

Under What Conditions Is Local Differential Privacy Preferable to Central Differential Privacy? A Synthetic Study of Online Advertising Campaign Measurement

Litong Xing¹

Received July 14, 2026

Accepted September 8, 2026

Electronic access October 15, 2026

The rapid development of online advertising systems in recent years has significantly increased their reliance on user data for analysis. Although personalized advertising enhances user experience and advertising effectiveness, large-scale information collection has made people concerned about user privacy and unauthorized user profiling. Differential privacy (DP) is a recent method used to protect individual information, while still allowing the resulting outputs to be utilized for statistical analysis. This paper analyzes the tradeoff between privacy protection and statistical utility to explore under what conditions local differential privacy (LDP) can outperform central differential privacy (CDP) in online advertising campaign measurement. A synthetic dataset of 10,000 users, eight campaigns, and 88,256 advertisement impressions is constructed, in which users differ in interest category, demographic characteristics, browsing history, device, and click probability, and each user contributes between four and twelve impressions. CDP is implemented with the Laplace and Gaussian mechanisms and LDP with randomized response, all under a stated user-level privacy budget. Performance is evaluated across privacy budgets from 0.25 to 4.0, training sizes from 500 to 8,000 users, and three click-through rate levels, averaged over 80 trials, using absolute and normalized error, calibration, campaign ranking, and a measured attribute-inference attack. At a user-level budget of $\epsilon = 1$, weighted campaign mean absolute error was 0.14 percentage points for CDP with Laplace noise, 0.57 for CDP with Gaussian noise, and 6.60 for LDP randomized response. LDP also required 70,660 randomization operations per release against eight for the central mechanisms, and lost accuracy as the released domain widened. The attack analysis found no mechanism-level privacy advantage for LDP against an observer of the released output, but LDP eliminated the exposure of raw records when the curator was inspected or compromised. Overall, this study shows that neither CDP nor LDP is 100% superior to the other. Instead, privacy protection solutions should be selected based on respective demands for privacy preservation and data utility, and these results apply only to the synthetic conditions tested.

Keywords: Differential Privacy, Local Differential Privacy (LDP), Central Differential Privacy (CDP), Online Advertising, Click-Through Rate, Randomized Response, Campaign Measurement, Attribute Inference, Privacy-Utility Trade-off.

Introduction

Background

Modern-day advertising systems seem magical: ads from YouTube can somehow always be things that the user is interested in. This is heavily based on users' data of preferences in different aspects, such as whether the user is interested in sports or make-up products. The dependence on data has raised growing concerns about user tracking, behavioral profiling, and data breaches. Measurement studies of the live web have found third-party tracking to be pervasive and concentrated in a small number of ad-tech firms¹, and tracking iden-

tifiers to propagate widely between exchanges after a single ad impression². At the same time, users' own privacy concerns are not fixed quantities: they are uncertain, context-dependent, and manipulable by the parties collecting the data³, which means user consent alone is a weak safeguard. A privacy model called differential privacy has been proposed in recent years as a rigorous privacy framework⁴.

The unique challenge for online advertising systems is that they have to achieve both accurate advertising placement and protection of users' privacy. Modern advertising platforms rely on machine learning models to predict users' preferences and their click behaviors. But during the process of collecting data, they may disclose sensitive information from individual devices to the central curator. Recent studies have explored prac-

¹ The Stony Brook School, New York, USA

tical differential privacy frameworks that are useful for on-line advertising, indicating that the privacy protection models could be applied to the online advertisement prediction systems, while maintaining a reasonable accuracy level. Notably, Sun et al. achieve this in a centralized setting — the differentially private Gradient Boosted Decision Trees (GBDT) and Field-aware Factorization Machines (FFM) models are trained by a curator that has already received user profiles⁵. Their result therefore establishes that DP-preserved CTR prediction is feasible, but leaves open the harder question of whether comparable accuracy survives when noise is applied on-device, which is the gap this paper addresses.

Differential privacy is a mathematical privacy framework that limits how much any one person's data can influence the output of an analysis. It does not simply hide or remove access to the original data. Instead, a differentially private method produces similar output distributions whether one person's information is included or excluded. Random noise is commonly used to achieve this protection. Increasing the amount of noise generally provides stronger privacy but reduces the accuracy of the released results⁴.

Central vs. Local Differential Privacy

Within the broad category of differential privacy, there are two major branches: central DP and local DP. Specifically, Central DP (CDP) has randomized noise added by a trusted central server after the raw data has been collected and sent to it. Users need to place high trust in the curator, since the raw data is collected and secured through it. Examples of CDP can be widely found in Census-style systems such as the US Census Bureau⁶. Conversely, Local DP (LDP) has noise added to the dataset locally on users' devices before it is sent to the curator for further analysis. In this case, users do not need to trust the curator that much because the data is already being secured before it's sent⁷. It is widely used by Apple to protect users' privacy while improving their system's performance⁸.

My work

The research gap addressed in this study is whether the usual conclusions drawn from simple one-response privacy examples continue to hold in an advertising setting with multiple campaigns, repeated impressions from the same user, different click probabilities, and different levels of trust in the curator. The study asks how CDP and LDP compare when the total privacy budget is measured per user rather than separately for every impression. It also asks how privacy budget, dataset size, and click sparsity affect campaign measurement and ranking. Finally, it examines whether LDP's main advantage comes from the privacy mechanism itself or from the fact that the curator does not store raw user responses. Based on

the known properties of the mechanisms, CDP was expected to produce lower estimation error, while LDP was expected to provide an architectural advantage when the curator could not be fully trusted.

Background and Related Work

Basic Differential Privacy

The concept of differential privacy (DP) was first introduced by Dwork et al. (2006)⁴. It guarantees that whether any single data point is included in or excluded from a dataset does not substantially influence the computational outcome. Specifically, DP does not anonymize the whole dataset directly; instead, it limits how much any single record can influence the released result by adding noise to the dataset. This technique addresses the shortcomings of traditional data security methods, which proved easier to be attacked if the way to secure data is by merely deleting some identifiers^{9, 10}. Those traditional methods relied on de-identification: stripping names, identification numbers and other explicit identifiers before a dataset was released. Sweeney showed that the quasi-identifiers left behind in such a release—a combination such as ZIP code, birth date and sex—are frequently enough to single out an individual once the release is linked against a public register⁹. Narayanan and Shmatikov demonstrated the same weakness on a real release, de-anonymizing subscribers in the published Netflix Prize ratings dataset by matching it against publicly available IMDb profiles¹⁰. The problem is not confined to sparse ratings data. De Montjoye et al. analyzed fifteen months of mobile-phone location traces for 1.5 million people and found that four approximate spatiotemporal points were enough to uniquely identify 95% of individuals, and that coarsening the data in space and time reduced this uniqueness only slowly¹¹. Differential privacy responds to these failures by abandoning de-identification in favor of a formal, provable bound on how much any one individual's record can influence a released result⁴.

A randomized mechanism M satisfies ϵ -differential privacy if,

$$Pr[M(D) \in S] \leq \exp(\epsilon) \times Pr[M(D') \in S].$$

In this definition, D and D' are two neighboring datasets that differ by an individual's data point. S denotes any possible output the mechanism can have.

Intuitively, the outputs from two slightly different datasets, where a small number of data points are either removed or included, should stay close to each other rather than being exactly the same. Through this mechanism, the published result makes it very hard, though not completely impossible, to determine whether an individual has participated in this dataset.

These features of differential privacy limit privacy risks and protect data security better than traditional methods.

The parameter ϵ , privacy budget, controls the trade-off between privacy protection and data utility¹². For a smaller value of ϵ , the two outputs would be forced to perform more similarly, which is known as offering stronger privacy protection. On the other hand, larger ϵ would allow a greater difference between the two outputs; this can improve the utility of the results, but the data security is weakened.

