Privacy-centric Cross-publisher Reach and Frequency Estimation Via Vector of Counts

Jiayu Peng, Scott Schneider, Joseph G. Knightbrook, Laura Book, Sheng Ma, Xichen Huang, Michael Daub, Yunwen Yang, Jason Frye, Craig Wright, Evgeny Skvortsov, Preston Lee, Ying Liu, Jim Koehler

Google Inc.

Dec 2020

Abstract

Reach and frequency are two of the most important metrics in advertising management. Ads are distributed to different publishers with a hope to maximize the reach at effective frequency. Reliable cross-publisher reach and frequency measurement is called for, to assess the actual performance of branding and to improve the budget allocation strategy. However, cross-publisher measurement is non-trivial particularly under strict differential-privacy restrictions.

This paper introduces the first locally-differentially-private solution in the literature to cross-publisher reach and frequency estimation. The solution consists of a family of algorithms based on a data structure called Vector of Counts (VoC). Complying with the standard definition of differential privacy, the solution prevents attackers from telling if any specific user is reached or not with a given level of confidence. The solution enjoys particularly high accuracy for the estimation between two publishers. For more than two publishers, the solution enjoys small variance, at a risk of having bias in the presence of cross-publisher correlation of user activity.

Contents

1 Introduction

Privacy-centric cross-publisher measurement has recently received a great deal of attention. Advertisers are strategically distributing ads over diverse publishers to maximize the efficiency of branding. They are calling for industry standard measurement across different publishers (World Federation of Advertisers 2019, Media Rating Council 2019, Incorporated Society of British Advertisers 2019, Association of National Advertisers 2020). On the other hand, tech companies are restricting data available for measurement (Apple 2019, Google 2020, Facebook 2019), and governments are imposing affirmative consent (Council of European Union 2016 a.k.a. GDPR, California State 2018 a.k.a. CCPA) for advertising data. With privacy restrictions, cross-publisher measurement is becoming difficult, and increasingly dependent on probabilistic/statistical methods.

This paper studies the most basic problem of cross-publisher measurement: reach and frequency estimation. Advertisers allocate budget to different publishers with the goal to reach every corner of the universe (of users). To see if this goal is achieved, one needs to estimate how many *unique* users the ad campaign has actually reached. This is important to advertisers because an ad campaign can have a high number of impressions (hence high cost) but reach the same people many times. Besides reach estimation, advertisers are also interested in the estimation of frequency distribution. Reach is maximized when each user is exposed to the ad campaign exactly once, i.e., when each user has a frequency of one. This is not ideal though. Often, an effective frequency is desired, which sufficiently influences users' perception of a brand before introducing wasteful exposures. It is desirable to know whether the frequency distribution is concentrated about the effective frequency.

Reach and frequency estimation is important to advertisers, but it is difficult across publishers, because different publishers cannot share any sensitive data with each other. To achieve privacy-preserving data sharing, there are two approaches at a high level: global versus local solutions. They differ in that a global solution processes data through a secure, central system, while a local solution does not. In terms of pros and cons, global solutions are accurate and computationally expensive, while local solutions are less accurate but economical. Local solutions are scalable for the decomposition of reach and frequency across flexible sets of publishers. This paper is focused on local solutions. We refer readers to Subsection 2.4 of this paper for

detailed descriptions of local and global solutions, and to Wright et al. (2020) for a state-of-the-art global solution.

In local solutions, each publisher releases a summary of their data. The summary ought to contain necessary information for reach and frequency estimation, and meanwhile be sufficiently aggregated and noised to meet certain privacy standards. We propose a summarizing algorithm called Vector of Counts (VoC) that satisfies the differential privacy (DP) standard, as well as algorithms to conduct reach and frequency estimation from these DP summaries. The estimation thus obtained is highly accurate for two publishers, with no bias and small variance. For more than two publishers, the proposed solution applies a "sequential" technique so that the algorithm is scalable and has small variance. It is unbiased under an "independent case" which is often seen or assumed in practice, but is potentially biased in non-independent cases.

There exist a number of data summarizing algorithms (or called sketching algorithms) in the literature, such as Bloom Filters (Bloom 1970), Hyperloglog (HLL, Flajolet et al. 2007), Probabilistic Counting with Stochastic Averaging (PCSA, Flajolet and Martin 1985) and Liquid Legions (a.k.a. Exponential Bloom Filters, Wright et al. 2020). These algorithms were proposed for the use cases without privacy restrictions or with only global-DP restrictions, and in these cases they provide highly accurate results. However, their utilities are no longer preserved after inserting local noise to guarantee local-DP. Compared to them, the proposed VoC solution is significantly less susceptible to local noise and thus more accurate under local-DP restrictions. Comparison results are available at World Federation of Advertisers (2020).

The remainder of this paper is organized as follows. Section 2 defines our problem: what are the inputs, outputs and requirements in the local-DP reach and frequency estimation. A solution of the problem consists of two components: summarizing algorithm and estimation algorithm. Section 3 introduces the proposed summarizing algorithm, that is, how to summarize raw datasets as VoCs. It is explained why this summarizing algorithm satisfies DP. The estimation algorithm varies in different use cases. Section 4 presents the proposed reach estimation algorithm for two publishers, and evaluates the accuracy of this estimator. The accuracy depends on a parameter of the summarizing algorithm, that is, the length of the VoC. Section 5 shows that a fixed VoC length (4096) is always enough for high accuracy (under minor conditions). This is called out, since a fixed and short length is highly desirable in practice. Section 6 presents the proposed reach estimator for more than two publishers, and its performance is evaluated in Section 7. Then, the frequency estimation algorithm is presented in Section 8. The next two sections are add-ons of VoC. Section 9 introduces a "clipping" technique that guarantees the consistency of estimation results. Section 10 introduces a protocol called Meta-VoC, which converts a Bloom Filter to VoC. Meta VoC is appealing in practice as a linkage between global and local solutions. We close this paper with concluding remarks in Section 11. Proofs of various theoretical results are provided in the appendix.

2 Problem description: locally-differentially-private, cross-publisher reach and frequency estimation

In this section, we formally describe our problem: developing a locally-differentially-private (local-DP) solution for cross-publisher reach and frequency estimation. At high level, we aim to develop a protocol as depicted below.

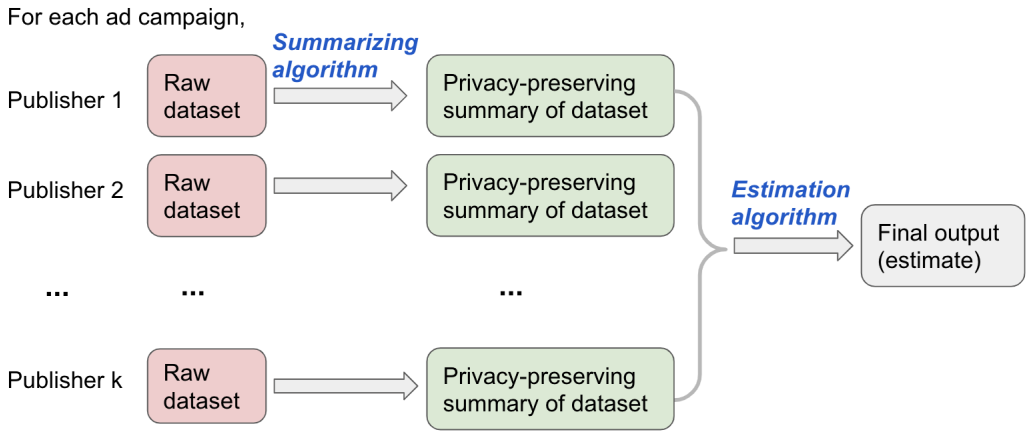

Figure 1: High level protocol of local-DP, cross-publisher reach and frequency estimation

Explicitly, each publisher has a raw dataset that carries information of the advertiser's interest. Of course, for privacy purposes, publishers cannot release the raw datasets to the advertiser. They have to summarize the dataset to meet a certain privacy standard, and release the summary. The advertiser can estimate the metrics of interest only based on the privacy-preserving summaries from each publisher.

Our objective, as shown in the above diagram, is to find proper summarizing and estimation algorithms that provide (i) solid privacy guarantee on the summary of the dataset, and meanwhile (ii) accurate final outputs that provide advertisers reliable assessment of the effectiveness of their ad campaigns.

We formulate the problem by defining each component of the above diagram. What raw dataset does each publisher have? What is the privacy requirement on the summary of the dataset? What are the desired final outputs? These will be answered in Subsections 2.1, 2.2, 2.3, respectively, for both reach and frequency estimation. Then, Subsection 2.4 illustrates the motivation of studying this local-DP protocol: what are its use cases, and its difference from the global-DP solution.

2.1 Raw dataset owned by each publisher

To describe the raw dataset that each publisher has, we first assume that the user IDs at different publishers share a common ID space. That is, the ID that each user has at publisher 1 matches the ID that they have at publisher 2. While this assumption is not 100% true in practice, it approximately holds with some data fusion techniques. One example approach is to construct the common ID space using "Virtual people"; see Skvortsov and Koehler (2019) for details.

Now, consider any specific ad campaign. Each publisher is able to observe the set of all user IDs they reached, from their log data. That is, the set of IDs that were exposed to the ad at least once at this publisher. Of course, this set is a subset of the common ID space.

From their data, each publisher is also able to observe the frequency of each reached user, i.e., the number of times each reached user is exposed to the ad. This allows each publisher to further break down the set of reached IDs into different frequency layers, namely, sets of users reached by frequency $= 1, 2, ...,$ up to a maximum frequency, say, 10.

The raw dataset of each publisher is formally described as follows.

Definition 1 (Raw dataset owned by each publisher)**.**

(1) Each publisher j owns a set $S_j = \{$users reached by publisher $j\}$. This is the raw dataset that is useful for reach estimation.

(2) Each publisher j furthers owns a break-down of S_j, which is a tuple of disjoint sets $(S_{j,1}, \cdots, S_{j,q-1}, S_{j,q+})$. Here $S_{j,t} = \{$users exposed to the ad t times at publisher $i\}$, $t = 1, \ldots, q-1$, and $S_{j,q+} = \{$users exposed to the ad q times or more, at publisher $j\}$; q is a maximum frequency of interest, say, 10. This is the raw dataset that is useful for frequency estimation.

2.2 Differential privacy

We require the summary of the dataset that each publisher releases to satisfy the ϵ-differential privacy (DP, Dwork et al. 2006). We start by introducing the general definition of ϵ-DP.

Definition 2 (General definition of ϵ-DP)**.** *Consider a random algorithm $\mathcal{A}$ that takes any dataset as input. Let $im(\mathcal{A})$ denote the image of $\mathcal{A}$. Algorithm $\mathcal{A}$ is said to provide ϵ-differential privacy, if for any two raw datasets D_1, D_2 that differ on the data of a single user, and any subset T of $im(\mathcal{A})$,*

$$e^{-\epsilon} \leq \frac{P[\mathcal{A}(D_1) \in T]}{P[\mathcal{A}(D_2) \in T]} \leq e^{\epsilon}, \tag{1}$$

where probability P is taken with respect to the randomness in algorithm $\mathcal{A}$.

We now explain this definition in the context of reach and frequency estimation. In the diagram of Figure 1, D_1 and D_2 in the above definition are two possible raw datasets in any single publisher, $\mathcal{A}$ is the summarizing algorithm, and $\mathcal{A}(D_1), \mathcal{A}(D_2)$ are privacy-preserving summaries of the two datasets. DP first requires the summary of a dataset to be random. And in view of inequality (1), DP further requires the two summaries $\mathcal{A}(D_1)$ and $\mathcal{A}(D_2)$ to have close enough probability density (or mass) functions at any point. That is, the summaries of the two datasets are hardly distinguishable in their probability distributions. The DP parameter ϵ describes how hard it is to distinguish the two summaries.

D_1 and D_2 are "two datasets that differ on the data of a single user". We clarify this following our specification of raw datasets (Definition 1). For reach estimation, a raw dataset is just a set of reached users. A change on "the data of a single user" means that this user is switched from reached to unreached, or vice versa. That is, D_1 and D_2 are two sets that differ by just one element. Explicitly, for reach estimation, a summarizing algorithm $\mathcal{A}$ has ϵ-DP if for any set S and element $x \notin S$, and any subset W of the common ID space,

$$e^{-\epsilon} \leq \frac{P[\mathcal{A}(S + \{x\}) = W]}{P[\mathcal{A}(S) = W]} \leq e^{\epsilon}.$$

For frequency estimation, a raw dataset of any publisher i is a tuple of disjoint sets $(S_{i,1}, \cdots, S_{i,q-1}, S_{i,q+})$ where $S_{i,j}$ means the set of reached users at frequency j and at publisher i. A change on "the data of a single user" means a switch in the frequency of a user. This will move one element from one set to another, among the q sets $(S_{i,1}, \cdots, S_{i,q-1}, S_{i,q+})$. A DP algorithm shall hide such move in the probability sense. This is rigorously described as follows (in a complicated manner). For frequency estimation, a summarizing algorithm $\mathcal{A}_f$ has ϵ-DP if for any integer $q \geq 2$ and any subset T of $\text{im}(\mathcal{A}_f)$,

$$e^{-\epsilon} \leq \frac{P[\mathcal{A}_f(S_1, S_2, \ldots, S_q) \in T]}{P[\mathcal{A}_f(S'_1, S'_2, \ldots, S'_q) \in T]} \leq e^{\epsilon}, \tag{2}$$

for any tuples of sets $(S_1, S_2, \ldots, S_q)$ and $(S'_1, S'_2, \ldots, S'_q)$ satisfying the conditions: (i) $S_1, S_2, \ldots, S_q$ are disjoint, and so are $S'_1, S'_2, \ldots, S'_q$; (ii) there exist $1 \leq i < j \leq n$, $i \neq j$, and element x, such that $S'_i = S_i - \{x\}$, $S'_j = S_j + \{x\}$, and $S'_k = S_k$ for $k \notin \{i, j\}$.

It seems not easy to check requirement (2) as it involves multiple sets, and actually, any number of sets. The following result simplifies our problem.

Proposition 1. *Let $\mathcal{A}$ be any $(\epsilon/2)$-DP summarizing algorithm for reach estimation. Explicitly, $e^{-\epsilon/2} \leq P[\mathcal{A}(S + \{x\}) = W]/P[\mathcal{A}(S) = W] \leq e^{\epsilon/2}$ for any set S, $x \notin S$, and subset W of the common ID space.*

For any disjoint sets $S_1, S_2, \ldots, S_q$, let $\mathcal{A}_f(S_1, S_2, \ldots, S_q) = [\mathcal{A}(S_1), \mathcal{A}(S_2), \ldots, \mathcal{A}(S_q)]$, that is, $\mathcal{A}_f$ applies algorithm $\mathcal{A}$ to each separate set. Then, $\mathcal{A}_f$ is an ϵ-DP summarizing algorithm for frequency estimation, i.e., $\mathcal{A}_f$ satisfies requirement (2).

This result is not difficult to prove (see, e.g., Vadhan 2017), following the definitions. The key here is that changing the data of a single user (only) affects a pair of counts in the frequency histogram — it shifts one count from a frequency layer to another. An $\epsilon/2$-DP for each count guarantees an ϵ-DP for the joint distribution of a pair of counts, and thus guarantees the ϵ-DP for the whole frequency histogram. With this result, it suffices to find a DP summarizing algorithm for reach estimation. The DP summarizing algorithm for frequency estimation follows from that for reach.

We briefly illustrate why we choose DP here instead of some weaker privacy criteria such as k-anonymity. We need the strict standard of DP to support "dynamic reporting". Suppose that an ad campaign reached a set S_0 of users at a publisher. Set S_0 includes the reached users in the whole universe, say, the US web population. In practice, advertisers would also like to know the reach in different slices of the universe, for example, the demographic slice "Female 18-34", or the geographic slice "users in Boston", or of course, "Female 18-34 in Boston". In dynamic reporting, advertisers would like to query the union reach for flexible slices, that is, the cardinality of $S_0 \cap G$ for flexible subset G of the universe. This requires each publisher to release the summaries $\mathcal{A}(S_0 \cap G)$ for different sets G. In this case, privacy attackers will correlate the summaries for different G and try to identify if an individual was reached. DP restricts the amount of information the attacker can gain by correlating the summaries of any two sets that differ by only one element. Explicitly, even if the attacker is able to construct queries G_1 and G_2 where G_2 includes just one more user x than G_1, the summaries $\mathcal{A}(S_0 \cap G_1)$ and $\mathcal{A}(S_0 \cap G_2)$ are hard to distinguish in the probability sense, so that the attacker cannot confidently tell if user x is reached or not.

