Generalised Differential Privacy for Text Document Processing

Natasha Fernandes[1,2(✉)], Mark Dras[1], and Annabelle McIver[1]

[1] Macquarie University, Sydney, Australia
natasha.fernandes@hdr.mq.edu.au
[2] Inria, Paris-Saclay and École Polytechnique, Palaiseau, France

Abstract. We address the problem of how to "obfuscate" texts by removing stylistic clues which can identify authorship, whilst preserving (as much as possible) the content of the text. In this paper we combine ideas from "generalised differential privacy" and machine learning techniques for text processing to model privacy for text documents. We define a privacy mechanism that operates at the level of text documents represented as "bags-of-words"—these representations are typical in machine learning and contain sufficient information to carry out many kinds of classification tasks including *topic identification* and *authorship attribution* (of the original documents). We show that our mechanism satisfies privacy with respect to a metric for semantic similarity, thereby providing a balance between utility, defined by the semantic content of texts, with the obfuscation of stylistic clues. We demonstrate our implementation on a "fan fiction" dataset, confirming that it is indeed possible to disguise writing style effectively whilst preserving enough information and variation for accurate content classification tasks. We refer the reader to our complete paper [15] which contains full proofs and further experimentation details.

Keywords: Generalised differential privacy · Earth Mover's metric · Natural language processing · Author obfuscation

1 Introduction

Partial public release of formerly classified data incurs the risk that more information is disclosed than intended. This is particularly true of data in the form of text such as government documents or patient health records. Nevertheless there are sometimes compelling reasons for declassifying data in some kind of "sanitised" form—for example government documents are frequently released as redacted reports when the law demands it, and health records are often shared to facilitate medical research. Sanitisation is most commonly carried out by hand but, aside from the cost incurred in time and money, this approach provides no guarantee that the original privacy or security concerns are met.

We acknowledge the support of the Australian Research Council Grant DP140101119.

F. Nielson and D. Sands (Eds.): POST 2019, LNCS 11426, pp. 123–148, 2019.
https://doi.org/10.1007/978-3-030-17138-4_6

To encourage researchers to focus on privacy issues related to text documents the digital forensics community PAN@Clef ([41], for example) proposed a number of challenges that are typically tackled using *machine learning*. In this paper our aim is to demonstrate how to use ideas from *differential privacy* to address some aspects of the PAN@Clef challenges by showing how to provide strong a priori privacy guarantees in document disclosures.

We focus on the problem of *author obfuscation*, namely to automate the process of changing a given document so that as much as possible of its original substance remains, but that the author of the document can no longer be identified. Author obfuscation is very difficult to achieve because it is not clear exactly what to change that would sufficiently mask the author's identity. In fact author properties can be determined by "writing style" with a high degree of accuracy: this can include author identity [28] or other undisclosed personal attributes such as native language [33,51], gender or age [16,27]. These techniques have been deployed in real world scenarios: native language identification was used as part of the effort to identify the anonymous perpetrators of the 2014 Sony hack [17], and it is believed that the US NSA used author attribution techniques to uncover the identity of the real humans behind the fictitious persona of Bitcoin "creator" Satoshi Nakamoto.[1]

Our contribution concentrates on the perspective of the "machine learner" as an adversary that works with the standard "bag-of-words" representation of documents often used in text processing tasks. A *bag-of-words* representation retains only the original document's words and their frequency (thus forgetting the order in which the words occur). Remarkably this representation still contains sufficient information to enable the original authors to be identified (by a stylistic analysis) *as well as* the document's topic to be classified, both with a significant degree of accuracy.[2] Within this context we reframe the PAN@Clef author obfuscation challenge as follows:

> Given an input bag-of-words representation of a text document, provide a mechanism which changes the input without disturbing its topic classification, but that the author can no longer be identified.

In the rest of the paper we use ideas inspired by $d_{\mathcal{X}}$*-privacy* [9], a metric-based extension of differential privacy, to implement an automated privacy mechanism which, unlike current ad hoc approaches to author obfuscation, gives access to both solid privacy and utility guarantees.[3]

[1] https://medium.com/cryptomuse/how-the-nsa-caught-satoshi-nakamoto-868affcef595.

[2] This includes, for example, the character n-gram representation used for author identification in [29].