Central Differential Privacy

Central Differential Privacy(CDP) is a traditional and widely studied DP model. Under CDP, users submit their raw data to a trusted central server, and the central server conducts statistical analysis on that raw data; finally, it applies differential privacy to the large dataset so that the result can be released to the public. In general, privacy is ensured through trusted servers instead of users themselves.

The major assumption made in CDP is that users have a trusted manager/server. The server has access to all users' raw data, but it's responsible for producing secured summary data to publish. Therefore, the viewers of the published data cannot determine whether any individual is included in the dataset because the generalized result is compiled from massive raw datasets. Since raw data are being compiled together to conduct differential privacy and produce results, CDP is generally more accurate than local differential privacy¹³, and it is more widely used in large-scale data analysis systems. This gap is not an artifact of any particular mechanism. Duchi et al. show that ϵ -local privacy reduces the effective sample size from n to approximately $n\epsilon^2$, so LDP estimators converge more slowly than their centralized counterparts by a factor that grows as ϵ shrinks¹³. The simulations reported in the Discussion should therefore be read as an empirical illustration of a proven asymptotic separation, not as independent evidence for it.

The technique used in CDP is the Laplace mechanism, in which noise is added to the dataset after it has been aggregated. The noisy output is defined as,

$$A(x) = f(x) + \text{Lap}(\Delta f / \epsilon),$$

where $f(x)$ is the true query result, Δf is the sensitivity of the query, and ϵ is the privacy budget. The noise added is scaled by the sensitivity of the query and the desired privacy level. In general, queries with lower sensitivities require less noise, which allows the published results to remain more accurate.

In the context of online advertisement, central differential privacy is used to calculate statistics such as user click-through rates and user interests. Since noise is added after the aggregation, the calculated result is generally more accurate than those produced from local differential privacy. As a result,

central differential privacy is usually considered the more effective way for advertising platforms to secure users' raw data, but this could only operate under a trusted server assumption.

However, the downside of central differential privacy is obvious, because the trusted server is only an assumption, so there is a possibility that raw data could be leaked on its way to the trusted server. The benefit of central differential privacy, which is generating more accurate results, is significantly less useful when the risk of raw data leakage is high. Therefore, while central differential privacy often provides higher utility and accuracy, local differential privacy provides the better security.

Large technological companies have unique challenges when they're collecting users' information, as they need a large amount of data to train their model for better optimization. And training machine learning models in a traditional centralized system requires users to send their raw data to the central curator¹⁴, a premise that federated learning was designed to avoid by keeping training data on-device. So, such a system is fully relying on the curator's security. When the number of users and the number of raw data points collected increase, the potential consequences of security issues rise. Therefore, for large-scale company data analysis, privacy protection technology has become more important¹⁵.

In addition to traditional user preference data, modern online platforms also collect contextual information such as location to enhance personalization and recommendation effectiveness. However, location information may reveal sensitive behavioral patterns of individuals, such as daily routines and frequently visited places. Therefore, privacy-preserving data collection methods must protect both explicit personal information and indirect information inferred from behavioral patterns. Research on privacy-preserving location systems explores methods to generate useful aggregated statistical information while minimizing the exposure of individual user information; Popa et al., for example, combine cryptographic aggregation with differential privacy to compute aggregate statistics over location data without revealing any individual user's path¹⁶. Notably, their design avoids the trusted curator that central differential privacy otherwise assumes, which indicates that the choice between central and local models is better understood as a spectrum of trust arrangements than as a binary.

Local Differential Privacy

Local differential privacy is a privacy model that doesn't need a trusted data server, as we mentioned before. Unlike central differential privacy, where users need to submit their raw data to the central curator, all the raw user data is secured on the user's own device and then sent to an aggregation server to be analyzed and to produce results. As a result, the server would

never have access to the original data, providing strong protection against data leakage and insider attacks as may appear possible in the case of central differential privacy.

The core idea of local differential privacy is that each user randomizes their own data using a randomized mechanism. Here, one of the earliest and most widely used methods is called the randomized response proposed by Warner (1965)¹⁷, which was later shown to be a differentially private mechanism and became a very important technique in local differential privacy⁷. Using this approach, the user might report a randomized incorrect answer, or intentionally change one of the responses with a certain probability or following a certain procedure. Therefore, it will be difficult for an observer to infer the users' exact information, while still giving reasonably accurate statistics when all the individual answers get aggregated.

As reflected from the equation:

$$RR(x) = \begin{cases} x & \text{with probability } \frac{e^\epsilon}{e^\epsilon + 1} \\ 1-x & \text{with probability } \frac{1}{e^\epsilon + 1} \end{cases}$$

Here, x represents the user's true response, and ϵ is the privacy budget as mentioned before. Smaller values of the privacy budget increase the privacy level by increasing the probability that the generated data differs from the original data, so the result would be more private. Consequently, as the required privacy level increases, the statistical accuracy decreases¹⁸.

Randomized response is not merely a convenient choice for binary data. Kairouz, Oh and Viswanath show that for a broad class of information-theoretic utility measures the optimal ϵ -locally-private mechanism belongs to a family of staircase mechanisms, and that binary randomized response is itself optimal in both the high-privacy and low-privacy regimes¹⁸. The mechanism used in this study is therefore a principled choice for the binary click outcome rather than a simplification adopted for convenience.

In online advertising systems, Local Differential Privacy can be used to collect user information on browsing behavior and advertisement preferences without exposing each individual's raw data. Specifically, the devices would first randomize the response before sending it to the advertising platform for further data analysis. So the user's actual preference for advertisements would still be private through this process. Although each individual's report may not be 100% accurate, the platform can still make a good estimate of users' overall preferences by aggregating a large number of user results.

A very famous example of local differential privacy's application is Google's RAPPOR, (Randomized Aggregatable Privacy Preserving Ordinal Response). This technology was developed to collect browser data while protecting users' pri-

vacacy¹⁹. Later, Apple developed similar local differential privacy techniques to collect users' preferences on emoji, word choice, etc. while keeping users' raw data on their individual devices⁸. Microsoft subsequently deployed LDP mechanisms for repeated telemetry collection across millions of Windows devices¹⁵, demonstrating that the model scales to production settings with recurring, rather than one-shot, data collection. All of these processes are done without sending users' raw data to the central servers. Those examples demonstrated that LDP can be used for large commercial systems. They do not, however, show that it comes without cost: RAPPOR requires very large client populations to recover accurate statistics, and Ding et al. show that the guarantee erodes under repeated collection¹⁵. Scale, in other words, is what makes LDP viable, and the accuracy cost is paid in the size of the population required.

Besides collecting statistical information, differential privacy is also being studied as a private Machine learning method. Many modern advertising platforms rely on machine learning algorithms for recommendation and optimization. Specifically, the privacy model must balance both statistical utility and model training performance. Research shows that balancing the tradeoff between model accuracy and the amount of information disclosed during the training process is key²⁰. That result is established for centralized training by a curator that already holds the raw data, however, so its accuracy figures are an upper bound on what an equivalent on-device method could reach, and the size of that gap is what the present study measures. This tradeoff is especially important for advertising platforms, because prediction accuracy directly affects advertising effectiveness²¹.

Comparison of CDP and LDP

In general, the difference between CDP and LDP is explained through Table 1 below.

The contrasts in Table 1 are cleaner in principle than in practice. Bernau et al. compared one central and two local mechanisms under a white-box membership inference attack and found that the empirical privacy-accuracy trade-off was comparable across the two families, despite their ϵ values differing by orders of magnitude²². Their conclusion is that the differential privacy upper bound sits far from actual susceptibility to inference attacks, so a small ϵ under CDP and a large ϵ under LDP can carry similar practical risk. Membership inference is the standard tool for measuring that susceptibility, following the shadow-model construction of Shokri et al.²³. This has a direct consequence for the present study: the measured attribute-inference comparison reported in the Discussion assumes a worst-case fully compromised curator, and that assumption—not the ϵ values themselves—is what drives the result.

