Privacy- and Utility-Preserving Textual Analysis via Calibrated Multivariate Perturbations

Oluwaseyi Feyisetan, Borja Balle, Thomas Drake, Tom Diethe
Amazon
{sey,pigem,draket,tdiethe}@amazon.com

ABSTRACT

Accurately learning from user data while providing quantifiable privacy guarantees provides an opportunity to build better ML models while maintaining user trust. This paper presents a formal approach to carrying out privacy preserving text perturbation using the notion of d_χ-privacy designed to achieve geo-indistinguishability in location data. Our approach applies carefully calibrated noise to vector representation of words in a high dimension space as defined by word embedding models. We present a privacy proof that satisfies d_χ-privacy where the privacy parameter ε provides guarantees with respect to a distance metric defined by the word embedding space. We demonstrate how ε can be selected by analyzing plausible deniability statistics backed up by large scale analysis on GloVe and fastText embeddings. We conduct privacy audit experiments against 2 baseline models and utility experiments on 3 datasets to demonstrate the tradeoff between privacy and utility for varying values of ε on different task types. Our results demonstrate practical utility (< 2% utility loss for training binary classifiers) while providing better privacy guarantees than baseline models.

CCS CONCEPTS

• **Security and privacy → Privacy protections**;

1 INTRODUCTION

Privacy-preserving data analysis is critical in the age of Machine Learning (ML) and Artificial Intelligence (AI) where the availability of data can provide gains over tuned algorithms. However, the inability to provide sufficient privacy guarantees impedes this potential in certain domains such as with user generated queries. As a result, computation over sensitive data has been an important goal in recent years [18, 25]. On the other hand, private data that has been inappropriately revealed carries a high cost, both in

terms of reputation damage and potentially fines, to data custodians charged with securing curated information. In this context, we distinguish between security and privacy breaches as follows: a security breach is unintended or unauthorized system usage, while a privacy breach is unintended or unauthorized data disclosure during intended system uses [6]. Unintended disclosures, or accidental publications which lead to re-identification have been two common causes of recent privacy breaches [2, 7, 18, 39, 41, 55, 56]. While it is possible to define rules and design access policies to improve data security, understanding the full spectrum of what can constitute a potential privacy infraction can be hard to predict *a priori*. As a result, solutions such as pattern matching, *ad hoc* filters and anonymization strategies are provably non-private. This is because such approaches cannot anticipate what side knowledge an attacker can use in conjunction with the released dataset. One definition that takes into account the limitations of existing approaches by preventing data reconstruction and protecting against any potential side knowledge is Differential Privacy.

Differential Privacy (DP) [21], which originated in the field of statistical databases, is one of the foremost standards for defining and dealing with privacy and disclosure prevention. At a high level, a randomized algorithm is differentially private if its output distribution is *similar* when the algorithm runs on two neighboring input databases. The notion of similarity is controlled by a parameter $\varepsilon \geq 0$ that defines the strength of the privacy guarantee (with $\varepsilon = 0$ representing absolute privacy, and $\varepsilon = \infty$ representing null privacy). Even though DP has been applied to domains such as geolocation [4], social networks [40] and deep learning [1, 50], less attention has been paid to adapting variants of DP to the context of Natural Language Processing (NLP) and the text domain [15, 59].

We approach the research challenge of preventing leaks of private information in text data by building on the quantifiable privacy guarantees of DP. In addition to these formal privacy requirements, we consider two additional requirements informed by typical deployment scenarios. First, the private mechanism must map text inputs to text outputs. This enables the mechanism to be deployed as a filter into existing text processing pipelines without additional changes to other components of the system. Such a requirement imposes severe limitations on the set of existing mechanisms one can use, and in particular precludes us from leveraging hash-based private data structures commonly used to identify frequent words [22, 54, 58]. The second requirement is that the mechanism must scale to large amounts of data and be able to deal with datasets that grow over time. This prevents us from using private data synthesis methods such as the ones surveyed in [10] because they suffer from severe scalability issues even in moderate-dimensional settings, and in general cannot work with datasets that grow over time. Together, these requirements push us towards solutions where each data

record is processed independently, similar to the setting in *Local DP (LDP)* [31]. To avoid the curse of dimensionality of standard LDP we instead adopt d_χ-privacy [3, 4, 14], a relaxed variant of local DP where privacy is defined in terms of the distinguishability level between inputs (see Sec. 2.3 for details).

Our main contribution is a scalable mechanism for *text analysis* satisfying d_χ-privacy. The mechanism operates on individual data records – adopting the *one user, one word* model as a baseline corollary to the *one user, one bit* model in the DP literature [31]. It takes a private input word x, and returns a privatized version $\hat{x}$ where the word in the original record has been 'perturbed'. The perturbation is obtained by first using a pre-determined *word embedding* model to map text into a high-dimensional vector space, adding noise to this vectorial representation, and the projecting back to obtain the perturbed word. The formal privacy guarantees of this mechanism can be interpreted as a degree of plausible deniability [8] conferred to the contents of x from the point of view of an adversary observing the perturbed output. We explore this perspective in detail when discussing how to tune the privacy parameters of our mechanism.

The utility of the mechanism is proportional to how well the semantics of the input text are preserved in the output. The main advantage of our mechanism in this context is to allow a higher degree of semantics preservation by leveraging the geometry provided by word embeddings when perturbing the data. In this work we measure semantics preservation by analyzing the performance obtained by using the privatized data on downstream ML tasks including binary sentiment analysis, multi-class classification, and question answering. This same methodology is typically used in NLP to evaluate unsupervised learning of word embeddings [49].

Our contributions in this paper can be summarized as follows:

1. We provide a formal approach to carrying out intent preserving text perturbation backed up by formal privacy analysis (Sec. 2).
2. We provide a principled way to select the privacy parameter ε for d_χ-privacy on text data based on geometrical properties of word embeddings (Sec. 3).
3. We conduct analysis on two embedding models, providing insights into words in the metric space (Sec. 4). We also show how the vectors respond to perturbations, connecting the geometry of the embedding with statistics of the d_χ-privacy mechanism.
4. We apply our mechanism to different experimental tasks, at different values of ε, demonstrating the trade-off between privacy and utility (Sec. 5).

2 PRIVACY PRESERVING MECHANISM

Consider a single word x submitted by a user interacting with an information system. *E.g.* x might represent a response to a survey request, or an elicitation of a fine grained sentiment. In particular, x will contain semantic information about the intent the user is trying to convey, but it also encodes an idiosyncratic representation of the user's word choices. Even though the word might not be explicitly *personally identifiable information* (PII) in the traditional sense of passwords and phone numbers, recent research has shown that the choice of words can serve as a *fingerprint* [12] via which tracing attacks are launched [52]. Our goal is to produce $\hat{x}$, a version of x that preserves the original intent while thwarting this attack vector. We start the section by giving a high-level description of the

rationale behind our mechanism and describing the threat model. Then we recall some fundamental concepts of d_χ-privacy. Finally, we provide a detailed description of our mechanism, together with a formal statement of its privacy guarantees.