[3] Our notion of utility here is similar to other work aiming at text privacy, such as [32,53].

We implement a mechanism K which takes b, b' bag-of-words inputs and produces "noisy" bag-of-words outputs determined by $K(b), K(b')$ with the following properties:

Privacy: If b, b' are classified to be "similar in topic" then, depending on a privacy parameter ϵ the *outputs* determined by $K(b)$ and $K(b')$ are also "similar to each other", irrespective of authorship.

Utility: Possible outputs determined by $K(b)$ are distributed according to a Laplace probability density function scored according to a semantic similarity metric.

In what follows we define *semantic similarity* in terms of the classic *Earth Mover's distance* used in machine learning for topic classification in text document processing.[4] We explain how to combine this with $d_{\mathcal{X}}$-privacy which extends privacy for databases to other unstructured domains (such as texts).

In Sect. 2 we set out the details of the bag-of-words representation of documents and define the Earth Mover's metric for topic classification. In Sect. 3 we define a generic mechanism which satisfies "$E_{d_{\mathcal{X}}}$-privacy" relative to the Earth Mover's metric $E_{d_{\mathcal{X}}}$ and show how to use it for our obfuscation problem. We note that our generic mechanism is of independent interest for other domains where the Earth Mover's metric applies. In Sect. 4 we describe how to implement the mechanism for data represented as real-valued vectors and prove its privacy/utility properties with respect to the Earth Mover's metric; in Sect. 5 we show how this applies to bags-of-words. Finally in Sect. 6 we provide an experimental evaluation of our obfuscation mechanism, and discuss the implications.

Throughout we assume standard definitions of probability spaces [18]. For a set $\mathcal{A}$ we write $\mathbb{D}\mathcal{A}$ for the set of (possibly continuous) probability distributions over $\mathcal{A}$. For $\eta \in \mathbb{D}\mathcal{A}$, and $A \subseteq \mathcal{A}$ a (measurable) subset we write $\eta(A)$ for the probability that (wrt. η) a randomly selected a is contained in A. In the special case of singleton sets, we write $\eta\{a\}$. If mechanism $K : \alpha \to \mathbb{D}\alpha$, we write $K(a)(A)$ for the probability that if the input is a, then the output will be contained in A.

2 Documents, Topic Classification and Earth Moving

In this section we summarise the elements from machine learning and text processing needed for this paper. Our first definition sets out the representation for documents we shall use throughout. It is a typical representation of text documents used in a variety of classification tasks.

Definition 1. *Let $\mathcal{S}$ be the set of all words (drawn from a finite alphabet). A* document *is defined to be a finite bag over $\mathcal{S}$, also called a* bag-of-words. *We denote the set of documents as $\mathbb{B}\mathcal{S}$, i.e. the set of (finite) bags over $\mathcal{S}$.*

[4] In NLP, this distance measure is known as the Word Mover's distance. We use the classic Earth Mover's here for generality.

Once a text is represented as a bag-of-words, depending on the processing task, further representations of the words within the bag are usually required. We shall focus on two important representations: the first is when the task is semantic analysis for eg. topic classification, and the second is when the task is author identification. We describe the representation for topic classification in this section, and leave the representation for author identification for Sects. 5 and 6.

2.1 Word Embeddings

Machine learners can be trained to classify the topic of a document, such as "health", "sport", "entertainment"; this notion of topic means that the words within documents will have particular semantic relationships to each other. There are many ways to do this classification, and in this paper we use a technique that has as a key component "word embeddings", which we summarise briefly here.

A *word embedding* is a real-valued vector representation of words where the precise representation has been experimentally determined by a neural network sensitive to the way words are used in sentences [38]. Such embeddings have some interesting properties, but here we only rely on the fact that when the embeddings are compared using a distance determined by a pseudometric[5] on $\mathbb{R}^n$, words with similar meanings are found to be close together as word embeddings, and words which are significantly different in meaning are far apart as word embeddings.

