Research Challenges in Designing Differentially Private Text Generation Mechanisms

Oluwaseyi Feyisetan, Abhinav Aggarwal, Zekun Xu, Nathanael Teissier
Amazon
{sey,aggabhin,zeku,natteis}@amazon.com

Abstract

Accurately learning from user data while ensuring quantifiable privacy guarantees provides an opportunity to build better Machine Learning (ML) models while maintaining user trust. Recent literature has demonstrated the applicability of a generalized form of Differential Privacy to provide guarantees over text queries. Such mechanisms add privacy preserving noise to vectorial representations of text in high dimension and return a text based projection of the noisy vectors. However, these mechanisms are sub-optimal in their trade-off between privacy and utility. In this proposal paper, we describe some challenges in balancing this trade-off. At a high level, we provide two proposals: (1) a framework called LAC which defers some of the noise to a privacy amplification step and (2), an additional suite of three different techniques for calibrating the noise based on the local region around a word. Our objective in this paper is not to evaluate a single solution but to further the conversation on these challenges and chart pathways for building better mechanisms.

Privacy has emerged as a topic of strategic consequence across all computational fields – from machine learning, to natural language processing and statistics. Whether it is to satisfy compliance regulations, or build trust among customers, there is a general consensus about the need to provide privacy guarantees to users whose datasets serve as inputs to arbitrary functions provided by external processors. Within the mathematical and statistical disciplines, Differential Privacy (Dwork et al. 2006) has emerged as a gold standard for evaluating theoretical privacy claims. 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. Similarly, it is possible to train differentially private deep learning (Abadi et al. 2016) models by extending the methods from the statistical literature to the universal function approximators in neural networks. However, while Differential Privacy (DP) comes with strong theoretical guarantees, and the related literature is quite mature, DP private mechanisms for generating text is less studied.

As a result, within the field of traditional and computational linguistics, the norm is to apply anonymization techniques such as k-anonymity (Sweeney 2002) and its variants. While this offers a more intuitive way of expressing privacy guarantees as a function of an aggregation parameter k, all such methods are provably non-private (Korolova et al. 2009). Nevertheless, recent works such as (Fernandes, Dras, and McIver 2019; Feyisetan, Diethe, and Drake 2019; Feyisetan et al. 2020) have attempted to directly adapt the methods of DP to Natural Language Processing (NLP) by borrowing ideas from the privacy methods used for location data (Andrés et al. 2013). In DP, one way privacy is attained by adding 'properly calibrated noise' to the output of a mechanism (Dwork et al. 2006), or to gradient computations for deep learning (Abadi et al. 2016). The premise of such 'DP for text' methods is predicated on adding noise to the vector representation of words in a high dimensional embedding space, and projecting the noisy vectors back to the discrete vocabulary space.

Unlike statistical queries however, language generation comes with a unique set of problems. Consider a simple counting query where the objective is to return the number of people who exhibit a certain property x. The sensitivity of such a query is 1 since a new individual can only increase the count by 1. With text however, the sensitivity is much larger and is driven by the richness of the vocabulary, and how it is represented in the metric space under consideration. In this paper we propose strategies for increasing the utility of these 'DP for text' mechanisms by reducing the noise required while maintaining the desired privacy.

Privacy Implications and Threat Model. We consider a system where users generate training data (as text) which is then made available to an analyst. The analyst's utility requirement is to assess the quality of a downstream metric (e.g., ML model accuracy) derived from this data. The analyst therefore requires clear form access to the input data (e.g., for aggregation or annotation) to continuously improve downstream utility. In this model however, it is possible that the analyst learns more information about the user e.g., their *identity*, or some *property*, than is required to play their role of improving the utility metric. An example where textual data was used for re-identification can be seen with the AOL data release (Barbaro, Zeller, and Hansell 2006).

Challenges in Designing Private Text. Consider a set of n users, each with data $x_i \in \mathcal{X}$. Each user wishes to release up to m messages in a privacy preserving manner while maximizing the utility gained from the release of the messages. One approach is for each user to submit their messages $(x_{i,1},\dots,x_{i,m})$ in clear form to a *trusted* curator. The curator then proceeds to apply a privacy preserving randomized mechanism $\mathcal{R}(*)$ to the analysis $\mathcal{A}(x)$ on the aggregated data. The privacy mechanism works by *injecting noise* to the results of the analysis. This technique corresponds to the curator model of DP (Dwork et al. 2006), however, it requires that the users *trust* the curator. This is the proposed approach for preserving privacy in the upcoming U.S. Census (Abowd 2018). The curator model results in high utility since noise is applied only once on the aggregated data; however, a parallel approach cannot be clearly drawn for private text synthesis.