2.1 Mechanism Overview

We start by providing a high-level description of our mechanism. Our mechanism applies a d_χ-privacy mechanism $\hat{x} = M(x)$ to obtain a replacement for the given word x. Such replacement is sampled from a carefully crafted probability distribution to ensure that $\hat{x}$ conveys a similar semantic to x while at the same time hiding any information that might reveal the identity of the user who generated x. Intuitively, the randomness introduced by d_χ-privacy provides plausible deniability [8] with respect to the original content submitted by the user. However, it also permits a curator of the perturbed words to perform aggregate sentiment analysis, or to cluster survey results without significant loss of utility.

2.2 Utility Requirements and Threat Model

When designing our mechanism we consider a threat model where a *trusted curator* collects a word from each user as $x^{(1)}, x^{(2)}, \ldots$ and wishes to make them available in clear form to an *analyst* for use in some downstream tasks (such as clustering survey responses or building ML models). The data is collected in the '*one user, one word*' model, and we do not seek to extend theoretical protections to aggregate user data in this model. Unfortunately, providing words in the clear presents the challenge of unwittingly giving the analyst access to information about the users interacting with the system. This could be either in the form of some shared side knowledge between the user and the analyst [33], or through an ML attack to learn which users frequently use a given set of words [51, 52]. Our working assumption is that the exact word is not necessary to effectively solve the downstream tasks of interest, although the general semantic meaning needs to be preserved to some extent; the experiments in Sec. 5 give several examples of this type of use case. Thus, we aim to transform each submission by using randomization to provide plausible deniability over any potential identifiers.

2.3 Privacy over Metric Spaces

Over the last decade, Differential Privacy (DP) [21] has emerged as a *de facto* standard for privacy-preserving data analysis algorithms. Several variants of DP have been proposed in the literature to address a variety of settings depending on whether, for example, privacy is defined with respect to aggregate statistics and ML models (*curator DP*) [21], or privacy is defined with respect to the data points contributed by each individual (*local DP*) [31].

Since our application involves privatizing individual words submitted by each user, LDP would be the ideal privacy model to consider. However, LDP has a requirement that renders it impractical for our application: it requires that the given word x has a non-negligible probability of being transformed into *any* other word $\hat{x}$, no matter how unrelated x and $\hat{x}$ are. Unfortunately, this constraint makes it virtually impossible to enforce that the *semantics* of x are approximately captured by the privatized word $\hat{x}$, since the space of words grows with the size of the language vocabulary, and the

number of words semantically related to x will have vanishingly small probability under LDP.

To address this limitation we adopt d_χ-privacy [3, 14], a relaxation of local DP that originated in the context of location privacy to address precisely the limitation described above. In particular, d_χ-privacy allows a mechanism to report a user's location in a privacy-preserving manner, while giving higher probability to locations which are close to the current location, and negligible probability to locations in a completely different part of the planet. d_χ-privacy was originally developed as an abstraction of the model proposed in [4] to address the privacy-utility trade-off in location privacy.

Formally, d_χ-privacy is defined for mechanisms whose inputs come from a set X equipped with a distance function $d : X \times X \to \mathbb{R}_+$ satisfying the axioms of a metric (*i.e.* identity of indiscernibles, symmetry and triangle inequality). The definition of d_χ-privacy depends on the particular distance function d being used and it is parametrized by a privacy parameter $\varepsilon > 0$. We say that a randomized mechanism $M : X \to \mathcal{Y}$ satisfies εd_χ-privacy if for any $x, x' \in X$ the distributions over outputs of $M(x)$ and $M(x')$ satisfy the following bound: for all $y \in \mathcal{Y}$ we have

$$\frac{\Pr[M(x) = y]}{\Pr[M(x') = y]} \le e^{\varepsilon d(x,x')} \quad . \tag{1}$$

We note that d_χ-privacy exhibits the same desirable properties of DP (*e.g.* composition, post-processing, robustness against side knowledge, *etc.*), but we won't use these properties explicitly in our analysis; we refer the reader to [14] for further details.

The type of probabilistic guarantee described by (1) is characteristic of DP: it says that the log-likelihood ratio of observing any particular output y given two possible inputs x and x' is bounded by $\varepsilon d(x, x')$. The key difference between d_χ-privacy and local DP is that the latter corresponds to a particular instance of the former when the distance function is given by $d(x, x') = 1$ for every $x \ne x'$. Unfortunately, this Hamming metric does not provide a way to classify some pairs of points in X as being closer than others. This indicates that local DP implies a strong notion of indistinguishability of the input, thus providing very strong privacy by "remembering almost nothing" about the input. Fortunately, d_χ-privacy is less restrictive and allows the indistinguishability of the output distributions to be scaled by the distance between the respective inputs. In particular, the further away a pair of inputs are, the more distinguishable the output distributions can be, thus allowing these distributions to remember more about their inputs than under the strictly stronger definition of local DP. An inconvenience of d_χ-privacy is that the meaning of the privacy parameter ε changes if one considers different metrics, and is in general incomparable with the ε parameter used in standard (local) DP (which can lead to seemingly larger privacy budget values as the dimensionality of the metric space increases). As a result, this paper makes no claim to provide privacy guarantees in the traditional sense of classical DP. Thus, in order to understand the privacy consequences of a given ε in d_χ-privacy one needs to understand the structure of the underlying metric d. For now we assume ε is a parameter given to the mechanism; we will return to this point in Sec. 3 where we analyze the meaning of this parameter for metrics on words derived from embeddings. All the metrics described in this work are

Euclidean. For discussions on d_χ-privacy over other metrics (such as Manhattan and Chebyshev, see [14])

2.4 Method Details

We now describe the proposed d_χ-privacy mechanism. The full mechanism takes as input a string x containing $|x|$ words and outputs a string $\hat{x}$ of the same length. To privatize x we use a d_χ-privacy mechanism $M : X \to X$, where $X = \mathcal{W}^\ell$ is the space of all strings of length ℓ with words in a dictionary $\mathcal{W}$. The metric between strings that we consider here is derived from a *word embedding model* $\phi : \mathcal{W} \to \mathbb{R}^n$ as follows: given $x, x' \in \mathcal{W}^\ell$ for some $\ell \ge 1$, we let $d(x, x') = \sum_{i=1}^{\ell} \|\phi(w_i) - \phi(w_i')\|$, where w_i (resp. w_i') denotes the ith word of x (resp. x'), and $\| \cdot \|$ denotes the Euclidean norm on $\mathbb{R}^n$. Note that d satisfies all the axioms of a metric as long as the word embedding ϕ is injective. We also assume the word embedding is independent of the data to be privatized; *e.g.* we could take an available word embedding like GloVe [44] or train a new word embedding on an available dataset. Our mechanism M works by computing the embedding $\phi(w)$ of each word $w \in x$, adding some properly calibrated random noise N to obtain a perturbed embedding $\hat{\phi} = \phi(w) + N$, and then replacing the word w with the word $\hat{w}$ whose embedding is closest to $\hat{\phi}$. The noise N is sampled from an n-dimensional distribution with density $p_N(z) \propto \exp(-\varepsilon\|z\|)$, where ε is the privacy parameter of the mechanism. The following pseudo-code provides implementation details for our mechanism.

