Web Privacy: A Formal Adversarial Model for Query Obfuscation

Florimond Houssiau, Thibaut Liénart, Julien Hendrickx, *Member, IEEE*, and Yves-Alexandre de Montjoye

***Abstract*— The queries we perform, the searches we make, and the websites we visit — this sensitive data is collected at scale by companies as part of the services they provide. Query obfuscation, intertwining the user queries with artificial queries, has been proposed as a solution to protect the privacy of individuals on the web. We here present a formal model and formulate through attack models three privacy requirements for obfuscators: 1) *indistinguishability*, that the user query should be hard to identify; 2) *coverage*, that its topic should be hard to identify; and 3) *imprecision*, that the query should still be hard to identify for an attacker with additional auxiliary information. The latter is needed to make the former two guarantees "future-proof". Using our framework, we derive two important results for obfuscators. First, we show that indistinguishability imposes strong bounds on the coverage and imprecision achievable by an obfuscator. Second, we prove an important tradeoff between coverage and imprecision, which inherently limits the strength and robustness of the privacy guarantees that an obfuscator can provide. We then introduce a family of obfuscators with provable indistinguishability guarantees, which we call $k-$ball obfuscators, and show, for a range of parameter values, the achievable coverage and imprecision. We show empirically that our theoretical tradeoff holds, and that its bound is not tight in practice: even in a simple idealized setting, there is a significant gap between practical coverage and imprecision guarantees, and the optimal bounds. While obfuscators have proven popular with the general public, all obfuscators currently available provide ad-hoc guarantees, and have been shown to be vulnerable to attacks, putting the data of users at risk. We hope this work to be a first step towards a robust evaluation of the properties of query obfuscators and the development of principled obfuscators.**

***Index Terms*— Privacy, obfuscation.**

I. Introduction

In THE past decades, the internet has grown to become the main source of information for billions of individuals. The richness and variety of the information available, as well as its ease of access, have made it a central part of our professional and personal lives.

However, as users browse the web, the trace of their activities, such as the websites they visit, the links they click on, and the search queries they perform, is recorded. For instance, many websites embed small pieces of code on other websites in order to track users, a practice known as "third-party tracking" that concerns 46% of websites in the Alexa Global Top 10,000 [1]. Similarly, Englehardt et al. showed in 2016 that Google and its subsidiaries had trackers on over 75% of 1 million popular websites [2], demonstrating the extent of the ability of the company to track users across the web. This browsing data is being used by companies to serve individuals personalized advertisement [3] but also by governments, for surveillance and counter-terrorism purposes [4]. This constant monitoring of online activity has led to what some call the "age of surveillance capitalism" [5].

This persistent and ubiquitous collection of fine-grained behavioral data raises serious privacy concerns. Web data can contain or be used to infer private information relative to, e.g., employment, financial and medical status [6], [7]. According to a 2014 report by BCG, 56% of Americans "consider [their surfing history] moderately to extremely private" [8]. It has also been shown to be trivially identifiable [9], leading to the worrying trend of users self-censoring the searches they make, thereby limiting their access to information [10].

Query obfuscation, the technique of polluting the user data by sending artificial queries in parallel to the queries of the user, has been proposed to alleviate this privacy issue for both web search [11], [12], [13], [14], [15], [16], [17], [18], [19] and general web traffic [20], [21], [22], [23], [24]. The idea is to "drown the original data in noise", such that anyone observing the queries is unable to learn information about the user. Query obfuscation is an approach popular with users as (1) it is easy to deploy, only requiring them to install a program on their machine, and (2) it does not require trust in a third-party (the data collector or other connected computers). Its popularity has increased further with the revocation of Net Neutrality by the FCC in 2017, with many independent developers proposing obfuscators [20], [21], [22], [23], [24].

However, while obfuscators are used in practice (e.g., TrackMeNot has over 40,000 downloads), the privacy guarantees they offer are often unclear, which puts their user at risk. TrackMeNot, for instance, has been shown to fall to simple attacks based on semantic similarity [25], the words used in queries [26] or even their capitalization [27]. The protection it promised on all queries of its users is now void, as all past obfuscated queries can be attacked and retrieved. More general

Manuscript received 22 March 2022; revised 26 January 2023; accepted 8 March 2023. Date of publication 27 March 2023; date of current version 7 April 2023. The work of Julien Hendrickx was supported in part by the "RevealFlight" Concerted Research Action (ARC) of the Fédération Wallonie-Bruxelles and in part by F.R.S.-FNRS via the Incentive Grant for Scientific Research (MIS) "Learning from Pairwise Comparisons" and the "KORNET Project." The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Chia-Mu Yu. *(Corresponding author: Florimond Houssiau.)*

Florimond Houssiau is with The Alan Turing Institute, NW1 2DB London, U.K., and also with Imperial College London, SW7 2AZ London, U.K. (e-mail: fhoussiau@turing.ac.uk).

Thibaut Liénart and Yves-Alexandre de Montjoye are with Imperial College London, SW7 2AZ London, U.K.

Julien Hendrickx is with ICTEAM institute, UCLouvain, 1348 Ottignies-Louvain-la-Neuve, Belgium.

This article has supplementary downloadable material available at https://doi.org/10.1109/TIFS.2023.3262123, provided by the authors.

Digital Object Identifier 10.1109/TIFS.2023.3262123

attacks, based on machine learning [28] or custom similarity metrics [29], have also been introduced and tested. In their review of obfuscators for web search, Balsa et al. concluded that all the obfuscators they evaluated were flawed and could be attacked to reveal sensitive user information [30].

In response to these concerns, some developers of obfuscators have argued that their goal is to protest large-scale data collection, rather than robustly protecting the privacy of their users. For instance, the authors of TrackMeNot state that "[TrackMeNot] *offers* control directly to those most motivated to seek reform, providing a relatively near-term even if imperfect solution." [11] We disagree with this argument. In practice, obfuscators are often marketed to users as an effective privacy protection. Users who care about the privacy of their data are told that installing the software will "significantly increas[e] the difficulty of aggregating [their queries] into accurate or identifying user profiles",[1] giving users a false sense of security. Similarly, Viejo et al state that their obfuscator "can be considered a proper option for those users who are concerned about their privacy." [13]

A. Our Contributions

In this paper, we introduce a fully adversarial formulation of privacy for query obfuscators, along with three privacy requirements. Building on existing literature, we formalize two requirements of obfuscators, which we call *indistinguishability* and *coverage*. We then introduce a third requirement, *imprecision*, relating to attackers with more knowledge of the behavior of the user. This last requirement measures the robustness of the defenses provided by the obfuscator to additional auxiliary knowledge (the topic of the query), and is required to make the obfuscator "future-proof". We formalize and quantify these requirements through attack models, and show the existence of optimal adversaries for each attack model.

We use our framework to derive important properties of query obfuscators. First, we prove that indistinguishability is a necessary requirement for the two others, and imposes strong bounds on the coverage and imprecision achievable by an obfuscator (Theorem 1). Second, we show that a tradeoff exists between coverage and imprecision, inherently limiting the strength and robustness of the privacy guarantees that an obfuscator can provide (Theorem 2).

We then introduce a family of obfuscators, called *key-based obfuscators*, for which one can prove perfect indistinguishability under some conditions (Proposition 5). We instantiate a particular class of key-based obfuscators, which we call $k-$ball obfuscators, which aim at distributing queries in the query space in order to achieve a satisfactory and configurable balance between coverage and imprecision. We implement a $k-$ball obfuscator in a simple setup and show empirically that (1) the user behavior limits the privacy that can be guaranteed by an obfuscator, and (2) the coverage-imprecision tradeoff is not tight in practice: there is a significant gap between the optimal guarantees and those that can be achieved in practice.

Our results show that there are fundamental limitations to the robustness and quality of the privacy provided by obfuscation, as obfuscator designers must choose between good privacy (coverage) and robustness to auxiliary knowledge (imprecision). Further, our empirical analysis shows that this coverage-imprecision tradeoff is even harder to meet in practical implementations. Taken together, our results suggest that developing obfuscators with provable robust, "future-proof" privacy guarantees is challenging in practice.

[1] According to https://www.trackmenot.io/

II. Obfuscator and Adversary Model

A. Setup and Assumptions

We consider a service S, to which a *user* sends *queries* taken from a countable set $\mathcal{Q}$. Such services can be, e.g., search engines where, the queries consist of the text the user is searching for, DNS servers resolving IP addresses from hostnames, or Internet Service Providers delivering general Internet traffic, where the queries are requests in specific protocols. Obfuscators aim to protect the user queries from an *adversary*, which can be the service provider (search engine, websites, ...) or an eavesdropper (third-party trackers, surveillance agency, ...).

A *block obfuscator* is a program installed by the user that sends a *block* of artificial queries along with every user query and filters the responses received to keep only the one related to the true query of the user. The obfuscator is user-centric, i.e., tailored to the user profile. We consider the following idealized setting:

1) The obfuscator works by sending a block of N *different* queries upon every user query, and that block always contains the genuine user query;
2) The adversary knows that the queries come from an obfuscator and how this obfuscator works, as well as the block size N;
3) The adversary knows the profile of the user (i.e., distribution of their queries);
4) The adversary is willing to allocate (unbounded) resources to cancel out the effect of the obfuscator;
5) The adversary is *honest-but-curious*: they only observe the block of queries and cannot interfere with the obfuscator.

Assumption 1) describes the model we use for obfuscators. We assume that an obfuscator will generate fake queries only when the user sends a query, and will then generate a block of N unique queries, containing the user query, and send it to the adversary. This assumption prevents the adversary from analyzing the timing of the queries to identify the original one. Note that, as part of this assumption, we do not consider methods that alter the true query.