DP describes the information gain from two query results, but how about multiple, say, 100 query results? The joint privacy of these queries can be guaranteed by combining DP with the notion of a privacy budget. As an conservative example, if algorithm $\mathcal{A}$ satisfies $(\epsilon/100)$-DP, then we can claim an ϵ-DP if advertisers are allowed to do 100 queries for each ad campaign (which allows advertisers to evaluate their reach in 100 different slices of interest). We refer readers to Dwork et al. (2014), Chapter 3 and Abadi et al. (2016) for theories and practices of privacy budget.

2.3 Outputs in reach and frequency estimation

The output of reach estimation is the deduplicated reach, or *union reach* across a set of publishers. Consider an example where an advertiser launched an ad campaign on 10 publishers. Suppose each publisher tells, from their raw data, that they reached 1 million users. In this case, the union reach is some number between 1 million and 10 million, which is a wide range. In the current practice, advertisers often have little clue where the actual union reach falls in this wide range — and thus little clue about whether their branding budget was spent effectively. They eagerly desire a reliable estimate of union reach.

The output of frequency estimation is a "histogram" that breaks down the reach estimation output, i.e., the union reach. An example is as follows.

Total frequency*	1	2	3	4	5	6	7	8	9	10+**
Reach with this total frequency	5000	5000	7000	10000	30000	20000	8000	3000	1000	500

* Total frequency indicates the total number of times that a user was exposed to the ad, across *all* the publishers.
** 10+ means 10 or more.

Frequency estimation provides advertisers additional insight beyond reach estimation. Advertisers often have an *effective frequency* in mind. Suppose that the effective frequency is 5, that is, 5 repetitions of the ad is necessary to cause the user to remember the product or other desired reactions. Then, a frequency

of 1 is not adequate; a frequency of 10 or more can be wastage or even annoying to users. In this case, an ideal frequency histogram should be concentrated around 5. If, on the contrary, the count of frequency$=1$ or 10+ were too high in the histogram, the advertiser would not be satisfied and would try to improve the frequency management strategy.

We formally describe the outputs of reach and frequency estimation as follows.

Definition 3 (Outputs of reach and frequency estimation)**.** *Consider any k publishers.*

(1) In reach estimation, we aim to estimate the union reach, or in other words, the number of users exposed to the ad campaign at least once across all the k publishers.

(2) In frequency estimation, we aim to estimate a histogram $(r_1, \ldots, r_{q-1}, r_{q+})$, where $r_t = \#\{$users exposed to the ad campaign t times across all the k publishers$\}$, $t = 1, \ldots, q-1$, and $r_{q+} = \#\{$users exposed to the ad campaign q times or more across all the k publishers$\}$; q is a maximum frequency of interest, say, 10.

Note that these outputs can be represented using the raw datasets S_t and $(S_{t,1}, \cdots, S_{t,q-1}, S_{t,q+})$ as in Definition 1. For example, the union reach is represented as $|S_1 \cup \cdots \cup S_k|$. As for frequency estimation, the histogram $(r_1, \ldots, r_{q-1}, r_{q+})$ can also be represented as a function of $(S_{t,1}, \cdots, S_{t,q-1}, S_{t,q+})$, $1 \le i \le k$ (details will be given in Section 8.1). So, if different publishers were allowed to share the raw datasets with each other, the reach and frequency estimation could be easily completed. But of course, this is not allowed under strict differential-privacy restrictions. Instead, we can only estimate the union reach and frequency histogram based on the DP summaries of each dataset. This becomes non-trivial.

Now that the reach and frequency estimation problems have been clarified, the high-level diagram of Figure 1 is specified further in the following two diagrams, for reach and frequency estimation, respectively.

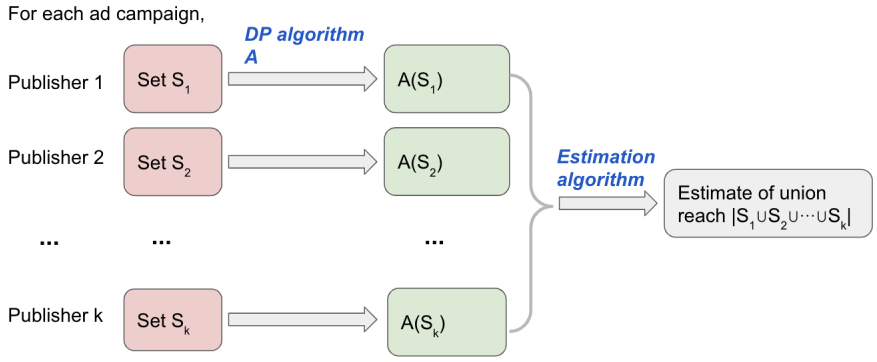

Figure 2: Specific protocol diagram for reach estimation (see Definitions 1, 2, 3 for notations)

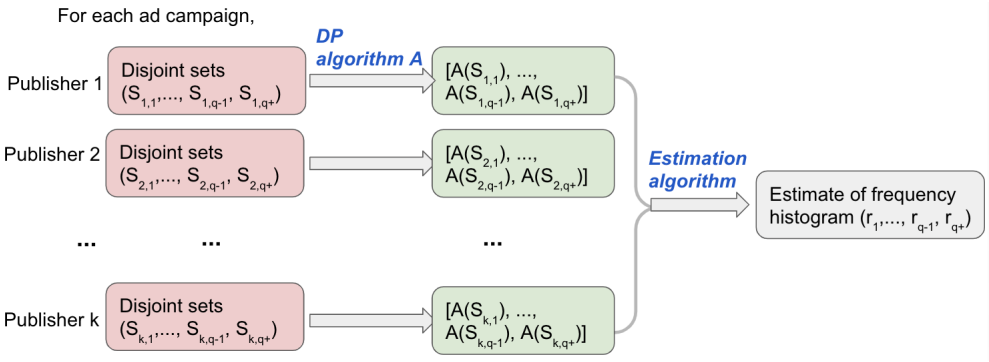

Figure 3: Specific protocol diagram for frequency estimation (see Definitions 1, 2, 3 for notations)

2.4 Local versus global DP

Before presenting the solutions, we briefly illustrate the motivation for developing the local-DP protocols in Figures 2 and 3. A local-DP solution requires each ad publisher to (locally) guarantee privacy, i.e., whatever is released from them are already DP. There is an alternative notion called global DP, of which the high-level diagram is shown below.

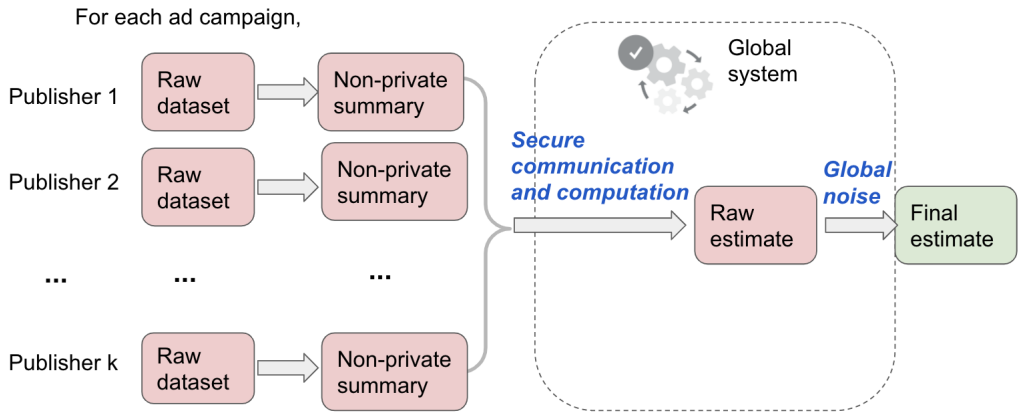

Figure 4: High-level diagram for global-DP solution

In a global solution, publishers are allowed to send non-DP information to a global system. The estimation is completed securely in a global system. The global system adds noise to the estimation results and then releases them. In Wright et al. (2020), this global system is realized using Secure Multiparty Computation (MPC). The MPC solution provides highly accurate reach and frequency estimates. However, it is not flexible. For example, suppose that an MPC has computed the estimate of union reach for publishers 1 - 10. Now, if the advertiser would like to add another publisher 11, the MPC system would have to completely rerun to compute the union reach of the 11 publishers. Likewise, if the advertiser removes publisher 10 from the list, the MPC would need to completely rerun to compute the union of the remaining 9 publishers. MPC is a complex cryptographic infrastructure, and rerunning the MPC causes substantial cost and latency.

The Local-DP solution, as studied in this paper, supports flexible reporting. In a local solution, each publisher releases privacy-preserving summaries of their raw datasets. As these summaries are public, one can send any subset of these summaries into the estimation algorithm, and obtain the union reach of the corresponding subset of users. This allows advertisers to have an interface with check-boxes of all the publishers. Advertisers can flexibly check and uncheck publishers based on their interest, and obtain the subset union reach with one click. Advertisers gain insights from the union reach of flexible subsets. For example, suppose that publishers 1 - 3 have a union reach of 1 million, and that publishers 1 - 4 also have a union reach of 1 million. It follows that publisher 4 has a zero "incremental reach" on the top of publishers 1 - 3. This result can guide the advertiser to reduce the budget on publisher 4 in the next campaign, and instead invest in the publishers with higher incremental reach.

However, there is no free lunch. While the local solution has appealing utility, it has an "uncertainty principle" on its accuracy. The paper Desfontaines et al. (2019) has a formidable title: "cardinality estimators do not preserve privacy." Here is our interpretation of their findings, in the context of reach and frequency estimation: with a large number of publishers, there does not exist a local-DP protocol that guarantees *unbiasedness* and low variance at the same time. Of course, the solution proposed in this paper does not break this uncertainty principle either. In fact, for multiple (> 2) publishers, the proposed estimation algorithm is partially model-based, which does a bias-variance trade-off. Yet for two publishers, the proposed algorithm enjoys both unbiasedness and small variance.

In a nutshell, this section defines the inputs, outputs and privacy requirement for the local-DP cross publisher reach and frequency estimation problem. In the following sections, we present (i) an ϵ-DP algorithm for summarizing any set of reached IDs, (ii) estimation algorithms that estimate the final output from the DP summaries of the inputs, for reach and frequency respectively.

3 Summarizing algorithm: Vector of Counts

We propose a differentially private algorithm, called Vector of Counts (VoC), for each publisher to summarize any set of reached IDs.

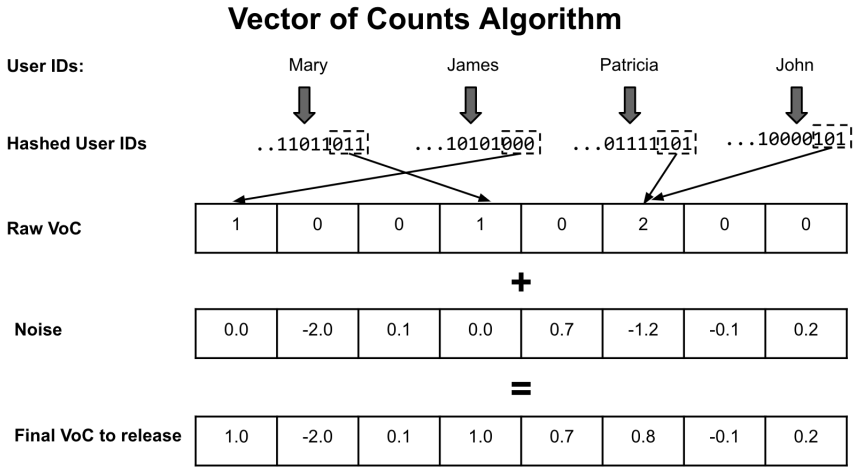

Figure 5: The VoC summarizing algorithm

Each step of the algorithm is explained as follows, using the example in the above Figure 5. Suppose that a publisher reached a set of four users $\{Mary, James, Patricia, John\}$. To summarize this set, the publisher will first hash each ID, which maps an ID to a long integer. Note that the hash function should be synchronized across different publishers. Then, take the last three digits of the binary representation of each hashed ID. This determines where the ID will fall in a vector of length $2^3 = 8$. For example, the hashed ID of James has the last three digits 000, so James is mapped to the 0th bucket of the length-8 vector (the buckets of this vector are indexed as $0, 1, \ldots, 7$). Mary has her last three digits of hashed ID being 011, which is the binary representation of 3. Subsequently, Mary is mapped to the 3rd bucket. Patricia and John both have the last three digits being 101 and are mapped to the 5th bucket. Note that the last three digits indicate the remainder of the hashed ID divided by the vector length 8.

The raw VoC is initialized as a vector of all zeros. Whenever a user is mapped to a bucket, that bucket of the raw VoC is increased by 1. As such, the raw VoC ends up counting the number of IDs mapped to each bucket — so it is called a Vector of Counts.

The raw VoC does not satisfy DP. The publisher further adds noise to each bucket of the raw VoC. To guarantee ϵ-DP, a Laplacian noise with mean 0 and scale $1/\epsilon$ is independently added to each bucket. The

noised VoC is then ready to be released. We formulate this procedure as follows.

Algorithm 1: Summarize a set as a VoC

Parameters: All publishers agree on a common VoC length m, common hash function h, and DP-parameter ϵ.

Input: Any set S of reached users. (Note that by definition, a set is already deduplicated, i.e., IDs in S are all different.)

Output: A vector of length m.

Initialization: Empty m-dimensional vector $\mathbf{c} = [0, \ldots, 0]$.

for $x \in S$ **do**

  index $\leftarrow h(x) \mod m$ // When $m = 2^s$, the "mod" operation is efficiently realized by taking the last s digits of $h(x)$

  $\mathbf{c}$[index] $\leftarrow \mathbf{c}$[index] $+ 1$

end

for index $= 1$ *to* m **do**

  $\mathbf{c}$[index] $\rightarrow \mathbf{c}$[index] $+$ Laplacian(scale $= 1/\epsilon$)

end

Return $\mathbf{c}$

In the following, we denote this algorithm as

$$\mathbf{VoC}_{m,h,\epsilon}, \text{ or shortened as } \mathbf{VoC}.$$

This algorithm is random because of the Laplacian noises added in the last step. Laplacian noises are classical approaches to guarantee DP of count data. Following Chapter 3 of Dwork et al. (2006), we have:

Proposition 2. *With any positive integer m, any hash function h and any $\epsilon > 0$, Algorithm* $\mathbf{VoC}_{m,h,\epsilon}$ *(i.e., Algorithm 1) satisfies the ϵ-DP in Definition 2.*

Now, each publisher j summarizes their reached set S_j as $\mathbf{VoC}_{m,h,\epsilon}(S_j)$ and releases it. The following Sections, 4, 5, and 6, describe how to estimate the union reach ($|S_1 \cup S_2 \cup \cdots \cup S_k|$) from the locally DP Vector of Counts ($\mathbf{VoC}(S_1), \mathbf{VoC}(S_2), \ldots, \mathbf{VoC}(S_k)$).

4 Reach estimation for two publishers

From this section, we study the estimation algorithms with VoCs as inputs. This section is devoted to the reach estimation for two publishers. The methodology for two publishers is the building brick of that for $k > 2$ publishers and also for frequency estimation. On the other hand, the two-publisher estimation result is important by itself. It shows which pairs of publishers are highly overlapped, and which pairs are reaching different corners of the population. The pairwise results are often intuitive and tell clear stories.

Consider any two publishers that reached sets S_1 and S_2 of users respectively. How to estimate $|S_1 \cup S_2|$ from $\mathbf{VoC}(S_1)$ and $\mathbf{VoC}(S_2)$? First, note that

$$|S_1 \cup S_2| = |S_1| + |S_2| - |S_1 \cap S_2|. \tag{3}$$

Now $|S_1|$ and $|S_2|$ can be estimated by the sums of their VoCs. In view of Figure 5, the sum of raw VoC exactly equals the per-publisher reach. Then, with Laplacian noises:

Proposition 3. *For any publisher j, the per-publisher reach estimate*

$$\widehat{|S_j|} = \text{sum}(\mathbf{VoC}_{m,h,\epsilon}(S_j))$$

has a mean of $|S_j|$ and a variance of $2m/\epsilon^2$. Here, sum$(\cdot)$ *indicates the sum of all elements of a vector.*

But how to estimate $|S_1 \cap S_2|$? Consider an extreme case where S_1 and S_2 are fully overlapped, i.e., the same set. Then, apparently, the raw VoCs of S_1 and S_2 are exactly the same. In other words, they have a *correlation* of 1. With Laplacian noises, $\mathbf{VoC}(S_1)$ and $\mathbf{VoC}(S_2)$ still have a correlation close to 1. On the other extreme, suppose that S_1 and S_2 are disjoint. The hash function maps each ID to different buckets approximately in a random and uniform manner (as will be explained later). This implies that the bucket indices of two different IDs are approximately independent. As such, for two disjoint sets, their VoCs are approximately uncorrelated.