Definition 2. *An n-dimensional word embedding is a mapping* $Vec : \mathcal{S} \to \mathbb{R}^n$. *Given a pseudometric dist on* $\mathbb{R}^n$ *we define a distance on words* $dist_{Vec} : \mathcal{S}\times\mathcal{S}\to\mathbb{R}_{\geq}$ *as follows:*

$$dist_{Vec}(w_1, w_2) \;:=\; dist(\,Vec(w_1),\, Vec(w_2))\;.$$

Observe that the property of a pseudometric on $\mathbb{R}^n$ carries over to $\mathcal{S}$.

Lemma 1. *If dist is a pseudometric on* $\mathbb{R}^n$ *then* $dist_{Vec}$ *is also a pseudometric on* $\mathcal{S}$.

Proof. Immediate from the definition of a pseudometric: i.e. the triangle equality and the symmetry of $dist_{Vec}$ *are inherited from dist.*

Word embeddings are particularly suited to language analysis tasks, including topic classification, due to their useful semantic properties. Their effectiveness depends on the quality of the embedding *Vec*, which can vary depending on the size and quality of the training data. We provide more details of the particular

[5] Recall that a pseudometric satisfies both the triangle inequality and symmetry; but different words could be mapped to the same vector and so $dist_{Vec}(w_1, w_2) = 0$ no longer implies that $w_1 = w_2$.

embeddings in Sect. 6. Topic classifiers can also differ on the choice of underlying metric *dist*, and we discuss variations in Sect. 3.2.

In addition, once the word embedding *Vec* has been determined, and the distance *dist* has been selected for comparing "word meanings", there are a variety of semantic similarity measures that can be used to compare documents, for us bags-of-words. In this work we use the "Word Mover's Distance", which was shown to perform well across multiple text classification tasks [31].

The *Word Mover's Distance* is based on the classic *Earth Mover's Distance* [43] used in transportation problems with a given distance measure. We shall use the more general Earth Mover's definition with *dist*[6] as the underlying distance measure between words. We note that our results can be applied to problems outside of the text processing domain.

Let $X, Y \in \mathbb{B}\mathcal{S}$; we denote by X the tuple $\langle x_1^{a_1}, x_2^{a_2}, \ldots, x_k^{a_k} \rangle$, where a_i is the number of times that x_i occurs in X. Similarly we write $Y = \langle y_1^{b_1}, y_2^{b_2}, \ldots, y_l^{b_l} \rangle$; we have $\sum_i a_i = |X|$ and $\sum_j b_j = |Y|$, the sizes of X and Y respectively. We define a *flow matrix* $F \in \mathbb{R}_{\geq 0}^{k \times l}$ where F_{ij} represents the (non-negative) amount of flow from $x_i \in X$ to $y_j \in Y$.

Definition 3 *(Earth Mover's Distance). Let $d_{\mathcal{S}}$ be a (pseudo)metric over $\mathcal{S}$. The Earth Mover's Distance with respect to $d_{\mathcal{S}}$, denoted by $E_{d_{\mathcal{S}}}$, is the solution to the following linear optimisation:*

$$E_{d_{\mathcal{S}}}(X, Y) \quad := \quad \min \sum_{x_i \in X} \sum_{y_j \in Y} d_{\mathcal{S}}(x_i, y_j) F_{ij} \ , \quad \textit{subject to:} \tag{1}$$

$$\sum_{i=1}^{k} F_{ij} = \frac{b_j}{|Y|} \quad \textit{and} \quad \sum_{j=1}^{l} F_{ij} = \frac{a_i}{|X|} \ , \quad F_{ij} \geq 0, \quad 1 \leq i \leq k, 1 \leq j \leq l \tag{2}$$

where the minimum in (1) is over all possible flow matrices F subject to the constraints (2). In the special case that $|X| = |Y|$, the solution is known to satisfy the conditions of a (pseudo)metric [43] which we call the Earth Mover's Metric.

In this paper we are interested in the special case $|X| = |Y|$, hence we use the term *Earth Mover's metric* to refer to $E_{d_{\mathcal{S}}}$.

We end this section by describing how texts are prepared for machine learning tasks, and how Definition 3 is used to distinguish documents. Consider the text snippet "The President greets the press in Chicago". The first thing is to remove all "stopwords" – these are words which do not contribute to semantics, and include things like prepositions, pronouns and articles. The words remaining are those that contain a great deal of semantic and stylistic traits.[7]

[6] In our experiments we take *dist* to be defined by the Euclidean distance.