Assumption 2) requires the obfuscator to be publicly available (i.e. we do not consider "*privacy by obscurity*", in accordance with the Kerckhoff principle [31]). The adversary knowing the block size N is a consequence of the block model, as all queries of a block are sent simultaneously.

Assumption 3) considers a worst-case scenario for the obfuscator, where the adversary knows the user and their query patterns. This means that our requirements are independent from the specific knowledge that the adversary has on the user, and remain valid against adversaries with less information.

Assumption 4) further states that the adversary will try to reverse the effect of the obfuscation, in order to obtain information on the user query. This is essential for robust guarantees, and assuming the contrary means that the obfuscator

lures users to a false sense of security (i.e. they are safe as long as no one tries to attack the system). We assume the attacker to have unbounded computational power.

Assumption 5) limits the capabilities of the adversary: they only observe the block of queries and cannot influence the generation of the obfuscated block. This assumption is natural in the setup we assume, where Web services are not able to run code on the machine of the user.

Note that our paper assumes an idealized setting for the obfuscator, where the attacker has access to only one query. In practice, an adversary could perform more powerful attacks exploiting the correlation between successive user queries, or with access to information pertaining to post-query behavior, e.g. recording where the user clicked. In section VII-D, we discuss how our model can be extended to account for correlations.

B. Formal Model

Fig. 1 summarizes our model and notations. Let $\mathcal{Q}$ be a countable set from which all queries are drawn (called the *query space* or *query set*). We denote by $\Delta(\mathcal{X})$ the set of distributions over a set $\mathcal{X}$. For a random function $\mathcal{F} : A \to \Delta(B)$, we will denote by $\mathcal{F}(a)$ the random variable taking values in B obtained by calling the function with argument $a \in A$. We write T_V the distribution of a random variable V, and $T_{V|U}$ the conditional distribution of a random variable V to random variable U. We write $\text{supp}(T_V)$ the support of a distribution T_V.

Our model considers one user query in isolation albeit it can be extended for a stream of queries. The user query is a random variable X taking values in $\mathcal{Q}$ according to some known distribution $T_X \in \Delta(\mathcal{Q})$. The obfuscator is a random function of the user query $\mathcal{O} : \mathcal{Q} \to \Delta(\mathcal{Q}^N)$ such that its result always contains that query. The *block* of queries sent to the adversary is the image of the user query by the obfuscator, and we denote it by $Z := \mathcal{O}(X)$. This allows us to equivalently write the obfuscator as the distribution $T_{Z|X}(z|x)$. As per assumptions 2) and 3), both $T_{Z|X}$ and T_X are known to the adversary. We also define the *index* I, a random variable that gives the position of the user query X in the block Z: $X = Z_I$ always holds. By assumption 1), it is well defined, since X is guaranteed to be in the block, and each query is unique in the block. Without loss of generality, we assume that the index variable I is *uniformly distributed*. This means that no position in the block is more likely a priori to be the user query. An important part of assumption (1) is that the queries in the block are *unique*: $\forall z \in \mathcal{Q}^N, T_Z(z) > 0, \forall i \neq j : z_i \neq z_j$.

C. Topics

In order to model semantic proximity between queries in $\mathcal{Q}$, we assume that a proper semantic distance d_S over $\mathcal{Q}$ exists. We define *topics* as closed balls of radius R in $\mathcal{Q}$: $B[x, R] = \{y \in \mathcal{Q} : d_S(x, y) \leq R\}$, where R represents the "specificity" of a topic. This gives a versatile definition of topics, as it naturally allows for various levels of specificity and overlap between topics. We do not define this semantic distance precisely, as it is not required by our argumentation. Our results hold for any choice of distance metric over $\mathcal{Q}$. This distance should be capable of measuring whether two queries relate to the same topic and have similar meaning. The choice of distance impacts the meaning of the privacy guarantees, and is thus part of the design of the obfuscator (i.e., an obfuscator will define the privacy it gives according to some particular distance).

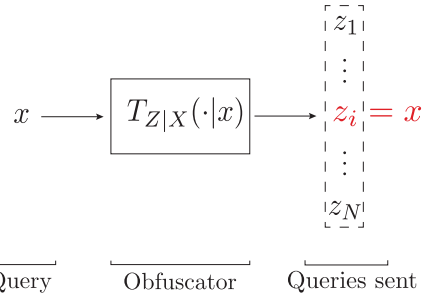

Fig. 1. *Model and notations for a block obfuscator*. x is the user query, from which a block z is generated by the obfuscator. i is the index of the user query, such that $z_i = x$. We here use lowercase letters to denote a realization of the variables.

D. Privacy Games

In the next section, we formulate our privacy requirements as *games* played against an adversary $\mathcal{M}$. We assume the following setting: the adversary receives a block of queries Z, and potentially some side information Q, and runs a probabilistic algorithm $\mathcal{M}$ in the goal of obtaining some information on the query. The nature of Q and the target information depend on the game. The games are defined for $\mathcal{O}$, an obfuscator of block size N on some countable query space $\mathcal{Q}$.

III. Privacy Requirements

We propose three necessary adversarial privacy requirements for block obfuscators. Our first two requirements build on metrics proposed in the literature, and state that an obfuscator should hide the user query and its intent. We then introduce a third, novel, requirement that imposes the obfuscator to be robust against additional auxiliary knowledge that an attacker could have access to.

A. Indistinguishability

The artificial queries generated by an obfuscator should be reasonably similar to the user query, so that one can't easily filter them out. Otherwise, the obfuscator is pointless: the noise it generates can be canceled out, and the adversary observes the user query in clear. In previous work, Balsa et al. [30] require artificial queries to be "indistinguishable" from the user query, and propose to measure the probability that an abstract "Dummy Classification Algorithm", run by the adversary, succeeds in identifying the user queries. Gervais et al. propose a similar idea, by measuring the accuracy of a specific classifier [28].

Building on this, the first requirement we propose is *indistinguishability* (Fig. 2): the true query should not be "easy" to distinguish from the artificially generated queries for any adversary without additional information. All queries in the block must be reasonably likely to have been written by the user. Intuitively, this requirement amounts to having the artificial queries be "sufficiently realistic" to user queries. Parallels can be drawn between indistinguishability and the

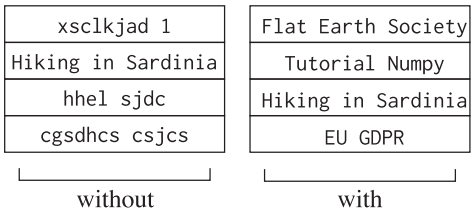

Fig. 2. *The indistinguishability requirement*: a possible block of web search queries resulting from an obfuscator (left) without the requirement; (right) with the requirement. In this example, the obfuscator on the left generates nonsensical queries, making them easy to distinguish from the true query, whereas all queries on the right look plausible in the absence of additional information.

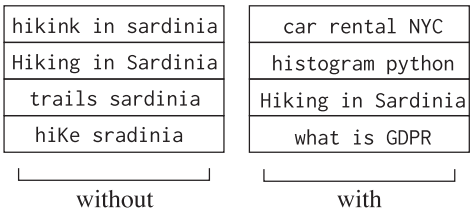

Fig. 3. *The coverage requirement*: a possible block of web search queries resulting from an obfuscator (left) without the requirement; (right) with the requirement. All queries on the left are variations of the user query, whose intent is thus easy to identify. On the right, the queries, all plausible user queries, cover different topics, making it harder to guess the intent of the user.

$k-$anonymity [32] privacy definition for datasets, in the sense that the real query ("record") is similar to $k-1$ others.

Privacy game: We formalize the intuitive description of indistinguishability through the following attack model. Let $\mathcal{M} : \mathcal{Q}^N \rightarrow \Delta(\{1, \ldots, N\})$ be an adversary. The adversary wins the indistinguishability game (i.e. $WinIND^{\mathcal{O}}_{\mathcal{M}}$ is true) if, given the block Z, they can identify the user query without additional information.

1) Let X be a user query;
2) Let $Z = \mathcal{O}(X)$ be the query block;
3) $WinIND^{\mathcal{O}}_{\mathcal{M}} = \left(Z_{\mathcal{M}(Z)} = X\right)$

Note that, by assumption, queries in the block are unique, so the last line can be equivalently written as $WinIND^{\mathcal{O}}_{\mathcal{M}} = (\mathcal{M}(Z) = I)$, where I is the index variable.

B. Coverage

Indistinguishability by itself is not sufficient to guarantee privacy – it protects the query itself, but not its semantics. For instance, one could build an obfuscator that replicates the user query and adds small typos, hence generating many different queries. While this makes the exact genuine query hard to identify, this obfuscator would not prevent an adversary from learning the intent of the user. This situation is illustrated by Figure 3 (left), where an obfuscator produces slightly modified versions of the user query "Hiking in Sardinia". Intuitively, the artificial queries should also protect the *intent* of the true query.

Many obfuscators developed in the literature are built with a constraint whose goal is to hide the meaning of the user query. Wang et al. propose to mask "topical intent", by requiring that the confidence of the adversary on the interest of the user in a topic does not increase too much after observing the block [33]. Viejo et al. use the (defunct) DMOZ project to define a topical tree, and require that queries in the block share parents at least at some distance up the tree [13]. Ahmad et al. similarly rely on a topic tree and language model to define semantics, and require that queries in the block come from different branches of the tree, with similar specificity within their topic [18], [19]. Murugesan and Clifton et al. specify "diversity" as a design principle for obfuscators, requiring that the queries in the block are topically diverse (as measured by a semantic distance) [14].