In general, after fixing $|S_1|$ and $|S_2|$, as the intersection reach $|S_1 \cap S_2|$ increases from zero to full intersection, the correlation of $\mathbf{VoC}(S_1)$ and $\mathbf{VoC}(S_2)$ increases accordingly. In fact, we found an unbiased estimator of $|S_1 \cap S_2|$ that is *proportional to the covariance* of the two VoCs, defined as follows.

Definition 4 (CenteredDotProduct of two VoCs)**.** *For any* $\mathbf{VoC}_1$ *and* $\mathbf{VoC}_2$ *with the same length* m*, define*

$$\begin{aligned}\text{CenteredDotProduct}(\mathbf{VoC}_1, \mathbf{VoC}_2) &:= (\mathbf{VoC}_1 - \text{sum}(\mathbf{VoC}_1)/m) \cdot (\mathbf{VoC}_2 - \text{sum}(\mathbf{VoC}_2)/m) \\ &= \sum_{i=0}^{m-1} (\mathbf{VoC}_1[i] - \text{sum}(\mathbf{VoC}_1)/m) \cdot (\mathbf{VoC}_2[i] - \text{sum}(\mathbf{VoC}_2)/m)\,, \end{aligned} \tag{4}$$

where for any vector $\mathbf{c}$*,* $\mathbf{c}[i]$ *indicates its* i*th element. Note that* $\text{CenteredDotProduct}(\mathbf{VoC}_1, \mathbf{VoC}_2) = (m-1) \times [$*sample covariance between* $\mathbf{VoC}_1$ *and* $\mathbf{VoC}_2]$*, when we view each of the vectors* $\mathbf{VoC}_1, \mathbf{VoC}_2$ *as a sample of size* m*.*

The following figure further illustrates how the CenteredDotProduct is calculated.

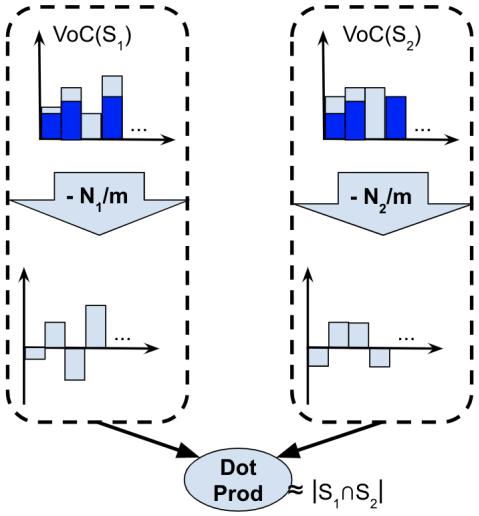

Figure 6: Estimation of intersection reach from two VoCs, using CenteredDotProduct

This turns out to give an unbiased estimator of the intersection reach:

Theorem 1 (Accuracy of two-way VoC intersection estimator)**.** *For any two sets* S_1 *and* S_2*, positive integer* m*, hash function* h *and* $\epsilon > 0$*, let*

$$\widehat{|S_1 \cap S_2|} = \text{CenteredDotProduct}(\mathbf{VoC}_{m,h,\epsilon}(S_1), \mathbf{VoC}_{m,h,\epsilon}(S_2)), \tag{5}$$

following Definition 4. Then,

$$\text{E}\left(\widehat{|S_1 \cap S_2|}\right) = |S_1 \cap S_2|$$

and

$$\mathrm{Var}\left(\widehat{|S_1 \cap S_2|}\right) = \frac{|S_1| \cdot |S_2| + |S_1 \cap S_2|^2}{m} + \frac{2(|S_1| + |S_2|)}{\epsilon^2} + \frac{4m}{\epsilon^4}. \tag{6}$$

The expectation and variance are taken over the probability space of

a. *Random Laplacian noises, and*

b. *Hash function h being viewed as a random mapping $\mathcal{H} : \{users\} \to \mathbb{Z}$ such that $(\mathcal{H}(x) \mod m)$ follows a uniform distribution on $\{0, 1, \ldots, m-1\}$.*

Here is a comment on the item b that constitutes the probability space in Theorem 1. In practice, the hash h is fixed for engineering feasibility, at least over a period, say, one day. Rigorously speaking, h is deterministic and cannot be viewed as the foregoing random mapping. Yet, we postulate that *the hash function has been designed in such a way that the hashed values closely resembled a uniform model of randomness, namely, bits of (binary representation) of hashed values are assumed to be independent and to have each probability* $1/2$ *of occurring* (Flajolet et al. 2007). Such random-uniformity assumption is commonly used for deriving the properties of classical cardinality estimators, such as the aforementioned HLL and PCSA. "Good hashes" such as SHA256 (see Section 4 of National Institute of Standards and Technology 2015) successfully resembles such random-uniformity in practice (see, e.g., Tchórzewski and Jakóbik 2019).

Now, we look back at the estimation of union reach $|S_1 \cup S_2|$. It is then estimated as

$$\begin{aligned} &\widehat{|S_1 \cup S_2|} = \widehat{|S_1|} + \widehat{|S_2|} - \widehat{|S_1 \cap S_2|} \\ =&\mathrm{sum}(\mathbf{VoC}_{m,h,\epsilon}(S_1)) + \mathrm{sum}(\mathbf{VoC}_{m,h,\epsilon}(S_2)) - \mathrm{CenteredDotProduct}(\mathbf{VoC}_{m,h,\epsilon}(S_1), \mathbf{VoC}_{m,h,\epsilon}(S_2)). \end{aligned} \tag{7}$$

The variance of $\widehat{|S_1 \cup S_2|}$ comes from the error of estimating both per-publisher reach and intersection reach, with the error for intersection reach dominating. The explicit result is as follows.

Proposition 4. *The estimator $\widehat{|S_1 \cup S_2|}$ given in (7) has*

$$\mathrm{E}\left(\widehat{|S_1 \cup S_2|}\right) = |S_1 \cup S_2|$$

and

$$\mathrm{Var}\left(\widehat{|S_1 \cup S_2|}\right) = \frac{|S_1| \cdot |S_2| + |S_1 \cap S_2|^2}{m} + \frac{2(|S_1| + |S_2| + 2m)}{\epsilon^2} + \frac{4m}{\epsilon^4}, \tag{8}$$

where the mean and variance are taken with respect to the probability space described in Theorem 1.

We take a close look at the variance formula (8). It is of the form variance $= f(m) = am + b/m + c$ for positive coefficients a, b, c. As m increases from zero to infinity, this function first decreases and then increases. There exists an optimal VoC length m that minimizes the variance. See Figure 7 below.

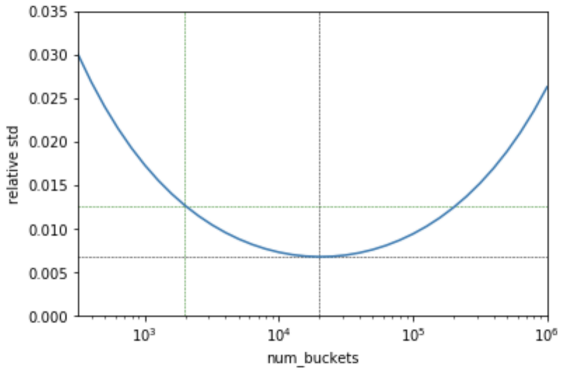

Figure 7: Relative standard deviation of $\widehat{|S_1 \cup S_2|}$ as the number of buckets increases, in an example where $|S_1| = |S_2| = 50,000$, $|S_1 \cap S_2| = 5,000$, and $\epsilon = \ln(3)$.

Here, the relative standard deviation means $\sqrt{\text{Var}(\widehat{|S_1 \cup S_2|})}$ divided by the true value of $|S_1 \cup S_2|$. The U-shape of Figure 7 is intuitively understandable. First, if we look at raw VoCs, a longer VoC carries more information. When the number of buckets is as large as the universe size, each user is roughly in their own bucket, and a raw VoC becomes a binary vector. With these long raw VoCs, reach can be precisely deduplicated across publishers. But to guarantee DP, a fixed amount of noise is needed for each bucket. The longer the VoC, the larger the amount of noise in total. As the VoC becomes very long, the total noise blurs the signal. There is thus a trade-off, and a moderate VoC length is best. This will be discussed in the next section.

5 Selection of VoC length

This section is devoted to the discussion and recommendation of the length, i.e., number of buckets of VoC. We first derive an optimal length which minimizes the variance. Such dynamic, optimal length is useful for one-off tasks. In real world cross-publisher reach estimation, we process a large number of campaigns everyday — thus, a fixed, short VoC length is needed for infrastructure capacity and stability. In Section 5.2, we show that a VoC length of about 4000 is sufficient for high accuracy, under mild conditions.

5.1 Optimal VoC length

Taking derivative of m in the variance formula (6), we have:

Proposition 5. *Given any sets S_1, S_2 and $\epsilon > 0$, the VoC variance in equation (6) attains the minimum when the number of buckets*

$$m_{\text{optimal}} = \sqrt{\frac{|S_1| \cdot |S_2| + |S_1 \cap S_2|^2}{4(\epsilon^{-2} + \epsilon^{-4})}}. \tag{9}$$

For example, when $\epsilon = \ln(3)$, $|S_1| = |S_2|$ and $|S_1 \cap S_2|$ is small compared to S_1, the optimal $m \approx |S_1|/2.5$. That is, the variance is minimized when there are on average 2-3 users in each bucket of VoC.

For one-time reach estimation tasks, two publishers can share (noised numbers of) $|S_1|$ and $|S_2|$, and insert a rough estimate of $|S_1 \cap S_2|$ into equation (9) to determine the optimal VoC length. But for frequent estimation tasks, it is infeasible to determine and vary the VoC length each time. And the optimal length for large campaigns can be millions, which is computationally expensive.

Can we choose a fixed, short VoC length and still achieve nearly optimal accuracy? Yes, as will be explained below.

5.2 Minimum set size and VoC length, given accuracy threshold

Recall that Figure 7 shows how the VoC length m affects the relative standard deviation (std) in an example with $|S_1| = |S_2| = 50,000$, $|S_1 \cap S_2| = 5,000$, and $\epsilon = \ln(3)$. (and that the relative std is with respect to the union.) The black dashed lines show that with $20,000$ buckets, the relative std reaches the minimum (0.67%). The green lines show that with just $2,000$ buckets, we can achieve a relative std that is only twice the minimum value. In fact, the relative std curve is rather flat around the optimum. We have a wide range of number of buckets that achieves nearly optimal accuracy.

How to decide what VoC length is enough, under general configurations? We introduce the following protocol based on visualization.

First, choose a target accuracy level, i.e., a threshold α of relative std. You may choose several candidates

such as $\alpha = 1\%, 2\%, 5\%$. From equation (6), the target accuracy is achieved when

$$\frac{|S_1| \cdot |S_2| + |S_1 \cap S_2|^2}{m} + \frac{2(|S_1| + |S_2| + 2m)}{\epsilon^2} + \frac{4m}{\epsilon^4} \leq \alpha^2(|S_1| + |S_2| - |S_1 \cap S_2|)^2$$

In the following, we fix $\epsilon = \ln(3)$. Theoretically, we can obtain the minimum value of m from the above inequality. But, this minimum m is a function of $|S_1|$, $|S_2|$ and $|S_1 \cap S_2|$, and the functional form is rather obscure. In practice, we cannot determine a minimum m case by case for different $|S_1|$, $|S_2|$ and $|S_1 \cap S_2|$. We need to reach an universal recommendation such as "for any campaigns with size greater than x, we recommend VoC length y, so that the relative std is within z." The plot below helps reach such a recommendation.

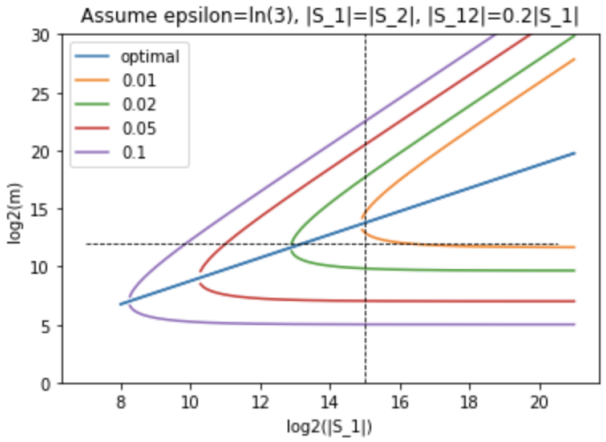

Figure 8: Plot to determine minimum length and minimum reach, for certain accuracy threshold. Here $|S_1| = |S_2|$ and $|S_1 \cap S_2| = 20\% \times |S_1|$.

This plot shows the contours of relative std, on the plane of VoC length m and reach $|S_1|$. m and $|S_1|$ are on log2 scales so the plot is compact. (Also, we prefer choosing m as a power of 2 for computational efficiency; see Figure 5.) In this plot, the two dashed lines indicate $\log_2(m) = 12$ and $\log_2(|S_1|) = 15$ respectively. It can be seen that the ray $(\log_2(m) = 12, \log_2(|S_1|) > 15)$ is along the contour of relative std= 0.01. That means, if we choose $m = 2^{12} = 4096$, then the relative std is always within 1% for any $|S_1| > 2^{15} \approx 33,000$.

The above conclusion is obtained under the condition that $|S_1| = |S_2|$ and $|S_1 \cap S_2| = 0.2|S_1|$. Does the conclusion still hold when $|S_1| \neq |S_2|$, and for other overlap rate $|S_1 \cap S_2|/|S_1|$? To see this, one can tune the ratios between $|S_1|, |S_2|, |S_2|$ and re-plot Figure 8. One can see the monotonicity of the relative std with respect to these ratios. A conclusion is:

Accuracy for low-overlap cases Set VoC length $m = 2^{12} = 4096$. Let $|S_1|, |S_2|$ be both at least $2^{15} \approx 33,000$. Then, as long as $|S_1 \cap S_2|$ is no greater than 20% of $\min(|S_1|, |S_2|)$, the two-way union estimator (12) has a relative std within 1%.

In case that the overlap rate is higher, the relative std increases, but not by much:

Accuracy for high-overlap cases Still, set VoC length $m = 2^{12} = 4096$, and let $|S_1|, |S_2|$ both be at least $2^{15} \approx 33,000$. Then for any $|S_1 \cap S_2|$, the two-way union estimator (12) has a relative std within 2.2%.

The highest relative std occurs in the full-overlap case when $|S_1 \cap S_2| = |S_1| = |S_2|$.

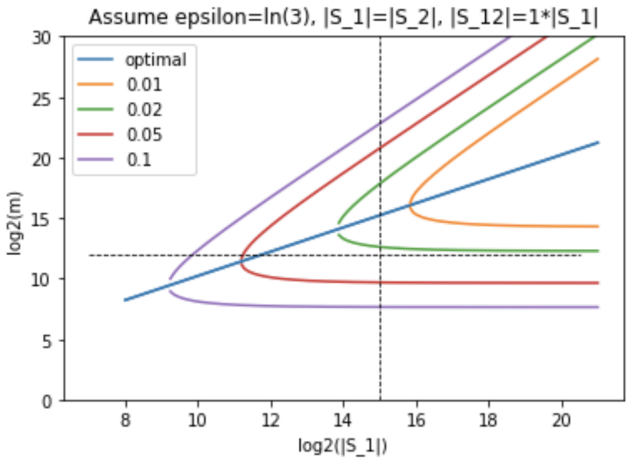

Figure 9: Re-plotting Figure 8 when $|S_1| = |S_2|$ and $|S_1 \cap S_2| = 100\% \times |S_1|$.

In summary, we recommend choosing VoC length as $2^{12} = 4096$. This length will be used throughout the following sections. For two publisher VoC, this length gives a relative std within 1%, under mild conditions.

6 Reach estimation for more than two publishers

In practice, advertisers often deliver ads to multiple publishers, with the hope to reach every corner of the population. Reach estimation becomes more difficult for more than two publishers. As mentioned in Subsection 2.4, there is a barrier of accuracy for any local-DP solution when estimating the union reach of a large number of publishers. A bias-variance trade-off is needed. In this section, we introduce two approaches that extends the two-way reach estimation algorithm to more than two publishers:

Dot products + inclusion-exclusion. This is a natural extension of two-way VoC. This approach is unbiased, but suffers from large variance for five or more publishers. It is unscalable in terms of computation. This approach is introduced in Subsection 6.1.

Sequential VoC (recommended approach). This is a partially model based approach which approximates the multiple-publisher union reach using a series of two-way VoC estimation. With modeling, it significantly reduces variance. The variance does not propagate as the number of publishers increases. In turn, this approach does not guarantee unbiasedness. It is scalable in terms of computation. This approach is introduced in Subsection 6.2.

Overall, we recommend the Sequential VoC approach, for its scalability and small variance. The Dot products + inclusion-exclusion approach is still introduced here, as it is a natural extension of the two-way technique and potentially useful for checking the bias of Sequential VoC in practice (as will be briefly discussed in Section 7).