Another theoretical approach is for each user to apply the encoding or randomizing mechanism $\mathcal{R}:\mathcal{X}\rightarrow\mathcal{Y}^m$ to their data. The resulting $n\cdot m$ messages $(y_{i,1},\dots,y_{i,m})=\mathcal{R}(x_i)$ for each user is then passed to the curator for analysis $\mathcal{A}:\mathcal{Y}^*\rightarrow\mathcal{Z}$. This corresponds to Local DP (LDP) (Kasiviswanathan et al. 2011), since each user randomizes their data *locally*. The model provides stronger privacy guarantees in the presence of an *untrusted* curator. However, it incurs more error than the curator model because it requires multiple local $\mathcal{R}(x_i)$ transformations (as opposed to one by the trusted curator). As a result, it has mainly been successfully adopted by companies with large user bases (such as Microsoft (Ding, Kulkarni, and Yekhanin 2017), Google (Erlingsson, Pihur, and Korolova 2014), and Apple (Team 2017)) which compensate for the error. The local model is more amenable for text (Fernandes, Dras, and McIver 2019) and the literature builds on this framework.

The error accrued in the local model is exacerbated by the output range of the randomization function $\mathcal{R}(x_i)$. As an example, for one-bit messages (e.g., a coin flip) where $f:\mathcal{X}\rightarrow\{0,1\}$, the overall error goes down faster as the number of users increase, given the small output size of 2. Using a die roll with 6 outputs, the noise smooths out a bit slower. However, for analysis over vector representations of words $f:\mathcal{X}\rightarrow\mathbb{R}^d$, where d is the dimensionality of a word embedding model, and the number of words in the vocabulary could exceed thousands, the resulting analysis leads to far more noisy outputs. The noise (and by extension, the error) increases because of the DP promise, i.e., to guarantee privacy and protect all outliers, there must be a non-zero probability for transforming any given x to *any* other x'. We loosely correlate this size of the output space with the *sensitivity* of the function f. Therefore, when the sensitivity is large, more noise is required to preserve privacy.

The challenge with designing privacy mechanisms for text stems from these aforementioned issues. We observe that unlike the natural distribution of values over the number line, the vector representation of words in an embedding space tends to be non-uniform. The distance between words carries information as to their semantic similarities, and as a result, there are sparse regions and dense regions. Conversely, the privacy guarantees from differential privacy extends to every word in the entire space (leading to the large noise required to ascertain worst-case protections). This problem is not unique to the text space, however, it has been better studied in the statistical privacy literature. For example, the theoretical sensitivity for computing the median of an arbitrary set of numbers is infinite, but, in most dataset scenarios, the sensitivity is smaller as values coalesce around the median (Nissim, Raskhodnikova, and Smith 2007). Similar considerations have also been explored in private release of graph statistics (Blocki et al. 2013; Kasiviswanathan et al. 2013).

In this exploratory paper we examine these challenges from different lenses:

1. Can we reduce the noise by deferring additional privacy guarantees to other amplification mechanisms that do not require noise (e.g., sub-sampling, shuffling, k-aggregation have all been proposed in the literature (Li, Qardaji, and Su 2012; Bittau et al. 2017));
2. Can we re-calibrate the noise added such that it varies for every word depending on the density of the space surrounding the current word – rather than resorting to a single global sensitivity?

To address (1), we propose framing the private data release problem within the central DP (Erlingsson et al. 2019) paradigm by recommending a generalized form of the ESA protocol of (Bittau et al. 2017) which we denote as LAC. For (2), we propose three different methods that can be adopted to directly reduce the noise: density modulated noise, calibrating the noise to data sensitivity, and truncating the noise using a variety of approaches.

Preliminaries and Current Methods. We now give some preliminaries before providing details of our proposals.

Definition 1 (Differential Privacy (Dwork et al. 2006))**.** *A randomized algorithm $\mathcal{A}:\mathcal{X}^n\rightarrow\mathcal{Z}$ is ε-differentially private if for every pair of adjacent datasets $x\sim x'\in\mathcal{X}^n$ and every $\mathcal{Z}\subseteq Range(\mathcal{A})$, it holds that*

$$\mathsf{Pr}[\mathcal{A}(x)\in\mathcal{Z}]\leq e^{\varepsilon}\mathsf{Pr}[\mathcal{A}(x')\in\mathcal{Z}].$$

A DP algorithm protects a user by ensuring that its output distribution is approximately the same, whether or not the user was in the dataset used as an input to the algorithm. DP is usually achieved by applying noise drawn from a Laplace distribution scaled by the sensitivity of the analysis function.