Our second requirement, which we call *coverage*, is more general: the topic of the user query should be "hard" to infer from the block by an adversary without additional information. That is, the obfuscator should be able to hide the intent of the user from the adversary. We choose the name coverage as it entails that an obfuscator should *cover* various topics. Following the parallel with $k-$anonymity, coverage can be seen as similar to $l-$diversity [34]: in addition to being indistinguishable from $k-1$ others, there must be diversity in the queries' topic (at least l topics).

Privacy game: We formalize the intuitive description of coverage through the following attack model. Let $\mathcal{M} : \mathcal{Q}^N \rightarrow \Delta(\mathcal{Q})$ be an adversary. The adversary wins the coverage game for some specificity R_{cov} if they can find a topic close to that of the user query.

1) Let X be a user query;
2) Let $Z = \mathcal{O}(X)$ be the query block;
3) $WinCOV^{\mathcal{O},R_{cov}}_{\mathcal{M}} = (d_S(\mathcal{M}(Z), X) \leq R_{cov})$

C. Imprecision

Indistinguishability and coverage protect the query and its meaning against an adversary that knows the user distribution, but has no information on the query outside of the block. As such, the guarantees they provide are not *future-proof*: should the adversary obtain additional information on the queries, both guarantees could cease to hold.

We introduce a novel requirement, which we call *imprecision*: the true query should be "hard" to find even if the adversary has prior information on the topic of the user query. Indeed, it can happen that the attacker knows the topic of the user query, and tries to identify the query using this additional piece of knowledge. This is illustrated in Fig. 4, where queries that are very topically diverse fail to protect the user query, given that an adversary knows its broad topic ("traveling in Europe"). We choose the name imprecision as it means the query must be imprecise within its topic. This third requirement is needed to ensure that the privacy offered by the obfuscator is "future-proof", i.e., robust to stronger attackers who gain access to additional information on the query.

While this requirement is novel in the context of query obfuscation, similar ideas have been used in the field of *program obfuscation*, where the code of a program is transformed into a semantically identical program that is hard to understand [35]. Goldwasser et al. showed, e.g., that obfuscating a program is hard in general when the attacker has access to prior information [36].

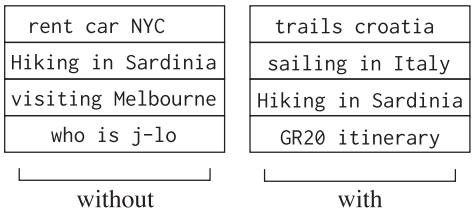

Fig. 4. *The imprecision requirement*: a possible block of web search queries resulting from an obfuscator (left) without the requirement; (right) with the requirement. Knowing that the user is interested in "traveling in Europe" makes the query easy to identify on the left, but does not help on the right, where all queries are related to this topic. Note that this is informal as, in general, the block would have to be designed to account for a set of plausible topics that the user query is part of.

Privacy game: Similarly to the two former requirements, we formalize imprecision as the following attack model. Let $\mathcal{M} : \mathcal{Q}^N \times \mathcal{Q} \to \Delta(\{1, \ldots, N\})$ be an adversary (with access to additional information). The adversary wins the game for some specificity R_{imp} if they can identify the user query, given a query (and hence, the topic of size R_{imp} centered on this query) close to it.

1) Let X be a user query;
2) Let $Z = \mathcal{O}(X)$ be the query block;
3) Let $Q \leftarrow B[X, R_{imp}]$ be the additional information;
4) $WinIMP_{\mathcal{M}}^{\mathcal{O},R_{imp}} = (Z_{\mathcal{M}(Z,Q)} = X)$

IV. Meeting the Requirements

Having formally defined each requirement through an attack model, we here develop a rigorous probabilistic definition for these requirements. We then prove important theoretical properties of our model, relating indistinguishability to the other requirements.

A. Probabilistic Formulation

We say that an obfuscator meets a requirement with some level α if *the probability that any adversary wins the corresponding game is less than α, for every possible adversary*. This probability is taken over (1) the user query ($\sim T_X$), (2) the obfuscator ($\sim T_{Z|X}$) and (3) randomness in the algorithm of the adversary. The games are all defined for one user of distribution T_X.

Let $\mathcal{O}$ be an obfuscator of block size N, and $R_{imp}, R_{cov} \in \mathbb{R}_0^+$. We define $\alpha_{ind}, \alpha_{cov}$ and α_{imp} such that $\mathcal{O}$ provides $\alpha_{ind}-$indistinguishability, $(\alpha_{cov}, R_{cov})-$coverage and $(\alpha_{imp}, R_{imp})-$imprecision as:

$$\alpha_{ind} = \sup_{\mathcal{M}:\mathcal{Q}^N \to \Delta(\{1,\ldots,N\})} \mathbb{P}\left[WinIND_{\mathcal{M}}^{\mathcal{O}}\right]$$
$$\alpha_{cov} = \sup_{\mathcal{M}:\mathcal{Q}^N \to \Delta(\mathcal{Q})} \mathbb{P}\left[WinCOV_{\mathcal{M}}^{\mathcal{O},R_{cov}}\right]$$
$$\alpha_{imp} = \sup_{\mathcal{M}:\mathcal{Q}^N \times \mathcal{Q} \to \Delta(\{1,\ldots,N\})} \mathbb{P}\left[WinIMP_{\mathcal{M}}^{\mathcal{O},R_{imp}}\right]$$

This formulation implies that no adversary can win at the indistinguishability (resp. imprecision, coverage) game with a probability higher than α_{ind} (resp. α_{imp}, α_{cov}).

Proposition 1 gives a lower bound to α_{ind}: for all block obfuscators, we must have that $\alpha_{ind} \geq 1/N$. Hence, we will say that an obfuscator provides *perfect indistinguishability* if it matches this lower bound, i.e. it is such that $\alpha_{ind} = 1/N$. In practice, this means that all queries in a block are equally likely to be the user query (and thus, all adversaries will perform as well as random selection).

Proposition 1 (Lower Bound on Indistinguishability): Let $\mathcal{Q}$ be a countable query space and $\mathcal{O}$ a block obfuscator over $\mathcal{Q}$ of block size $N \in \mathbb{N}_0$ that provides $\alpha_{ind}-$indistinguishability. Then:

$$\frac{1}{N} \leq \alpha_{ind}$$

Proof: Define the "random" adversary $\mathcal{M}^R$ for the indistinguishability game that selects a query uniformly at random:

$$\mathcal{M}^R(Z) \sim Unif(\{1, \ldots, N\})$$

We have that (with the probability taken over Z, I and $\mathcal{M}^R(Z)$):

$$\mathbb{P}\left[WinIND_{\mathcal{M}^R}^{\mathcal{O}}\right] = \mathbb{P}\left[\mathcal{M}^R(Z) = I\right] = \frac{1}{N}$$

And thus, by definition of α_{ind},

$$\alpha_{ind} \geq \mathbb{P}\left[WinIND_{\mathcal{M}^R}^{\mathcal{O}}\right] = 1/N$$

□

B. Optimal Adversaries

It is possible to find *optimal adversaries* for our attack models, whose probability of success exactly matches the corresponding α_* (for α_*-*guarantee*). Proposition 2 shows that at least one optimal adversary exists for indistinguishability. In practice, this adversary is hard to compute exactly, as it requires to know the exact user distribution T_X and to be able to compute the obfuscator distribution $T_{Z|X}$. This is however a powerful mathematical tool to evaluate properties of obfuscators.

Proposition 2 (Existence of an Optimal Adversary for Indistinguishability): Let $\mathcal{Q}$ be a query space, and an obfuscator $\mathcal{O}$ of block size $N \in \mathbb{N}_0$ providing $\alpha_{ind}-$indistinguishability. Any adversary $\mathcal{M} : \mathcal{Q}^N \mapsto \Delta(\{1, \ldots, N\})$ is such that:

$$\mathbb{P}\left[WinIND_{\mathcal{M}}^{\mathcal{O}}\right] \leq \mathbb{P}\left[WinIND_{\mathcal{M}*}^{\mathcal{O}}\right]$$

where $\mathcal{M}^*$ (the optimal adversary) is defined as:

$$\mathcal{M}^*(z) \in \arg\max_i T_{I|Z}(i|z)$$

Proof: Developing conditionally to Z and I for $\mathcal{M}$ gives:

$$\mathbb{P}\left[WinIND_{\mathcal{M}}^{\mathcal{O}}\right] = \sum_{z \in \mathcal{Q}} T_Z(z) \sum_{i=1}^{N} T_{I|Z}(i|z) \cdot \mathbb{P}\left[\mathcal{M}(z) = i\right]$$

Denote $p_i^{\mathcal{M}}(z) = \mathbb{P}\left[\mathcal{M}(z) = i\right]$. Now, observe that $\sum_{i=1}^{N} T_{I|Z}(i|z) \cdot p_i^{\mathcal{M}}(z)$ is a convex combination of $T_{I|Z}(i|z)$, and thus, $\forall z \in \mathcal{Q}$:

$$\sum_{i=1}^{N} T_{I|Z}(i|z) \cdot p_i^{\mathcal{M}}(z) \leq \max_i T_{I|Z}(i|z)$$

We have $\forall z \in \mathcal{Q}$, $p_i^{\mathcal{M}^*}(z) = I\{i = \arg\max_j T_{I|Z}(j|z)\}$ by definition, and thus:

$$\sum_{i=1}^{N} T_{I|Z}(i|z) \cdot p_i^{\mathcal{M}^*}(z) = \max_i T_{I|Z}(i|z)$$

Hence, $\mathcal{M}^*$ achieves the upper bound, which concludes the proof. □

Under perfect indistinguishability, we can also find closed-form optimal adversaries for coverage and imprecision. Proofs for these propositions are in the appendix.