6.1 Estimation using dot products and inclusion-exclusion formula

In the two-way VoC of Section 4, we estimated the union reach using the formula $|S_1 \cup S_2| = |S_1| + |S_2| - |S_1 \cap S_2|$. For $k > 2$ publishers, a natural idea is to use the general include-exclusion formula

$$\begin{aligned} |S_1 \cup \cdots \cup S_k| = &\sum_{j=1}^{k} |S_j| - \sum_{1 \le j_1 < j_2 \le k} |S_{j_1} \cap S_{j_2}| + \cdots \\ &+ (-1)^k \sum_{1 \le j_1 < \cdots < j_{k-1} \le k} |S_{j_1} \cap \cdots \cap S_{j_{k-1}}| + (-1)^{k+1} |S_1 \cap \cdots \cap S_k|. \end{aligned} \tag{10}$$

Then, how to estimate each intersection term? As a straightforward extension of the two-way CenteredDotProduct (Definition 4), the three-way CenteredDotProduct

$$\begin{aligned} &\widehat{|S_1 \cap S_2 \cap S_3|} \\ &= \sum_{i=0}^{m-1} \{\mathbf{VoC}_1[i] - \mathrm{sum}(\mathbf{VoC}_1)/m\} \{\mathbf{VoC}_2[i] - \mathrm{sum}(\mathbf{VoC}_2)/m\} \{\mathbf{VoC}_3[i] - \mathrm{sum}(\mathbf{VoC}_3)/m\} \end{aligned} \tag{11}$$

turns out to be an unbiased estimator of the three-way intersection, where $\mathbf{VoC}_j = \mathbf{VoC}(S_j)$, $j = 1, 2, 3$. See Appendix A for more details. Then, the three-way union can be estimated as $\widehat{|S_1 \cup S_2 \cup S_3|} = \widehat{|S_1|} + \widehat{|S_2|} + \widehat{|S_3|} - \widehat{|S_1 \cap S_2|} - \widehat{|S_1 \cap S_3|} - \widehat{|S_2 \cap S_3|} + \widehat{|S_1 \cap S_2 \cap S_3|}$. However, this extension is generally infeasible for larger k, especially for up to $k = 100$ publishers in the real world. The reasons are summarized as follows.

- Concern on the scalability
  - The inclusion-exclusion formula (10) involves $2^k - 1$ terms, which is computationally infeasible for $k \ge 20$.
  - When $k \ge 4$, the k-way intersection can no longer be estimated *directly* estimated by a dot product like in (11). It can be estimated by a *combination* of various dot-products, and the formula gets complicated. We have derived the estimators for $k = 4, 5, 6$-way intersections (details are omitted here). The number of terms involved in the k-way intersection estimator appears to grow exponentially as well, which is unscalable.
- Concerns on the variance. The variance is acceptable for $k = 3$ publishers, with slightly longer VoC than the two-way case, say, with $m = 2^{14} = 16392$. As k increases, the variance quickly increases, and becomes certainly unacceptable for $k > 5$ publishers.

In the following, we propose an alternative approach that resolves the above concerns.

6.2 Sequential VoC

We propose a scalable approach called Sequential VoC. It is called sequential, as it consists of a sequence of two-way VoC estimation. See the flowchart below.

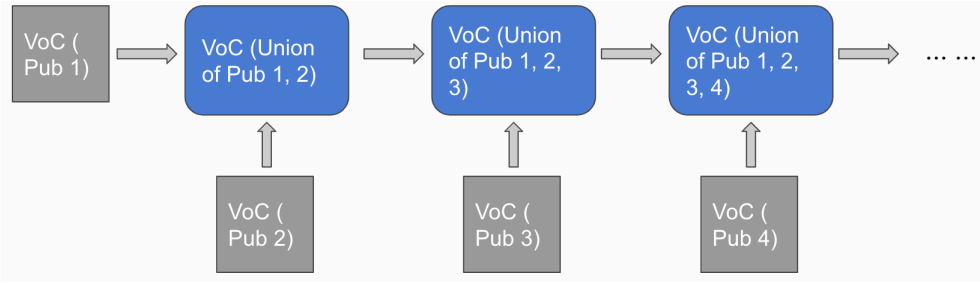

Figure 10: Flowchart of Sequential VoC

In the above plot, "Pub" stands for publisher. The grey boxes are real VoCs from each publisher, and the blue boxes are approximated VoCs.

How is VoC(Union of Pub 1,2) approximated? It is approximated by

$$[\text{VoC(Pub 1)} + \text{VoC(Pub 2)}] \times \left(1 - \frac{\widehat{|S_1 \cap S_2|}}{\widehat{|S_1|} + \widehat{|S_2|}}\right), \tag{12}$$

where $\widehat{|S_1 \cap S_2|}$ is the intersection estimate given by (4), and $\widehat{|S_1|}, \widehat{|S_2|}$ are estimates of per-publisher reach, given by the sum of VoCs. This approximation assumes that the overlap rates in different buckets are homogeneous — and hence equal the total overlap rate, i.e., $\widehat{|S_1 \cap S_2|}/(\widehat{|S_1|} + \widehat{|S_2|})$.

With Voc(Union of Pub 1, 2) obtained, we view the union of publishers 1 and 2 as a single pseudo publisher and Voc(Union of Pub 1, 2) as its VoC. Then VoC(Union of Pub 1, 2, 3) can be approximated by further deduplicating this pseudo publisher with publisher 3. So on so forth, for any number of publishers, we obtain a VoC representing all the IDs reached across these publishers. The sum of this VoC provides an estimate of union reach. The algorithm is formally described as follows.

Algorithm 2: Sequential VoC

Input: $\mathbf{VoC}(S_j)$, for $j = 1, \ldots, k$. Each VoC has length m.
Output: Estimate of union reach $|S_1 \cup S_2 \cup \cdots \cup S_k|$.
Initialization: Empty m-dimensional vector $\mathbf{c} := [0, 0, \ldots, 0]$.
for $j = 1$ *to* k **do**
  Obtain intersection estimate $\hat{n}_{\text{intersect}} = \text{CenteredDotProduct}(\mathbf{c}, \mathbf{VoC}(S_j))$;
  Update

$$\mathbf{c} \leftarrow (\mathbf{c} + \mathbf{VoC}(S_j)) \times \left(1 - \frac{\hat{n}_{\text{intersect}}}{\text{sum}(\mathbf{c}) + \text{sum}(\mathbf{VoC}(S_j))}\right).$$

end
Return union reach estimate as $\text{sum}(\mathbf{c})$.

Remark 1. *In Sequential VoC, the order of publishers matters. That is, shuffling the indices* $(1, 2, \ldots, k)$ *will affect the output of Algorithm 2. In practice, we can estimate the union reach using a number of (say, 5) different orders. If the results from these orders are close enough (say, within* 5% *range), we can take the average as the final estimate. If, however, different orders give rather inconsistent results, we shall conclude that the model behind Sequential VoC is not suitable for this particular ad campaign. In this case, advertisers may switch to the precise but more computationally expensive global solution.*

7 Some properties of Sequential VoC

In this section we claim that Sequential VoC has the following properties: (1) Scalability (2) Unbiasedness under an independence assumption (3) Biasedness in some cases (4) Small and stable relative standard deviation.

First, scalability. For reach estimation of k publishers, the flowchart of Figure 10 clearly has $O(k)$ steps, and thus requires $O(k)$ times of VoC dot products and additions. Subsequently, Sequential VoC has linear computational complexity.

As for the accuracy, we first theoretically describe the properties, and then show some empirical results at the end of this section. We start with the bias. Sequential VoC is an approximate estimator and generally has bias. Nevertheless, it is unbiased in the following special cases.

Proposition 6. *Sequential VoC is unbiased for (but not only for)*

1. *Disjoint campaigns, i.e., when the reached set $S_1, S_2, \ldots, S_k$ are disjoint, where S_j again represents the set of reached users at publisher j;*
2. *Fully overlapped campaigns, i.e., when $S_1 = S_2 = \cdots = S_k$;*
3. *Independent campaigns, where within a targeted universe with size U,*

$$\frac{|S_{j_1} \cap S_{j_2} \cap \cdots \cap S_{j_r}|}{U} = \frac{|S_{j_1}|}{U} \times \frac{|S_{j_2}|}{U} \times \cdots \times \frac{|S_{j_r}|}{U} \tag{13}$$

for any subset $\{j_1, j_2, \ldots, j_r\} \subset \{1, 2, \ldots, k\}$.

Note that the third case, independent campaigns, has been used as a model of reach deduplication for decades. It is also known as the Sainsbury formula (Caffyn and Sagovsky 1963). It assumes no correlation between the exposures to one publisher and another. In past practices of cross-media reach estimation, the conventional use of independence model requires specifying a universe size, i.e., U in equation (13). If the universe size were not correctly specified, the result of the independence model would be misleading. Sequential VoC provides an alternative, parameter-free approach — it does not require specifying a universe size. Instead, it automatically detects the correct universe size from the pairwise intersection estimates and applies the independence model in a scalable manner. In this sense, Sequential VoC serves as a good replacement of the conventional independence model.

When different publishers are not independent, Sequential VoC usually has bias. How large is the bias? We have a theoretical result for three publishers.

Proposition 7. *For $k = 3$, the union estimate $|S_1 \cup \widehat{S_2 \cup S_3}|$ given by Sequential VoC has bias*

$$\mathrm{E}(|S_1 \cup \widehat{S_2 \cup S_3}|) - |S_1 \cup S_2 \cup S_3| \approx n_{12} \times \frac{n_{13} + n_{23}}{n_1 + n_2} - n_{123},$$

where each $n_i = |S_i|$, the per-publisher reach; $n_{ij} = |S_i \cap S_j|$, the two-way intersection reach; $n_{ijk} = |S_i \cap S_j \cap S_k|$, the three-way intersection reach. Here the Sequential VoC uses the natural merging order of publishers $1 \to 2 \to 3$.

Note that under the independence model, it is easy to see that $n_{123} = n_{12}(n_{13}+n_{23})/(n_1+n_2)$, and then Sequential VoC is unbiased. In the non-independent case, the ratio between n_{123} and $n_{12}(n_{13}+n_{23})/(n_1+n_2)$ is an indicator of the correlation. The union estimate from Sequential VoC has positive bias when this ratio is smaller than 1, and negative bias when the ratio is greater than 1. In practice, this ratio can be approximated, with n_{123} estimated from the three-way VoC (Subsection 6.1) and each n_{ij} estimated from two-way VoC. Empirical study shows that the ratio thus approximated has reasonable accuracy with long enough VoC

($2^{14} = 16392$ buckets). Then, with $k \geq 3$ publishers, one possible approach to assess the bias is to examine this ratio for every triplet of publishers. If the ratios substantially deviate from 1 for most triplets, then we conclude that Sequential VoC can have significant bias for this campaign. On the other hand, if the ratios are close to 1 for most triplets, we would have confidence (but without a 100% guarantee) that Sequential VoC has small bias. This is just one idea of assessing the Sequential VoC bias, and a future work is to detail and evaluate it.

Sequential VoC does have a bias-variance trade-off. As discussed in Section 2.4, following the spirit of Desfontaines et al. (2019), any local-DP cardinality estimator has a limit in accuracy. Any unbiased local-DP solution will have variance propagating as the number of publishers increases. Sequential VoC guarantees a stable and small variance, at a cost of potential bias. We provide a proof of the stable variance in the Appendix, for the special case of disjoint campaigns. For other cases, we believe that it can be proved likewise with heavier mathematics. Here is a rough explanation of this stability. Sequential VoC does just $O(k)$ steps of two-way estimations with k publishers (see Figure 10), and thus the standard deviation (std) of Sequential VoC typically increases at an order of $O(\sqrt{k})$. On the other hand, the true union reach typically increases at a similar speed, so that the relative std (std divided by the true union) is at a constant magnitude.

We close this section by presenting simulation results under two typical scenarios. It is shown that Sequential VoC has stable variance, no bias in an independent case, and significant bias in a heavily correlated case.

Example 1. *This simulation scenario is set up based on a user-level mechanism, in which we control the similarity of user activities between different publishers. We believe that this setup provides practical insights, and will re-use this setup in the simulation of frequency estimation (Subsection 8.3).*

Consider a universe with $U = 2 \times 10^6$ users. At each publisher, different users have different activity levels, and thus have different probabilities of being exposed to an ad. For any ad impression (i.e., one exposure of the ad) at any publisher j, suppose that each user u has probability $p_{u,j}$ to receive this impression, $u = 1, 2, \ldots, U$. Often, $p_{u,j}$ is closely associated with user u's online time at this publisher. In the simulation, we assume that $p_{u,j}$ decays exponentially from the most to the least active users. That is, if users are ranked from the most to the least active, then each $p_{u,j}$ is proportional to $\exp(-au/U)$, a being a decay rate. We set $a = 5$ here, which means that the most active user has $\exp(5) \approx 150$ times chance than the least active user to receive an ad impression.

Now, consider two scenarios of how the user activities are correlated between different publishers:

Scenario A, independent activity. *The activity of a user at one publisher is independent with their activity at other publishers. This is realized in the following procedure.*

$U \leftarrow 2 \times 10^6$; $a \leftarrow 5$;
for $j = 1$ *to* k **do**
  Shuffle the indices of users, i.e., generate a random permutation $\pi : \{1, \ldots, U\} \to \{1, \ldots, U\}$;
  for $u = 1$ *to* U **do**
    $p_{u,j} \leftarrow \exp(-a \cdot \pi(u)/U)$;
    $p_{u,j} \leftarrow p_{u,j} / \sum_{u=1}^{U} p_{u,j}$;
  end
end

Scenario B, identical activity. *In this scenario, we do not shuffle the indices of users:*

$U \leftarrow 2 \times 10^6$; $a \leftarrow 5$;
for $j = 1$ *to* k **do**
  for $u = 1$ *to* U **do**
    $p_{u,j} \leftarrow \exp(-au/U)$;
    $p_{u,j} \leftarrow p_{u,j} / \sum_{u=1}^{U} p_{u,j}$;
  end
end

In this way, the most active user at publisher 1 is also the most active at publisher 2, and similarly for the least active user.

With $p_{u,j}$ generated for each user u and publisher j, we simulate the reach by randomly assigning a certain amount of impressions according to $p_{u,j}$. Suppose that the advertiser launches the campaign on 20 publishers and delivers 2×10^5 impressions on each publisher. The data generation procedure is:

NumImpressions $\leftarrow 2 \times 10^5$; $k \leftarrow 20$;
Initialization: $S_j \leftarrow \varnothing$, $1 \le j \le k$
for $j = 1$ *to* k **do**
  for $v = 1$ *to* NumImpressions **do**
    Randomly pick one user u according to the probability $p_{u,j}$, $1 \le u \le U$;
    if $u \notin S_j$ **then**
      Add u to S_j;
    end
  end
end
Return $(S_j)_{1 \le j \le k}$

This generates the raw dataset of each publisher. We then summarize each S_j with VoC and estimate the union reaches from these VoCs. The relative error, i.e., (estimate - truth) / truth is assessed as the metric of accuracy. We repeated the experiment 50 times and obtained the following distributions of relative errors.

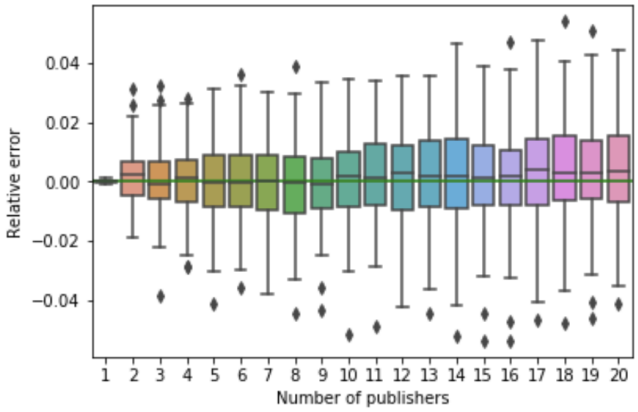

Figure 11: Performance of Sequential VoC under the above simulation scenario A (colors are used only to differentiate the different boxplots; they do not indicate another variable)

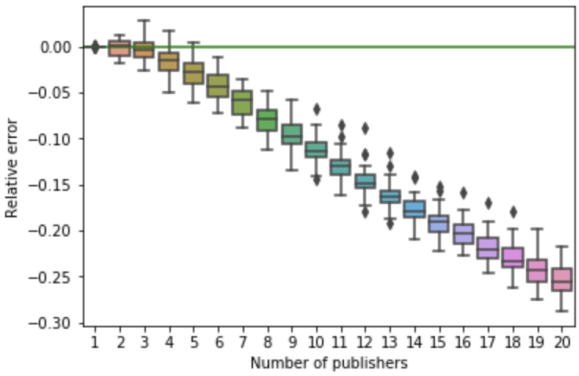

Figure 12: Performance of Sequential VoC under the above simulation scenario B (colors are used only to differentiate the different boxplots; they do not indicate another variable)

Figures 11 and 12 show the boxplots of relative errors of Sequential VoC, for estimating the union reach of 1 - 20 publishers, under scenarios A and B respectively. Under the independent scenario A, Sequential VoC gives unbiased estimates of the union reach. And the relative std, indicated by the width of boxes, is quite stable: for 1 - 20 publishers, all relative errors are within $\pm 5\%$*. Under the correlated scenario B, Sequential VoC underestimates the union reach. The bias is within* 5% *for* $k \leq 5$ *publishers. It goes up to* 10% *for* 10 *publishers, and* 25% *for* 20 *publishers. And the relative std is still stable and small.*

8 Stratified VoC for frequency estimation

In this section, we study the frequency estimation problem. Recall that for frequency estimation, the raw dataset from each publisher is a tuple of disjoint sets $(S_{j,1}, \ldots, S_{j,q-1}, S_{j,q+})$. We aim to find DP summaries of these raw datasets, and a way to estimate the desired output $(r_1, \ldots, r_{q-1}, r_{q+})$ from the DP summaries, where each r_t is the number of users reached with a total frequency of t across all publishers. See Section 2 for detailed description of the problem.