Several pieces of research have demonstrated generalized DP (also known as $d_{\mathcal{X}}$ privacy) for different metric spaces and distance functions (Chatzikokolakis et al. 2013; Andrés et al. 2013; Chatzikokolakis, Palamidessi, and Stronati 2015; Feyisetan et al. 2020; Fernandes, Dras, and McIver 2019; Feyisetan, Diethe, and Drake 2019). For example, (Chatzikokolakis et al. 2013) demonstrated how the Manhattan distance metric was used to preserve privacy when releasing the number of days from a reference point. Similarly, the Chebyshev metric (chessboard distance) was adapted to perturb the output of smart meter readings (Chatzikokolakis et al. 2013) providing privacy with respect to TV channels being viewed. Further, the Euclidean distance was utilized by (Andrés et al. 2013; Chatzikokolakis, Palamidessi, and Stronati 2015) in a 2 dimensional coordinate system to privately report the location of users, and finally, (Fernandes, Dras, and McIver 2019) applied the Wasserstein metric in higher dimensions to demonstrate privacy preserving textual analysis using the metric space realized by word embeddings.

This work focuses on preserving privacy in high dimensional metric spaces equipped with the Euclidean metric. To achieve this form of metric differential privacy ($d_{\mathcal{X}}$ privacy), using a corollary to the Laplace mechanism, noise is sampled from an $n-$dimensional Laplacian and added to the output of the desired mechanism.

Proposal 1: Deferred Amplification

Our mechanism starts with a protocol similar to the privacy strategy of the local model. Given a set of n users, each with m data submissions $x_i \in \mathcal{X}$. Each user applies the $d_{\mathcal{X}}$ privacy mechanism $\mathcal{L} : \mathcal{X} \rightarrow \mathcal{Y}^m$ to their data. The resulting $n \cdot m$ messages $(y_{i,1}, \ldots, y_{i,m}) = \mathcal{L}(x_i)$ for each user is then passed to the curator $\mathcal{C} : \mathcal{Y}^* \rightarrow \mathcal{Z}$.

Our proposal includes an additional step that amplifies the privacy guarantees. Between the local noise injection $\mathcal{L}(x)$ and the curator analysis $\mathcal{C}(y)$, we introduce a privacy amplification step $\mathcal{A} : \mathcal{Y}^* \rightarrow \mathcal{Y}^*$ which takes in the result of the message perturbations from all the users $\mathcal{A}(\cup_{i=1}^{n} \mathcal{L}(x_i))$, amplifies the privacy and outputs it to the curator.

To get an intuition on how LAC can be used to improve utility while preserve privacy, consider the standard randomized response of (Warner 1965). Given a bit $b \in \{0, 1\}$ and privacy parameter ε. To output a privatized bit $\hat{b}$, we set $\hat{b} = b$ with probability $p = \frac{e^\varepsilon}{1+e^\varepsilon}$, otherwise $\hat{b} = 1 - b$. To improve the utility of this mechanism, we need to increase ε. However, in the local model, an adversary can map the output $\{\hat{b_1}, \ldots, \hat{b_n}\}$ to the n corresponding users. Therefore, the parameter p has to be close to $\frac{1}{2}$ otherwise $\hat{b} \approx b$ and the privacy guarantees are meaningless. Thus, to maintain the original (privacy) guarantees (while improving the utility), we need an additional mechanism that's different from the bit flipping noise addition. The desired property is such that the privacy guarantees are still meaningful when $p \ll \frac{1}{2}$.

In building composite DP algorithms, tools for *privacy amplification* are used to design mechanisms that provide additional guarantees than the initial privacy protocol.

Probably the most studied technique is privacy 'amplification by sub-sampling' (Chaudhuri and Mishra 2006; Kasiviswanathan et al. 2011), which states in its basic form that an ε-DP mechanism applied to a q fraction sub-sample of the initial population, yields an ε'-DP mechanism, where $\varepsilon' \approx q\varepsilon$. Other approaches such as (Li, Qardaji, and Su 2012) and (Feyisetan et al. 2019) have proposed augmenting sub-sampling with a $k-$anonymity parameter. Another class of amplification is by *contractive iteration* (Feldman et al. 2018) for privacy preserving ML models.