Algorithm 1: Privacy Preserving Mechanism

Input: string $x = w_1 w_2 \cdots w_\ell$, privacy parameter $\varepsilon > 0$
for $i \in \{1, \ldots, \ell\}$ **do**
  Compute embedding $\phi_i = \phi(w_i)$
  Perturb embedding to obtain $\hat{\phi}_i = \phi_i + N$ with noise density $p_N(z) \propto \exp(-\varepsilon\|z\|)$
  Obtain perturbed word $\hat{w}_i = argmin_{u \in \mathcal{W}} \|\phi(u) - \hat{\phi}_i\|$
  Insert $\hat{w}_i$ in ith position of $\hat{x}$
release $\hat{x}$

See Sec. 2.6 for details on how to sample noise from the multivariate distribution p_N for different values of ε.

2.5 Privacy Proof

The following result states that our mechanism M satisfies εd_χ-privacy with respect to the metric d defined above.

THEOREM 1. *For any $\ell \ge 1$ and any $\varepsilon > 0$, the mechanism $M : \mathcal{W}^\ell \to \mathcal{W}^\ell$ satisfies εd_χ-privacy with respect to d.*

PROOF. The intuition behind the proof is to observe that M can be viewed as a combination of the generic exponential mechanism construction for the metric d together with a post-processing strategy that does not affect the privacy guarantee of the exponential mechanism. However, we chose not to formalize our proof in those terms; instead we provide a self-contained argument leading to a more direct proof without relying on properties of d_χ-privacy established elsewhere.

To start the proof, we first consider the case $\ell = 1$ so that $x = w \in \mathcal{W}$ and $x' = w' \in \mathcal{W}$ are two inputs of length one. For any

possible output word $\hat{w} \in \mathcal{W}$ we define a set $C_{\hat{w}} \subset \mathbb{R}^n$ containing all the feature vectors which are closer to the embedding $\phi(\hat{w})$ than to the embedding of any other word. Formally, we have

$$C_{\hat{w}} = \left\{ z \in \mathbb{R}^n \ : \ \|z - \phi(\hat{w})\| < \min_{u \in \mathcal{W}\backslash\{\hat{w}\}} \|z - \phi(u)\| \right\} \ .$$

The set $C_{\hat{w}}$ is introduced because it is directly related to the probability that the mechanism M on input $x = w$ produces $\hat{w}$ as output. Indeed, by the description of M we see that we get $M(w) = \hat{w}$ if and only if the perturbed feature vector $\hat{\phi} = \phi(w) + N$ is closer to $\phi(\hat{w})$ than to the embedding of any other word in $\mathcal{W}$. In particular, letting $p_{\phi(w)+N}(z)$ denote the density of the random variable $\phi(w) + N$, we can write the probability of this event as follows:

$$\Pr[M(w) = \hat{w}] = \Pr[\phi(w) + N \in C_{\hat{w}}] = \int_{C_{\hat{w}}} p_{\phi(w)+N}(z)dz$$

$$= \int_{C_{\hat{w}}} p_N(z - \phi(w))dz \propto \int_{C_{\hat{w}}} \exp(-\varepsilon\|z - \phi(w)\|)dz \ ,$$

where we used that $\phi(w) + N$ has exactly the same distribution of N but with a different mean. Now we note that the triangle inequality for the norm $\|\cdot\|$ implies that for any $z \in \mathbb{R}^n$ we have the following inequality:

$$\exp(-\varepsilon\|z - \phi(w)\|) = \frac{\exp(-\varepsilon\|z - \phi(w)\|)}{\exp(-\varepsilon\|z - \phi(w')\|)} \exp(-\varepsilon\|z - \phi(w')\|)$$

$$= \exp(\varepsilon(\|z - \phi(w')\| - \|z - \phi(w)\|))$$

$$\times \ \exp(-\varepsilon\|z - \phi(w')\|)$$

$$\leq \exp(\varepsilon\|\phi(w) - \phi(w')\|) \exp(-\varepsilon\|z - \phi(w')\|)$$

$$= \exp(\varepsilon d(w, w')) \exp(-\varepsilon\|z - \phi(w')\|) \ .$$

Combining the last two derivations and observing the the normalization constants in $p_N(z)$ and $p_{\phi(w)+N}(z)$ are the same, we obtain

$$\frac{\Pr[M(w) = \hat{w}]}{\Pr[M(w') = \hat{w}]} = \frac{\int_{C_{\hat{w}}} \exp(-\varepsilon\|z - \phi(w)\|)dz}{\int_{C_{\hat{w}}} \exp(-\varepsilon\|z - \phi(w')\|)dz} \leq \exp(\varepsilon d(w, w')) \ .$$

Thus, for $\ell = 1$ the mechanism M is εd_χ-privacy preserving.

Now we consider the general case $\ell > 1$. We claim that because the mechanism treats each word in $x = w_1 \cdots w_\ell$ independently, the result follows directly from the analysis for the case $\ell = 1$. To see this, we note the following decomposition allows us to write the output distribution of the mechanism on strings of length $\ell > 1$ in terms of the output distributions of the mechanism on strings of length one: for $x, \hat{x} \in \mathcal{W}^\ell$ we have

$$\Pr[M(x) = \hat{x}] = \prod_{i=1}^{\ell} \Pr[M(w_i) = \hat{w}_i] \ .$$

Therefore, using that M is d_χ-privacy preserving with respect to d on strings of length one, we have that for any pair of inputs $x, x' \in \mathcal{W}^\ell$ and any output $\hat{x} \in \mathcal{W}^\ell$ the following is satisfied:

$$\frac{\Pr[M(x) = \hat{x}]}{\Pr[M(x') = \hat{x}]} = \prod_{i=1}^{\ell} \left(\frac{Pr[M(w_i) = \hat{w}_i]}{Pr[M(w'_i) = \hat{w}_i]} \right)$$

$$\leq \prod_{i=1}^{\ell} \exp(\varepsilon d(w_i, w'_i)) = \exp(\varepsilon d(x, x')) \ ,$$

where we used that the definition of d is equivalent to $d(x, x') = \sum_{i=1}^{\ell} d(w_i, w'_i)$. The result follows.   □

2.6 Sampling from the Noise Distribution

To sample from p_N, first, we sample a vector-valued random variable $\mathbf{v} = [v_1 \ldots v_n]$ from the multivariate normal distribution:

$$p(x; \mu, \Sigma) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu) \right).$$

where n is the dimensionality of the word embedding, the mean μ is centered at the origin and the covariance matrix Σ is the identity matrix. The vector $\mathbf{v}$ is then normalized to constrain it in the unit ball. Next, we sample a magnitude l from the Gamma distribution

$$p(x; n, \theta) = \frac{x^{n-1}e^{-x/\theta}}{\Gamma(n)\theta^n}.$$

where $\theta = 1/\varepsilon$ and n is the embedding dimensionality. A sample noisy vector at the privacy parameter ε is therefore output as $l\mathbf{v}$. More details on the approach can be found in [60, Appendix E].