Proposition 3: Let $\mathcal{Q}$ be a query space, and an obfuscator $\mathcal{O}$ of block size $N \in \mathbb{N}_0$ providing $1/N-$indistinguishability. Any adversary $\mathcal{M} : \mathcal{Q}^N \mapsto \Delta(\mathcal{Q})$ is such that:

$$\mathbb{P}\left[WinCOV_{\mathcal{M}}^{\mathcal{O},R_{cov}}\right] \leq \mathbb{P}\left[WinCOV_{\mathcal{M}_C^*}^{\mathcal{O},R_{cov}}\right]$$

where $\mathcal{M}_C^*$ (the optimal adversary) is defined as:

$$\mathcal{M}_C^*(z) \in \arg\max_{r \in \mathcal{Q}} |B[r, R_{cov}] \cap z|$$

Proposition 4: Let $\mathcal{Q}$ be a query space, and an obfuscator $\mathcal{O}$ of block size $N \in \mathbb{N}_0$ providing $1/N-$indistinguishability. Any adversary $\mathcal{M} : \mathcal{Q}^N \times \mathcal{Q} \mapsto \Delta(\{1, \ldots, N\})$ is such that:

$$\mathbb{P}\left[WinIMP_{\mathcal{M}}^{\mathcal{O},R_{imp}}\right] \leq \mathbb{P}\left[WinIMP_{\mathcal{M}_I^*}^{\mathcal{O},R_{imp}}\right]$$

where $\mathcal{M}_I^*$ (the optimal adversary) is defined as:

$$\mathcal{M}_I^*(z, q) \in \arg\min_{\{i : d_S(z_i, q) \leq R_{imp}\}} \left|B[z_i, R_{imp}]\right|$$

C. Indistinguishability Is the Weakest Requirement

Theorem 1 shows that if an obfuscator does not provide some degree of *indistinguishability*, it will fail at guaranteeing any form of privacy. Formally, we show that α_{ind} is a lower bound to α_{cov} and α_{imp}. Hence, if there exists an attacker that can identify the true query with probability at least α_{ind}, there exist attackers for coverage and imprecision whose probability of success will be at least α_{ind}.

Theorem 1: Let $\mathcal{Q}$ be a countable query set with a given semantic distance d_S over $\mathcal{Q}$. Every obfuscator providing $\alpha_{ind}-$indistinguishability, $\alpha_{cov}-$coverage and $\alpha_{imp}-$imprecision satisfies:

$$\alpha_{ind} \leq \alpha_{cov}$$
$$\alpha_{ind} \leq \alpha_{imp}$$

Proof: Let $\mathcal{M}^*$ be an optimal adversary for indistinguishability, whose existence is guaranteed by proposition 2. We define an adversary $\mathcal{M}^C$ for the coverage game that selects as topic the topic of the best guess query:

$$\mathcal{M}^C(z) = z_{\mathcal{M}^*(z)}$$

Observe that, for any given block z, when $\mathcal{M}^*$ is correct, then so is $\mathcal{M}^C$. Indeed, for every block on which $\mathcal{M}^*$ correctly guesses the user query, then $\mathcal{M}^C$ returns a topic containing the user query. That is, the following always holds ($\forall z \in \mathcal{Q}^N$):

$$WinIND_{\mathcal{M}^*}^{\mathcal{O}} \Rightarrow WinCOV_{\mathcal{M}^C}^{\mathcal{O},R_{cov}}$$

Which implies, by definition of α_{ind} and α_{cov}:

$$\alpha_{ind} = \mathbb{P}\left[WinIND_{\mathcal{M}^*}^{\mathcal{O}}\right] \leq \mathbb{P}\left[WinCOV_{\mathcal{M}^C}^{\mathcal{O},R_{cov}}\right] \leq \alpha_{cov}$$

Similarly, we define another adversary $\mathcal{M}^I$ for the imprecision game, that selects as user query the best guess:

$$\mathcal{M}^I(z, q) = \mathcal{M}^*(z)$$

For a given block $z \in \mathcal{Q}^N$, $\mathcal{M}^I$ is correct if and only if $\mathcal{M}$ is (as it selects the same, correct, query). In other words, $\forall z \in \mathcal{Q}^N$:

$$WinIND_{\mathcal{M}^*}^{\mathcal{O}} \Leftrightarrow WinIMP_{\mathcal{M}^I}^{\mathcal{O},R_{imp}}$$

And thus:

$$\alpha_{ind} = \mathbb{P}\left[WinIND_{\mathcal{O}}^{\mathcal{M}^*}\right] = \mathbb{P}\left[WinIMP_{\mathcal{M}^I}^{\mathcal{O},R_{imp}}\right] \leq \alpha_{imp}$$

□

Theorem 1 shows that the coverage and imprecision of an obfuscator admit as a lower bound its indistinguishability. Intuitively, this means that if an adversary is able to re-identify the user query with reasonable certainty, that adversary can also break the other requirements. Indeed, if the true query is easy to estimate, then the obfuscator is useless: the adversary can filter out the artificial queries. Indistinguishability is a *key requirement* for the obfuscator to provide privacy guarantees. As a consequence of proposition 1, this also implies that $\alpha_{cov} \geq 1/N$, and $\alpha_{imp} \geq 1/N$.

V. Coverage-Imprecision Tradeoff

The requirements we introduced in the previous sections are described by a few parameters: the probabilities of successful attacks $(\alpha_{ind}, \alpha_{cov}, \alpha_{imp})$ of the guarantees and the topic specificities (R_{cov} and R_{imp}). These parameters describe formally the nature of the guarantees given by the obfuscator. Ideally, one would like that no attacker can do better than random for any of our attack models, i.e. $\alpha_{ind} = \alpha_{cov} = \alpha_{imp} = 1/N$, meaning that the bounds given by Theorem. 1 and Proposition. 1 are tight. We show in Theorem 2 below that there are inherent limitations to the privacy that can be provided by an obfuscator, making it impossible for an obfuscator to optimally provide indistinguishability, coverage, and imprecision.

In particular, we show that in a best-case scenario ($\alpha_{ind} = 1/N$), coverage and imprecision need to be balanced one against the other: for every $1/N-$indistinguishable obfuscator that guarantees $\alpha_{cov}-$coverage and $\alpha_{imp}-$imprecision, one must have that $\alpha_{cov} \cdot \alpha_{imp} \geq 1/N$. Hence, no obfuscator can guarantee simultaneously optimally good coverage and optimally good imprecision.

Theorem 2 (Coverage-Imprecision Tradeoff): Let $\mathcal{Q}$ be a query set with a proper semantic distance d_S, a distribution T_X over $\mathcal{Q}$, and $\mathcal{O}$ an obfuscator of block size $N \in \mathbb{N}^+$ and distribution $T_{Z|X}$ that provides perfect indistinguishability, $(\alpha_{cov}, R_{cov})-$coverage for a radius $R_{cov} \in \mathbb{R}_0^+$ and $(\alpha_{imp}, R_{imp})-$imprecision for a radius $R_{imp} \in \mathbb{R}_0^+$. Then, if $R_{imp} \leq R_{cov}$, the block size N of this obfuscator must satisfy:

$$\alpha_{cov} \cdot \alpha_{imp} \geq \frac{1}{N}$$

Proof: We give below a sketch of the proof, which is presented in full in the appendix. Define the query count of a set $\mathcal{S}$ as $N_{\mathcal{S}}(Z) = |\{i \mid Z_i \in \mathcal{S}\}|$. We first show that under

perfect indistinguishability, the query index I is independent from Z, i.e. $I|Z$ is uniform (proposition 6). This implies that for every set $\mathcal{S}$ (Lemma 1):

$$\mathbb{P}[X \in \mathcal{S} \mid Z = z] = \frac{N_{\mathcal{S}}(z)}{N}$$

Since $\mathcal{O}$ provides $\alpha_{cov}-$coverage, every adversary for the coverage game wins with probability at most α_{cov}. We define one specific adversary, that selects the ball with highest query count, $\mathcal{M}_C(Z) := \arg\max_y N_{B[y,R_{cov}]}(Z)$. From Lemma 1, this obfuscator finds the user query with probability $\frac{1}{N}\max_y N_{B[y,R_{cov}]}$ (the probability that the user query is one of the $\max_y N_{B[y,R_{cov}]}$ queries in the ball). Since $\mathcal{M}_C$ wins with probability at most α_{cov}, we obtain the following expression:

$$\frac{1}{N}\mathbb{E}_Z\left[\max_y N_{B[y,R_{cov}]}(Z)\right] \leq \alpha_{cov} \tag{1}$$

Similarly, we define an adversary for the imprecision game, which selects a query at random in the topic of the user query topic: $\mathcal{M}_I(Z, Q^B) \leftarrow Z \cup B[Q^B, R_{imp}]$. The probability that this adversary finds the exact query is $N^{-1}_{B[Q^B,R_{imp}]}$. Since $\mathcal{O}$ provides $\alpha_{imp}-$imprecision, $\mathcal{M}_I$ wins with probability at most α_{imp}, which gives:

$$\mathbb{E}_{Z,Q^B}\left[\left(N_{B[Q^B,R_{imp}]}(Z)\right)^{-1}\right] \leq \alpha_{imp}$$

By Jensen's inequality:

$$\mathbb{E}_{Z,Q^B}\left[N_{B[Q^B,R_{imp}]}(Z)\right]^{-1} \leq \mathbb{E}_{Z,Q^B}\left[\left(N_{B[Q^B,R_{imp}]}(Z)\right)^{-1}\right] \tag{2}$$

Since $R_{imp} \leq R_{cov}$, we have that

$$\mathbb{E}_{Z,Q^B}\left[\left(N_{B[Q^B,R_{imp}]}(Z)\right)\right] \leq \mathbb{E}_Z\left[\max_y N_{B[y,R_{cov}]}(Z)\right].$$

This allows to group equations 1 and 2 to obtain:

$$\frac{1}{\alpha_{imp}} \leq \mathbb{E}_Z\left[\max_y N_{B[y,R_{cov}]}(Z)\right] \leq \alpha_{cov} N$$

□

Theorem 2 shows a fundamental incompatibility in the privacy requirements we identified for obfuscators. Even assuming that artificial queries are indistinguishable from user queries, there are limits to the extent one can jointly protect the *intent of the user query* (coverage) and *the query itself, given some information on its topic* (imprecision), as the block size is in practice limited by broadband and query rate constraint.