Amplification Model Spotlight: The Shuffler. In this work, we highlight the *shuffle* mechanism (Bittau et al. 2017; Erlingsson et al. 2019; Cheu et al. 2019; Balle et al. 2019) to amplify the privacy guarantees. While shuffling on its own offers no DP guarantees (unlike sub-sampling, which does), when combined with LDP, it has the advantage of maintaining the underlying statistics of the dataset by not 'throwing away' any of the data. The shuffler de-links data by masking its origin. For shuffling to be a viable amplification model, the Analyzer and Randomizer outputs must be amenable to shuffling, and not rely on any discriminating characteristics that link an individual to their contributions.

Algorithm 1: Composite privacy mechanism

```
// Localizer
Input: word $w \in \mathcal{W}$, parameters $m$, for each $n$ users
Output: word $\hat{w} \in \mathcal{W}$
for $i \in \{1, \ldots, m\}$ do
    Noise $\eta \sim$ Lap($\Delta_f / \varepsilon$)
    $\hat{\phi} = \phi(w) + \eta$
release $\hat{w}$
// Amplifier
Input: Multiset $\{\hat{w_i}\}_{i \in [n]}$, outputs of randomizers
Output: Multiset $\{\hat{w_i}\}_{i \in [n]}$, uniform permutes of [n]
for $i \in \{n-1, \ldots, 1\}$ do
    $j \leftarrow$ random integer such that $0 \leq j \leq i$
    exchange $w_i$ and $w_j$
release $\{w\}$
// Curator
Input: Multiset $\{y_i\}_{i \in [n]}$, with $y_i \in \mathcal{Y}$
Compute $z = \mathcal{A}(y)$
release $z$
```

The pseudo-code in Alg. 1 provides a high level overview of the composite privacy mechanism using a shuffler. Each user contributes their data which passes through a local privacy randomizer. The noisy outputs are then passed to a shuffler which permutes the order of the source of the perturbed data. The overall protocol $\mathcal{P}$, thus, consists of ($\mathcal{L}$, $\mathcal{A}$, $\mathcal{C}$) and is modeled around the *Encode, Shuffle, Analyze* (ESA) architecture of (Bittau et al. 2017).

In principle, shuffling can be implemented via multi-party computation, mixnets, running on secure hardware or via a trusted third party (Cheu et al. 2019; Bittau et al. 2017).

Selecting a Privacy Amplification Model. We provide some high level proposals:

Shuffler: can be used to generate text that's fed into linear classifiers with high utility. For example, a mechanism that outputs a sentiment class based on private perturbed data can still yield high utility on user de-linked and shuffled data.

Sub-sampler: For other use cases such as personalization which require some form of user linked data, a sub-sampler can be used instead of a shuffler. This will be more suitable if the data is reasonably uniform (without outliers).

K-threshold: with randomized sub-sampling can be used for cases where the underlying data follows a long tail distribution such as for annotating data in crowdsourcing or training generalized ML models with user data.

Proposal 2: Improved Randomizers

The randomizer $\mathcal{R}$ is based on the $d_{\mathcal{X}}$ metric privacy mechanism described by (Feyisetan et al. 2020) on word embeddings where the distance between word vectors is represented as the Euclidean metric. A similar mechanism was also proposed by (Fernandes, Dras, and McIver 2019), however, the distance metric was the Earth mover distance. Similarly, (Feyisetan, Diethe, and Drake 2019) extended the model to demonstrate preserving privacy using noise sampled from Hyperbolic space. The metric space of interest is as defined by word embedding models which organize discrete words in a continuous space such that the similarity in the space reflects their semantic affinity. Models such as WORD2VEC (Mikolov et al. 2013), GLOVE (Pennington, Socher, and Manning 2014), and FASTTEXT (Bojanowski et al. 2017) create such a mapping $\phi : \mathcal{W} \to \mathbb{R}^d$, where the distance function is expressed as $d : \mathcal{W} \times \mathcal{W} \to [0, \infty)$. The distance $d(w, w')$ between a pair of words is therefore given as $\|\phi(w) - \phi(w')\|$, where $\|\cdot\|$ is the Euclidean norm on $\mathbb{R}^d$.

This mechanism however leads to sub-optimal accuracies due to a lack of uniformity in the embedding space. In particular, to achieve a certain level of privacy protection, the amount of noise is controlled by the worst-case word, which roughly corresponds to the word whose embedding is farther apart from any other word (i.e., the global sensitivity). Therefore, at a given level of ε, a unique word like *nudiustertian* will be perturbed similarly to a common word like *drunk* which has over $2,000$ possible synonyms[1]. To improve on this, we propose a variation of the original mechanism that can provide a fixed level of plausible deniability (Bindschaedler, Shokri, and Gunter 2017), measured in terms of the proxy statistics of (Feyisetan et al. 2020) with less noise, thus yielding more accuracy. In other words, the improved mechanisms should provide the same level of plausible deniability as the original mechanism, but under a larger value of ε. To achieve this goal, we propose:

1. Defining a prior to account for the space variability.
2. Calibrating the noise to the local sensitivity of the space.
3. Adopting a truncated noise mechanism.