3 STATISTICS FOR PRIVACY CALIBRATION

In this section we present a methodology for calibrating the ε parameter of our d_χ-privacy mechanism M based on the geometric structure of the word embedding ϕ used to define the metric d. Our strategy boils down to identifying a small number of statistics associated with the output distributions of M, and finding a range of parameters ε where these statistics behave as one would expect from a mechanism providing a prescribed level of plausible deniability. We recall that the main reason this is necessary, and why the usual rules of thumb for calibrating ε in traditional (*i.e.* hamming distance based) DP cannot be applied here, is because the meaning of ε in d_χ-privacy depends on the particular metric being used and is not transferable across metrics. We start by making some qualitative observations about how ε affects the behavior of mechanism M. For the sake of simplicity we focus the discussion on the case where x is a single word $x = w$, but all our observations can be directly generalized to the case $|x| > 1$.

3.1 Qualitative Observations

The first observation is about the behavior at extreme values of ε. As $\varepsilon \to 0$ we have $M(w)$ converging to the a fixed distribution over $\mathcal{W}$ independent of w. This distribution will not be uniform across $\mathcal{W}$ since the probability $\lim_{\varepsilon \to 0} \Pr[M(w) = \hat{w}]$ will depend on the relative size of the event $C_{\hat{w}}$ defined in the proof of Thm. 1. However, since this distribution is the same regardless of the input w, we see that $\varepsilon \to 0$ provides absolute privacy as the output produced by the mechanism becomes independent of the input word. Such a mechanism will not provide preserve semantics as the output is essentially random. In contrast, the regime $\varepsilon \to \infty$ will yield a mechanism satisfying $M(w) = w$ for all inputs, thus providing null privacy, but fully preserving the semantics. As expected, by tuning the privacy parameter ε we can trade-off privacy *vs.* utility. Utility for our mechanism comes in the form of some semantic-preserving properties; we will measure the effect of ε on the utility when we use the outputs in the context of a ML pipeline in Sec. 5. Here we focus on exploring the effect of ε on the privacy provided

by the mechanism, so as to characterize the minimal values of the parameter that yield acceptable privacy guarantees.

Our next observation is that for any finite ε, the distribution of $M(w)$ has full support on $\mathcal{W}$. In other words, for any possible output $\hat{w} \in \mathcal{W}$ we have a non-zero probability that $M(w) = \hat{w}$. However, we know from our discussion above that for $\hat{w} \neq w$ these probabilities vanish as $\varepsilon \to \infty$. A more precise statement can be made if one tries to compare the rate at which the probabilities $\Pr[M(w) = \hat{w}]$ for different outputs $\hat{w}$. In particular, given two outputs with $d(w, \hat{w}) \ll d(w, \hat{w}')$, by the definition of M we will have $\Pr[M(w) = \hat{w}] \gg \Pr[M(w) = \hat{w}']$ for any fixed ε. Thus, taking the preceding observation and letting ε grow, one obtains that $\Pr[M(w) = \hat{w}]$ goes to zero much faster for outputs $\hat{w}$ far from w than for outputs close to it. We can see from this argument that, essentially, as ε grows, the distribution of $M(w)$ concentrates around w and the words close to w. This is good from a utility point of view – words close to w with respect to the metric d will have similar meanings by the construction of the embeddings – but too much concentration degrades the privacy guarantee since it increases the probability $\Pr[M(w) = w]$ and makes the effective support of the distribution of $M(w)$ too small to provide plausible deniability.

3.2   Plausible Deniability Statistics

Inspired by the discussion above, we define two statistics to measure the amount of plausible deniability provided by a choice of the privacy parameter ε. Roughly speaking, in the context of text redaction applications, plausible deniability measures the likelihood of making correct inferences about the input given a sample from the privatization mechanism. In this sense, plausible deniability can be achieved by making sure the original word has low probability of being released unperturbed, and additionally making sure that the words that are frequently sampled given some input word induce enough variation on the sample to hide which what the input word was. A key difference between LDP and d_χ-privacy is that the former provides a stronger form of plausible deniability by insisting that almost every outcome is possible when a word is perturbed, while the later only requires that we give enough probability mass to words close to the original one to ensure that the output does not reveal what the original word was, although it still releases information about the neighborhood where the original word was.

More formally, the statistics we look at are the probability $N_w = \Pr[M(w) = w]$ of not modifying the input word w, and the (effective) support of the output distribution S_w (*i.e.* number of possible output words) for an input w. In particular, given a small probability parameter $\eta > 0$, we define S_w as the size of the smallest set of words that accumulates probability at least $1 - \eta$ on input w:

$$S_w = \min |\{S \subseteq \mathcal{X} \;:\; \Pr[M(w) \notin S] \leq \eta\}| \;.$$

Intuitively, a setting of ε providing plausible deniability should have N_w small and S_w large for (almost) all words in $w \in \mathcal{W}$.

These statistics can also be related to the two extremes of the Rényi entropy [46], thus providing an additional information-theoretic justification for the settings of ε that provide plausible deniability in terms of large entropy. Recall that for a distribution p over $\mathcal{W}$

with $p_w = \Pr_{W \sim p}[W = w]$, the Rényi entropy of order $\alpha \geq 0$ is

$$H_\alpha(p) = \frac{1}{1 - \alpha} \log \left(\sum_{w \in \mathcal{W}} p_w^\alpha \right) \;.$$

By taking the extremes $\alpha = 0$ and $\alpha = \infty$ one obtains the so-called min-entropy $H_0(p) = \log |\mathrm{supp}(p)|$ and max-entropy $H_\infty(p) = \log(1/\max_w p_w)$, where $\mathrm{supp}(p) = \{w : p_w > 0\}$ denotes the support of p. This implies that we can see the quantities S_w and N_w as proxies for the two extreme Rényi entropies through the approximate identities $H_0(M(w)) \approx \log S_w$ and $H_\infty(M(w)) \approx \log 1/N_w$, where the last approximation relies on the fact that (at least for small enough ε), w should be the most likely word under the distribution of $M(w)$. Making these two quantities large amounts to increasing the entropy of the distribution. In practice, we prefer to work with the statistics S_w and N_w than with the extreme Rényi entropies since the former are easier to estimate through simulation.

4   ANALYSIS OF WORD EMBEDDINGS

A word embedding $\phi : \mathcal{W} \to \mathbb{R}^n$ maps each word in some vocabulary to a vector of real numbers. An approach for selecting the model parameters is to posit a conditional probability $p(o|w)$ of observing a word o given a nearby word or context w by taking the soft-max over all contexts in the vocabulary as: $p(o|w) = \exp\left(\phi(o)^\top \phi(w)\right) / \sum_{w' \in \mathcal{W}} \exp\left(\phi(w')^\top \phi(w)\right)$ [26]. Such models are usually trained using a skip-gram objective [38] to maximize the average log probability of words w given the surrounding words as a context window of size m scans through a large corpus of words $w_1, \ldots, w_T$: $\frac{1}{T} \sum_{t=1}^{T} \sum_{-m \leq j \leq m, j \neq 0} \log p(w_{t+j}|w_t)$.

