2022.

[SKR+23] Rylan Schaeffer, Mikail Khona, Zachary Robertson, Akhilan Boopathy, Kateryna Pistunova, Jason W. Rocks, Ila Rani Fiete, and Oluwasanmi Koyejo. Double Descent Demystified: Identifying, Interpreting & Ablating the Sources of a Deep Learning Puzzle, 2023.

[SRTP+21] James Scheibner, Jean Louis Raisaro, Juan Ramón Troncoso-Pastoriza, Marcello Ienca, Jacques Fellay, Effy Vayena, and Jean-Pierre Hubaux. Revolutionizing Medical Data Sharing Using Advanced Privacy-Enhancing Technologies: Technical, Legal, and Ethical Synthesis. *Journal of Medical Internet Research*, 23(2):e25120, Feb 2021.

[SVBV23] Robin Staab, Mark Vero, Mislav Balunović, and Martin Vechev. Beyond Memorization: Violating Privacy Via Inference with Large Language Models, 2023.

[SVT25] Ali Satvaty, Suzan Verberne, and Fatih Turkmen. Undesirable Memorization in Large Language Models: A Survey, 2025.

[Swe02] Latanya Sweeney. k-anonymity: a model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5):557–570, October 2002.

[SY24] Kiarash Sedghighadikolaei and Attila A Yavuz. Privacy-preserving and trustworthy deep learning for medical imaging, 2024.

[TMO23] The New York Times Company, Microsoft Corporation, and OpenAI. The New York Times Company v. Microsoft Corporation and OpenAI. *District Court, S.D. New York*, 1:23-cv-11195, 2023. Case Law.

[TSBP22] Anvith Thudi, Ilia Shumailov, Franziska Boenisch, and Nicolas Papernot. Bounding Membership Inference, 2022.

[WMW+24] Orion Weller, Marc Marone, Nathaniel Weir, Dawn Lawrie, Daniel Khashabi, and Benjamin Van Durme. "According to ...": Prompting Language Models Improves Quoting from Pre-Training Data, 2024.

[XFZ+23] Fuzhao Xue, Yao Fu, Wangchunshu Zhou, Zangwei Zheng, and Yang You. To Repeat or Not To Repeat: Insights from Scaling LLM under Token-Crisis, 2023.

[YDvdS20] Jinsung Yoon, Lydia N Drumright, and Mihaela van der Schaar. Anonymization Through Data Synthesis Using Generative Adversarial Networks (ADS-GAN). *IEEE journal of biomedical and health informatics*, 24(8):2378—2388, August 2020.

[YRC23] Oleksandr Yermilov, Vipul Raheja, and Artem Chernodub. Privacy- and Utility-Preserving NLP with Anonymized Data: A case study of Pseudonymization, 2023.

[ZIL+23] Chiyuan Zhang, Daphne Ippolito, Katherine Lee, Matthew Jagielski, Florian Tramèr, and Nicholas Carlini. Counterfactual Memorization in Neural Language Models, 2023.

[ZWC+23] Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson. Universal and Transferable Adversarial Attacks on Aligned Language Models, 2023.

[ZXCS23] Zhenhong Zhou, Jiuyang Xiang, Chaomeng Chen, and Sen Su. Quantifying and Analyzing Entity-level Memorization in Large Language Models, 2023.

[ZZA+23] Sicheng Zhu, Ruiyi Zhang, Bang An, Gang Wu, Joe Barrow, Zichao Wang, Furong Huang, Ani Nenkova, and Tong Sun. AutoDAN: Interpretable Gradient-Based Adversarial Attacks on Large Language Models, 2023.

Appendix

I. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Fedor Stomakhin**,
*(*author's name)

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

   reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

   Framework for Privacy-Preserving Synthesis of Textual Data,
   *(*title of thesis)

   supervised by Sven Laur and Liina Kamm.
   *(*supervisor's name)

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Fedor Stomakhin
15/05/2025