Density-Modulated Noise. We observe that the algorithm from (Feyisetan et al. 2020) can be interpreted as an instance of the exponential mechanism (McSherry and Talwar 2007) together with a post-processing step. Further, noise sampling via the exponential mechanism assumes a base measure $\mu(z)$ with a uniform distribution over the feasible range. Accordingly, the algorithm can be expanded as $p_N(z) \propto \mu(z) \times \exp(-\varepsilon\|z\|)$. However, the distribution of words in $\mathbb{R}^d$ is not uniform over the embedding space. As a consequence of Zipf's law, some words occur more frequently in a dataset and are surrounded by dense regions of similar words in the embedding space.

A natural way to "bias" an exponential mechanism without changing its privacy properties is to modulate it with a public "prior" $\mu(z)$. For example, such a prior can be constructed over a publicly available corpus such as Wikipedia or Common Crawl. The question we address in this section is whether we can design an appropriate, potentially unnormalized, prior such that the resulting exponential mechanism that samples from $p_N(z) \propto \mu(z) \times \exp(-\varepsilon\|z\|)$ provides more accurate answers than the original mechanism under similar privacy constraints. An important research challenge in this direction is that by incorporating this correction to improve accuracy, we might end up with a mechanism that is computationally hard to sample from.

To obtain a prior that will solve the non-uniformity in the privacy mechanism using a vanilla word embedding is to modulate the distribution by a prior that captures the distribution of words in $\mathbb{R}^d$ induced by the word embedding. By introducing a prior that assigns high probability to dense areas of the embedding and low probability to sparse areas of the embedding, we can achieve the same level of plausible deniability statistics with smaller values of ε, hence, mitigating the worst-case effect that is observed in the unmodulated mechanism around sparse areas of the embedding.

One way to produce this prior measure $\mu(z)$ is to take a kernel density estimator with Radial Basis Function kernels on the resulting embedding, i.e., $\mu(z) \propto \sum_{u \in \mathcal{W}} \exp\left(-\|z - \phi(u)\|^2 / 2\sigma^2\right)$ for some tuned variance σ^2. However, it is not immediately clear how to sample from the modulated mechanism that on input w has density $p_N(z) \propto \mu(z) \times \exp(-\varepsilon\|z\|)$ for $\mu(z)$ defined above. Rather than sampling directly, we can either opt for an approximation to the distribution, or adopt indirect sampling strategies such as the Metropolis–Hastings algorithm.

Another observation is that we don't need to pay the cost of an expensive sampling every time we want to use the mechanism. Instead, by introducing the projection step of the sampled vector to the closest word embedding, we can represent the mechanism by a $W \times W$ matrix containing the probabilities $\Pr[M(w) = w']$, where M is the complete mechanism. We can precompute and store these $||W||^2$ probabilities and then use this matrix to define the output distribution every time we run the mechanism.

Calibrating Noise to Data Sensitivity. In the proof of (Feyisetan et al. 2020), $\hat{w}$ is calibrated at the worst-case distance T from w and w' which is analogous to the global sensitivity. We can however, have a data dependent sensitivity definition over the metric space:

Definition 2 (Local sensitivity (Nissim, Raskhodnikova, and Smith 2007))**.** *The local sensitivity of a function $f : \mathcal{X}^n \to \mathbb{R}^d$ is given for $x \sim x' \in \mathcal{X}$ as,*

$$\Delta_{\mathcal{L}_f} = \max_{x' : d(x,x')=1} \|f(x) - f(x')\|_1$$

The local sensitivity of f with respect to x is how much $f(x')$ can differ from $f(x)$ for any x' adjacent to the input x (and not any possible entry x). We observe that the global sensitivity $\Delta_{\mathcal{G}_f} = \max_x \Delta_{\mathcal{L}_f}(x)$. However, a mechanism that adds noise scaled to the local sensitivity does not preserve DP as the noise magnitude can leak information (Nissim, Raskhodnikova, and Smith 2007). To address this, for example, (Nissim, Raskhodnikova, and Smith 2007) adds noise calibrated to a *smooth bound* on the local sensitivity. The noise is typically sampled from the Laplace distribution. Thus, if we consider $\hat{w}$ at a distance $0 < t < T$, then the local sensitivity $\Delta_{\mathcal{L}_f}$ is:

$$\Delta_{\mathcal{L}_f}{}^{(t)} = \max_{w' : d(w,w') \leq t} \Delta_{\mathcal{L}_f}. \qquad (1)$$