The geometry of the resulting embedding model has a direct impact on defining the output distribution of our redaction mechanism. To get an intuition for the structure of these metric spaces – *i.e.*, how words cluster together and the distances between words and their neighbors – we ran several analytical experiments on two widely available word embedding models: GLOVE [44] and FASTTEXT [9]. We selected $319,000$ words that were present in both the GLOVE and FASTTEXT embeddings. Though we present findings only from the common $319,000$ words in the embedding vocabularies, we carried out experiments over the entire vector space (*i.e.*, $400,000$ for GLOVE and $2,519,370$ for FASTTEXT).

Our experiments provide: (i) insights into the distance $d(x, x')$ that controls the privacy guarantees of our mechanism for different embedding models detailed below; and (ii) empirical evaluation of the plausible deniability statistics S_w and N_w described in Sec. 4.1 for the mechanisms obtained using different embeddings.

We analyzed the distance between each of the $319,000$ words and its k closest neighbors. The k values were $1, 5, 10, 20, 50, 100, 200, 500,$ and 1000. We computed the Euclidean distance between each word vector and its k neighbors. We then computed 5th, 20th, 50th, 80th, and 95th percentile of the distances for each of the k values. The line chart in Fig. 1 summarizes the results across the percentiles values by presenting a logarithmic view of the increasing k values.

The line plot results in Fig. 1 give insights into how different embedding models of the same vector dimension can have different distance distributions. The words in FASTTEXT have a smoother distance distribution with a wider spread across percentiles.
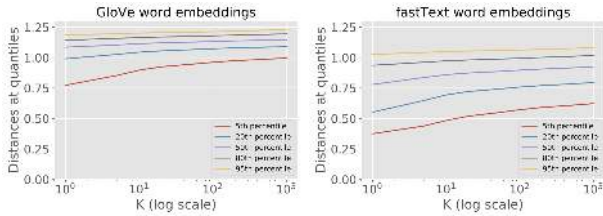
Figure 1: Distribution of distances between a given vector and its k closest neighbors for GloVe and fastText

4.1 Word Distribution Statistics

We ran the mechanism $1,000$ times on input w to compute the plausible deniability statistics S_w and N_w at different values of ε for each word embedding model. For each word w and the corresponding list of $1,000$ new words W' from our d_χ perturbation, we recorded: (i) the probability $N_w = \Pr[M(w) = w]$ of not modifying the input word w (estimated as the empirical frequency of the event $M(w) = w$); and (ii) the (effective) support of the output distribution S_w (estimated by the distinct words in W').

The results presented in Fig. 2 provide a visual way of selecting ε for task types of different sensitivities. We can select appropriate values of ε by selecting our desired *worst case* guarantees, then observing the extreme values of the histograms for N_w and S_w. For example, at $\varepsilon = 5$, no word yields fewer than 300 distinct new words (S_w graph), and no word is ever returned more than 500 times in the worst case (N_w graph). Therefore, by looking at the worst case guarantees of S_w and N_w over different values of ε, we can make a principled choice on how to select ε for a given embedding model.

4.2 Selecting Between Different Embeddings

Our analysis gives a reasonable approach to selecting ε (i.e., via *worst case* guarantees) by means of the proxies provided by the plausible deniability statistics. In general, tuning privacy parameters in d_χ-privacy is still a topic under active research [30], especially with respect to what ε means for different applications.

With regards to the same embedding model with different dimensionalities, Fig. 3 suggests that they do provide the same level of average case guarantees (at '*different*' values of ε). Therefore, selecting a model becomes a function of utility on downstream tasks. Fig. 3 further underscores the need to interpret the notion of ε in d_χ-privacy within the context of the metric space.

In Fig. 4, we present the average values of S_w and N_w statistics for GloVe and fastText. However, the average values are not sufficient to make a conclusive comparison between embedding models since different distributions can result in the same entropy – therefore, we recommend setting worst case guarantees. For further discussions on the caveats of interpreting entropy based privacy metrics see [57].

Table 1 presents examples of word perturbations on similar mechanisms calibrated on GloVe and fastText. The results show that as the average values of N_w increase (corresponding to higher values of ε), the resulting words become more similar to the original word.

5 ML UTILITY EXPERIMENTS

We describe experiments we carried out to demonstrate the trade-off between privacy and utility for three downstream NLP tasks.

5.1 Datasets

We ran experiments on three textual datasets, each representing a common task type in ML and NLP. The datasets are: IMDb movie reviews (binary classification) [36], Enron emails (multi-class classification) [32], and InsuranceQA (question answering) [23]. Each dataset contains contributions from individuals making them suitable dataset choices.

5.2 Setup for utility experiments

For each dataset, we demonstrated privacy vs. utility at:

Training time: we trained the models on perturbed data, while testing was carried out on plain data. This simulates a scenario where there is more access to private training data.

Test time: here, we trained the models completely on the available training set. However, the evaluation was done on a privatized version of the test sets.

5.3 Baselines for utility experiments

All models in our experiments use $300d$ GloVe embeddings (hence the seemingly *larger* values of ε. See discourse in Sec. 2.3 and Fig. 3) on Bidirectional LSTM (biLSTM) models [27].

IMDb movie reviews The training set was split to as in [36]. The privacy algorithm was run on the partial training set of $15,000$ reviews. The evaluation metric used was classification accuracy.

Enron emails Our test set was constructed by sampling a random 10% subset of the emails of 10 selected authors in the dataset. The evaluation metric was also classification accuracy.

InsuranceQA We replicated the results from [53] with Geometric mean of Euclidean and Sigmoid Dot product (GESD) as similarity scores. The evaluation metrics used Mean Average Precision (MAP) and Mean Reciprocal Rank (MRR).

The purpose of our baseline models was not to advance the state of the art for those specific tasks. They were selected to provide a standard that we could use to compare our further experiments.

5.4 Results for utility experiments

We now discuss the individual results from running the d_χ algorithm on machine learning models trained on the 3 datasets presented in Fig. 5. We start with the binary sentiment classification task on the IMDb dataset. Across the 3 experiments, we observe the expected privacy utility trade-off. As ε increases (greater privacy loss), the utility scores improve. Conversely, at smaller values of ε, we record worse off scores. However, this observation varies across the tasks. For the binary classification task, at training and test time, the model remains robust to the injected perturbations. Performance degrades on the other 2 tasks with the question answering task being the most sensitive to the presence of noise.