Table 1 Comparison between Central DP and Local DP

Feature	Central Differential Privacy	Local Differential Privacy
Trusted curator required	Yes	No
Raw data visible to server	Yes	No
Where noise is added	After data aggregation	Before Data aggregation
Statistical accuracy	Higher in the campaign-measurement experiments	Lower in the repeated-report experiments
Privacy protection	Depends on the privacy parameters and threat model	Depends on the privacy parameters and threat model
Vulnerability to server breach	Higher raw-record exposure because the curator receives the original records	Lower raw-record exposure if the curator stores only randomized reports
Typical applications	Google, US Census	Apple, Google RAPPOR
Best suited for	Applications that require high statistical accuracy and can use a trusted curator	Applications where reducing trust in the curator is more important than statistical accuracy

The split is also less binary than the table suggests. Naseri et al. evaluated both local and central differential privacy inside federated learning and found that the two defend against different attacks with different protection-utility trade-offs, rather than one simply dominating the other²⁴. Which model is preferable therefore depends on the threat being defended against rather than on a general ranking—a point the attribute-inference experiment reported in the Discussion makes concrete by separating the threats: against an observer of the released output the two mechanisms are comparable, and LDP's advantage appears only once the curator itself is compromised.

Although those differences between local differential privacy and central differential privacy have been well established, it still remains uncertain how factors such as privacy budget, dataset size, and data sparsity influence the tradeoff between data utility and accuracy in the context of online advertising systems.

Open Problems and Limitations of Prior Work

The comparisons above establish that central and local differential privacy differ in trust assumptions and in accuracy. They do not settle three questions that a deployed advertising system must answer, and it is these that the present study is positioned against.

Composition and Repeated Release

A privacy budget spent once is not the situation an advertising platform faces. Users receive many impressions per day, and the guarantee that matters is the composed one. Basic composition is linear— k queries at ϵ each yield $k\epsilon$ —but this bound is loose. Dwork, Rothblum and Vadhan proved an advanced composition theorem giving a bound growing roughly as $\sqrt{k}$ rather than k , at the cost of introducing a δ term²⁵. Kairouz, Oh and Viswanath later established the exact optimal composition bound for ϵ -differentially private mechanisms, showing

that even advanced composition is not tight²⁶. Subsequent relaxations were developed specifically to track composition more sharply: concentrated differential privacy²⁷, Rényi differential privacy²⁸, and Gaussian differential privacy²⁹ each replace the ϵ - δ pair with an accounting object that composes cleanly. These are not merely theoretical refinements—the 2020 Census disclosure avoidance system is built on zero-concentrated differential privacy⁶, and the experiments in this paper use the same machinery to convert a user-level ρ -zCDP budget into the reported (ϵ, δ) guarantee.

Dimensionality

The accuracy gap between the two models widens as the released quantity grows in dimension. Under ϵ -local privacy the effective sample size falls from n to approximately $n\epsilon^2$ per estimated coordinate¹³, so the sample requirement compounds across attributes. A substantial literature addresses this directly. Bassily and Smith gave the first protocols achieving optimal error for succinct histograms over large domains³⁰; Wang et al. systematized and optimized frequency-estimation protocols, showing large constant-factor differences between mechanisms that are asymptotically equivalent³¹; Ye and Barg established optimal schemes for discrete distribution estimation under local privacy³²; and Cormode, Kulkarni and Srivastava studied marginal release, which is the operation a platform actually needs when correlating several user attributes³³. For genuinely high-dimensional crowdsourced data, Ren et al. combine dimensionality reduction with local randomization³⁴, and Wang et al. address multidimensional numeric and categorical collection jointly³⁵.

Dimensionality is also where local randomization becomes expensive in resources rather than only in accuracy. Because each client must randomize and transmit its own report, uplink cost scales with the encoding rather than with the size of the released statistic: a unary encoding of a d -category domain sends d bits per report, against the $\lceil \log_2 d \rceil$ bits a central upload requires. Bassily and Smith's protocols were motivated

in part by this, achieving optimal error while reducing per-user communication to a single bit³⁰, and Wang et al. show that mechanisms which are asymptotically equivalent in error differ substantially in the constants that determine practical cost³¹. Client-side computation scales the same way, since randomization occurs once per report rather than once per released coordinate. Industrial deployments reflect these constraints: RAPPOR requires very large client populations to recover accurate statistics¹⁹, and Ding et al. found single-round guarantees degrading rapidly enough under repeated collection that mechanisms had to be redesigned specifically for recurring telemetry¹⁵.

The private release studied here is a campaign-count vector over eight campaigns, which is multidimensional but far smaller than a realistic advertising profile of tens to hundreds of attributes. The Discussion reports measured communication, computation, and dimensionality costs for this release, and the comparison presented below should therefore be read as favorable to local differential privacy relative to what full-profile collection would demand.

Differential Privacy in Deployed Advertising Systems

The question this paper asks is no longer hypothetical. As third-party cookies are deprecated, browser vendors have begun shipping differentially private advertising measurement APIs, and the local-versus-central choice is being made in production. Tullii et al. survey the resulting research agenda from an industry perspective and identify utility loss from per-impression local randomization as the principal open obstacle³⁶. Delaney et al. give a formal treatment of ad conversion measurement under differential privacy, characterizing which combinations of attribution rule, adjacency relation and contribution bounding are operationally valid³⁷. Tholoniati et al. analyze the measurement APIs proposed by Google, Apple, Meta and Mozilla, replacing their budgeting components with an on-device scheme that admits more queries at equivalent protection³⁸. Sun et al. remain the closest work to the present study, but operate entirely in the central model⁵. What is missing is a controlled, like-for-like comparison of the two trust models on an identical advertising workload, with privacy budget, population size and click sparsity varied systematically.

Unresolved Questions

Three problems persist across this literature. First, ϵ is not comparable across trust models: an ϵ of 1 under central and under local differential privacy describe different adversaries, and Dwork, Kohli and Mulligan document how inconsistently ϵ is chosen and reported across deployed systems, making cross-system comparison unreliable³⁹. Second, the theoretic

cal guarantee and the empirical risk diverge: Jayaraman and Evans show that differentially private machine learning implementations in practice either provide little useful privacy or destroy utility, depending on the relaxation used⁴⁰, echoing Bernau et al.'s finding at the mechanism level²². Third, and most relevant here, the trust assumption is usually treated as binary—curator trusted or not—whereas real deployments sit on a spectrum, with partial trust, opt-in subpopulations⁴¹ and anonymizing infrastructure⁴² all changing the calculus. This study addresses the third by making the adversary an explicit experimental variable rather than a background premise, evaluating an output-only observer, an honest-but-curious curator, and a fully compromised curator separately. To explore this question further, the following section will present a series of simulation experiments using Python code to compare the performance of local differential privacy and central differential privacy.

Simulations and Experiments

Experimental Design

This study used a synthetic, impression-level advertising experiment to compare CDP and LDP. The main dataset contains 10,000 users, of whom 8,000 were used for campaign measurement and 2,000 were held out for evaluation. The users generated 88,256 impressions across eight advertising campaigns during a simulated seven-day period. The outcome of each impression was a binary variable called clicked, where 1 represented a click and 0 represented no click. Click-through rate (CTR) was defined consistently as the number of clicks divided by the number of impressions.

The privacy unit was one user over the complete seven-day study. Each user was limited to no more than 12 total impressions and no more than two impressions per day. These limits were applied before the privacy mechanisms because a user who contributes an unlimited number of records could have an unlimited influence on the results. Campaign assignments and impression information were treated as information already known to the advertising platform. The protected information in this experiment was the vector of each user's click outcomes.