However, for our rare word $w =$*nudiustertian*, the local sensitivity might still leak information on output $\hat{w}$. As a result, we can construct the smooth sensitivity $\Delta_{\mathcal{S}_f}$ as a $\beta-$smooth upper bound (Nissim, Raskhodnikova, and Smith 2007) on the local sensitivity. The desired properties of the bound include that:

[1] https://www.mhpbooks.com/books/drunk/

$$(1)\ \forall w \in \mathcal{W} : \qquad \Delta_{\mathcal{S}_f}(w) \geq \Delta_{\mathcal{L}_f}(w)$$
$$(2)\ \forall w, w' \in \mathcal{W} : \qquad \Delta_{\mathcal{S}_f}(w) \leq e^{\beta} \cdot \Delta_{\mathcal{S}_f}(w')$$

Observe that the smooth bound is equal to the global sensitivity $\Delta_{\mathcal{G}_f}$ when $\beta = 0$. Therefore, the smallest function $\Delta_{\mathcal{S}^*_{f,\beta}}$ that satisfies the two stated properties is the smooth sensitivity of the underlying function f and can be stated as: $\Delta_{\mathcal{S}^*_{f,\beta}}(w) = \max_{w':d(w,w')\leq t} \left(\Delta^{(t)}_{\mathcal{L}_f}(w') \cdot e^{-\beta d(w,w')}\right)$

However, we cannot describe the local and smooth sensitivity this way since the local sensitivity construction in Def 2 was defined for integer-valued metrics (such as the Hamming distance). To translate this to real-valued metrics as is required for $d_{\mathcal{X}}$-privacy, we can adopt the approach of (Laud, Pankova, and Pettai 2020) for defining the local sensitivity in metric spaces.

First, we consider each word embedding vector as a point in some Banach space. A Banach space is a vector space with a metric that allows the computation of vector length and distance between vectors. For example, our $n-$dimensional Euclidean space of word embeddings, with the Euclidean norm is a Banach space.

Next, we observe from (Kasiviswanathan et al. 2013) that the local sensitivity of a function is similar to its derivative (e.g., taking the limits in Eqn 1 as $t \to 0$). Therefore, the aim is to find an analog of a suitable derivative for continuous functions. One option is for the local sensitivity to be defined as the *Fréchet derivative* in Banach spaces. (Laud, Pankova, and Pettai 2020) described an approach for this and they demonstrated how to apply noise sampled from the Cauchy distribution to satisfy the DP guarantees. Additional research would be needed to explore the direct application of the method of local (and then smooth) sensitivity calibration to embedding spaces.

Truncated Noise Mechanisms The standard $d_{\mathcal{X}}$-privacy mechanisms were designed by borrowing ideas from the privacy methods used for location data (Andrés et al. 2013). One of the proposed approaches in that work was to truncate the mechanism to report only points within the limits of the area of interest. To achieve this, they define an 'acceptable area' of admissible points $\mathcal{A} \subset \mathbb{R}^2$ (i.e., location privacy in $2-d$ space) beyond which results are truncated to the closest point in $\mathcal{A}$. Other truncation mechanisms have been explored in traditional DP including the truncated Laplacian (Geng et al. 2018), and truncated geometric mechanism within the context of $d_{\mathcal{X}}$-privacy (Chatzikokolakis et al. 2013).

Designing a corollary for text based $d_{\mathcal{X}}$-privacy requires an approach to setting the truncation bounds while maintaining the privacy guarantees. We identify 2 potential ways of achieving this: (1) Distance based truncation; and (2) K-nearest neighbor based truncation. To achieve (1) we can define a distance based limit similar to (Andrés et al. 2013). In this approach, a word can only get perturbed to words within the distance-defined admissible area $\mathcal{A} \in \mathcal{U}$. The maximum distance τ between a word and the farthest word in $\mathcal{A}$ is defined and fixed a-priori. To handle words that fall outside the noise limits, (Andrés et al. 2013) proposes a discretization step to select the closest word in $\mathcal{A}$. Another option is for the mechanism to concentrate the probability of selecting a word to the admissible area $\mathcal{A}$, while assigning a residual probability to satisfy the DP guarantee on the entire set $\mathcal{U}$. Therefore, when the noise exceeds the distance τ, a replacement is randomly drawn from the set $\mathcal{U} - \mathcal{A}$.