6 ML PRIVACY EXPERIMENTS

We now describe how we evaluate the privacy guarantees from our approach against two query scrambling methods from literature.
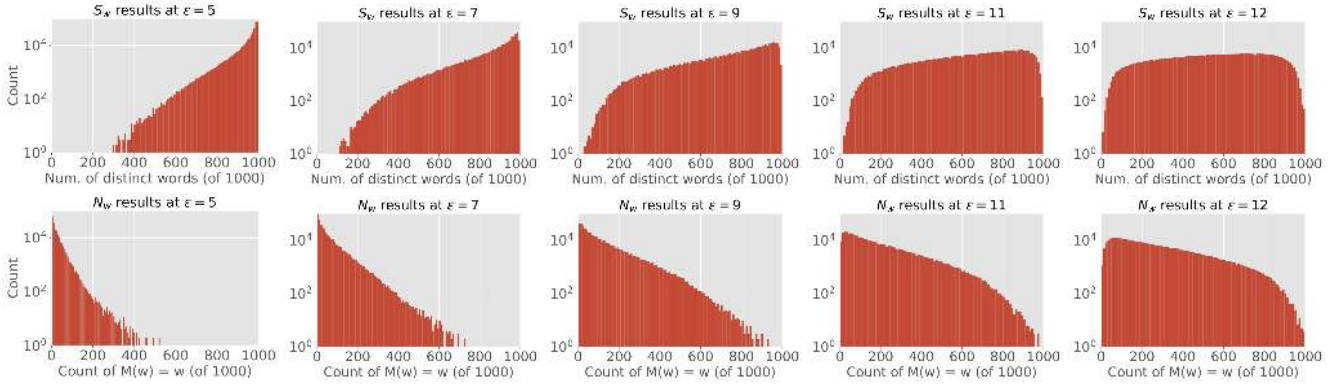
Figure 2: Empirical S_w and N_w statistics for 50 dimensional GloVe word embeddings as a function of ε.

		w = encryption		w = hockey		w = spacecraft	
ε	Avg. N_w	GloVe	fastText	GloVe	fastText	GloVe	fastText
	50	freebsd	ncurses	stadiumarena	dampener	telemeter	geospace
		multibody	vpns	futsal	popel	deorbit	powerup
		56-bit	tcp	broomball	decathletes	airbender	skylab
		public-key	isdn	baseballer	newsweek	aerojet	unmanned
increasing ε, better semantics	100	ciphertexts	plaintext	interleague	basketball	laser	voyager
		truecrypt	diffie-hellman	usrowing	lacrosse	apollo-soyuz	cassini-huygens
		demodulator	multiplexers	football	curlers	agena	adrastea
		rootkit	cryptography	lacrosse	usphl	phaser	intercosmos
	200	harbormaster	cryptographic	players	goaltender	launch	orbited
		unencrypted	ssl/tls	ohl	ephl	shuttlecraft	tatooine
		cryptographically	authentication	goaltender	speedskating	spaceborne	flyby
		authentication	cryptography	defenceman	eishockey	interplanetary	spaceborne
	300	decryption	encrypt	nhl	hockeygoalies	spaceplane	spaceship
		encrypt	unencrypted	hockeydb	hockeyroos	spacewalk	spaceflights
		encrypted	encryptions	hockeyroos	hockeyettan	spaceflights	satellites
		encryption	encrypted	hockey	hockey	spacecraft	spacecraft

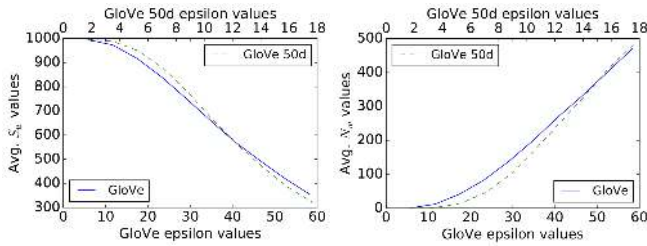Table 1: Output $\hat{w} = M(w)$ on topic model words from the 20 Newsgroups dataset. Selected words w are from [34]

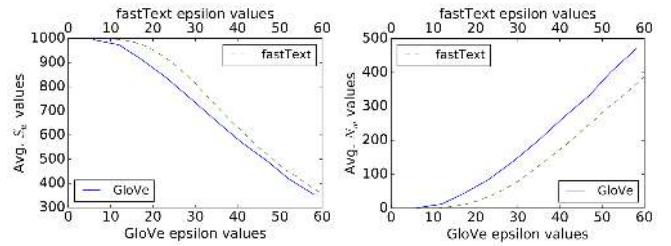Figure 3: Average S_w and N_w statistics: GloVe $50d$ and $300d$

Figure 4: Average S_w and N_w statistics: GloVe and fastText

6.1 Baselines for privacy experiments

We evaluated our approach against the following baselines:

Versatile [5] – using the 'semantic' and 'statistical' query scrambling techniques. Sample queries were obtained from the paper.

Incognito [37] – using perturbations 'with' and 'without' noise. Sample queries were also obtained from the paper.

6.2 Datasets

Search logs – The two evaluation baselines [5, 37] sampled data from [43] therefore, we also use this as the dataset for our approach.

6.3 Setup for privacy experiments

We evaluate the baselines and our approach using the privacy auditor described in [52]. We modeled our experiments after the paper as follows: From the search logs dataset [43], we sampled users with between 150 and 500 queries resulting in 8, 670 users. We randomly sampled 100 users to train and another 100 users (negative examples) to test the privacy auditor system.

For each evaluation baseline, we dropped an existing user, then created a new user and injected the scrambled queries using the baseline's technique. The evaluation metrics are: Precision, Recall, Accuracy and Area Under Curve (AUC). The metrics are computed
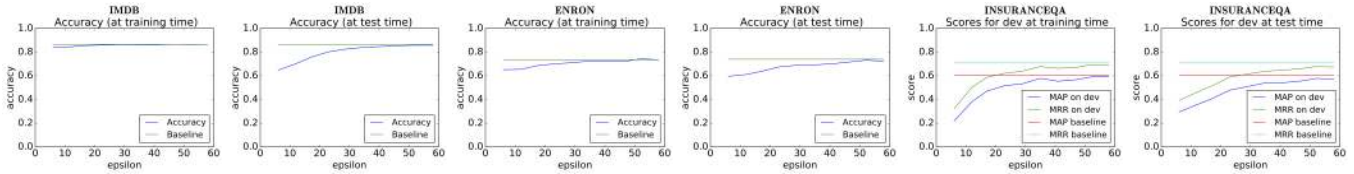
Figure 5: d_χ-privacy scores against utility baseline

over the ability of the privacy auditor to correctly identify queries *used* to train the system, and queries *not used* to train the system.

6.4 Results for privacy experiments

The metrics in Table 2 depict privacy loss (*i.e.* lower is better). The results highlight that existing baselines fail to prevent attacks by the privacy auditor. The auditor is able to perfectly identify queries that were perturbed using the baseline techniques regardless of whether they were actually used to train the system or not.

Model	Precision	Recall	Accuracy	AUC
Original queries	1.0	1.0	1.0	1.0
Versatile (semantic)	1.0	1.0	1.0	1.0
Versatile (statistical)	1.0	1.0	1.0	1.0
Incognito (without noise)	1.0	1.0	1.0	1.0
Incognito (with noise)	1.0	1.0	1.0	1.0
d_χ-privacy (at $\varepsilon = 23$)	**0.0**	**0.0**	**0.5**	**0.36**

Table 2: Results: scores measure privacy loss (lower is better)

Conversely, our d_χ approach in the last line of Table 2 and expanded in Table 3, show we are able provide tunable privacy guarantees (over 3x greater than baselines for $\varepsilon \leq 23$ on AUC scores). Across all metrics (at $\varepsilon \leq 23$), our privacy guarantees is better than chance.