[7] In fact the way that stopwords are used in texts turn out to be characteristic features of authorship. Here we follow standard practice in natural language processing to remove them for efficiency purposes and study the privacy of what remains. All of our results apply equally well had we left stopwords in place.

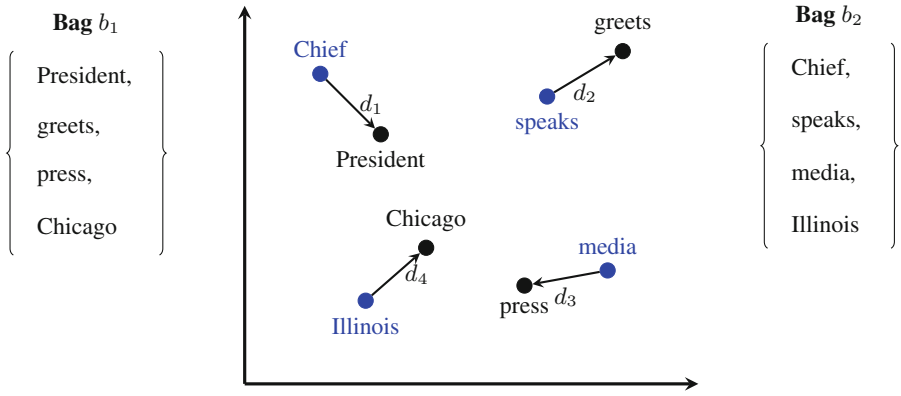

Fig. 1. Earth Mover's metric between sample documents.

In this case we obtain the bag:

$$b_1 \quad := \quad \langle \text{President}^1,\ \text{greets}^1,\ \text{press}^1,\ \text{Chicago}^1 \rangle\ .$$

Consider a second bag: $b_2 := \langle \text{Chief}^1, \text{speaks}^1, \text{media}^1, \text{Illinois}^1 \rangle$, corresponding to a different text. Figure 1 illustrates the optimal flow matrix which solves the optimisation problem in Definition 3 relative to $d_{\mathcal{S}}$. Here each word is mapped completely to another word, so that $F_{i,j} = 1/4$ when $i = j$ and 0 otherwise. We show later that this is always the case between bags of the same size. With these choices we can compute the distance between b_1, b_2:

$$\begin{aligned} E_{d_{\mathcal{S}}}(b_1, b_2) = \frac{1}{4}(&d_{\mathcal{S}}(\text{President}, \text{Chief}) + d_{\mathcal{S}}(\text{greets}, \text{speaks}) + \\ &d_{\mathcal{S}}(\text{press}, \text{media}) + d_{\mathcal{S}}(\text{Chicago}, \text{Illinois})) \\ = 2.816\ .& \end{aligned} \tag{3}$$

For comparison, consider the distance between b_1 and b_2 to a third document, $b_3 := \langle \text{Chef}^1, \text{breaks}^1, \text{cooking}^1, \text{record}^1 \rangle$. Using the same word embedding metric,[8] we find that $E_{d_{\mathcal{S}}}(b_1, b_3) = 4.121$ and $E_{d_{\mathcal{S}}}(b_2, b_3) = 3.941$. Thus b_1, b_2 would be classified as semantically "closer" to each other than to b_3, in line with our own (linguistic) interpretation of the original texts.

3 Differential Privacy and the Earth Mover's Metric

Differential Privacy was originally defined with the protection of individuals' data in mind. The intuition is that privacy is achieved through "plausible deniability", i.e. whatever output is obtained from a query, it could have just as

[8] We use the same word2vec-based metric as per our experiments; this is described in Sect. 6.

easily have arisen from a database that does not contain an individual's details, as from one that does. In particular, there should be no easy way to distinguish between the two possibilities. Privacy in text processing means something a little different. A "query" corresponds to releasing the topic-related contents of the document (in our case the bag-of-words)—this relates to the utility because we would like to reveal the semantic content. The privacy relates to investing *individual documents* with plausible deniability, rather than *individual authors* directly. What this means for privacy is the following. Suppose we are given two documents b_1, b_2 written by two distinct authors A_1, A_2, and suppose further that b_1, b_2 are changed through a privacy mechanism so that it is difficult or impossible to distinguish between them (by any means). Then it is also difficult or impossible to determine whether the authors of the original documents are A_1 or A_2, or some other author entirely. This is our aim for obfuscating authorship whilst preserving semantic content.