The downside of this approach is seen in regions of sharply varying density e.g., in the embedding space where one word has $2,000$ synonyms which potentially fall within $\mathcal{A}$ and the rare word with no neighbors in $\mathcal{A}$. Therefore to achieve (2) rather than having a fixed distance from each word, we can also define the (randomized) $k-$closest words as delineating our acceptable area. One potential benefit of this is, in dense spaces, we can select closer candidates while simultaneously guaranteeing that isolated words are replaced by one of their $k-$nearest neighbors regardless of how far off it is.

Implementing either of these mechanisms however come with their own set of challenges. For example, there isn't a direct way to set a maximum distance when drawing the multivariate laplacian noise that was proposed by (Feyisetan et al. 2020). One option will be to fix τ, or the distance to the max randomized k as the local sensitivity. Another option will be to rethink the entire design of the randomizers such that the noise is not added to the vector representation of the words, but to these τ distances.

Connections to Related Work The traditional DP literature contains techniques to limit the privacy preserving noise added to a mechanism. In one work, (Nissim, Raskhodnikova, and Smith 2007) introduced the notion of smooth sensitivity where a smooth upper bound on the local sensitivity is used to determine how much noise is added.

Similarly, (Dwork and Lei 2009) introduced a paradigm called *Propose-Test-Release* (PTR) where: the algorithm proposes a bound on sensitivity, tests the adequacy of the bound on the dataset, and halts if the sensitivity is too high.

In related work, (Kasiviswanathan et al. 2013) extended the notion of limiting the noise for private graph analysis where the degree bound (a function of the number of nodes in the graph) can be arbitrary. To achieve this, they set a D bound on the graph which aims to keep the sensitivity low while retaining as large a fraction of the graph as possible.

These all describe principled approaches to limit the magnitude of noise applied to a privacy preserving mechanism in the contexts of statistical and graph analysis by redefining the sensitivity that controls the noise. As opposed to the reviewed techniques, our representations are within a metric space defined by word embeddings.

Conclusion and Future work In this proposal paper, we surveyed some of the challenges of building differentially private mechanisms for generating text based on word embeddings. We investigated approaches built on the $d_{\mathcal{X}}$-privacy framework in Euclidean space. The core issues stem from the non-uniformity of the metric space defined by embeddings and the need to provide worst case guarantees for outliers as required by differential privacy. This necessitates a large amount of noise thus leading to utility impacts on downstream tasks that rely on the generated text as input.

Our approach was to explore the resulting utility issues from different perspectives: first, considering methods of reducing the required noise by deferring additional guarantees to other privacy amplification mechanisms that do not require noise (such as shuffling). We then proposed three ways to reduce the needed noise by accounting for the density around the word under consideration. These included introducing a prior, re-calibrating the noise, or truncating the noise. In future work, we plan to explore these approaches in detail and provide a study on what works, when it works, and why. Our aim is to provide a principled approach to studying these mechanisms in order to accelerate the research and drive adoption.

References

Abadi, M.; Chu, A.; Goodfellow, I.; McMahan, H. B.; Mironov, I.; Talwar, K.; and Zhang, L. 2016. Deep learning with differential privacy. In *ACM SIGSAC CCS*, 308–318.

Abowd, J. M. 2018. The US census bureau adopts differential privacy. In *ACM SIGKDD*, 2867–2867. ACM.

Andrés, M. E.; Bordenabe, N. E.; Chatzikokolakis, K.; and Palamidessi, C. 2013. Geo-indistinguishability: Differential privacy for location-based systems. In *ACM CCS*, 901–914.

Balle, B.; Bell, J.; Gascón, A.; and Nissim, K. 2019. The privacy blanket of the shuffle model. In *CRYPTO*, 638–667.

Barbaro, M.; Zeller, T.; and Hansell, S. 2006. A face is exposed for AOL searcher no. 4417749. *New York Times*.

Bindschaedler, V.; Shokri, R.; and Gunter, C. A. 2017. Plausible deniability for privacy-preserving data synthesis. *VLDB Endowment* 10(5):481–492.

Bittau, A.; Erlingsson, Ú.; Maniatis, P.; Mironov, I.; Raghunathan, A.; Lie, D.; Rudominer, M.; et al. 2017. Prochlo: Strong privacy for analytics in the crowd. In *SOSP*.