	ε for GLOVE $300d$								
Metric	6	12	17	23	29	35	41	47	52
Precision	0.00	0.00	0.00	0.00	0.67	0.90	0.93	1.00	1.00
Recall	0.00	0.00	0.00	0.00	0.02	0.09	0.14	0.30	0.50
Accuracy	0.50	0.50	0.50	0.50	0.51	0.55	0.57	0.65	0.75
AUC	0.06	0.04	0.11	0.36	0.61	0.85	0.88	0.93	0.98

Table 3: d_χ-privacy results: all scores measure privacy loss

7 DISCUSSION

We have described how to achieve formal privacy guarantees in textual datasets by perturbing the words of a given query. Our experiments on machine learning models using different datasets across different task types have provided empirical evidence into the feasibility of adopting this technique. Overall, our findings demonstrate the tradeoffs between desired privacy guarantees and the achieved task utility. Previous work in data mining [11, 35] and privacy research [24, 28] have described the cost of privacy and the need to attain tunable utility results. Achieving optimal privacy as described in *Dalenius's Desideratum* [20] will yield a dataset that confers no utility to the curator. While techniques such

as homomorphic encryption [25] hold promise, they have not been developed to the point of practical applicability.

8 RELATED WORK

Text redaction for privacy protection is a well understood and widely studied problem [13] with good solutions currently found wanting [29]. This is amplified by the fact that redaction needs vary. For example, with transactional data (such as search logs), the objective is *anonymity* or *plausible deniability* so that the identity of the person performing the search cannot be ascertained. On the other hand, with plain text data (such as emails and medical records), the objective might be *confidentiality* so that an individual is not associated with an entity. Our research is designed around conferring plausible deniability in search logs while creating a mechanism that can be extended to other plain text data types.

The general approach to text redaction in literature follows two steps: (i) detection of sensitive terms, and (ii) obfuscation of the identified entities. Our approach differs from related works in these two tasks. With respect to (i) in transactional data, [37] is predicated on defining private queries and sensitive terms based on uniformity, uniqueness and linkability to predefined PII such as names and locations. This approach however doesn't provide privacy guarantees for queries that fall outside this definition. Other methods such as [19, 42, 48] bypass the detection of sensitive terms and inject additional keywords into the initial search query. It has been shown in [45] that this model is susceptible to de-anonymisation attacks. On the other hand, techniques such as [5] do not focus on (i), but source (ii) i.e., replacement entities from related web documents while we use word embedding models for this step.

Similarly, for plain text data, approaches such as [16, 17] address (i) by using models to 'recognize several classes of PII' such as names and credit cards, while [47] focuses on (ii) that is, sanitizing an entity c by removing all terms t that can identify c individually or in aggregate in a knowledge base K. Indeed, any privacy preserving algorithm that places *a priori* classification on sensitive data types assume boundaries on an attackers side knowledge and a finite limit on potentially new classes of personal identifiers. Our approach with d_χ-privacy aims to do away with such assumptions to provide tunable privacy guarantees.

9 CONCLUSION

In this paper, we presented a formal approach to carrying out privacy preserving text perturbation using d_χ-privacy. Our approach applied carefully calibrated noise to vector representations of words in a high dimension space as defined by word embedding models. We presented a theoretical privacy proof that satisfies d_χ-privacy where the parameter ε provides guarantees with respect to a metric

$d(x, x')$ defined by the word embedding space. Our experiments demonstrated that our approach provides tunable privacy guarantees over 3 times greater than the baselines, while incurring $< 2\%$ utility loss on training binary classifiers (among other task types) for a range of ε values. By combining the results of our privacy and utility experiments, with our guidelines on selecting ε by using worst-case guarantees from our plausible deniability statistics, data holders can make a rational choice in applying our mechanism to attain a suitable privacy-utility tradeoff for their tasks.

REFERENCES

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC CCS*. ACM, 308–318.
[2] John M Abowd. 2018. The US Census Bureau Adopts Differential Privacy. In *Proceedings of the 24th ACM SIGKDD*. ACM, 2867–2867.
[3] Mário Alvim, Konstantinos Chatzikokolakis, Catuscia Palamidessi, and Anna Pazii. 2018. Local Differential Privacy on Metric Spaces: optimizing the trade-off with utility. In *Computer Security Foundations Symposium (CSF)*.
[4] Miguel E Andrés, Nicolás E Bordenabe, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. 2013. Geo-indistinguishability: Differential privacy for location-based systems. In *Proceedings of the 2013 ACM SIGSAC CCS*. ACM, 901–914.
[5] Avi Arampatzis, George Drosatos, and Pavlos S Efraimidis. 2015. Versatile query scrambling for private web search. *Info. Retrieval Journal* 18, 4 (2015), 331–358.
[6] Derek E Bambauer. 2013. Privacy versus security. *J. Crim. L. & Criminology* 103 (2013), 667.
[7] Michael Barbaro, Tom Zeller, and Saul Hansell. 2006. A face is exposed for AOL searcher no. 4417749. *New York Times* 9, 2008 (2006), 8.
[8] Vincent Bindschaedler, Reza Shokri, and Carl A Gunter. 2017. Plausible deniability for privacy-preserving data synthesis. *VLDB Endowment* 10, 5 (2017), 481–492.
[9] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *TACL* 5 (2017).
[10] Claire McKay Bowen and Fang Liu. 2016. Comparative study of differentially private data synthesis methods. *arXiv preprint arXiv:1602.01063* (2016).
[11] Justin Brickell and Vitaly Shmatikov. The cost of privacy: destruction of data-mining utility in anonymized data publishing. In *ACM SIGKDD*.
[12] Mark Bun, Jonathan Ullman, and Salil Vadhan. 2018. Fingerprinting codes and the price of approximate differential privacy. *SIAM J. Comput.* (2018).
[13] Declan Butler. 2004. US intelligence exposed as student decodes Iraq memo. (2004).
[14] Konstantinos Chatzikokolakis, Miguel E Andrés, Nicolás Emilio Bordenabe, and Catuscia Palamidessi. 2013. Broadening the scope of differential privacy using metrics. In *Intl. Symposium on Privacy Enhancing Technologies Symposium*.
[15] Maximin Coavoux, Shashi Narayan, and Shay B. Cohen. 2018. Privacy-preserving Neural Representations of Text. In *EMNLP*.
[16] Chad Cumby and Rayid Ghani. 2010. Inference control to protect sensitive information in text documents. In *ACM SIGKDD WISI*. ACM, 5.
[17] Chad M Cumby and Rayid Ghani. 2011. A Machine Learning Based System for Semi-Automatically Redacting Documents.. In *IAAI*.
[18] Irit Dinur and Kobbi Nissim. 2003. Revealing information while preserving privacy. In *ACM Symposium on Principles of Database Systems*. ACM, 202–210.
[19] Josep Domingo-Ferrer, Agusti Solanas, and Jordi Castellà-Roca. 2009. h (k)-Private information retrieval from privacy-uncooperative queryable databases. *Online Information Review* 33, 4 (2009), 720–744.
[20] Cynthia Dwork. 2011. A firm foundation for private data analysis. *Commun. ACM* 54, 1 (2011), 86–95.
[21] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *TCC*. Springer, 265–284.
[22] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. 2014. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *ACM SIGSAC CCS*.
[23] Minwei Feng, Bing Xiang, Michael R. Glass, Lidan Wang, and Bowen Zhou. 2015. Applying deep learning to answer selection: A study and an open task. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding*. 813–820.
[24] Quan Geng and Pramod Viswanath. 2014. The optimal mechanism in differential privacy. In *2014 IEEE Intl. Symposium on Information Theory (ISIT)*. 2371–2375.
[25] Craig Gentry and Dan Boneh. 2009. *A fully homomorphic encryption scheme*. Vol. 20. Stanford University Stanford.
[26] Yoav Goldberg and Omer Levy. 2014. word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method. *arXiv:1402.3722* (2014).
[27] Alex Graves, Santiago Fernández, and Jürgen Schmidhuber. 2005. Bidirectional LSTM networks for improved phoneme classification and recognition. In *International Conference on Artificial Neural Networks*. Springer, 799–804.