Our approach to obfuscating documents replaces words with other words, governed by probability distributions over possible replacements. Thus the type of our mechanism is $\mathbb{B}\mathcal{S} \to \mathbb{D}(\mathbb{B}\mathcal{S})$, where (recall) $\mathbb{D}(\mathbb{B}\mathcal{S})$ is the set of probability distributions over the set of (finite) bags of $\mathcal{S}$. Since we are aiming to find a careful trade-off between utility and privacy, our objective is to ensure that there is a high probability of outputting a document with a similar topic as the input document. As explained in Sect. 2, topic similarity of documents is determined by the Earth Mover's distance relative to a given (pseudo)metric on word embeddings, and so our privacy definition must also be relative to the Earth Mover's distance.

Definition 4 *(Earth Mover's Privacy).* *Let $\mathcal{X}$ be a set, and $d_{\mathcal{X}}$ be a (pseudo)metric on $\mathcal{X}$ and let $E_{d_{\mathcal{X}}}$ be the Earth Mover's metric on $\mathbb{B}\mathcal{X}$ relative to $d_{\mathcal{X}}$. Given $\epsilon \geq 0$, a mechanism $K : \mathbb{B}\mathcal{X} \to \mathbb{D}(\mathbb{B}\mathcal{X})$ satisfies $\epsilon E_{d_{\mathcal{X}}}$-privacy iff for any $b, b' \in \mathbb{B}\mathcal{X}$ and $Z \subseteq \mathbb{B}\mathcal{X}$:*

$$K(b)(Z) \quad \leq \quad e^{\epsilon E_{d_{\mathcal{X}}}(b,b')} K(b')(Z) \ . \tag{4}$$

Definition 4 tells us that when two documents are measured to be very close, so that $\epsilon E_{d_{\mathcal{X}}}(b, b')$ is close to 0, then the multiplier $e^{\epsilon E_{d_{\mathcal{X}}}(b,b')}$ is approximately 1 and the outputs $K(b)$ and $K(b')$ are almost identical. On the other hand the more that the input bags can be distinguished by $E_{d_{\mathcal{X}}}$, the more their outputs are likely to differ. This flexibility is what allows us to strike a balance between utility and privacy; we discuss this issue further in Sect. 5 below.

Our next task is to show how to implement a mechanism that can be proved to satisfy Definition 4. We follow the basic construction of Dwork et al. [12] for lifting a differentially private mechanism $K : \mathcal{X} \to \mathbb{D}\mathcal{X}$ to a differentially private mechanism $\underline{K}^\star : \mathcal{X}^N \to \mathbb{D}\mathcal{X}^N$ on *vectors* in $\mathcal{X}^N$. (Note that, unlike a bag, a vector imposes a fixed order on its components.) Here the idea is to apply K independently to each component of a vector $v \in \mathcal{X}^N$ to produce a random output vector, also in $\mathcal{X}^N$. In particular the probability of outputting some vector v' is

the product:

$$\underline{K}^\star(v)\{v'\} \quad = \quad \prod_{1\leq i\leq N} K(v_i)\{v'_i\}\ . \tag{5}$$

Thanks to the compositional properties of differential privacy when the underlying metric on $\mathcal{X}$ satisfies the triangle inequality, it's possible to show that the resulting mechanism $\underline{K}^\star$ satisfies the following privacy mechanism [13]:

$$\underline{K}^\star(v)(Z) \quad \leq \quad e^{M_{d_\mathcal{X}}(v,v')}\underline{K}^\star(v')(Z)\ , \tag{6}$$

where $M_{d_\mathcal{X}}(v,v') := \sum_{1\leq i\leq N} d_\mathcal{X}(v_i,v'_i)$, the Manhattan metric relative to $d_\mathcal{X}$.

However Definition 4 does not follow from (6), since Definition 4 operates on bags of size N, and the Manhattan distance between any vector representation of bags is *greater* than $N \times E_{d_\mathcal{X}}$. Remarkably however, it turns out that $K^\star$ –the mechanism that applies K independently to each item in a given bag– in fact satisfies the much stronger Definition 4, as the following theorem shows, provided the input bags have the same size as each other.