The privacy budgets were $\epsilon = 0.25, 0.5, 1.0, 2.0$, and 4.0 . These values were selected to study the change from stronger to weaker privacy and are not presented as recommended values for real systems. Training-set sizes of 500, 1,000, 2,500, 5,000, and 8,000 users were tested to show how the methods behave as the amount of data increases. Each condition was repeated 80 times to measure the effect of the mechanisms' randomness. The Gaussian mechanism used $\delta = 10^{-6}$. Click sparsity was tested by changing the click-generation log odds, producing realized CTRs of approximately 2.31%, 5.75%, and

13.62%.

Synthetic Dataset Generation

The synthetic advertising dataset was generated in Python. The full generator, all experiments, and the scripts that produce Figures 1 through 5 are available in the accompanying GitHub repository at https://github.com/litongxing11/Differential_Privacy_Examples⁴³.

Each synthetic user was assigned an age group, geographic region, preferred device, general response tendency, eight interest scores, and browsing-history counts for eight advertising categories: sports, technology, fashion, travel, gaming, finance, food, and health. Each campaign was assigned a category, a starting response probability, a quality effect, a target age group, a preferred device, and a delivery weight.

Each user received between four and twelve advertising impressions over seven simulated days. Campaign assignment was influenced by the user's interest in the campaign category. Each impression also included the device, advertisement placement, time of day, and number of earlier impressions from the same campaign. These features affected the probability that the user would click.

A logistic function was used to combine the user, campaign, and context features into a different click probability for every impression. The actual click outcome was then randomly generated from that probability. Therefore, clicked represents a response to an advertisement impression rather than a static statement of interest. Campaign CTR was calculated as clicks divided by impressions.

As a check that the added features contained meaningful information rather than noise, a non-private prediction model using the user and impression features was compared with a campaign-only baseline on the held-out users. The campaign-only model achieved a ROC AUC of 0.536, while the feature model achieved a ROC AUC of 0.641. This prediction model was used only to check the synthetic data and was not presented as a differentially private training method.

Central DP Implementation

Two central mechanisms were implemented against the same released statistic. In both, the CDP query released a vector containing the click count for each of the eight campaigns. Because one user could contribute no more than 12 impressions, the user-level L1 sensitivity of this vector was limited to 12. For the Laplace mechanism, independent Laplace noise with scale $12/\epsilon$ was added to each campaign's click count.

A Gaussian CDP mechanism was also tested. It used a conservative user-level L2 sensitivity of 12 and zero-concentrated differential privacy (zCDP) accounting²⁷. The zCDP privacy parameter was selected so that the mechanism satisfied the re-

ported (ϵ, δ) bound, where $\delta = 10^{-6}$. This mechanism was included to examine a method with clearer accounting for repeated releases. The Gaussian and Laplace results do not represent exactly the same formal privacy guarantee and should therefore be compared cautiously.

After noise was added, estimated click counts were limited to the valid range from zero to the campaign's number of impressions. The campaign measurement estimate was the limited click count divided by the campaign's impression count. Jeffreys smoothing was used only when a valid probability was needed for log loss and other prediction metrics.

Local DP Implementation

The local mechanism moves the same randomization from the curator to the client. Under LDP, each click outcome was randomized on the user's device before being sent to the curator. If a report received privacy budget ϵ_{report} , the probability of reporting the true value was ϵ raised to ϵ_{report} divided by ϵ raised to ϵ_{report} plus 1. Otherwise, the reported value was changed from 0 to 1 or from 1 to 0. The randomized-response formula introduced in the Local Differential Privacy subsection above therefore uses the per-report portion of the budget, not the per-user total.

Because each user contributed several impressions, the complete user-level privacy budget could not be used separately for every report. Under the main equal-allocation method, a user with m impressions assigned ϵ_{user}/m to each report. The privacy budgets of that user's reports therefore added to the stated total user-level privacy budget. The randomized reports were corrected using the standard unbiased randomized-response estimator, in which the reported proportion is mapped back to an estimate as $(\hat{p} - (1 - p)) / (2p - 1)$, and then aggregated by campaign.

A supplementary utility experiment also assigned larger portions of the same total privacy budget to campaigns with higher public priority weights. The allocation used only public campaign information and did not use private clicks, interests, demographics, or measured errors. This experiment tested whether utility could be redistributed toward an objective chosen in advance without increasing any user's total privacy budget.

Evaluation Metrics

Campaign measurement was evaluated using mean absolute error (MAE), root mean squared error (RMSE), bias, and estimator variance. Absolute error was calculated on the proportion scale by comparing estimated CTR with true CTR. Ordinary relative error was calculated as the absolute difference between estimated and true CTR divided by true CTR. This is equivalent to calculating the same relative error on the

click-count scale when the true count is greater than zero.

Ordinary relative error was treated as undefined when the true click count was zero. It was not set equal to zero. The sparsity experiment therefore also reported symmetric mean absolute percentage error (sMAPE), which uses both the estimate and true value in its denominator and remains usable in zero-event conditions. A value of zero was assigned only when both the estimate and true value were zero.

Unless stated otherwise, campaign MAE is impression-weighted, so that error on a high-volume campaign contributes proportionally more than the same error on a small one. The released campaign rates were also evaluated on the held-out users using log loss, Brier score, and campaign calibration error. Campaign ranking was evaluated using normalized discounted cumulative gain (NDCG) and Spearman rank correlation. Precision, recall, and F1 score were calculated as secondary measures by selecting the top 25% of campaigns according to the released campaign rates. The results were averaged across 80 mechanism runs, and 95% Monte Carlo intervals were calculated from the variation across those runs.

Measured Attribute-Inference Analysis

The privacy comparison was rebuilt around measured attacks rather than an assumed worst case, and the former fixed information leakage score was removed. Privacy exposure was instead evaluated through 200,000 simulated Bayes-optimal attribute-inference attacks. This attack benchmark used a single binary target attribute so that the attacker's prior probabilities and likelihoods could be calculated clearly. The assumed attribute prevalence was 0.30, and the privacy budget was $\epsilon = 1$.

Three threat models were evaluated. In the output-only model, the attacker observed either the released CDP statistic or the user's LDP randomized report. For the CDP attack, the attacker was also assumed to know the non-target records, which allowed those records to be removed from the released count. In the honest-but-curious-curator model, the CDP curator observed the raw stored attribute, while the LDP curator observed only the randomized report. In the fully compromised-curator model, the attacker received whatever was stored by the curator: the raw CDP record or the randomized LDP report.

In every threat model, the target was the selected user's binary attribute. The prior-only attacker guessed using only the assumed prevalence. Attack success was measured using accuracy advantage above the prior-only baseline, Bayes risk, log loss, and mutual information. Only the output-only model was used to compare the privacy mechanisms themselves. The curator models were used to compare the system architectures and the consequences of storing raw rather than randomized records.

Discussion

Effect of Privacy Budget

The effect of privacy budget was tested using ϵ values of 0.25, 0.5, 1.0, 2.0, and 4.0 with 8,000 training users and 2,000 held-out users. The total privacy budget was measured per user. Figure 1 reports weighted campaign MAE and held-out log loss. The points show the means across 80 mechanism runs, and the shaded areas show 95% Monte Carlo intervals.

Increasing ϵ generally improved the utility of all three private methods because the mechanisms introduced less randomization. CDP with Laplace noise produced the lowest error under the tested conditions. At $\epsilon = 1$, weighted campaign MAE was 0.14 percentage points for CDP with Laplace noise, 0.57 percentage points for CDP with Gaussian noise, and 6.60 percentage points for LDP randomized response. Held-out log loss at $\epsilon = 1$ was approximately 0.223 for both CDP methods and 0.353 for LDP. LDP had much higher error because every user's total privacy budget had to be divided across that user's repeated impressions.