[28] Xi He, Ashwin Machanavajjhala, and Bolin Ding. 2014. Blowfish privacy: Tuning privacy-utility trade-offs using policies. In *Proc. of the 2014 ACM SIGMOD*.
[29] Steven Hill, Zhimin Zhou, Lawrence Saul, and Hovav Shacham. 2016. On the (in) effectiveness of mosaicing and blurring as tools for document redaction. *Proceedings on Privacy Enhancing Technologies* 2016, 4 (2016), 403–417.
[30] Justin Hsu, Marco Gaboardi, Andreas Haeberlen, Sanjeev Khanna, Arjun Narayan, Benjamin C Pierce, and Aaron Roth. Differential privacy: An economic method for choosing epsilon. In *Computer Security Foundations Symposium*.
[31] Shiva Kasiviswanathan, Homin Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. 2011. What can we learn privately? *SIAM J. Comput.* 40, 3 (2011).
[32] Bryan Klimt and Yiming Yang. 2004. The Enron corpus: A new dataset for email classification research. In *European Conf. on Machine Learning*. Springer, 217–226.
[33] Aleksandra Korolova, Krishnaram Kenthapadi, Nina Mishra, and Alexandros Ntoulas. 2009. Releasing search queries and clicks privately. In *WebConf*. ACM.
[34] Hugo Larochelle and Stanislas Lauly. 2012. A neural autoregressive topic model. In *NeurIPS*. 2708–2716.
[35] Tiancheng Li and Ninghui Li. 2009. On the tradeoff between privacy and utility in data publishing. In *Proceedings of the 15th ACM SIGKDD*. ACM, 517–526.
[36] Andrew L Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the ACL*. 142–150.
[37] Rahat Masood, Dinusha Vatsalan, Muhammad Ikram, and Mohamed Ali Kaafar. 2018. Incognito: A Method for Obfuscating Web Data. In *WebConf*. 267–276.
[38] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*. 3111–3119.
[39] Arvind Narayanan and Vitaly Shmatikov. 2008. Robust de-anonymization of large sparse datasets. In *IEEE Symposium on Security and Privacy*. 111–125.
[40] Arvind Narayanan and Vitaly Shmatikov. 2009. De-anonymizing social networks. In *Security and Privacy, 2009 30th IEEE Symposium on*. IEEE, 173–187.
[41] Vijay Pandurangan. 2014. On taxis and rainbows: Lessons from NYC's improperly anonymized taxi logs. (2014).
[42] HweeHwa Pang, Xuhua Ding, and Xiaokui Xiao. 2010. Embellishing text search queries to protect user privacy. *VLDB Endowment* 3, 1-2 (2010), 598–607.
[43] Greg Pass, Abdur Chowdhury, and Cayley Torgeson. 2006. A picture of search.. In *InfoScale*, Vol. 152. 1.
[44] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*. 1532–1543.
[45] Albin Petit, Thomas Cerqueus, Sonia Ben Mokhtar, Lionel Brunie, and Harald Kosch. 2015. PEAS: Private, efficient and accurate web search. In *Trustcom*. IEEE.
[46] Alfréd Rényi. 1961. *On measures of entropy and information*. Technical Report. HUNGARIAN ACADEMY OF SCIENCES Budapest Hungary.
[47] David Sánchez and Montserrat Batet. 2016. C-sanitized: A privacy model for document redaction and sanitization. *JAIST* 67, 1 (2016), 148–163.
[48] David Sánchez, Jordi Castellà-Roca, and Alexandre Viejo. 2013. Knowledge-based scheme to create privacy-preserving but semantically-related queries for web search engines. *Information Sciences* 218 (2013), 17–30.
[49] Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims. 2015. Evaluation methods for unsupervised word embeddings. In *EMNLP*. 298–307.
[50] Reza Shokri and Vitaly Shmatikov. 2015. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC CCS*. ACM, 1310–1321.
[51] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *SP*. IEEE.
[52] Congzheng Song and Vitaly Shmatikov. 2019. Auditing Data Provenance in Text-Generation Models. In *ACM SIGKDD*. https://arxiv.org/pdf/1811.00513.pdf
[53] Ming Tan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2015. LSTM-based deep learning models for non-factoid answer selection. *arXiv:1511.04108* (2015).
[54] Abhradeep Guha Thakurta, Andrew H Vyrros, Umesh S Vaishampayan, Gaurav Kapoor, Julien Freudiger, Vivek Rangarajan Sridhar, and Doug Davidson. 2017. Learning new words. (2017). US Patent 9,594,741.
[55] Anthony Tockar. 2014. Riding with the stars: Passenger privacy in the NYC taxicab dataset. *Neustar Research, September* 15 (2014).
[56] Giridhari Venkatadri, Athanasios Andreou, Yabing Liu, Alan Mislove, Krishna Gummadi, Patrick Loiseau, and Oana Goga. 2018. Privacy Risks with Facebook's PII-based Targeting: Auditing a Data Broker's Advertising Interface. In *IEEE SP*.
[57] Isabel Wagner and David Eckhoff. 2018. Technical privacy metrics: a systematic survey. *ACM Computing Surveys (CSUR)* 51, 3 (2018), 57.
[58] Tianhao Wang, Jeremiah Blocki, Ninghui Li, and Somesh Jha. 2017. Locally differentially private protocols for frequency estimation. In *USENIX*. 729–745.
[59] Benjamin Weggenmann and Florian Kerschbaum. 2018. SynTF: Synthetic and Differentially Private Term Frequency Vectors for Privacy-Preserving Text Mining. In *The 41st International ACM SIGIR Conference (SIGIR '18)*. ACM, 305–314.
[60] Xi Wu, Fengan Li, Arun Kumar, Kamalika Chaudhuri, Somesh Jha, and Jeffrey Naughton. 2017. Bolt-on differential privacy for scalable stochastic gradient descent-based analytics. In *Proceedings of the 2017 ACM SIGMOD*. 1307–1322.