The VoC techniques introduced in previous sections have paved the way for an approach of frequency estimation. We summarize each single set using VoC. Then, the tuple of sets $(S_{j,1}, \ldots, S_{j,q-1}, S_{j,q+})$ is summarized as a tuple of VoCs. Explicitly,

$$\left(\mathbf{VoC}_{m,h,\epsilon/2}(S_{j,1}), \ldots, \mathbf{VoC}_{m,h,\epsilon/2}(S_{j,q-1}), \mathbf{VoC}_{m,h,\epsilon/2}(S_{j,q+})\right) \tag{14}$$

is the summary to be released from each publisher j. Note that each single VoC in this tuple has $(\epsilon/2)$-DP, which is achieved by using Laplacian errors with scale $2/\epsilon$. Then, by Proposition 1, the tuple of VoCs satisfies ϵ-DP. We call this summarizing algorithm as Stratified VoC, and the summary in equation (14) as a stratified VoC tuple.

Given stratified VoC tuples from all publishers, how to estimate the desired $(r_1, \ldots, r_{q-1}, r_{q+})$? Like in reach, we estimate this in a sequential and thus scalable procedure as shown in Figure 13.

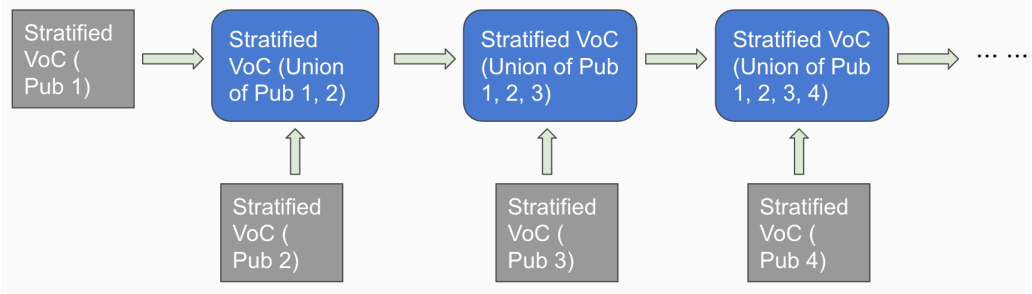

Figure 13: Flowchart of Sequential Stratified VoC for frequency estimation

It then suffices to have an approach of estimating $(r_1, \ldots, r_{q-1}, r_{q+})$ for two publishers. We will show that $(r_1, \ldots, r_{q-1}, r_{q+})$ can be represented as a function of the raw datasets of the two publishers, using a series of set operations. We approximate these set operations with certain "VoC operations", and then $(r_1, \ldots, r_{q-1}, r_{q+})$ can be estimated. These ideas will be illustrated in Subsection 8.1. Then, the estimate algorithm is formally described in Subsection 8.2, and its performance is assessed in Subsection 8.3.

8.1 Basic ideas

This subsection illustrates that for two publishers, the desired output $(r_1, \ldots, r_{q-1}, r_{q+})$ can be represented using set operations of the raw datasets $(S_{j,1}, \ldots, S_{j,q-1}, S_{j,q+})$, $j = 1, 2$. The proposed frequency estimation can be derived from there.

Consider $q = 3$ for instance. We have $r_1 = |R_1|$, $R_1 = \{\text{users with total frequency } 1\}$, and that

$$\begin{aligned} R_1 &= \{\text{users with } f_1 = 1 \text{ and } f_2 = 0\} + \{\text{users with } f_1 = 0 \text{ and } f_2 = 1\} \\ &= (S_{1,1} \cap S_{2,0}) + (S_{1,0} \cap S_{2,1}) \\ &= [S_{1,1} - (S_{2,1} + S_{2,2} + S_{2,3+})] + [S_{2,1} - (S_{1,1} + S_{1,2} + S_{1,3+})]. \end{aligned}$$

Here, "+" indicates the disjoint union, i.e., union of disjoint sets, and f_j indicates the frequency at publisher j, $j = 1, 2$. The last equality above is explained as follows. $S_{2,0}$ is the set of users who have frequency=0, i.e., who have not been reached at publisher 2. $S_{2,0}$ itself is not obtainable by the publisher, because the publisher does not know all the users in the universe (they only know the users they have ever reached). The trick here is to replace $S_{2,0}$ by the complement of $S_{2,1+} = S_{2,1} + S_{2,2} + S_{2,3+}$. In this way, r_1 is represented as an explicit function of the raw datasets $(S_{j,1}, S_{j,2}, S_{j,3+})$, $j = 1, 2$.

Likewise, $r_2 = |R_2|$, $R_2 = \{\text{users with total frequency } 2\}$, and

$$\begin{aligned} R_2 &= (S_{1,2} \cap S_{2,0}) + (S_{1,1} \cap S_{2,1}) + (S_{1,0} \cap S_{2,2}) \\ &= [S_{1,2} - (S_{2,1} + S_{2,2} + S_{2,3+})] + (S_{1,1} \cap S_{2,1}) + [S_{2,2} - (S_{1,1} + S_{1,2} + S_{1,3+})]. \end{aligned} \tag{15}$$

$r_3 = |R_3|$, $R_3 = \{\text{users with total frequency 3 or more}\}$, and

$$R_3 = (S_{1,1+} \cup S_{2,1+}) - R_1 - R_2,$$

with $S_{j,1+} = S_{j,1} + S_{j,2} + S_{j,3+}$, $j = 1, 2$. In summary, in frequency estimation, the output (r_1, r_2, r_{3+}) can be expressed using the raw datasets with the following set operations (between two sets): (i) disjoint union + (ii) non-disjoint union $\cup$ (iii) intersection $\cap$ (iv) set difference $-$.

These set operations can all be approximated with the operations on VoCs. First, disjoint union: for any two disjoint sets A, B, $\mathbf{VoC}(A + B) = \mathbf{VoC}(A) + \mathbf{VoC}(B)$. This is exact for raw VoCs and approximately true for noised VoCs. Second, non-disjoint union: $\mathbf{VoC}(A \cup B)$ can be approximated with equation (12). The intersection and set difference operations can be approximated likewise, as will be formulated below.

8.2 Formal description of the estimation algorithm

We define the following operations between any two VoCs.

Definition 5. *Given any* $\mathbf{VoC}_1 = \mathbf{VoC}(S_1)$ *and* $\mathbf{VoC}_2 = \mathbf{VoC}(S_2)$ *of the same length* m*, we define three operations*

$$\textit{VoC union } \sqcup, \textit{ VoC intersection } \sqcap, \textit{ VoC difference } \ominus$$

that map $(\mathbf{VoC}_1, \mathbf{VoC}_2)$ *to another VoC of length* m*:*

(1)

$$\textit{Intersection } \mathbf{VoC}_1 \sqcap \mathbf{VoC}_2 := (\mathbf{VoC}_1 + \mathbf{VoC}_2) \times \frac{\text{CenteredDotProduct}(\mathbf{VoC}_1, \mathbf{VoC}_2)}{\text{sum}(\mathbf{VoC}_1) + \text{sum}(\mathbf{VoC}_2)}.$$

This approximates the VoC of the intersection set $S_1 \cap S_2$*.*

(2) Union $\mathbf{VoC}_1 \sqcup \mathbf{VoC}_2 := \mathbf{VoC}_1 + \mathbf{VoC}_2 - \mathbf{VoC}_1 \sqcap \mathbf{VoC}_2$*. This approximates the VoC of the union set* $S_1 \cup S_2$*.*

(3) Difference $\mathbf{VoC}_1 \ominus \mathbf{VoC}_2 := \mathbf{VoC}_1 - \mathbf{VoC}_1 \sqcap \mathbf{VoC}_2$*. This approximates the VoC of the set difference* $S_1 - S_2$*.*

Of course, another natural operation between two VoCs is the regular vector addition $+$. When S_1 and S_2 are disjoint, $\mathbf{VoC}(S_1) + \mathbf{VoC}(S_2)$ represents the VoC of the disjoint union $S_1 + S_2$. With these operations, we can merge the stratified VoC tuples from any two publishers as below.

Algorithm 3: Merge two stratified VoC tuples

Input: Two stratified VoC tuples $(\mathbf{VoC}(S_{1,1}), \ldots, \mathbf{VoC}(S_{1,q-1}), \mathbf{VoC}(S_{1,q+}))$ and $(\mathbf{VoC}(S_{2,1}), \ldots, \mathbf{VoC}(S_{2,q-1}), \mathbf{VoC}(S_{2,q+}))$, both having the same VoC length m and maximum frequency q.

Output: A merged stratified VoC tuple, $(\mathbf{c}_1, \ldots, \mathbf{c}_{q-1}, \mathbf{c}_{q+})$.

Initialization: Empty m-dimensional vector $\mathbf{c}_t = [0, \ldots, 0]$, for all $t = 1, ..., q-1$ and $t = q+$.

Obtain $\widehat{\mathbf{VoC}}(S_{1,1+}) \leftarrow \mathbf{VoC}(S_{1,1}) + \cdots + \mathbf{VoC}(S_{1,q-1}) + \mathbf{VoC}(S_{1,q+})$ and $\widehat{\mathbf{VoC}}(S_{2,1+}) \leftarrow \mathbf{VoC}(S_{2,1}) + \cdots + \mathbf{VoC}(S_{2,q-1}) + \mathbf{VoC}(S_{2,q+})$;

for $t = 1$ *to* $q-1$ **do**

  for $r = 1$ *to* $t-1$ **do**

    //Skip this loop if $t = 1$

    $\mathbf{c}_t \leftarrow \mathbf{c}_t + \mathbf{VoC}(S_{1,r}) \sqcap \mathbf{VoC}(S_{2,t-r})$;

  end

  $\mathbf{c}_t \leftarrow \mathbf{c}_t + \mathbf{VoC}(S_{1,t}) \ominus \widehat{\mathbf{VoC}}(S_{2,1+})$;

  $\mathbf{c}_t \leftarrow \mathbf{c}_t + \mathbf{VoC}(S_{2,t}) \ominus \widehat{\mathbf{VoC}}(S_{1,1+})$;

end

$\mathbf{c}_{q+} \leftarrow \widehat{\mathbf{VoC}}(S_{1,1+}) \sqcup \widehat{\mathbf{VoC}}(S_{2,1+}) - \mathbf{c}_1 - \mathbf{c}_2 - \cdots - \mathbf{c}_{q-1}$;

if $\text{sum}(\mathbf{c}_{q+}) < 0$ **then**

  $\mathbf{c}_{q+} \leftarrow [0, \ldots, 0]$;

end

Return $(\mathbf{c}_1, \ldots, \mathbf{c}_{q-1}, \mathbf{c}_{q+})$

The reach at total frequency t, i.e., r_t can then be estimated as $\text{sum}(\mathbf{c}_t)$, for $t = 1, \ldots, q-1$ and $q+$. For $k \geq 3$ publishers, the frequency estimation can be conducted in the following sequential procedure.

Algorithm 4: Merge $k \geq 3$ stratified VoC tuples

Input: k stratified VoC tuples $(\mathbf{VoC}(S_{j,1}), \ldots, \mathbf{VoC}(S_{j,q-1}), \mathbf{VoC}(S_{j,q+}))$, $1 \leq j \leq k$, each having VoC length m and maximum frequency q.

Output: A merged stratified VoC tuple $(\mathbf{c}_1, \ldots, \mathbf{c}_{q-1}, \mathbf{c}_{q+})$.

Initialization: Empty m-dimensional vector $\mathbf{c}_j = [0, \ldots, 0]$, for all $j = 1, ..., q-1$ and $j = q+$;

for $j = 1$ *to* k **do**

  $(\mathbf{c}_1, \ldots, \mathbf{c}_{q-1}, \mathbf{c}_{q+}) \leftarrow \text{Merge}[(\mathbf{c}_1, \ldots, \mathbf{c}_{q-1}, \mathbf{c}_{q+}), (\mathbf{VoC}(S_{j,1}), \ldots, \mathbf{VoC}(S_{j,q-1}), \mathbf{VoC}(S_{j,q+}))]$ using Algorithm 3;

end

Return $(\mathbf{c}_1, \ldots, \mathbf{c}_{q-1}, \mathbf{c}_{q+})$

Again, the final output r_t can then be estimated as $\text{sum}(\mathbf{c}_t)$, for $t = 1, \ldots, q-1$ and $q+$.

8.3 Some properties of the estimation algorithm

We evaluate the performance of the proposed frequency estimation algorithm following the reach simulation scenarios in Example 1.

Example 2. *We generate raw datasets following the scenarios A (independent case) and B (highly correlated case) in Example 1. We generate the stratified VoC tuples from these datasets and estimate the histogram $(r_1, \ldots, r_{q-1}, r_{q+})$ using Algorithm 4, for 50 replicates. The results for $k = 10$ publishers and $q = 10$ maximum frequency are summarized as follows.*

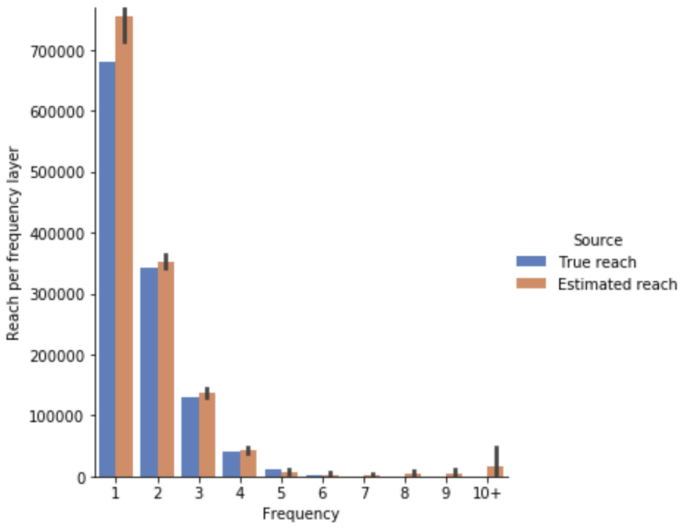

Figure 14: Performance of frequency estimation when $k = 10$ and $q = 10$, under scenario A

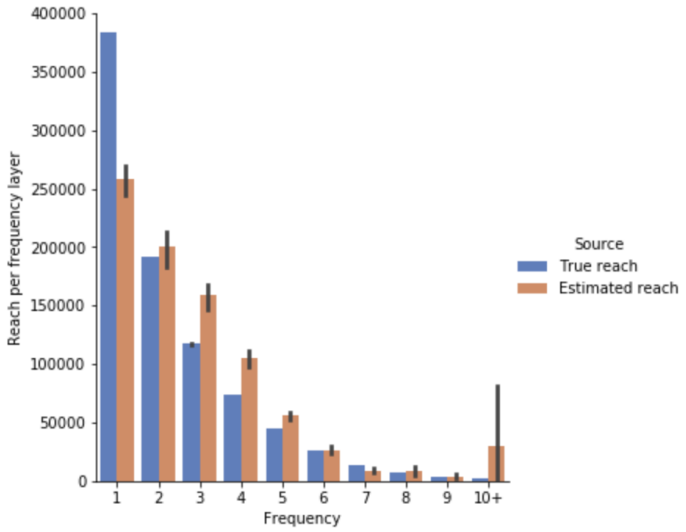

Figure 15: Performance of frequency estimation when $k = 10$ and $q = 10$, under scenario B

The plots are read in the following way. The blue bars indicate the true reach, and the orange bars indicate the estimated reach, at different layers of total frequency. For example, in Figure 15, at the frequency layer 1, the true reach is about 380,0000, while the estimated reach is only around 250,000. The whisker on the orange bar indicates the variation in the estimated results, for different replicates. As such, Figure 15 shows that the proposed algorithm has bias in the highly correlated case while preserving small variance (except for the last frequency layer), under simulation scenario B. Figure 14 shows that the proposed algorithm has little bias and still small variance, under scenario A.