Additional Utility Measures

The full set of utility measures at a user-level privacy budget of $\epsilon = 1$ is reported in Tables 3 and 4. Table 3 covers campaign measurement error and Table 4 covers held-out prediction and ranking quality.

Precision, recall, and F1 were secondary operational measures based on selecting the top 25% of campaigns. Because clicks were uncommon, log loss, Brier score, calibration error, and ranking quality provide more useful measures of the released campaign probabilities. The Spearman rank correlation is the sharpest indicator of LDP's cost to campaign ranking: it fell to 0.072 against 0.707 for CDP with Laplace noise, meaning the LDP-released ordering of campaigns carried almost no rank information.

Effect of Dataset Size

The dataset-size experiment used 500, 1,000, 2,500, 5,000, and 8,000 training users. The total user-level privacy budget was fixed at $\epsilon = 1$, and the same set of 2,000 users was used for held-out evaluation. Figure 2 reports weighted campaign MAE and held-out campaign NDCG.

Weighted campaign MAE generally decreased as the number of training users increased. CDP with Laplace noise had the lowest error at every tested size. LDP benefited from additional users, but it remained much less accurate because the privacy budget was divided across repeated reports. Campaign NDCG did not change perfectly smoothly because the experiment ranked only eight campaigns and small changes could alter their order. Therefore, the results support the conclusion

Table 2 Threat models evaluated in the attribute-inference analysis.

Threat model	Attacker's observation	CDP observation	LDP observation
Output-only	Released private output	Noisy CDP count after known non-target records were removed	Selected user's randomized report
Honest-but-curious curator	Data available to the curator	Raw stored attribute	Randomized report
Fully compromised curator	All records stored by the curator	Raw stored attribute	Randomized report

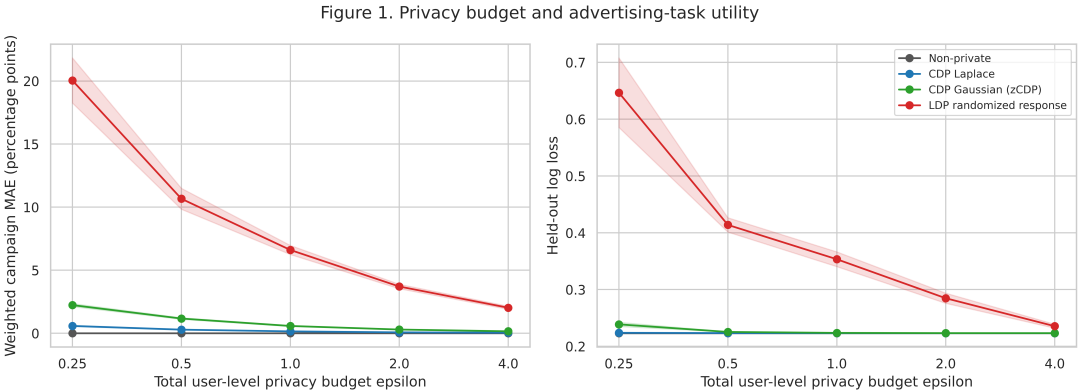


Figure. 1 Effect of total user-level privacy budget on weighted campaign MAE and held-out log loss. Results use 8,000 training users and 2,000 held-out users. Points are means across 80 mechanism runs, and shaded areas are 95% Monte Carlo intervals.

Table 3 Campaign measurement metrics at $\epsilon = 1$.

Method	Weighted MAE (pp)	Unweighted MAE (pp)	RMSE (pp)	Bias (pp)	Variance (pp ²)
CDP Laplace	0.143	0.145	0.199	0.006	0.044
CDP Gaussian	0.572	0.576	0.705	−0.011	0.529
LDP randomized response	6.599	6.645	8.218	2.247	66.907
Non-private	0.000	0.000	0.000	0.000	0.000

Table 4 Held-out prediction and ranking metrics at $\epsilon = 1$.

Method	Log loss	Brier	Calibration error	NDCG	Spearman	Precision	Recall	F1
CDP Laplace	0.2232	0.0553	0.0045	0.9887	0.7065	0.0664	0.2806	0.1074
CDP Gaussian	0.2237	0.0553	0.0072	0.9728	0.4452	0.0641	0.2741	0.1038
LDP randomized response	0.3534	0.0623	0.0657	0.9531	0.0717	0.0596	0.2525	0.0964
Non-private	0.2232	0.0553	0.0042	0.9972	0.6905	0.0669	0.2824	0.1082

that larger samples improve measurement accuracy, but they do not show that LDP becomes as accurate as CDP within the tested range.

Effect of Click Sparsity

Click sparsity was tested by applying three fixed shifts to the click-generation log odds while keeping the user, campaign, impression, and context generation process the same. The resulting non-private training CTRs were approximately 2.31%,

5.75%, and 13.62%. The total user-level privacy budget was fixed at $\epsilon = 1$, with 5,000 training users and 1,500 held-out users.

Figure 3 reports both absolute error and the zero-safe normalized measure sMAPE. The results show why both measures are necessary. LDP's weighted campaign MAE increased from approximately 6.56 percentage points at a CTR of 2.31% to 9.16 percentage points at a CTR of 13.62%. However, its sMAPE decreased from approximately 150.6%

Figure 2. Dataset size, campaign measurement, and ranking

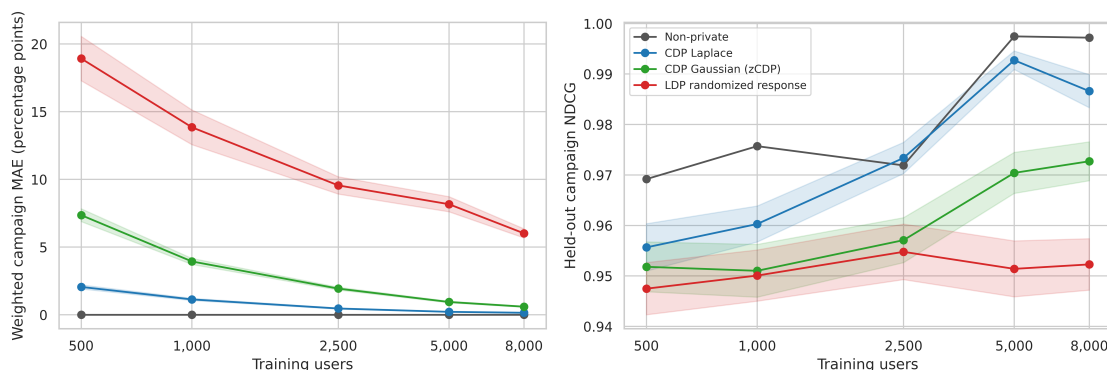


Figure. 2 Effect of training-set size on weighted campaign MAE and held-out campaign NDCG at a total user-level privacy budget of $\epsilon = 1$. Points are means across 80 mechanism runs, and shaded areas are 95% Monte Carlo intervals.