Further, from theorem 1, we know that $\alpha_{cov}, \alpha_{imp} \geq \alpha_{ind} = 1/N$. Hence, the strongest form of the requirements are mutually exclusive: an obfuscator that ensures perfect coverage ($\alpha_{cov} = 1/N$) will be left undefended to imprecision attacks ($\alpha_{imp} = 1$), and vice versa.

Recall that coverage requires that the obfuscator protects the topic of the user query, while imprecision guarantees some level of robustness against auxiliary knowledge. This tradeoff means that users must choose between the strength of their privacy protection, and this protection withstanding an attacker with some information on a query. In some sense, this means that there is "no free lunch" for obfuscation privacy: no obfuscator will protect optimally the queries now, and forever.

Assumptions: Theorem 2 relies on two important assumptions. Firstly, it requires that the obfuscator provides perfect indistinguishability. This assumption places the obfuscator in a "best-case scenario", in that the adversary can *not* exploit the likelihood of queries to be genuine for the coverage or imprecision attack. Secondly, the theorem requires that the topic specificity of coverage and imprecision are ordered: $R_{cov} \geq R_{imp}$. We argue that this ordering is natural: R_{cov} describes the most specific topic we want to protect, while R_{imp} describes the least specific topic that the adversary can know without being able to identify the user query.

VI. Empirical Illustration

We illustrate our theoretical results empirically in a simple toy setup. We first introduce *key-based obfuscators*, a family of obfuscators for which one can prove perfect indistinguishability (Proposition 5). We then propose a key-based obfuscator, the $k-$ball obfuscator, with parameters that can be adjusted to achieve a wide range of imprecision or coverage. Finally, we apply this obfuscator to a simple setup ($\mathcal{Q} \subset [0,1]^2$, $|\mathcal{Q}| = 10^6$), and compute its coverage and imprecision guarantees.

A. Key-Based Obfuscators

Key-based obfuscators generate queries by first sampling a *key* C in some set $\mathcal{C}$, then generating $N-1$ queries based on this key. The key conditions the dependency of artificial queries Y on the user query X. In order to ensure query uniqueness within a block, for a choice of C, blocks are drawn repeatedly until all queries sampled differ from each other and from X (rejection sampling).

Definition 1 (Key-Based Obfuscator): *A* key-based obfuscator *over $\mathcal{Q}$ is an obfuscator defined by a triplet $\langle \mathcal{C}, T_{C|X}, T_{Y|C} \rangle$ composed of a set of keys $\mathcal{C}$, a key-generating distribution $T_{C|X}$, and a keyed query-generating distribution $T_{Y|C}$. The obfuscator generates queries as, given the user query $x \in \mathcal{Q}$ and a block size N:*

$$c \leftarrow_{T_{C|X}(\circ|x)} \mathcal{C}$$

$$y_1, \ldots, y_{N-1} \leftarrow_{T_{Y|C}(\circ|c),\ i.i.d.} \mathcal{Q}\ s.t.\ y_i \neq x \wedge \forall i \neq j, y_i \neq y_j$$

$$Z = shuffle\,(x, y_1, \ldots, y_{N-1})$$

Equivalently, for an index $I \leftarrow \{1, \ldots, N\}$, the distribution of the obfuscator is $\forall z \in \mathcal{Q}^N, i \in \{1, \ldots, N\}$, where we define $U(x,c)$ to be the probability that all generated queries are unique and different to X, $U(x,c) := \mathbb{P}\left[\forall i \neq j : Y_i \neq Y_j \wedge Y_i \neq x\right]$ for $Y_1, \ldots, Y_{N-1} \sim T_{Y|C}(\circ|c)$:

$$T_{Z|X,I}(z|x,i) = I\{z_i = x\} \cdot \sum_{c \in \mathcal{C}} T_{C|X}(c|x) \frac{\prod_{j \neq i} T_{Y|C}(z_j|c)}{U(x,c)}$$

Intuitively, if the conditional distribution of the user query X to C is equal to that of synthetic queries $T_{Y|C}$, then the user query is indistinguishable from the other queries. We formalize this intuition in Prop. 5.

Proposition 5: Let $\mathcal{O} = \langle \mathcal{C}, T_{C|X}, T_{Y|C} \rangle$ be a key-based obfuscator over $\mathcal{Q}$ of block size N. If $\forall c \in supp\,(T_C), \forall x \in supp\,(T_X)$:

$$\frac{T_{C|X}(c|x) \cdot T_X(x)}{T_C(c)} = \frac{T_{Y|C}(x|c) \cdot U(x,c)}{\sum_{y \in \mathcal{Q}} T_{Y|C}(y|c) \cdot U(y,c)},$$

then $\mathcal{O}$ provides perfect indistinguishability.

Proof: From proposition 2, an optimal adversary for indistinguishability is given by

$$\mathcal{M}^*(z) = \arg\max_i T_{I|Z}(i|z)$$

We now show that for a $T_{I|Z}(i|z) = \frac{1}{N}$, which by proposition 1 concludes the proof. For this, we first develop $T_{Z|I}$ as:

$$\begin{aligned}
T_{Z|I}(z|i) &= \mathbb{P}\left[X = z_i \wedge Y = z_{-i}\right] \\
&= \mathbb{P}\left[X = z_i\right] \cdot \mathbb{P}\left[Y = z_{-i} | X = z_i\right] \\
&= T_X(z_i) \sum_{c \in \mathcal{C}} \mathbb{P}\left[Y = z_{-i} | C = c\right] \mathbb{P}\left[C = c | X = z_i\right] \\
&= \sum_{c \in \mathcal{C}} \underbrace{\frac{T_X(z_i) T_{C|X}(c|z_i)}{U(x,c)}}_{=, \text{ by assumption}} \prod_{j \neq i} T_{Y|C}(z_j|c) \\
&= \sum_{c \in \mathcal{C}} \overbrace{\frac{T_C(c)\, T_{Y|C}(z_i|c)}{\sum_{y \in \mathcal{Q}} T_{Y|C}(y|c) \cdot U(y,c)}} \cdot \prod_{j \neq i} T_{Y|C}(z_j|c) \\
&= f(z)
\end{aligned}$$

Hence, $T_{Z|I}(z|i) = f(z) = T_Z(z)$. Since I is uniform over $\{1, \ldots, N\}$, we find that $T_{I|Z}(i|z) = \frac{T_{Z|I}(z|i) \cdot \frac{1}{N}}{T_Z(z)} = \frac{1}{N}$, which concludes the proof. □

Proposition 5 gives a constructive framework to design perfectly indistinguishable obfuscators, assuming T_X is known. In particular, if the probability of sampling duplicate queries is negligible ($U(x,c) \approx 1$), the equation can be simplified to $T_{Y|C}(x|c) = \frac{T_{C|X}(c|x) \cdot T_X(x)}{T_C(c)}$, and allows to find a *query-generating distribution* for any *key-generating distribution* such that the resulting obfuscator guarantees indistinguishability. Other properties – coverage and imprecision – can thus be obtained by designing $T_{C|X}$.

We present a practical instance of a key-based obfuscator that can be tuned to give a wide range of coverage and precision, under the simplifying assumption that $U(x,c) \approx 1$. This obfuscator samples artificial queries in k random balls of radius ρ in $\mathcal{Q}$, one of which contains X. This follows the intuition that, for an obfuscator to achieve a balance between coverage and imprecision, it should aim at spreading the queries in $\mathcal{Q}$ while ensuring that queries are not isolated.

Definition 2 ($k-$Ball Obfuscator): The $k-$ball obfuscator of radius ρ in $\mathcal{Q}$ is the key-based obfuscator with $\mathcal{C} = \mathcal{Q}^k$, the key $C = (C_1, \ldots, C_k)$ is sampled as

$$\begin{aligned}
j &\leftarrow \{1, \ldots, k\} \\
c_j &\leftarrow B\left[x, \rho\right] \\
c_l &\leftarrow_{T_X} \mathcal{Q} \qquad \forall l = 1, \ldots, k, \quad l \neq j
\end{aligned}$$

which implies

$$T_{C|X}\left((c_1, \ldots, c_k)|x\right) = \frac{1}{k} \sum_{j=1}^{k} \frac{I\{c_j \in B[x,\rho]\}}{|B[x,\rho]|} \cdot \prod_{l \neq j} T_X(c_j)$$

and $T_{Y|C}$ defined accordingly to satisfy proposition 5.