As a summary of the above example, the Stratified VoC frequency estimation algorithm has similar properties as the Sequential VoC reach estimation algorithm: both are scalable, of small variance, (roughly) unbiased for independent cases, and biased for highly correlated cases. However, compared to Sequential VoC, Stratified VoC has larger variance, because q times more terms are involved in frequency estimation. Explicitly, Sequential VoC involves $O(k)$ steps of estimation (by CenteredDotProduct), while Stratified VoC (Algorithm 2) involves $O(k \times q)$ steps of estimation. As q increases, the variance propagates. As such, the algorithm has poor performance for $q = 20$ or more. The estimates for high frequency layers (such as the $q+$ reach) are especially susceptible to the inflation of variance. In practice, smoothing the estimates at different frequency layers may lead to more robust results.

9 Clipped VoC

Through Sections 3 to 8 we have presented a complete solution based on VoC to the problems defined in Section 2. This section introduces an add-on to this solution. We will illustrate the add-on methodology step-by-step, and give a summary at the end of this section.

The whole VoC solution relies on the two-way intersection estimator given by the CenteredDotProduct (5). The estimate thus obtained is dispersed about the true intersection. When the true intersection is small, the estimate can be negative. See the example below under the configuration that $|S_1| = |S_2| = 10^5$, $|S_1 \cap S_2| = 2000$, $m = 4096$ and $\epsilon = \ln(3)$.

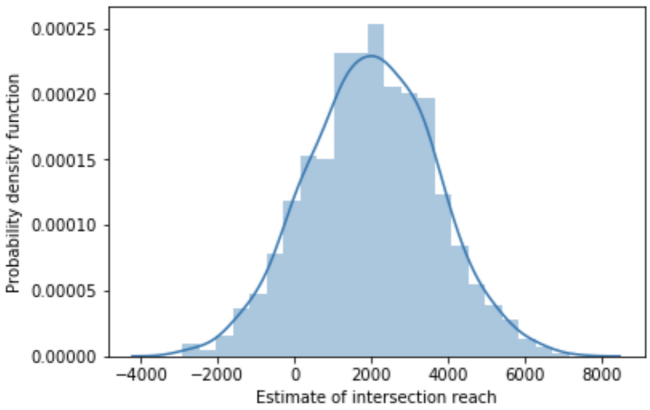

Figure 16: VoC can give negative estimate of intersection

In practice, it is often desirable to guarantee a non-negative intersection estimate, or in other words, a consistency in union reach — for any two publishers, the union reach is always less than or equal to the sum of per-publisher reaches.

To guarantee this consistency, a simple remedy is to truncate all the negative estimates, i.e., round them up to zero. However, this introduces bias. Consider a simple example when the true intersection is zero. Before clipping, VoC gives negative and positive estimates with equal chance, and they average to be zero. After naive clipping, the estimate is always positive, which introduces a positive bias.

To mitigate this bias, we adopt a "symmetric" clipping, based on hypothesis testing. We clip not only all the negative estimates, but also the positive estimates that are "close enough" to zero. The closeness to zero is determined by the Z-score

$$Z = \frac{\text{Intersection estimate} - 0}{\text{Standard error of the intersection estimate}}. \tag{16}$$

Note that the Z-score can be calculated using equations (4) and (6). We then compare the obtained Z-score with a threshold α, and clip the estimate to zero whenever $Z < \alpha$.

What threshold α should we choose? There is a trade-off: a higher α reduces the bias when the true intersection is close to zero, but increases the bias for larger true intersection. We recommend

$$\alpha = 1.2$$

which minimizes the overall bias in certain senses (see the theory in Appendix A).

On the other hand, when the true intersection is so large that it is close to the per-publisher reaches $|S_1|$ and $|S_2|$, the estimated intersection from the original VoC estimator can exceed $|S_1|$ or $|S_2|$, which is unreasonable. We clip such large estimates likewise. For each estimate, evaluate

$$Z = \frac{\text{Intersection estimate} - \min(|S_1|, |S_2|)}{\text{Standard error of the intersection estimate}}. \tag{17}$$

Then compare this Z-score with a negative threshold, -1.2. Clip the estimate to $\min(|S_1|, |S_2|)$ whenever $Z > -1.2$.

Clipping is also needed for per-publisher reaches. Recall that the per-publisher reach is estimated by summing up the VoC. This estimate has the mean being the true per-publisher reach and the standard deviation being $\sqrt{2m}/\epsilon$, with ϵ-DP and m buckets in the VoC. When the true per-publisher reach is small, the sum of VoC could be negative, which is unreasonable but also introduces computational error (say, in

Sequential VoC). To avoid this, we evaluate

$$Z = \frac{\text{sum(VoC)} - 0}{\sqrt{2m}/\epsilon} \tag{18}$$

for each VoC from each publisher. If $Z < 1.2$, we clip this VoC to a vector of all zeros — essentially, this VoC will no longer be used in the reach nor frequency estimation.

We summarize how the estimation algorithms given in the previous sections (in particular, Algorithms 2 and 4) are modified with the foregoing Clipped VoC technique. First, in the initialization step, we evaluate the Z-score in equation (18) for every single VoC, including the stratified VoC. Clip a VoC to a vector of zeros, if it has $Z < 1.2$. In the following steps, whenever a two-way intersection estimate is obtained, evaluate two Z-scores using equations (16) and (17) respectively. Clip this intersection estimate to zero if $Z < 1.2$ in equation (16), and to $\min(|S_1|, |S_2|)$ if $Z > -1.2$ in (17). Here $|S_1|$ and $|S_2|$ are the per-publisher reaches corresponding to the intersection.

10 Meta VoC

This section describes a special use case of VoC, that is, VoC can be used as a privacy-preserving summary of bloom-filter-type sketches. Bloom filter (BF, Bloom 1970) is a classical space-efficient, probabilistic data structure for tracking distinct items in a set, and can naturally be used in (non-privacy-preserving) reach and frequency estimation. BF is extended to Liquid Legions (LL). Compared to BF, LL is more favorable in the sense that its accuracy is much less sensitive to the set cardinality and thus it supports reach and frequency estimation for large campaigns. See Wright et al. (2020) for details.

Here is a high-level description of BF and LL. Like VoC, BF and LL also assign different user IDs to different buckets through a hash function. The buckets in BF and LL are called "registers". A register is active (or labeled as 1) if it contains at least one ID, and is inactive (or labeled as 0) if it does not contain any ID. As such, BF and LL are binary vectors, unlike VoC which tracks counts. BF and LL often have 10^5 or more registers, and are longer than VoC.

At each publisher, BF and LL are computationally easier to obtain than VoC. BF and LL can be directly obtained from the stream of ad impressions. Whenever a publisher receives a ping of ad impression, they just extract the user ID of this impression, find the register that the ID falls in, and then activate this register. On the other hand, VoC has to be generated from a set of reached users. It means that for each campaign, each publisher has to first save all the impressions of this campaign into a dataset and then "select distinct users" from the dataset, so as to produce a VoC.

BF and LL are computationally efficient, but can they be used for privacy-preserving reach and frequency estimation? LL is used in Wright et al. (2020) as a global-DP solution. However, it is not a viable local-DP solution (see Subsection 2.4 for a description of global and local solutions). This is because the accuracy of LL is highly susceptible to local noise (see, e.g., World Federation of Advertisers 2020). On the other hand, the accuracy of VoC is much more robust to local noise.

So if publishers already have a pipeline to generate LL for global-DP reach and frequency estimation, do they need to build another, computationally slower pipeline to generate VoC for local-DP estimation? The idea of Meta (most effective tactics available) VoC is to convert an LL into VoC. In this way, each publisher can just generate LL and send it to the global (MPC) system. The global system estimates the union reach of all publishers, and meanwhile converts each LL into a VoC, adds local noise, and saves these VoCs for ad-hoc, local-DP estimation tasks (see Subsection 2.4).

In Meta VoC, we use VoC to track the set of "reached registers" in LL or BF, instead of reached users. That is, treat the locations of LL registers as user IDs, and summarize the locations of active registers in the LL or BF of each publisher as a VoC. One can use the VoCs thus obtained to estimate the "union reach

of registers", i.e., the number of active registers in the merged LL, for any subset of publishers. The number of reached users can then be inferred from the number of active registers. See the following diagram.

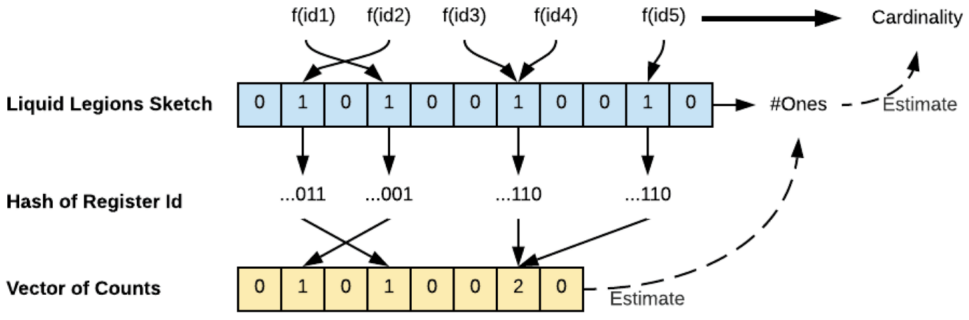

Figure 17: Diagram of Meta VoC

Each publisher can convert their LL to VoC following this diagram. Yet, it is not super efficient if each publisher builds a separate pipeline for Meta VoC. Ideally, the global MPC system handles all the LLs and converts them to VoCs using a common pipeline. This is, however, non-trial in terms of privacy. A careful design is needed to make sure that individual information is not leaked to each "worker" in MPC (see Wright et al. 2020). A high-level design for this purpose is as follows.

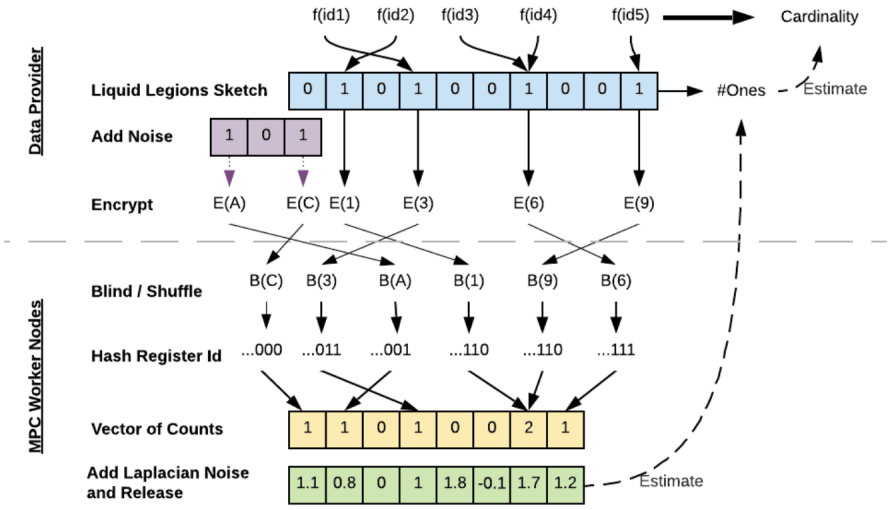

Figure 18: A Design of Meta VoC in the MPC system

There exist a number of candidate protocols that could realize this design. The above diagram purposefully lacks specifics as they can differ among protocols. A future work is to specify each step and rigorously evaluate the privacy of each candidate protocols.

We close this section with a caveat on the accuracy of Meta VoC. With the VoC of reached registers from each publisher, we use Sequential VoC (Algorithm 2) to estimate the union reach of registers. As mentioned in Section 5, Sequential VoC can have a large bias in highly correlated cases. Here, the reached registers of LL are indeed heavily correlated between publishers. The correlation comes from the fact that different registers in LL are not equally active. The most active registers are very likely to be reached by all the publishers, while the least active registers are unlikely to be reached by any publisher. Even if the reached

users are independent between publishers, the reached registers are not independent. In fact, the correlation here is equivalent to that in the simulation scenario B of Example 1. The performance of Meta VoC is thus similar to that in Figure 12, which has severe bias for 10 or more publishers. Therefore, we conclude that the current Meta VoC technique is only useful for a subset of no more than five publishers. Also, note that the current Meta VoC technique does not support frequency estimation.

11 Concluding remarks

In this paper, we propose a series of algorithms, based on a data structure called Vector of Counts (VoC), for privacy-centric reach and frequency estimation across ad publishers. To the best of our knowledge, this is the first locally-differentially-private (local-DP) solution in the literature for reach and frequency estimation. For two publishers, the proposed solution gives unbiased estimates, and the relative standard deviation is as small as 1% under mild conditions (each per-publisher reach $\geq 33,000$ and intersection $\geq 20\%$ of each per-publisher reach), with DP parameter $\epsilon = \ln(3)$. For more than two publishers, the solution does a bias-variance trade-off: it still guarantees small variance, at the risk of having large bias with > 5 publishers if the user activities are correlated between publishers.

Below, we discuss some thoughts on the practical application of this paper. As described in Subsection 2.4 and Section 10, in our vision, the proposed local-DP solution is used in combination with a global solution, for example, that of Wright et al. (2020). For any ad campaign, each publisher sends raw sketches (say, Liquid Legions) to the MPC system. The MPC system computes the estimation result across all publishers, and further releases the local-DP VoCs (say, through Meta VoC). Estimation results for any (proper) subset of publishers are not pre-computed. They are only computed upon the advertiser's request/query. As explained before, if we want to compute these queries in MPC, the advertiser may have to pay for the additional computational cost, and tolerate the lag time for every new query. On the other hand, the VoC solution allows the advertiser to freely choose (under a privacy budget) subsets of publishers and to receive results immediately.

Compared to the global solution, the potential bias in the proposed VoC solution is worth concern. Here are some thoughts on how to mitigate this issue. We can check the bias of VoC against the global solution result. Just use Sequential VoC to estimate the union reach across all the publishers, and see if the result aligns with that from the global solution. If we see obvious bias in VoC, here are two options. First, scale up or down the VoC results. For reach estimation across any j publishers, multiply the VoC estimation of union reach by a coefficient λ_j. For two publishers, we know that VoC is unbiased, so λ_2 should be 1. For all the k publishers, λ_k is determined by the ratio of global and VoC results. The middle-layer coefficients can be interpolated, for example, by the power law $\lambda_j = \lambda_k^{(j-2)/(k-2)}$. The second option is to raise a caveat and to let the advertiser decide if they are willing to pay and wait for the global solution to obtain a precise result. Empirical results (World Federation of Advertisers 2020) show that the VoC bias is quite small for $3, 4, 5$ publishers, even under stress tests. As such, we may raise the caveat just for a query of more than five publishers.

Besides the global solution result, the three-way CenteredDotProduct in Subsection 6.1 provides an additional tool for detecting and correcting the bias in Sequential VoC. As for the frequency estimation, the potential bias can be handled using similar heuristics, but the performance is more concerning. These will be investigated in the future.

We would like to emphasize a novel aspect of VoC sketches relative to conventional sketches. Sketches derived from the Flajolet-Martin algorithm, such as HLL, and K Min Values (KMV, Sparka et al. 2018) are constructed based on data from a representative sample of users. Essentially, the individual-level information of the sampled users are preserved, which carries information about cardinality. Because of this, raw HLL/KMV sketches expose information about sampled users and they need a large amount of noise to guarantee DP. On the other hand, VoC sketchesare constructed from aggregates. A raw VoC is not quite anonymized, but they are inherently much easier to make DP. The cardinality is estimable based on the

fact that the correlations (of user presence between publishers) are preserved after aggregation. The ideas of aggregation and correlation can be potentially used in other privacy-preserving measurements, not just for reach and frequency.

Acknowledgement

We would like to thank Tony Fagan, Penny Chu and Lu Zhang for their leadership, encouragement and support, and Arthur Asuncion, Damien Desfontaines, Aiyou Chen and Jim Dravillas for their insightful discussion and constructive suggestions.

References

Abadi, M., A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang (2016). Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 308–318.

Apple (2019). Intelligent tracking prevention 2.3. https://webkit.org/blog/9521/intelligent-tracking-prevention-2-3.

Association of National Advertisers (2020). The necessity of cross-media measurement. https://www.ana.net/miccontent/showvideo/id/v-roi-sep20v-proctergamble.

Bloom, B. H. (1970). Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM 13*(7), 422–426.

Caffyn, J. M. and M. Sagovsky (1963). Net audiences of British newspapers – a comparison of the Agostini and Sainsbury methods. *Journal of Advertising Research 3*(1), 21–25.