Figure 3. Corrected sparsity analysis: absolute and zero-safe normalized error

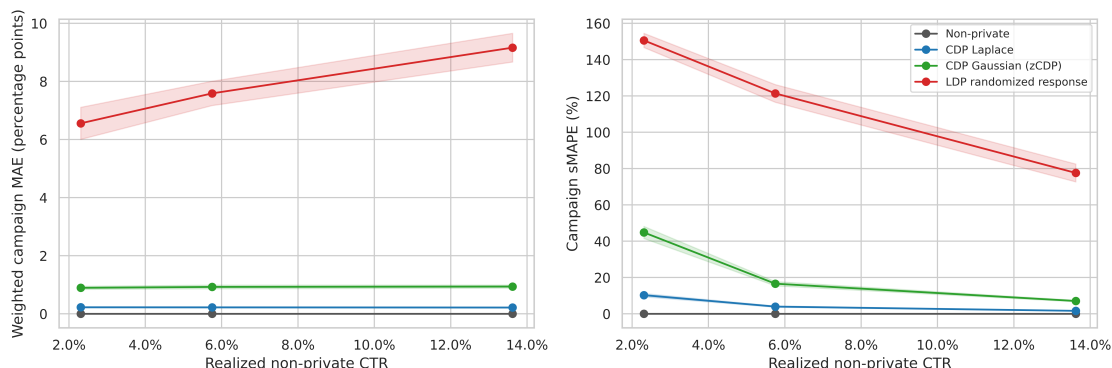


Figure. 3 Effect of realized non-private CTR on weighted campaign MAE and campaign sMAPE at $\epsilon = 1$. Absolute and normalized errors are shown separately because a reduction in normalized error does not necessarily mean that absolute accuracy improved.

to 77.6% because the true CTR in the denominator became larger. Therefore, higher CTR reduced normalized error but did not improve LDP's absolute accuracy. CDP with Laplace noise remained near 0.22 percentage points of absolute error across the three conditions.

The rise in LDP's absolute error is not a property of the randomized-response estimator itself, whose unclipped error stayed essentially flat near 10 percentage points across all three conditions. It follows instead from clipping the released estimate to the feasible interval. When the true rate lies close to zero, clipping truncates noise toward the truth and removed 37.2% of the error at the lowest tested CTR, against 26.4% at the middle level and only 9.0% at the highest. Sparsity therefore helps LDP in absolute terms through a boundary effect while hurting it in normalized terms, which is why the two scales must be reported together rather than one standing in for the other. None of the realized campaign counts in these runs were zero; the zero-event rule was nonetheless fixed be-

fore the experiments, with ordinary relative error recorded as undefined when the true count was zero while absolute error and sMAPE were still reported.

These error magnitudes are not purely statistical quantities. Johnson et al. found that ads served to users who had opted out of behavioral targeting fetched 52% less revenue on an ad exchange than comparable ads for users who permitted it²¹, establishing that the quality of user-preference estimates carries measurable commercial consequences. Their comparison is between full targeting and none rather than between two noise levels, so it does not translate into a price for any particular estimation error. What it does establish is the scale of what is at stake: the accuracy gap between the two mechanisms in Figures 1 through 3 falls somewhere within a range whose extreme is worth roughly half of exchange revenue. That is why identifying the conditions under which the gap narrows—chiefly large user populations—is a practical question rather than only a theoretical one.

Measured Attribute-Inference Results

Figure 4 reports the measured Bayes attribute-inference accuracy advantage above a prior-only baseline. The assumed attribute prevalence was 0.30, giving the prior-only attacker an accuracy of approximately 69.80%.

In the output-only model, the CDP Laplace attack achieved an accuracy of 72.14%, or an advantage of 2.35 percentage points above the prior-only baseline. The LDP randomized-response attack achieved an accuracy of 73.09%, or an advantage of 3.30 percentage points. Therefore, this experiment does not support the claim that the LDP mechanism leaked less information than the CDP mechanism at the same record-level ϵ .

The result changed when the curator could inspect or lose its stored records. The attacker could recover the raw CDP attribute with 100% accuracy, producing an advantage of 30.20 percentage points above the prior-only baseline. The LDP curator stored only the randomized report, so the LDP attack remained at 73.09% accuracy and an advantage of 3.30 percentage points. This is an architectural advantage of not storing raw records. It is outside CDP's trusted-curator threat model and should not be described as proof that the LDP mechanism always leaks less information.

The Bayes-risk and log-loss results followed the same pattern. In the output-only model, Bayes risk was 0.279 for CDP and 0.269 for LDP. When the raw CDP record was available, its Bayes risk fell to zero because the attacker knew the target exactly. These values are descriptive results from the stated attack model. No claim of statistical significance is made.

Communication, Computation, and Release Dimensionality

Accuracy is not the only cost of local randomization. Uplink is measured in bits per user per release window under the contribution bound of twelve impressions. Binary randomized response costs the same as a central upload at 48 bits, because both transmit one campaign index and one bit per impression. A unary encoding of the campaign domain, of the kind used by frequency oracles over large domains, costs twice the central upload at eight campaigns and 93 times as much at 1,024. Computation scales with reports rather than with released coordinates: one release required 70,660 randomization operations under LDP against eight under either central mechanism. Restricting the released domain to 2, 4, 6, and 8 campaigns at a fixed budget shows weighted campaign MAE growing from 2.78 to 6.25 percentage points under LDP while central Laplace holds at 0.14 throughout. These three costs all worsen as the released domain widens, which is the practical form of the dimensionality argument developed in the literature review.

Conclusion

This simulation compared user-level CDP and LDP for measuring and ranking eight synthetic advertising campaigns with repeated impressions from each user. The study evaluated changes in privacy budget, training-set size, and click sparsity using campaign measurement, prediction, ranking, and attribute-inference metrics.

CDP with Laplace noise provided the highest utility under the tested conditions. At $\epsilon = 1$ with 8,000 training users, its weighted campaign MAE was approximately 0.14 percentage points, compared with 0.57 percentage points for CDP with Gaussian noise and 6.60 percentage points for LDP randomized response. LDP remained less accurate because every user's total privacy budget was divided across several impression reports. Increasing the number of users improved LDP utility, but LDP did not become as accurate as CDP within the tested range. The sparsity experiment also showed that decreasing normalized error does not necessarily mean that absolute error decreased.

The measured output-only attack did not show a mechanism-level privacy advantage for LDP at the same record-level ϵ . LDP's main advantage appeared when the curator could inspect or lose its stored records. In that situation, the CDP curator contained raw attributes, while the LDP curator contained only randomized reports. This reduces the consequences of curator compromise, but it is a system-architecture advantage rather than proof that LDP always leaks less information.

These results do not identify one method as universally superior. CDP was preferable when accurate campaign measurement was the main goal and the curator could be trusted to protect raw records. LDP may be preferable when reducing the curator's access to raw user information is more important and the application can accept a substantial loss of statistical utility. The correct choice therefore depends on the privacy unit, contribution limits, total privacy budget, required accuracy, and threat model.

In this paper, central and local differential privacy are studied as separate approaches. Recent research has explored hybrid privacy models that combine the advantage from local differential privacy and central differential privacy⁴¹. So, hybrid methods can reduce the accuracy loss introduced by local randomization, while the amount of trust is still relying on central servers. A related line of work reaches a similar goal by a different route: the shuffle model shows that anonymizing locally randomized reports amplifies their privacy guarantee when it is accounted for centrally, so an LDP deployment may already be more private than its stated ϵ suggests⁴². These approaches are especially useful for online advertising systems, because for those platforms, statistical utility is the key. Future advertising platforms, which have both large user populations

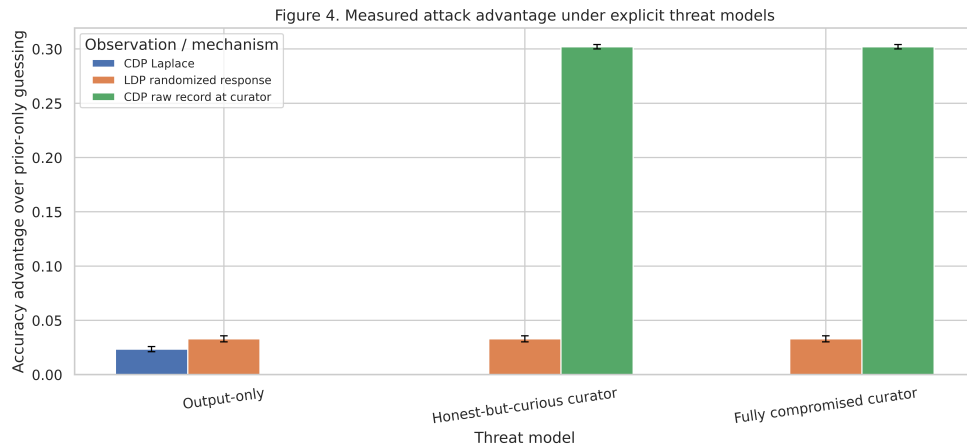


Figure. 4 Measured Bayes attribute-inference accuracy advantage above prior-only guessing under three threat models. Results are based on 200,000 simulated attacks with prior prevalence 0.30 and $\epsilon = 1$. Only the output-only condition provides a mechanism-level comparison. Error bars show paired 95% Monte Carlo intervals.