B. Application

We illustrate our framework in a simple setting. We build a $k-$ball obfuscator over a simple query space, and show how its privacy guarantees change with the parameters (k, ρ). We validate the tradeoff of theorem 2, and show that its bound is not tight even in a simple setting.

As query space, we consider a finite, random subset of $\mathbb{R}^d$, for $d = 2$: $\mathcal{Q}_{[0,1]^2} = (P_1, \ldots, P_r) \subset \mathbb{R}^d$, with $P_i \leftarrow [0, 1[$ uniformly at random. In our experiments, we use $r = \left|\mathcal{Q}_{[0,1]^2}\right| = 10^6$. As semantic distance, we use the wrap-around L_2 distance, i.e. warping the boundaries such that $(0, x)$ and $(1, x)$ (resp. $(x, 0)$ and $(x, 1)$) are the same point $\forall x \in [0, 1]$, such that all points in the space are equivalent:

$$\begin{aligned}
&d_{L_2,\text{wrap}}(x, y) \\
&\quad = \sqrt{\sum_{i=1}^{d} \left(\min\left(|x_i - y_i|, |x_i + 1 - y_i|, |y_i + 1 - x_i|\right)\right)^2}
\end{aligned}$$

This can be seen as a simplistic approximation of text embeddings, where (discrete) texts are projected to a continuous space where the L_2 distance can be used to evaluate text similarity [37].

We take the user distribution to be uniform over the vocabulary: $T_X(x) = \left|\mathcal{Q}_{[0,1]^2}\right|^{-1}$, $\forall x \in \mathcal{Q}_{[0,1]^2}$. We then define a $k-$ball obfuscator over $\mathcal{Q}_{[0,1]^2}$, with block size $N = 10$, number of balls k ranging from 1 and 100, and radius[2] $\rho \in \left[5 \times 10^{-3}, \sqrt{2}/2\right]$.

For each choice of parameters, we build the optimal coverage and imprecision adversaries of the obfuscator (under the assumption of perfect indistinguishability, guaranteed by proposition 5), given in propositions 3 and 4. We set $R_{cov} = R_{imp} = 5 \times 10^{-2}$. We then measure the privacy characteristics α_{cov} and α_{imp} of the obfuscator as the probability for the optimal adversary to successfully attack the obfuscator. For this, we sample user queries X, generate the obfuscated query blocks Z with the obfuscator, and use the optimal adversaries in the Coverage and Imprecision attack models. We repeat the experiment 10,000 times for each choice of parameters (k, ρ) to get reliable estimates of the success probability for both attacks.

In figure 5, we present the coverage-imprecision curves obtained for different values of the ball radius ρ and number of balls k. Each point represents the success rate of the optimal coverage and imprecision obfuscators for an obfuscator, defined by its parameters (ρ, k). These points are grouped by radius ρ in curves, with varying free parameter k. Note that for $\rho = \frac{\sqrt{2}}{2}$, since $B[x, \frac{\sqrt{2}}{2}] = \mathcal{Q}_{[0,1]^2}$, the number of balls k does not matter (all balls are identical), and the curve is just a single point. In this case, the obfuscator samples artificial queries from $\mathcal{Q}_{[0,1]^2}$ according to the user distribution T_X, and is thus uniquely conditioned by the user behavior.

Intuitively, as the number of balls k increases, queries become less concentrated and more dispersed in the query space. Our results indeed show that, as the number of balls k increases, the obfuscator becomes less vulnerable to coverage attacks, but more vulnerable to imprecision attacks. Further,

[2]The value $\sqrt{2}/2$ is the largest possible wrapped-L_2 distance between two points of $\mathcal{Q}_{[0,1]^2}$.

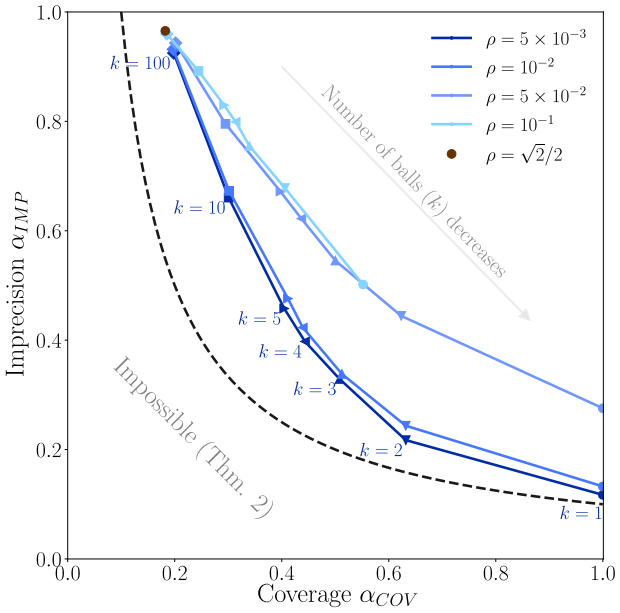

Fig. 5. **Imprecision-Coverage tradeoff in a toy setting**: Each point represents the coverage and imprecision of one $k-$ball obfuscator, for different values of the number of balls k and their radius ρ, over a discrete sample of $[0, 1)^2$ of size $|\mathcal{Q}_{[0,1]^2}| = 10^6$. Curves group points for obfuscators with the same radius ρ, and decreasing number of balls k. The value of k used for a specific point is given by its marker. All values for $\rho = \sqrt{2}/2$ are aggregated into one point, since the obfuscator is identical for all values of k. The zone below the dashed line contains values of $(\alpha_{cov}, \alpha_{imp})$ that are impossible according to Thm. 2 (such that $\alpha_{cov} \cdot \alpha_{imp} < \frac{1}{N}$). We observe that the theoretical tradeoff holds, but the bound it puts on coverage and imprecision is not tight.

we observe that the obfuscator performs better for radii ρ much smaller than the coverage and imprecision radii $R_{cov} = R_{imp}$. Hence, $k-$ball obfuscators are more resilient to attacks for smaller values of the radius, and the number of balls can then be used to change the balance between coverage and imprecision.

Finally, we see that the tradeoff of theorem 2 holds, but it is not tight for this obfuscator and user distribution. Indeed, there is a significant margin, especially for lower values of α_{cov}. The closest the obfuscator gets to the $\alpha_{cov}\alpha_{imp} = \frac{1}{N}$ line is for $k = 1$ and $\rho = 5 \times 10^{-3}$, where all artificial queries are located in a small ball including the user query. In such a setup, the obfuscator offers almost perfect imprecision – given the topic, all queries are equally likely – but no coverage, since the query topic can be immediately inferred.

VII. Discussion

A. Alternative Definitions of Privacy

Many obfuscators in the literature are designed around information-theoretic metrics, with the aim to provably prevent any information leakage on the user or query to the adversary. These typically involve measuring the distance between the distribution of the user query (input) T_X and the distribution of a query in the block (output) T_{Z_j}, or some target distribution. This distance is measured with mutual information [15], Kullback-Leibler divergence [16], equivocation [30], or cosine similarity [12]. Domingo-Ferrer also proposed to measure the entropy of the output of the obfuscator [17]. A strong limitation of such metrics is that the practical privacy protection they provide (i.e., limits on information leakage) is hard to understand, especially in an adversarial setting. For instance, the mutual information between the obfuscated block Z and the user query X is not a good indicator of what the adversary can actually learn from the block [15]. Additionally, these measures also obscure the difficulty of sampling from distributions in large query spaces, such as text queries to a search engine.

Obfuscators based on information-theoretic metrics over distributions typically aim to prevent the adversary from building a "profile" of the interests of the user, capturing what Gervais et al. call 'semantic privacy" [28]. Since the query set is untractably large, this profile is usually defined as a multinomial distribution over a predefined set of topics [15], [16], [30] or over the set of keywords [12], [17]. In a similar context, Mac Aonghusa et al. [38] proposed to measure obfuscator privacy as plausible deniability over a set of sensitive topics, by observing the advertisement offered by the search engine, and reporting those relating to a sensitive topic. When trying to prevent profiling, these methods make assumptions on the type of profile the adversary is trying to build, as well as on the auxiliary information available to them. In doing so, they can underestimate the risks of a real-world adversary. For this reason, our framework considers a worst-case attacker, by explicitly assuming that the adversary knows the exact user profile T_X (assumption 3).

Finally, the idea of burying sensitive information among "chaff" entries has been used extensively as a building block of cryptographic tools. A prominent example is Oblivious RAM (ORAM) [39], where the memory entries used by a sensitive program are hidden in a large set of bogus lookups. While this research is not immediately applicable to our setup, definitions and tools from it could help inform the design and implementation of obfuscators.

B. Alternatives to Query Obfuscation

Solutions to query a service while protecting the privacy of the user broadly fall in three categories: *server-based*, *network-based*, and *user-based*.

Server-based solutions rely on a specific protocol implemented on the server, which guarantees limited information leakage on the user query. The most famous example of such solutions are Private Information Retrieval (PIR) methods [40], which allow users to query a database with information theoretic guarantees that the server cannot identify which record they queried. However, these methods usually do not scale for very large query spaces.

Network-based solutions aim at hiding the identity of the user (usually, their IP address) in communications with the server. These solutions usually rely on networks of users exchanging queries before sending them to the server, such as in onion routing [41]. Mix networks [42], where intermediate servers "mix" the queries from several users before sending them to the service, are a particularly promising alternative to obfuscation. These methods require the existence of a network that needs to be at least partially trusted, and can lead to high latencies.