California State (2018). California consumer privacy act of 2018. http://leginfo.legislature.ca.gov/faces/codes_displayText.xhtml?division=3.&part=4.&lawCode=CIV&title=1.81.5.

Council of European Union (2016). Regulation (EU) 2016/679 of the European parliament and of the council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46. https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679.

Desfontaines, D., A. Lochbihler, and D. Basin (2019). Cardinality estimators do not preserve privacy. *Proceedings on Privacy Enhancing Technologies 2019*(2), 26–46.

Dwork, C., F. McSherry, K. Nissim, and A. Smith (2006). Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference*, pp. 265–284. Springer.

Dwork, C., A. Roth, et al. (2014). The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science 9*(3-4), 211–407.

Facebook (2019). Data portability and privacy. https://iapp.org/media/pdf/fb_whitepaper_sep_2019.pdf.

Flajolet, P., É. Fusy, O. Gandouet, and F. Meunier (2007, June). HyperLogLog: the analysis of a near-optimal cardinality estimation algorithm. In P. Jacquet (Ed.), *AofA: Analysis of Algorithms*, Volume DMTCS Proceedings vol. AH, 2007 Conference on Analysis of Algorithms (AofA 07) of *DMTCS Proceedings*, Juan les Pins, France, pp. 137–156. Discrete Mathematics and Theoretical Computer Science.

Flajolet, P. and G. N. Martin (1985). Probabilistic counting algorithms for data base applications. *Journal of Computer and System Sciences 31*(2), 182–209.

Google (2020). Google Chrome privacy whitepaper. https://www.google.com/chrome/privacy/whitepaper.html.

Incorporated Society of British Advertisers (2019). Origin – the UK cross media measurement programme. https://originmediameasurement.com.

Media Rating Council (2019). MRC cross-media audience measurement standards. http://mediaratingcouncil.org/MRC%20Issues%20Final%20Version%20of%20Cross-Media%20Audience%20Measurement%20Standards%20For%20Video%20-%20FINAL.pdf.

National Institute of Standards and Technology (2015). Secure hash standard. https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf.

Skvortsov, E. and J. Koehler (2019). Virtual people: Actionable reach modeling. Technical report, Google Inc. https://research.google/pubs/pub48387.

Sparka, H., F. Tschorsch, and B. Scheuermann (2018). P2kmv: a privacy-preserving counting sketch for efficient and accurate set intersection cardinality estimations.

Tchórzewski, J. and A. Jakóbik (2019). Theoretical and experimental analysis of cryptographic hash functions. *Journal of Telecommunications and Information Technology 1*, 125–133.

Vadhan, S. (2017). The complexity of differential privacy. In *Tutorials on the Foundations of Cryptography*, pp. 347–450. Springer.

World Federation of Advertisers (2019). Cross-media measurement system for reach and frequency. https://wfanet.org/l/library/download/urn:uuid:2647d566-42fb-45d5-8f64-c62356efc46d/cross_media_measurement_system_for_reach_and_frequency.pdf?format=save_to_disk&ext=.pdf.

World Federation of Advertisers (2020). Cardinality estimation evaluation framework. https://github.com/world-federation-of-advertisers/cardinality_estimation_evaluation_framework.

Wright, C. W., B. Kreuter, , E. S. Skvortsov, R. Mirisola, and Y. Wang (2020). Privacy-preserving secure cardinality and frequency estimation. Technical report, Google Inc. https://research.google/pubs/pub49177.

Appendix A: Theoretical results that provide additional insights

Estimation of three-way intersection using CenteredDotProduct of VoCs

The three-way CenteredDotProduct is an unbiased estimator of the three-way intersection:

Proposition 8. *For* $|S_1 \cap \widehat{S_2 \cap} S_3|$ *defined in equation (11),*

$$\mathrm{E}\left[|S_1 \cap \widehat{S_2 \cap} S_3|\right] \approx |S_1 \cap S_2 \cap S_3|.$$

where the expectation is taken over the probability space in Theorem 1.

The variance can be derived following the proof of Theorem 1, with heavy mathematics. To provide insight, we show the clean result under a special case: disjoint reach.

Proposition 9. *When the true $S_1 \cap S_2 = S_1 \cap S_3 = S_2 \cap S_3 = \varnothing$ (hence $S_1 \cap S_2 \cap S_3$ is also empty),*

$$\mathrm{Var}\left[|S_1 \cap \widehat{S_2 \cap S_3}|\right] \approx \frac{|S_1||S_2||S_3|}{m^2} + \frac{2(|S_1||S_2| + |S_1||S_3| + |S_2||S_3|)}{m\epsilon^2} + \frac{4(|S_1| + |S_2| + |S_3|)}{\epsilon^4} + \frac{8m}{\epsilon^6},$$

and

$$\begin{aligned}\mathrm{Var}\left[|S_1 \cup \widehat{S_2 \cup S_3}|\right] \approx & \mathrm{Var}\left[|S_1 \cap \widehat{S_2 \cap S_3}|\right] + \frac{|S_1||S_2| + |S_1||S_3| + |S_2||S_3|}{m} \\ & + \frac{2(2|S_1| + 2|S_2| + 2|S_3| + 3m)}{\epsilon^2} + \frac{6m}{\epsilon^4},\end{aligned} \tag{19}$$

where $|S_1 \cup \widehat{S_2 \cup S_3}|$ is given by the inclusion-exclusion formula (10).

Propositions 8 and 9 can be proved following the proof of Theorem 1. We compare the accuracy of two-way and three-way intersection estimators using formulas (12) and (19):

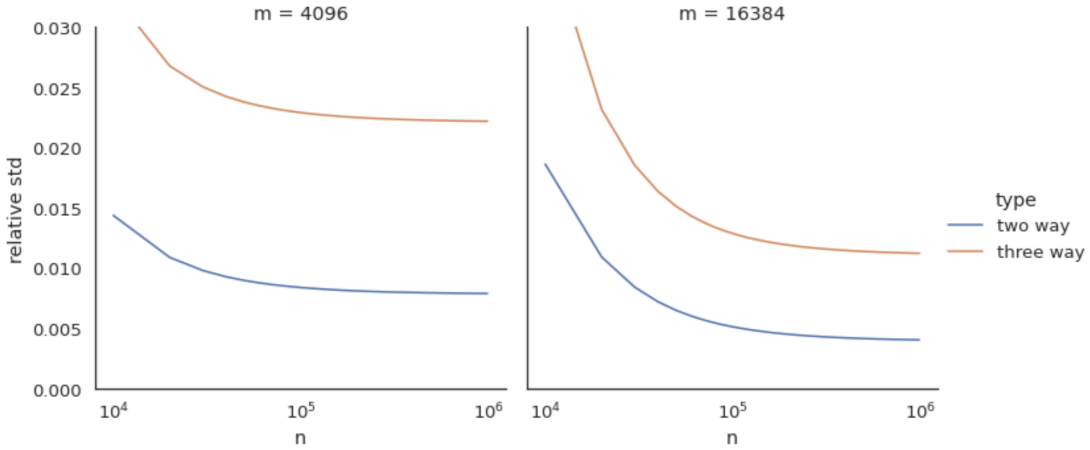

Figure 19: Relative std of two-way and three-way union estimators using CenteredDotProducts, in the special case that $S_1 \cap S_2 = S_1 \cap S_3 = S_2 \cap S_3 = \varnothing$, when $|S_1| = |S_2| = |S_3| = n$ increases from 10^4 to 10^6, and when $m = 2^{12}$ and 2^{14} respectively

Here, the relative std is still with respect to the union. In the plot, the relative std for three-way is always less than three times that for two-way. The three-way accuracy is thus acceptable. With VoC length being $2^{12} = 4096$, the three-way relative std is above 2%. If we increase the length to $2^{14} = 16384$, the relative std can be controlled under 1.5%, given a reasonable minimum reporting threshold of the per-publisher reach.

Sequential VoC has stable variance

We claim that the relative std of Sequential VoC does not propagate as the number of publishers increases, and this claim is supported by empirical results. Here is some theoretical support. Still, for simplicity, consider the special case of disjoint reach. Explicitly:

Proposition 10. *Let S_j be the set of reached users at publisher j, $j = 1, 2, \ldots$. Suppose that $S_i \cap S_j = \varnothing$ for any $i \neq j$. For any positive integer k, let ξ_k be the union estimate given by Sequential VoC up to k publishers. That is, ξ_k is the output of Algorithm 2 with the input being $[\mathbf{VoC}_{m,h,\epsilon}(S_j)]_{1 \le j \le k}$. Then:*

$$\mathrm{Var}(\xi_k) \approx \frac{2mk}{\epsilon^2} + \sum_{1 \le i < j \le k} V_{ij},$$

where V_{ij} is the variance for estimating the intersection between publishers i and j, which is

$$V_{ij} = \frac{|S_1||S_2|}{m} + \frac{2(|S_1| + |S_2|)}{\epsilon^2} + \frac{4m}{\epsilon^4}$$

in the disjoint case.

This result can be proved following the proofs of Theorem 1 and Proposition 7; details are omitted here. It suggests that roughly speaking, $\mathrm{Var}(\xi_k)$ grows at the order $\Omega(k^2)$ as the number of publishers k increases. Therefore, the standard deviation grows at $\Omega(k)$. For disjoint reach, the true union grows at the order $\Omega(k)$ as well, and therefore the relative std is at the order $\Omega(1)$. That is, the relative std is stable, no matter how large k is.

Tuning the threshold in clipped VoC

The following result helps in tuning the threshold:

Proposition 11. *Suppose that X is an unbiased estimator of a true value μ. Suppose that X is normal distributed with standard deviation σ. Let Y be the clipped X using the proposed Z-score methodology with threshold z^*, that is, $Y = 0$ if $X/\sigma < z^*$ and $= X$ otherwise. Then, Y has a bias of*

$$\mathrm{E}(Y) - \mu = \sigma\varphi(z^* - \mu/\sigma) - \mu\Phi(z^* - \mu/\sigma)$$

for estimating μ, where φ and Φ are the probability density function and cumulative distribution function, respectively, of the standard normal distribution.

In our context, μ is the true intersection $|S_1 \cap S_2|$, X is the original two-way VoC estimator, and Y is the clipped estimator. σ^2 is then given by equation (6). Our goal is to minimize the maximum possible absolute value of bias, $|\mathrm{E}(Y) - \mu|$, as $\mu = |S_1 \cap S_2|$ increases from zero to $\min(|S_1|, |S_2|)$.

Note that $|\mathrm{E}(Y) - \mu| = \sigma\,|\varphi(z^* - \mu/\sigma) - (\mu/\sigma)\Phi(z^* - \mu/\sigma)|$. It can be shown that for the min-max problem as described above, we only need to consider small $\mu = |S_1 \cap S_2|$ (say, $\mu < (1/5)\min(|S_1|, |S_2|)$) , for which σ is roughly a constant (not depending on μ). That is, $|\mathrm{E}(Y) - \mu|$ is proportional to $|\varphi(z^* - \mu/\sigma) - (\mu/\sigma)\Phi(z^* - \mu/\sigma)|$. It then suffices to solve

$$\arg\min_{z^*>0} \max_{s\geq 0} |\varphi(z^* - s) - s\Phi(z^* - s)|\,.$$

Via theoretical determination of a range that z^* falls in, plus a numerical search of z^* in the range, we found that $z^* = 1.189$ is the solution of the above min-max problem. Thus, we recommend $z^* = 1.2$ as the threshold in the clipped VoC.

Appendix B: Proofs

Proof of Theorem 1. Consider a universe of users $\{1, \ldots, U\}$. The reached set S_1, S_2 are subsets of this universe. Let $x_u := I(\text{user } u \text{ is reached by publisher 1})$, $y_u := I(\text{user } u \text{ is reached by publisher 2})$, for $1 \leq u \leq U$, where $I(\cdot)$ is the indicator function. On the other hand, let $I_{u,i} = I(\text{user } u \text{ is in bucket i})$, for $1 \leq u \leq U$ and $0 \leq i \leq m-1$.

Let $\mathbf{c}_1$ and $\mathbf{c}_2$ denote the raw VoCs of publishers 1 and 2, respectively. Then, $\mathbf{c}_1[i] = \sum_{u=1}^{U} x_u I_{u,i}$ and $\mathbf{c}_1[i] = \sum_{u=1}^{U} y_u I_{u,i}$, for $0 \leq i \leq m-1$. Then centered VoCs then have their ith element being

$$\mathbf{c}_1[i] - \mathrm{sum}(\mathbf{c}_1)/m = \sum_{u=1}^{U} x_u \delta_{u,i} \text{ and } \mathbf{c}_2[i] - \mathrm{sum}(\mathbf{c}_2)/m = \sum_{u=1}^{U} y_u \delta_{u,i},$$

where $\delta_{u,i} = I_{u,i} - 1/m$, which equals $1 - 1/m$ with probability $1/m$ and $-1/m$ with probability $1 - 1/m$. The CenteredDotProduct of $\mathbf{c}_1$ and $\mathbf{c}_2$ is then

$$\begin{aligned}
&\sum_{i=0}^{m-1} (\mathbf{c}_1[i] - \text{sum}(\mathbf{c})/m)(\mathbf{c}_2[i] - \text{sum}(\mathbf{c})/m) \\
&= \sum_{i=0}^{m-1} \left(\sum_{u=1}^{U} x_u \delta_{u,i} \cdot \sum_{u=1}^{U} y_u \delta_{u,i} \right) \\
&= \sum_{i=0}^{m-1} \left(\sum_{u=1}^{U} x_u y_u \delta_{u,i}^2 + 2 \sum_{0 \le u < v \le U} x_u y_u \delta_{u,i} \delta_{v,i}. \right)
\end{aligned} \tag{20}$$

The mean and variance of this quantity can be derived from the moments of the random variables $\delta_{u,i} = I_{u,i} - 1/m$. Under the "random hash" probability space specified in the theorem, we have:

- Each $I_{u,i} = 1$ with probability $1/m$ and $= 0$ otherwise.
- $I_{u,i}$ is independent with $I_{v,j}$ for any $u \neq v$ and any i, j.
- $I_{u,i} I_{u,j}$ for any u and any $i \neq j$.

It is then easy to show that

- Each $\mathrm{E}(\delta_{u,i}) = 0$.
- Each $\mathrm{E}(\delta_{u,i}^2) = (1/m)(1 - 1/m) \approx 1/m$.
- In general, $\mathrm{E}(\delta_{u,i}^r) = \approx 1/m$ for any integer $r \geq 2$.
- $\delta_{u,i}$ is independent with $\delta_{v,j}$ for any $u \neq v$ and any i, j.
- $E(\delta_{u,i}\delta_{u,j}) = -1/m^2$ and $E(\delta_{u,i}^2 \delta_{u,j}^2) \approx 2/m^3$ for any u and any $i \neq j$.

Then by equation (20),

$$\begin{aligned}
&\mathrm{E}\left[\text{CenteredDotProduct}(\mathbf{c}_1, \mathbf{c}_2)\right] \\
&= \sum_{i=0}^{m-1} \left[\sum_{u=1}^{U} x_u y_u \,\mathrm{E}(\delta_{u,i}^2) + 2 \sum_{0 \le u < v \le U} x_u y_u \,\mathrm{E}(\delta_{u,i})\,\mathrm{E}(\delta_{v,i}) \right] \\
&\approx \sum_{i=0}^{m-1} \sum_{u=1}^{U} x_u y_u (1/m) = \sum_{u=1}^{U} x_u y_u.
\end{aligned}$$

By the definition of x_u, y_u, $x_u y_u = 1$ if user i is reached by both publisher 1 and 2. Thus, $\sum_{u=1}^{U} x_u y_u$ equals the intersection reach $|S_1 \cap S_2|$. This proves that CenteredDotProduct$(\mathbf{c}_1, \mathbf{c}_2)$ is an unbiased estimator of $|S_1 \cap S_2|$.