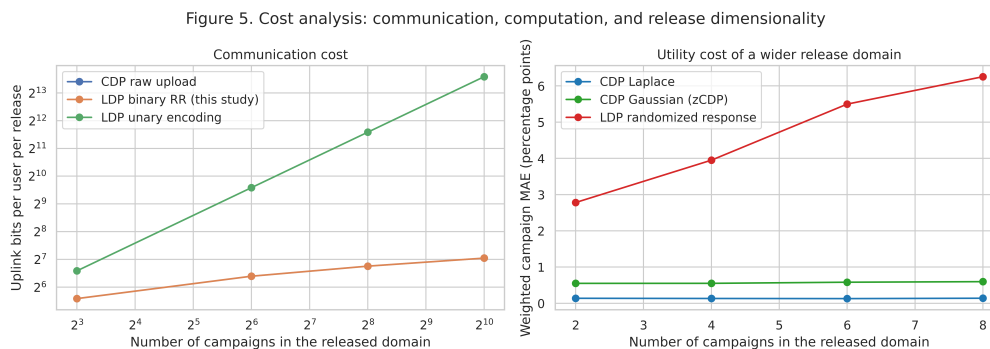


Figure. 5 Cost analysis. Left: uplink bits per user per release under three encodings, on log axes. Right: weighted campaign MAE as the released campaign domain widens, at a fixed user-level $\epsilon = 1.0$.

and a subset of consenting users, may rely on combinations of central and local differential privacy.

This study has several limitations. First, the data are synthetic and were not fitted to the traffic of a real advertising platform. The number of impressions, contribution limits, campaign effects, and public priority weights are simulated design choices. Second, campaign assignments and impression information are treated as already known to the advertising platform. The privacy mechanisms protect click outcomes rather than campaign exposure, browsing history, demographic characteristics, or every other part of an advertising system. Third, the non-private feature model was used only to check that the synthetic features contained predictive information; private machine-learning training was not studied. Fourth, the Gaussian and pure-DP mechanisms provide different formal guarantees, so their ϵ values should not be treated as perfectly equivalent. Fifth, the attack experiment targets one binary at-

tribute and does not represent every possible privacy attack. Sixth, the released statistic is a vector over eight campaigns, which is far smaller than a realistic advertising profile, so the comparison presented here should be read as favorable to local differential privacy relative to what full-profile collection would demand. Finally, the Python code is an educational simulation rather than production privacy software. Although communication and computational costs were measured for the released statistic, a real deployment would additionally require a tested differential-privacy library, secure randomness, privacy accounting, and access controls.

Code Availability

All simulation code and the scripts that generate Figures 1 through 5 are available at https://github.com/litongxing11/Differential_Privacy_Examples.

References

- 1 S. Englehardt, A. Narayanan. Online tracking: a 1-million-site measurement and analysis. in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* pg. 1388–1401, ACM, Vienna Austria, 2016. <https://doi.org/10.1145/2976749.2978313>.
- 2 M. A. Bashir, C. Wilson. Diffusion of user tracking data in the online advertising ecosystem. *Proceedings on Privacy Enhancing Technologies*. Vol. 2018, pg. 85–103, 2018, <https://doi.org/10.1515/popets-2018-0033>.
- 3 A. Acquisti, L. Brandimarte, G. Loewenstein. Privacy and human behavior in the age of information. *Science*. Vol. 347, pg. 509–514, 2015, <https://doi.org/10.1126/science.aaa1465>.
- 4 C. Dwork, F. McSherry, K. Nissim, A. Smith. Calibrating noise to sensitivity in private data analysis. in *Theory of Cryptography* Vol. 3876 pg. 265–284, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006. https://doi.org/10.1007/11681878_14.
- 5 J. Sun, L. Zhao, Z. Liu, Q. Li, X. Deng, Q. Wang, Y. Jiang. Practical differentially private online advertising. *Computers Security*. Vol. 112, pg. 102504, 2022, <https://doi.org/10.1016/j.cose.2021.102504>.
- 6 J. M. Abowd, R. Ashmead, R. Cumings-Menon, S. Garfinkel, M. Heinicke, C. Heiss, R. Johns, D. Kifer, P. Leclerc, A. Machanavajjhala, B. Moran, W. Sexton, M. Spence, P. Zhuravlev. The 2020 census disclosure avoidance system topdown algorithm. *Harvard Data Science Review*. Vol. Special Issue 2, 2022.
- 7 S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, A. Smith. What can we learn privately? *SIAM Journal on Computing*. Vol. 40, pg. 793–826, 2011, <https://doi.org/10.1137/090756090>.
- 8 G. Cormode, S. Jha, T. Kulkarni, N. Li, D. Srivastava, T. Wang. Privacy at scale: local differential privacy in practice. in *Proceedings of the 2018 International Conference on Management of Data* pg. 1655–1658, ACM, Houston TX USA, 2018. <https://doi.org/10.1145/3183713.3197390>.
- 9 L. Sweeney. K-anonymity: a model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*. Vol. 10, pg. 557–570, 2002, <https://doi.org/10.1142/S0218488502001648>.
- 10 A. Narayanan, V. Shmatikov. Robust de-anonymization of large sparse datasets. in *2008 IEEE Symposium on Security and Privacy (sp 2008)* pg. 111–125, IEEE, Oakland, CA, USA, 2008. <https://doi.org/10.1109/SP.2008.33>.
- 11 Y.-A. de Montjoye, C. A. Hidalgo, M. Verleysen, V. D. Blondel. Unique in the crowd: the privacy bounds of human mobility. *Scientific Reports*. Vol. 3, pg. 1376, 2013, <https://doi.org/10.1038/srep01376>.
- 12 J. Hsu, M. Gaboardi, A. Haeberlen, S. Khanna, A. Narayan, B. C. Pierce, A. Roth. Differential privacy: an economic method for choosing epsilon. in *2014 IEEE 27th Computer Security Foundations Symposium* pg. 398–410, IEEE, Vienna, 2014. <https://doi.org/10.1109/CSF.2014.35>.
- 13 J. C. Duchi, M. I. Jordan, M. J. Wainwright. Minimax optimal procedures for locally private estimation. *Journal of the American Statistical Association*. Vol. 113, pg. 182–201, 2018, <https://doi.org/10.1080/01621459.2017.1389735>.
- 14 B. McMahan, E. Moore, D. Ramage, S. Hampson, B. Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics* Vol. 54 pg. 1273–1282, PMLR, 2017.
- 15 B. Ding, J. Kulkarni, S. Yekhanin. Collecting telemetry data privately. in *Advances in Neural Information Processing Systems* Vol. 30, Curran Associates, Inc., 2017.
- 16 R. A. Popa, A. J. Blumberg, H. Balakrishnan, F. H. Li. Privacy and accountability for location-based aggregate statistics. in *Proceedings of the 18th ACM conference on Computer and communications security* pg. 653–666, ACM, Chicago Illinois USA, 2011. <https://doi.org/10.1145/2046707.2046781>.
- 17 S. L. Warner. Randomized response: a survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*. Vol. 60, pg. 63–69, 1965, <https://doi.org/10.1080/01621459.1965.10480775>.
- 18 P. Kairouz, S. Oh, P. Viswanath. Extremal mechanisms for local differential privacy. *Journal of Machine Learning Research*. Vol. 17, pg. 1–51, 2016.
- 19 Ú. Erlingsson, V. Pihur, A. Korolova. RAPPOR: randomized aggregatable privacy-preserving ordinal response. in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security* pg. 1054–1067, ACM, Scottsdale Arizona USA, 2014. <https://doi.org/10.1145/2660267.2660348>.
- 20 M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, L. Zhang. Deep learning with differential privacy. in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* pg. 308–318, 2016. <https://doi.org/10.1145/2976749.2978318>.
- 21 G. A. Johnson, S. K. Shriver, S. Du. Consumer privacy choice in online advertising: who opts out and at what cost to industry? *Marketing Science*. Vol. 39, pg. 33–51, 2020, <https://doi.org/10.1287/mksc.2019.1198>.
- 22 D. Bernau, J. Robl, P. W. Grassal, S. Schneider, F. Kerschbaum. Comparing local and central differential privacy using membership inference attacks. in *Data and Applications Security and Privacy XXXV* Vol. 12840 pg. 22–42, Springer International Publishing, Cham, 2021. https://doi.org/10.1007/978-3-030-81242-3_2.
- 23 R. Shokri, M. Stronati, C. Song, V. Shmatikov. Membership inference attacks against machine learning models. in *2017 IEEE Symposium on Security and Privacy (SP)* pg. 3–18, IEEE, San Jose, CA, USA, 2017. <https://doi.org/10.1109/SP.2017.41>.
- 24 M. Naseri, J. Hayes, E. De Cristofaro. Local and central differential privacy for robustness and privacy in federated learning. in *Proceedings 2022 Network and Distributed System Security Symposium* Internet Society, San Diego, CA, USA, 2022. <https://doi.org/10.14722/ndss.2022.23054>.
- 25 C. Dwork, G. N. Rothblum, S. Vadhan. Boosting and differential privacy. in *2010 IEEE 51st Annual Symposium on Foundations of Computer Science* pg. 51–60, IEEE, Las Vegas, NV, USA, 2010. <https://doi.org/10.1109/FOCS.2010.12>.
- 26 P. Kairouz, S. Oh, P. Viswanath. The composition theorem for differential privacy. *IEEE Transactions on Information Theory*. Vol. 63, pg. 4037–4049, 2017, <https://doi.org/10.1109/TIT.2017.2685505>.
- 27 M. Bun, T. Steinke. Concentrated differential privacy: simplifications, extensions, and lower bounds. in *Theory of Cryptography* (eds M. Hirt A. Smith) Vol. 9985 pg. 635–658, Springer Berlin Heidelberg, Berlin, Heidelberg, 2016. https://doi.org/10.1007/978-3-662-53641-4_24.
- 28 I. Mironov. Rényi differential privacy. in *2017 IEEE 30th Computer Security Foundations Symposium (CSF)* pg. 263–275, IEEE, Santa Barbara, CA, 2017. <https://doi.org/10.1109/CSF.2017.11>.
- 29 J. Dong, A. Roth, W. J. Su. Gaussian differential privacy. *Journal of the Royal Statistical Society Series B: Statistical Methodology*. Vol. 84, pg. 3–37, 2022, <https://doi.org/10.1111/rssb.12454>.
- 30 R. Bassily, A. Smith. Local, private, efficient protocols for succinct histograms. in *Proceedings of the forty-seventh annual ACM symposium on Theory of Computing* pg. 127–135, ACM, Portland Oregon USA, 2015. <https://doi.org/10.1145/2746539.2746632>.
- 31 T. Wang, J. Blocki, N. Li, S. Jha. Locally differentially private protocols for frequency estimation. in *26th USENIX Security Symposium (USENIX Security 17)* pg. 729–745, USENIX Association, Vancouver, BC, 2017.