User-based solutions aim at protecting privacy of the user through algorithms implemented solely on their machine. Query obfuscation is the most popular of such solutions.

Researchers also proposed query scrambling [43], [44], where the user queries are modified before being sent to the server, possibly as part of a block of artificial queries. These systems also provide techniques to recombine/filter the query results in order to provide useful answers. While interesting, query scrambling techniques assume that the results of the true query can be obtained from the modified queries, which leads to limited utility in practice.

C. Evaluating the Privacy of Existing Obfuscators

While a thorough evaluation of the privacy of previously developed methods is beyond the scope of this paper, we briefly comment on the guarantees that these methods could satisfy in our framework. General attacks to filter out artificial queries have been proposed by Petit et al. [29] and Gervais et al. [28], which can be used to evaluate upper bounds on the indistinguishability of obfuscators. These have successfully been applied to TrackMeNot [11] and GooPIR [17], suggesting these obfuscators do not provide meaningful privacy guarantees (theorem 1). Specialized attacks have further been applied to TrackMeNot [25], [26], [27], with similarly high accuracy. In their review [30], Balsa et al. describe high-level attacks against six previously developed obfuscators, either filtering artificial queries, or undoing the distortion of the user profile caused by the obfuscator (allowing the query topic to be inferred, i.e., a coverage attack). This work suggests that the obfuscators considered [11], [12], [14], [15], [16], [17] do not provide meaningful indistinguishability or coverage in practice. Finally, all obfuscators from prior work ignore the possibility of attacker auxiliary information, and thus probably do not provide meaningful imprecision. In Table I, we summarize previously published obfuscators and whether they can satisfy our requirements based on current research. For each obfuscator and requirement, we write ✓ if there is a proof that the obfuscator could satisfy the requirement (or the privacy definition used by the obfuscator implies satisfaction of the requirement), ✗ if an attack has been demonstrated against the obfuscator, ✗* if the obfuscator cannot satisfy the requirement by design, and **?** otherwise. Note, however, that most of these obfuscators are theoretical constructions, and that implementing them in practice will likely lead to degraded privacy. Indeed, pen-and-paper guarantees might not translate to resistance to attacks in practice, especially when these guarantees require the obfuscator to sample exactly from a distribution over the set of queries. Further work is needed to formally evaluate obfuscators through the development of specialized attacks, although the lack of open implementation and under-specification of these methods can make this hard in practice.

D. Extension to Multiple Queries

Our model considers one query in isolation, and highlights the challenges of robust, secure query obfuscation in this simplified setting. In practice, additional challenges stem from the fact that users are likely to issue multiple correlated queries in succession making it easier for an attacker to (possibly retroactively) identify the user queries, as discussed in [19]. Extending the framework most likely requires to make some assumptions on the behavior of the user. These assumptions will then guide the design of further games. We identify two possible extensions of our model:

1) Assuming that the user samples queries $(X_1, \ldots, X_i, \ldots)$ according to a Markovian process of order $k \geq 1$. The user behavior is then described by the distribution $T_{X_i|X_{i-1},\ldots,X_{i-k}}$. The three privacy games should then be adapted to add knowledge of prior (and, potentially, future) queries to the adversary.
2) The user query stream can be divided into independent, disjoint *search tasks* of successive queries [45]. In that case, we can then choose $\mathcal{Q}$ to be the set of search tasks, and our results apply, at least in theory. In practice, this is complex, because it requires an obfuscator generating search tasks of potentially different lengths rather than queries, as well as a robust definition of a search task.

E. Empirical Evaluation

In Section VI, we empirically evaluate the privacy guarantees of an obfuscator in an idealised context. However, our analysis is made simple by the fact that the optimal adversaries have a closed form expression that can be computed exactly. In general, auditing the privacy properties of an arbitrary obfuscator over natural language is challenging. It involves computing provably optimal adversaries, which requires to evaluate and sample from distributions over a high-dimensional space.

A solution is to *approximately* evaluate the guarantees using a large number of non-optimal attacks. By applying many attacks for each attack model, one can obtain a lower bound for the relevant privacy parameter α. This approach has been applied to evaluate the privacy of, e.g., machine learning models [46] and synthetic data [47]. Attacks from prior work in obfuscation [28], [29] can already be applied against obfuscators in the indistinguishability attack model. Further work on generic attacks that can be applied to all obfuscators is however required for this approach to give useful lower bounds.

Finally, the development of a large library of attacks against obfuscators could help to design more secure obfuscators. A promising direction for the field is iterative attack-defense games, where one party designs the obfuscator while the other builds (sub-optimal) adversaries for that obfuscator. This process is repeated until no attack designed by the adversary can reach a given privacy level. This could help build trust in obfuscation in contexts where theoretical guarantees are hard to prove. Our work enables this approach by formalizing the attack models that query obfuscators should protect against.

F. Future Work

We identify three main lines of research to extend the framework introduced in this work.

1) Extend the analysis to sequences of queries, as we explain in section VII-D.
2) Our framework considers a strong attacker, and it is likely that some of our assumptions can be relaxed in order to get better guarantees in practice.
3) Our theoretical analyses, and in particular our central result (theorem 2), assume perfect indistinguishability. It is likely that some version of the results

TABLE I

OBFUSCATORS FROM PRIOR WORK. FOR EACH OBFUSCATOR PUBLISHED IN PRIOR WORK, WE SUMMARIZE ITS MECHANISM AND PRIVACY DEFINITION. WE ALSO EVALUATE WHETHER EACH OBFUSCATOR COULD SATISFY OUR REQUIREMENTS. WE MARK ✓ WHEN THERE IS A PROOF THAT THE OBFUSCATOR COULD SATISFY A REQUIREMENT OR THE PRIVACY DEFINITION USED BY THE OBFUSCATOR IS EQUIVALENT TO ONE THE REQUIREMENT (ALTHOUGH THIS APPLIES MOSTLY TO THEORETICAL DEFINITIONS AND NOT TO IMPLEMENTATIONS), ✗ WHEN AN ATTACK HAS BEEN SUCCESSFULLY APPLIED IN PREVIOUS WORK, ✗* WHEN THE OBFUSCATOR CANNOT SATISFY THE REQUIREMENT BY DESIGN, AND ? WHEN OTHERWISE

Name	Query Generation Method	Privacy Metric	IND	COV	IMP
TrackMeNot [11]	Modify sentences from recent news headlines with simple heuristics.	None.	✗ [25]–[29]	⇒✗	⇒✗
PraW [12]	Mix keywords from an internal glossary, as well as keywords generated from the user profile (T_X).	Divergence between the real and obfuscated profiles of the user.	?	?	?
Viejo et al. [13]	Generate queries with varying levels of semantic similarity to the real query.	Mutual information between X and Z.	?	?	?
Murugesan et al. [14]	Send only queries from a set of "canonical queries" that cover different topics with a similar level of specificity.	Each query in Z is equally likely to be the user query, on a different topic, and equally likely to have produced Z.	?	?	✗*
Ye et al. [15]	Sample queries from a well-designed (theoretical) distribution.	Mutual information between X and $Y = Z_J$ with $J \sim Unif(\{1, \ldots, n\})$.	✓	?	?
Rebollo et al. [16]	Sample queries from a well-designed (theoretical) distribution.	Similarity between the obfuscated profile and the average population.	?	?	?
GooPIR [17]	Select keywords from a glossary with similar frequency in a text corpus to the keywords of the user query.	Entropy of the user query after observing the block, $H(X\|Z)$.	✗ [28], [29]	⇒✗	⇒✗
Ahmad et al. (2016) [18]	Use topic modeling to generate queries in different topics with similar entropy to the user query.	Mutual information between artificial and real user queries, and divergence between the real and obfuscated profiles of the user.	?	?	✗*
Ahmad et al. (2018) [19]	Use a language model to generate (sessions of) queries in topics related to the topic of the user query and with the same level of specificity.	Distance between the prior and posterior beliefs on the topic of X of an attacker.	?	✓	?

still hold for imperfect indistinguishability. In particular, we conjecture that when an obfuscator provides $\alpha_{ind}-$indistinguishability, the tradeoff of theorem 2 becomes: $\alpha_{ind} \leq \alpha_{cov} \cdot \alpha_{imp}$.

VIII. CONCLUSION

Query obfuscation, generating artificial data aimed at fooling an adversary in order to hide the content of the user, is a popular idea for web privacy. However, the guarantees provided by existing obfuscators are often unclear or based on ad-hoc measures. These obfuscators lack an adversarial framework that accounts for the capabilities of, and auxiliary information available to, a real-world adversary. Indeed, most obfuscators from the literature have been shown to be vulnerable to attacks [30]. In this paper, we present an adversarial framework to define the privacy of query obfuscation, centered around three privacy requirements that a block obfuscator must satisfy. We present two requirements based on previous work, indistinguishability and coverage, and introduce a novel requirement, imprecision. The latter is essential to make the privacy guarantees of an obfuscator resilient against adversaries with additional auxiliary information ("future-proof"). Our adversarial framework provides a rigorous basis on which to evaluate and compare existing obfuscators. Using the framework, we prove fundamental limitations to the protection that an obfuscator can offer, and in particular a tradeoff between privacy requirements which implies that making obfuscators more resilient to auxiliary knowledge ("future-proof") has an inherent cost. Finally,

we introduce a family of obfuscators that provably satisfy perfect indistinguishability. Using such an obfuscator in a simple setting, we show empirically that the bound of the coverage-imprecision tradeoff holds and is not tight: there is a significant difference between optimal guarantees and those that can be achieved in practice. Our work is a first step towards designing new obfuscators guaranteeing future-proof privacy, and developing a stronger characterization of adversarial privacy on the web.