Now, we derive the variance of this estimator. Let $\xi :=$ CenteredDotProduct$(\mathbf{c}_1, \mathbf{c}_2)$. Then,

$$\begin{aligned}
\mathrm{E}(\xi^2) &= \mathrm{E}\left[\sum_{i=0}^{m-1} \left(\sum_{u=1}^{U} x_u \delta_{u,i} \right) \left(\sum_{u=1}^{U} y_u \delta_{u,i} \right) \right]^2 \\
&= \sum_{i,j,u_1,u_2,u_3,u_4} x_{u_1} y_{u_2} x_{u_3} y_{u_4} \,\mathrm{E}(\delta_{u_1,i} \delta_{u_2,i} \delta_{u_3,j} \delta_{u_4,j}).
\end{aligned}$$

Denote $E(\delta_{u_1,i}\delta_{u_2,i}\delta_{u_3,j}\delta_{u_4,j})$ as A. From the above properties of $\delta_{u,i}$, it is not difficult to show that

(1) When $i = j$,

$$A = \begin{cases} 1/m & \text{if } u_1 = u_2 = u_3 = u_4 \\ 1/m^2 & \text{if } u_1 = u_2 \neq u_3 = u_4 \text{ or } u_1 = u_3 \neq u_2 = u_4 \text{ or } u_1 = u_4 \neq u_2 = u_3 \\ 0 & \text{otherwise.} \end{cases}$$

(2) When $i \neq j$,

$$A = \begin{cases} \mathrm{E}(\delta_{u,i}^2\delta_{u,j}^2) = 2/m^3 & \text{if } u_1 = u_2 = u_3 = u_4 \\ \mathrm{E}(\delta_{u,i}^2)\,\mathrm{E}(\delta_{v,j}^2) = 1/m^2 & \text{if } u_1 = u_2 \neq u_3 = u_4 \\ \mathrm{E}(\delta_{u,i}\delta_{u,j})\,\mathrm{E}(\delta_{v,i}\delta_{v,j}) = 1/m^4 & \text{if } u_1 = u_3 \neq u_2 = u_4 \\ \mathrm{E}(\delta_{u,i}\delta_{u,j})\,\mathrm{E}(\delta_{v,i}\delta_{v,j}) = 1/m^4 & \text{if } u_1 = u_4 \neq u_2 = u_3 \\ 0 & \text{otherwise.} \end{cases}$$

Thus,

$$\begin{aligned}
\mathrm{E}(\xi^2) =& \sum_{i,j,u_1,u_2,u_3,u_4} x_{u_1}y_{u_2}x_{u_3}y_{u_4}\,\mathrm{E}(\delta_{u_1,i}\delta_{u_2,i}\delta_{u_3,j}\delta_{u_4,j}) \\
=& m^{-1}\sum_{i,u} x_u^2y_u^2 + m^{-2}\sum_{i,u\neq v}(x_uy_ux_vy_v + x_uy_vx_uy_v + x_uy_vx_vy_u) \\
&+ 2m^{-3}\sum_{i\neq j,u} x_u^2y_u^2 + m^{-2}\sum_{i\neq j,u\neq v} x_uy_ux_vy_v + m^{-4}\sum_{i\neq j,u\neq v}(x_uy_vx_uy_v + x_uy_vx_vy_u) \\
=& m^{-1}m\sum_u x_u^2y_u^2 + m^{-2}m\sum_{u\neq v}(x_u^2y_v^2 + 2x_ux_vy_uy_v) \\
&+ 2m^{-3}m(m-1)\sum_u x_u^2y_u^2 + m^{-2}m(m-1)\sum_{u\neq v} x_ux_vy_uy_v + 2m^{-4}m(m-1)\sum_{u\neq v}(x_u^2y_v^2 + x_uy_vx_vy_u) \\
=& [1 + 2m^{-1} + o(m^{-1})]\sum_u x_u^2y_u^2 + [m^{-1} + o(m^{-1})]\sum_{u\neq v} x_u^2y_v^2 + [1 + m^{-1} + o(m^{-1})]\sum_{u\neq v} x_ux_vy_uy_v.
\end{aligned}$$

We have

$$\begin{aligned}
\sum_{u\neq v} x_ux_vy_uy_v =& \sum_{u,v} x_ux_vy_uy_v - \sum_{u=v} x_ux_vy_uy_v \\
=& \sum_u x_uy_u \sum_v x_vy_v - \sum_u x_u^2y_u^2 \\
=& \left(\sum_u x_uy_u\right)^2 - \sum_u x_uy_u \\
=& |S_1 \cap S_2|^2 - |S_1 \cap S_2|.
\end{aligned}$$

Similarly, it can be shown that

$$\begin{aligned}
\sum_u x_u^2y_u^2 &= |S_1 \cap S_2|, \\
\sum_{u\neq v} x_u^2y_v^2 &= |S_1||S_2| - |S_1 \cap S_2|.
\end{aligned}$$

Hence,

$$\mathrm{E}(\xi^2) = o(m^{-1})|S_1 \cap S_2| + [1 + m^{-1} + o(m^{-1})]|S_1 \cap S_2|^2 + [m^{-1} + o(m^{-1})]m^{-1} + o(m^{-1})].$$

And then

$$\begin{aligned}\mathrm{Var}(\xi) =& \mathrm{E}(\xi^2) - [E(\xi)]^2 = \mathrm{E}(\xi^2) - |S_1 \cap S_2|^2 \\ =& (|S_1||S_2| + |S_1 \cap S_2|^2)[m^{-1} + o(m^{-1})] \\ \approx& \frac{|S_1||S_2| + |S_1 \cap S_2|^2}{m},\end{aligned}$$

This proves that the CenteredDotProduct of two *raw* VoCs has variance $(|S_1||S_2| + |S_1 \cap S_2|^2)/m$.

Now, for *noised* VoCs, denote ζ the estimator CenteredDotProduct($\mathbf{VoC}_{m,h,\epsilon}(S_1), \mathbf{VoC}_{m,h,\epsilon}(S_2)$). We have

$$\zeta = \sum_{i=0}^{m-1} (\mathbf{c}_1[i] - \mathrm{sum}(\mathbf{c}_1)/m + \gamma_{1,i} - \bar{\gamma}_1)(\mathbf{c}_2[i] - \mathrm{sum}(\mathbf{c}_2)/m + \gamma_{2,i} - \bar{\gamma}_2) \tag{21}$$

where all $\gamma_{1,i}$ and $\gamma_{2,i}$ are independent Laplacian random variables with variance $2/\epsilon^2$, $\bar{\gamma}_j = \sum_{i=0}^{m-1} \gamma_{j,i}/m$, $j = 1, 2$. First, the noise does not introduce bias. By the independence between different terms and the fact that $\mathrm{E}(\gamma_{j,i}) = 0$, we have

$$\begin{aligned}&\mathrm{E}\left[(\mathbf{c}_1[i] - \mathrm{sum}(\mathbf{c}_1)/m + \gamma_{1,i} - \bar{\gamma}_1)(\mathbf{c}_2[i] - \mathrm{sum}(\mathbf{c}_2)/m + \gamma_{2,i} - \bar{\gamma}_2)\right] \\ =& \mathrm{E}\left[(\mathbf{c}_1[i] - \mathrm{sum}(\mathbf{c}_1)/m)(\mathbf{c}_2[i] - \mathrm{sum}(\mathbf{c}_2)/m)\right] + \mathrm{E}(\mathbf{c}_1[i] - \mathrm{sum}(\mathbf{c}_1)/m)\,\mathrm{E}(\gamma_{2,i} - \bar{\gamma}_2) \\ &+ \mathrm{E}(\gamma_{1,i} - \bar{\gamma}_1)\,\mathrm{E}(\mathbf{c}_2[i] - \mathrm{sum}(\mathbf{c}_2)/m) + \mathrm{E}(\gamma_{1,i} - \bar{\gamma}_1)\,\mathrm{E}(\gamma_{2,i} - \bar{\gamma}_2) \\ =& \mathrm{E}\left[(\mathbf{c}_1[i] - \mathrm{sum}(\mathbf{c}_1)/m)(\mathbf{c}_2[i] - \mathrm{sum}(\mathbf{c}_2)/m)\right],\end{aligned}$$

and therefore $\mathrm{E}(\zeta) = \sum_{i=0}^{m-1} \mathrm{E}\left[(\mathbf{c}_1[i] - \mathrm{sum}(\mathbf{c}_1)/m)(\mathbf{c}_2[i] - \mathrm{sum}(\mathbf{c}_2)/m)\right]$ still equals $|S_1 \cap S_2|$.

To evaluate the variance of the ζ in equation (21), we first make an approximation by dropping two terms $\bar{\gamma}_1$ and $\bar{\gamma}_2$, that is, approximate ζ as

$$\zeta \approx \sum_{i=0}^{m-1} (\mathbf{c}_1[i] - \mathrm{sum}(\mathbf{c}_1)/m + \gamma_{1,i})(\mathbf{c}_2[i] - \mathrm{sum}(\mathbf{c}_2)/m + \gamma_{2,i}).$$

This is reasonable as the $\mathrm{Var}(\bar{\gamma}_1) = \mathrm{Var}(\bar{\gamma}_2) = 2/(m\epsilon^2)$, which introduces negligible error. (And this is rigorizable with heavy mathematics, which are omitted here.) Then,

$$\zeta \approx A + B + C + D,$$

$$\begin{aligned}\text{where } A =& \sum_{i=0}^{m-1} (\mathbf{c}_1[i] - \mathrm{sum}(\mathbf{c}_1)/m)(\mathbf{c}_2[i] - \mathrm{sum}(\mathbf{c}_2)/m), \\ B =& \sum_{i=0}^{m-1} (\mathbf{c}_1[i] - \mathrm{sum}(\mathbf{c}_1)/m) \cdot \gamma_{2,i}, \\ C =& \sum_{i=0}^{m-1} \gamma_{1,i} \cdot (\mathbf{c}_2[i] - \mathrm{sum}(\mathbf{c}_2)/m), \\ D =& \sum_{i=0}^{m-1} \gamma_{1,i} \cdot \gamma_{2,i}.\end{aligned}$$

From the fact that (i) $\gamma_{1,i}$ is independent with $\gamma_{2,j}$; (ii) $\gamma_{1,i}$ is independent with $\mathbf{c}_1, \mathbf{c}_2$; (iii) $\gamma_{2,i}$ is independent with $\mathbf{c}_1, \mathbf{c}_2$; (iv) $\mathrm{E}(\gamma_{1,i}) = \mathrm{E}(\gamma_{2,i}) = 0$, it is not difficult to see that A, B, C, D have zero covariance between each other. Then,

$$\mathrm{Var}(\zeta) \approx \mathrm{Var}(A) + \mathrm{Var}(B) + \mathrm{Var}(C) + \mathrm{Var}(D).$$

As shown above, $\mathrm{Var}(A) \approx (|S_1||S_2| + |S_1 \cap S_2|^2)/m$. The variance of B, C, D can be obtained by the law of total variance. Explicitly,

$$\begin{aligned}
\mathrm{Var}(B) =& \mathrm{E}(\mathrm{Var}(B|\mathbf{c}_1)) + \mathrm{Var}(\mathrm{E}(B|\mathbf{c}_1)) \\
=& \mathrm{E}\left[\sum_{i=0}^{m-1} (\mathbf{c}_1[i] - \mathrm{sum}(\mathbf{c}_1)/m)^2 \cdot \frac{2}{\epsilon^2}\right] + 0 \\
=& \frac{2}{\epsilon^2} \mathrm{E}\left[\mathrm{CenteredDotProduct}(\mathbf{c}_1, \mathbf{c}_1)\right] \\
=& \frac{2}{\epsilon^2}|S_1 \cap S_1| = \frac{2}{\epsilon^2}|S_1|.
\end{aligned}$$

Similarly, it can be shown that

$$\begin{aligned}
\mathrm{Var}(C) &= \frac{2}{\epsilon^2}|S_2| \\
\mathrm{Var}(D) &= \frac{4m}{\epsilon^4}.
\end{aligned}$$

Then, $\mathrm{Var}(\zeta) = \mathrm{Var}(A) + \mathrm{Var}(B) + \mathrm{Var}(C) + \mathrm{Var}(D)$ equals the result in equation (6), which completes the proof.

This proof can be simplified by assuming that different elements of each VoC are uncorrelated with each other (the above rigorous proof essentially shows that this assumption introduces a $o(1)$ multiplicative error to the result). Proportions 8 and 9 can be proved in the same way, with more complicated calculations. □

Proof of Proposition 7. In Algorithm 2, we use $\hat{\mathbf{c}} := (\mathbf{VoC}_1 + \mathbf{VoC}_2) \times \{1 - \hat{n}_{12}/[\mathrm{sum}(\mathbf{VoC}_1) + \mathrm{sum}(\mathbf{VoC}_2)]\} \approx (\mathbf{VoC}_1 + \mathbf{VoC}_2) \times [1 - \hat{n}_{12}/(n_1 + n_2)]$ to approximate the VoC of $|S_1 \cup S_2|$, where $\mathbf{VoC}_j = \mathbf{VoC}(S_j)$, $j = 1, 2, 3$, and $\hat{n}_{12} = \mathrm{CenteredDotProduct}(\mathbf{VoC}_1, \mathbf{VoC}_2)$ is the estimate of n_{12}. Then,

$$\begin{aligned}
|\widehat{S_1 \cup S_2 \cup S_3}| =& \mathrm{sum}(\hat{\mathbf{c}}) + \mathrm{sum}(\mathbf{VoC}_3) - \mathrm{CenteredDotProduct}(\hat{\mathbf{c}}, \mathbf{VoC}_3) \\
\approx& [\mathrm{sum}(\mathbf{VoC}_1) + \mathrm{sum}(\mathbf{VoC}_2)] \left(1 - \frac{\hat{n}_{12}}{n_1 + n_2}\right) + n_3 \\
& - \left(1 - \frac{\hat{n}_{12}}{n_1 + n_2}\right) \times \mathrm{CenteredDotProduct}\left(\mathbf{VoC}_1 + \mathbf{VoC}_2, \mathbf{VoC}_3\right) \\
=& n_1 + n_2 - \hat{n}_{12} + n_3 - \left(1 - \frac{\hat{n}_{12}}{n_1 + n_2}\right) \times (\hat{n}_{13} + \hat{n}_{23}) \\
=& n_1 + n_2 + n_3 - \hat{n}_{12} - \hat{n}_{13} - \hat{n}_{23} + \frac{\hat{n}_{12}(\hat{n}_{13} + \hat{n}_{23})}{n_1 + n_2}.
\end{aligned}$$

It follows that

$$\mathrm{E}(|\widehat{S_1 \cup S_2 \cup S_3}|) - |S_1 \cup S_2 \cup S_3| \approx \mathrm{E}\left(\frac{\hat{n}_{12}(\hat{n}_{13} + \hat{n}_{23})}{n_1 + n_2}\right) - n_{123}.$$

It suffices to show that $\mathrm{E}(\hat{n}_{12}\hat{n}_{13}) \approx n_{12}n_{13}$ and $\mathrm{E}(\hat{n}_{12}\hat{n}_{23}) \approx n_{12}n_{23}$. (Essentially, show that $\hat{n}_{12}, \hat{n}_{13}, \hat{n}_{23}$

are weakly correlated with each other.) Following the notations in the proof of Theorem 1,

$$\begin{aligned}
\mathrm{E}(\hat{n}_{12}\hat{n}_{13}) =& \mathrm{E}\left[\sum_{i=0}^{m-1}\left(\sum_{u=1}^{U} x_u\delta_{u,i}\sum_{u=1}^{U} y_u\delta_{u,i}\right)\sum_{i=0}^{m-1}\left(\sum_{u=1}^{U} x_u\delta_{u,i}\sum_{u=1}^{U} z_u\delta_{u,i}\right)\right] \\
=& \sum_{i,j,u_1,u_2,u_3,u_4} x_{u_1}y_{u_2}x_{u_3}z_{u_4}\,\mathrm{E}(\delta_{u_1,i}\delta_{u_2,i}\delta_{u_3,j}\delta_{u_4,j}) \\
=& \sum_{i,j,u_1,u_2,u_3,u_4} x_{u_1}y_{u_2}x_{u_3}z_{u_4}\,\mathrm{E}(\delta_{u_1,i}\delta_{u_2,i}\delta_{u_3,j}\delta_{u_4,j}) \\
=& m^{-1}\sum_{i,u} x_u^2 y_u z_u + m^{-2}\sum_{i,u\neq v}(x_u y_u x_v z_v + x_u y_v x_u z_v + x_u y_v x_v z_u) \\
&+ 2m^{-3}\sum_{i\neq j,u} x_u^2 y_u z_u + m^{-2}\sum_{i\neq j,u\neq v} x_u y_u x_v z_v + m^{-4}\sum_{i\neq j,u\neq v}(x_u y_v x_u z_v + x_u y_v x_v z_u) \\
=& n_{12}n_{13}[1+o(m^{-1})],
\end{aligned}$$

where $z_u = I(\text{user } u \text{ is reached by publisher 3})$ Same argument, $\mathrm{E}(\hat{n}_{12}\hat{n}_{23}) = n_{12}n_{23}[1+o(m^{-1})]$. This completes the proof. □

Proof of Proposition 6. Consider three publishers first. Under the independence assumption, $n_{12} = n_1 n_2/U$, $n_{13} = n_1 n_2/U$, $n_{23} = n_2 n_3/U$, and $n_{23} = n_1 n_2 n_3/U^2$. It follows that

$$\frac{n_{12}(n_{13}+n_{23})}{n_1+n_2} = n_{123}.$$

Then by Proposition 7, Sequential VoC is almost unbiased. The unbiasedness for $k > 3$ publishers can be proved by mathematical induction.

The disjoint and fully-overlapped cases are special cases of independence. So, Sequential VoC is almost unbiased under these cases. □