- 32 M. Ye, A. Barg. Optimal schemes for discrete distribution estimation under locally differential privacy. *IEEE Transactions on Information Theory*. Vol. 64, pg. 5662–5676, 2018, <https://doi.org/10.1109/TIT.2018.2809790>.
- 33 G. Cormode, T. Kulkarni, D. Srivastava. Marginal release under local differential privacy. in *Proceedings of the 2018 International Conference on Management of Data* pg. 131–146, ACM, Houston TX USA, 2018. <https://doi.org/10.1145/3183713.3196906>.
- 34 X. Ren, C.-M. Yu, W. Yu, S. Yang, X. Yang, J. A. McCann, P. S. Yu. LoPub: high-dimensional crowdsourced data publication with local differential privacy. *IEEE Transactions on Information Forensics and Security*. Vol. 13, pg. 2151–2166, 2018, <https://doi.org/10.1109/TIFS.2018.2812146>.
- 35 N. Wang, X. Xiao, Y. Yang, J. Zhao, S. C. Hui, H. Shin, J. Shin, G. Yu. Collecting and analyzing multidimensional data with local differential privacy. in *2019 IEEE 35th International Conference on Data Engineering (ICDE)* pg. 638–649, IEEE, Macao, China, 2019. <https://doi.org/10.1109/ICDE.2019.00063>.
- 36 M. Tullii, S. Gaucher, H. Richard, E. Diemert, V. Perchet, A. Rakotomamonjy, C. Calauzènes, M. Vono. Open research challenges for private advertising systems under local differential privacy. in *Web Information Systems Engineering – WISE 2024* (eds M. Barhamgi, H. Wang X. Wang) Vol. 15440 pg. 107–122, Springer Nature Singapore, Singapore, 2025. https://doi.org/10.1007/978-981-96-0576-7_9.
- 37 J. Delaney, B. Ghazi, C. Harrison, C. Ilvento, R. Kumar, P. Manurangsi, M. Pál, K. Prabhakar, M. Raykova. Differentially private ad conversion measurement. *Proceedings on Privacy Enhancing Technologies*. Vol. 2024, pg. 124–140, 2024, <https://doi.org/10.56553/popets-2024-0044>.
- 38 P. Tholoniati, K. Kostopoulou, P. McNeely, P. S. Sodhi, A. Varanasi, B. Case, A. Cidon, R. Geambasu, M. Lécuyer. Cookie monster: efficient on-device budgeting for differentially-private ad-measurement systems. in *Proceedings of the ACM SIGOPS 30th Symposium on Operating Systems Principles* pg. 693–708, ACM, Austin TX USA, 2024. <https://doi.org/10.1145/3694715.3695965>.
- 39 C. Dwork, N. Kohli, D. Mulligan. Differential privacy in practice: expose your epsilons! *Journal of Privacy and Confidentiality*. Vol. 9, 2019, <https://doi.org/10.29012/jpc.689>.
- 40 B. Jayaraman, D. Evans. Evaluating differentially private machine learning in practice. in *28th USENIX Security Symposium (USENIX Security 19)* pg. 1895–1912, USENIX Association, Santa Clara, CA, 2019.
- 41 B. Avent, A. Korolova, D. Zeber, T. Hovden, B. Livshits. BLENDER: enabling local search with a hybrid differential privacy model. *Journal of Privacy and Confidentiality*. Vol. 9, 2019, <https://doi.org/10.29012/jpc.680>.
- 42 Ú. Erlingsson, V. Feldman, I. Mironov, A. Raghunathan, K. Talwar, A. Thakurta. Amplification by shuffling: from local to central differential privacy via anonymity. in *Proceedings of the 2019 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)* pg. 2468–2479, Society for Industrial and Applied Mathematics, 2019. <https://doi.org/10.1137/1.9781611975482.151>.
- 43 L. Xing. Differential privacy examples. https://github.com/litongxing11/Differential_Privacy_Examples 2026.