ACKNOWLEDGMENT

The authors would like to thank the Computational Privacy Group at Imperial College London, and in particular Ana-Maria Creţu and Stefano Marrone, for their valuable feedback and inspirational discussions.

REFERENCES

[1] T.-C. Li, H. Hang, M. Faloutsos, and P. Efstathopoulos, "TrackAdvisor: Taking back browsing privacy from third-party trackers," in *Proc. Int. Conf. Passive Act. Netw. Meas.* Cham, Switzerland: Springer, 2015, pp. 277–289.

[2] S. Englehardt and A. Narayanan, "Online tracking: A 1-million-site measurement and analysis," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 1388–1401.

[3] T. O'reilly, *What is Web 2.0*. Sebastopol, CA, USA: O'Reilly Media, 2005.

[4] M. Catalano, "My family's Google searching got us a visit from counterterrorism police," Guardian, May 2013. [Online]. Available: https://www.theguardian.com/commentisfree/2013/aug/01/government-tracking-google-searches

[5] S. Zuboff, "Big other: Surveillance capitalism and the prospects of an information civilization," *J. Inf. Technol.*, vol. 30, no. 1, pp. 75–89, Mar. 2015.

[6] R. Jones, R. Kumar, B. Pang, and A. Tomkins, "'I know what you did last summer': Query logs and user privacy," in *Proc. 16th ACM Conf. Conf. Inf. Knowl. Manage.*, Nov. 2007, pp. 909–914.

[7] G. Eysenbach, "Infodemiology: Tracking flu-related searches on the web for syndromic surveillance," in *Proc. AMIA Annu. Symp.*, vol. 2006. Bethesda, MD, USA: American Medical Informatics Association, 2006, p. 244.

[8] J. Rose, C. Barton, R. Souza, and J. Platt. (Feb. 2014). Data Privacy by the Numbers. Boston Consulting Group, Accessed: Mar. 28, 2023. [Online]. Available: https://www.bcg.com/publications/2014/data-privacy-numbers

[9] M. Barbaro, T. Zeller, and S. Hansell, "A face is exposed for AOL searcher No. 4417749," *New York Times*, vol. 9, no. 2008, p. 8, 2006.

[10] A. Marthews and C. E. Tucker. (2017). *Government Surveillance and Internet Search Behavior*," [Online]. Available: https://ssrn.com/abstract=2412564

[11] D. C. Howe and H. Nissenbaum, "TrackMeNot: Resisting surveillance in web search," in *Lessons From the Identity Trail: Anonymity, Privacy, and Identity in a Networked Society*, vol. 23. London, U.K.: Oxford Univ. Press, 2009, pp. 417–436.

[12] B. Shapira, Y. Elovici, A. Meshiach, and T. Kuflik, "PRAW—A PRivAcy model for the Web," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 56, no. 2, pp. 159–172, Jan. 2005.

[13] A. Viejo, J. Castella-Roca, O. Bernadó, and J. M. Mateo-Sanz, "Single-party private web search," in *Proc. 10th Annu. Int. Conf. Privacy, Secur. Trust*, Jul. 2012, pp. 1–8.

[14] M. Murugesan and C. Clifton, "Providing privacy through plausibly deniable search," in *Proc. SIAM Int. Conf. Data Mining*, Apr. 2009, pp. 768–779.

[15] S. Ye, F. Wu, R. Pandey, and H. Chen, "Noise injection for search privacy protection," in *Proc. Int. Conf. Comput. Sci. Eng.*, vol. 3, 2009, pp. 1–8.

[16] D. Rebollo-Monedero and J. Forne, "Optimized query forgery for private information retrieval," *IEEE Trans. Inf. Theory*, vol. 56, no. 9, pp. 4631–4642, Sep. 2010.

[17] J. Domingo-Ferrer, A. Solanas, and J. Castellà-Roca, "$h(k)$-private information retrieval from privacy-uncooperative queryable databases," *Online Inf. Rev.*, vol. 33, no. 4, pp. 720–744, Aug. 2009.

[18] W. U. Ahmad, M. M. Rahman, and H. Wang, "Topic model based privacy protection in personalized web search," in *Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2016.

[19] W. U. Ahmad, K.-W. Chang, and H. Wang, "Intent-aware query obfuscation for privacy protection in personalized web search," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 285–294.

[20] D. Schultw. *Internet Noise*. Accessed: Mar. 28, 2023. [Online]. Available: http://makeinternetnoise.com/index.html

[21] I. Hury. *Noisy*. Accessed: Mar. 28, 2023. [Online]. Available: https://github.com/1tayH/noisy

[22] P. Price. *Needl*. Accessed: Mar. 28, 2023. [Online]. Available: https://github.com/eth0izzle/Needl

[23] E. Capuano. *Web-Traffic-Generator*. Accessed: Mar. 28, 2023. [Online]. Available: https://github.com/ecapuano/web-traffic-generator

[24] *Ruin My Search History*. Accessed: Mar. 28, 2023. [Online]. Available: https://proprivacy.com/tools/ruinmysearchhistory

[25] R. Al-Rfou', W. Jannen, and N. Patwardhan, "TrackMeNot-so-good-after-all," 2012, *arXiv:1211.0320*.

[26] S. T. Peddinti and N. Saxena, "On the privacy of web search based on query obfuscation: A case study of TrackMeNot," in *Proc. Int. Symp. Privacy Enhancing Technol. Symp.* Cham, Switzerland: Springer, 2010, pp. 19–37.

[27] R. Chow and P. Golle, "Faking contextual data for fun, profit, and privacy," in *Proc. 8th ACM Workshop Privacy Electron. Soc.*, Nov. 2009, pp. 105–108.

[28] A. Gervais, R. Shokri, A. Singla, S. Capkun, and V. Lenders, "Quantifying web-search privacy," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Nov. 2014, pp. 966–977.

[29] A. Petit et al., "SimAttack: Private web search under fire," *J. Internet Services Appl.*, vol. 7, no. 1, pp. 1–17, Dec. 2016.

[30] E. Balsa, C. Troncoso, and C. Diaz, "OB-PWS: Obfuscation-based private web search," in *Proc. IEEE Symp. Secur. Privacy*, May 2012, pp. 491–505.

[31] D. R. Stinson, *Cryptography: Theory and Practice*. Boca Raton, FL, USA: CRC Press, 2005.

[32] L. Sweeney, "K-anonymity: A model for protecting privacy," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 10, no. 5, pp. 557–570, 2002.

[33] P. Wang and C. V. Ravishankar, "On masking topical intent in keyword search," in *Proc. IEEE 30th Int. Conf. Data Eng.*, Mar. 2014, pp. 256–267.

[34] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "L-diversity: Privacy beyond k-anonymity," *ACM Trans. Knowl. Discovery Data*, vol. 1, no. 1, p. 3, 2007.

[35] B. Barak et al., "On the (Im) possibility of obfuscating programs," in *Proc. Annu. Int. Cryptol. Conf.* Cham, Switzerland: Springer, 2001, pp. 1–18.

[36] S. Goldwasser and Y. T. Kalai, "On the impossibility of obfuscation with auxiliary input," in *Proc. 46th Annu. IEEE Symp. Found. Comput. Sci. (FOCS)*, 2005, pp. 553–562.

[37] R. Kiros et al., "Skip-thought vectors," in *Proc. 28th Adv. Neural Inf. Process. Syst.*, vol. 2, Dec. 2015, pp. 3294–3302.

[38] P. Mac Aonghusa and D. J. Leith, "Plausible deniability in web search—From detection to assessment," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 4, pp. 874–887, Apr. 2018.

[39] O. Goldreich and R. Ostrovsky, "Software protection and simulation on oblivious RAMs," *J. ACM*, vol. 43, no. 3, pp. 431–473, 1996.

[40] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, "Private information retrieval," in *Proc. 36th Annu. Symp. Found. Comput. Sci.*, 1995, pp. 41–50.

[41] D. Goldschlag, M. Reed, and P. Syverson, "Onion routing," *Commun. ACM*, vol. 42, no. 2, pp. 39–41, 1999.

[42] D. L. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Commun. ACM*, vol. 24, no. 2, pp. 84–90, 1981.

[43] A. Arampatzis, G. Drosatos, and P. S. Efraimidis, "A versatile tool for privacy-enhanced web search," in *Proc. Eur. Conf. Inf. Retr.* Cham, Switzerland: Springer, 2013, pp. 368–379.

[44] H. Pang, X. Ding, and X. Xiao, "Embellishing text search queries to protect user privacy," *Proc. VLDB Endowment*, vol. 3, nos. 1–2, pp. 598–607, Sep. 2010.

[45] R. Jones and K. L. Klinkner, "Beyond the session timeout: Automatic hierarchical segmentation of search topics in query logs," in *Proc. 17th ACM Conf. Inf. Knowl. Manage.*, Oct. 2008, pp. 699–708.

[46] M. Nasr, S. Songi, A. Thakurta, N. Papernot, and N. Carlin, "Adversary instantiation: Lower bounds for differentially private machine learning," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2021, pp. 866–882.

[47] F. Houssiau et al., "TAPAS: A toolbox for adversarial privacy auditing of synthetic data," in *Proc. NeurIPS Workshop Synth. Data Empowering ML Res.*, 2022, pp. 1–8.