Blocki, J.; Blum, A.; Datta, A.; and Sheffet, O. 2013. Differentially private data analysis of social networks via restricted sensitivity. In *ITCS*, 87–96. ACM.

Bojanowski, P.; Grave, E.; Joulin, A.; and Mikolov, T. 2017. Enriching word vectors with subword information. *TACL* 5.

Chatzikokolakis, K.; Andrés, M. E.; Bordenabe, N. E.; and Palamidessi, C. 2013. Broadening the scope of differential privacy using metrics. In *PETS*.

Chatzikokolakis, K.; Palamidessi, C.; and Stronati, M. 2015. Constructing elastic distinguishability metrics for location privacy. *PETS*.

Chaudhuri, K., and Mishra, N. 2006. When random sampling preserves privacy. In *CRYPTO*, 198–213. Springer.

Cheu, A.; Smith, A.; Ullman, J.; Zeber, D.; and Zhilyaev, M. 2019. Distributed differential privacy via shuffling. In *EUROCRYPT*, 375–403. Springer.

Ding, B.; Kulkarni, J.; and Yekhanin, S. 2017. Collecting telemetry data privately. In *NeurIPS*.

Dwork, C., and Lei, J. 2009. Differential privacy and robust statistics. In *STOC*, volume 9, 371–380.

Dwork, C.; McSherry, F.; Nissim, K.; and Smith, A. 2006. Calibrating noise to sensitivity in private data analysis. In *TCC*, 265–284. Springer.

Erlingsson, Ú.; Feldman, V.; Mironov, I.; Raghunathan, A.; Talwar, K.; and Thakurta, A. 2019. Amplification by shuffling: From local to central differential privacy via anonymity. In *ACM-SIAM*.

Erlingsson, Ú.; Pihur, V.; and Korolova, A. 2014. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *ACM SIGSAC CCS*.

Feldman, V.; Mironov, I.; Talwar, K.; and Thakurta, A. 2018. Privacy amplification by iteration. In *FOCS*. IEEE.

Fernandes, N.; Dras, M.; and McIver, A. 2019. Generalised differential privacy for text document processing. *POST*.

Feyisetan, O.; Drake, T.; Balle, B.; and Diethe, T. 2019. Privacy-preserving active learning on sensitive data for user intent classification. *arXiv preprint arXiv:1903.11112*.

Feyisetan, O.; Balle, B.; Drake, T.; and Diethe, T. 2020. Privacy- and utility-preserving textual analysis via calibrated multivariate perturbations. In *ACM WSDM*.

Feyisetan, O.; Diethe, T.; and Drake, T. 2019. Leveraging hierarchical representations for preserving privacy and utility in text. In *IEEE ICDM*.

Geng, Q.; Ding, W.; Guo, R.; and Kumar, S. 2018. Truncated laplacian mechanism for approximate differential privacy. *arXiv preprint arXiv:1810.00877*.

Kasiviswanathan, S.; Lee, H.; Nissim, K.; Raskhodnikova, S.; and Smith, A. 2011. What can we learn privately? *SIAM Journal on Computing* 40(3).

Kasiviswanathan, S. P.; Nissim, K.; Raskhodnikova, S.; and Smith, A. 2013. Analyzing graphs with node differential privacy. In *TCC*, 457–476. Springer.

Korolova, A.; Kenthapadi, K.; Mishra, N.; and Ntoulas, A. 2009. Releasing search queries and clicks privately. In *WebConf*. ACM.

Laud, P.; Pankova, A.; and Pettai, M. 2020. A framework of metrics for differential privacy from local sensitivity. *POPETS* 2020(2):175–208.

Li, N.; Qardaji, W.; and Su, D. 2012. On sampling, anonymization, and differential privacy or, k-anonymization meets differential privacy. In *ASIA-CCS*, 32–33.

McSherry, F., and Talwar, K. 2007. Mechanism design via differential privacy. In *FOCS*, volume 7, 94–103.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *NeurIPS*, 3111–3119.

Nissim, K.; Raskhodnikova, S.; and Smith, A. 2007. Smooth sensitivity and sampling in private data analysis. In *STOC*, 75–84. ACM.

Pennington, J.; Socher, R.; and Manning, C. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Sweeney, L. 2002. k-anonymity: A model for protecting privacy. *IJUFKS* 10(05):557–570.

Team, A. D. P. 2017. Learning with privacy at scale. *Apple Machine Learning Journal* 1(9).

Warner, S. L. 1965. Randomized response: A survey technique for eliminating evasive answer bias. *JASA* 60(309):63–69.