Theorem 1. *Let $d_\mathcal{X}$ be a pseudo-metric on $\mathcal{X}$ and let $K : \mathcal{X} \to \mathbb{D}\mathcal{X}$ be a mechanism satisfying $\epsilon d_\mathcal{X}$-privacy, i.e.*

$$K(x)(Z) \quad \leq \quad e^{\epsilon d_\mathcal{X}(x,x')}K(x')(Z)\ ,\ \textit{for all } x,x' \in \mathcal{X},\ Z \subseteq \mathcal{X}. \tag{7}$$

Let $K^\star : \mathbb{B}\mathcal{X} \to \mathbb{D}(\mathbb{B}\mathcal{X})$ be the mechanism obtained by applying K independently to each element of X for any $X \in \mathbb{B}\mathcal{X}$. Denote by $K^\star\!\downarrow N$ the restriction of $K^\star$ to bags of fixed size N. Then $K^\star\!\downarrow N$ satisfies $\epsilon N E_{d_\mathcal{X}}$-privacy.

Proof (Sketch). The full proof is given in our complete paper [15]; here we sketch the main ideas.

Let b, b' be input bags, both of size N, and let c a possible output bag (of $K^\star$). Observe that both output bags determined by $K^\star(b_1), K^\star(b_2)$ and c also have size N. We shall show that (4) is satisfied for the set containing the singleton element c and multiplier ϵN, from which it follows that (4) is satisfied for all sets Z.

By Birkhoff-von Neumann's theorem [26], in the case where all bags have the same size, the minimisation problem in Definition 3 is optimised for transportation matrix F where all values F_{ij} are either 0 or $1/N$. This implies that the optimal transportation for $E_{d_\mathcal{X}}(b,c)$ is achieved by moving each word in the bag b to a (single) word in bag c. The same is true for $E_{d_\mathcal{X}}(b',c)$ and $E_{d_\mathcal{X}}(b,b')$. Next we use a vector representation of bags as follows. For bag b, we write $\underline{b}$ for a vector in $\mathcal{X}^N$ such that each element in b appears at some $\underline{b}_i$.

Next we fix $\underline{b}$ and $\underline{b'}$ to be vector representations of respectively b, b' in $\mathcal{X}^N$ such that the optimal transportation for $E_{d_\mathcal{X}}(b,b')$ is

$$E_{d_\mathcal{X}}(b,b') \quad = \quad 1/N \times \sum_{1\leq i\leq N} d_\mathcal{X}(\underline{b}_i,\underline{b}'_i) \quad = \quad M_{d_\mathcal{X}}(\underline{b},\underline{b}')/N\ . \tag{8}$$

The final fact we need is to note that there is a relationship between $K^\star$ acting on bags of size N and $\underline{K}^\star$ which acts on vectors in $\mathcal{X}^N$ by applying K

independently to each component of a vector: it is characterised in the following way. Let b, c be bags and let $\underline{b}, \underline{c}$ be any vector representations. For permutation $\sigma \in \{1 \ldots N\} \to \{1 \ldots N\}$ write $\underline{c}^\sigma$ to be the vector with components permuted by σ, so that $\underline{c}^\sigma_i = \underline{c}_{\sigma(i)}$. With these definitions, the following equality between probabilities holds:

$$K^\star(b)\{c\} \quad = \quad \sum_\sigma \underline{K}^\star(\underline{b})\{\underline{c}^\sigma\} \ , \tag{9}$$

where the summation is over all permutations that give distinct vector representations of c. We now compute directly:

$$\begin{array}{lll} & K^\star(b)\{c\} & \\ = & \sum_\sigma \underline{K}^\star(\underline{b})\{\underline{c}^\sigma\} & \text{“(9) for } b, c\text{”} \\ \leq & \sum_\sigma e^{\epsilon M_d(\underline{b},\underline{b}')} \underline{K}^\star(\underline{b}')\{\underline{c}^\sigma\} & \text{“(6) for } \underline{b}, \underline{b}', \underline{c}\text{”} \\ = & e^{\epsilon N E_d(\underline{b},\underline{b}')} \sum_\sigma \underline{K}^\star(\underline{b}')\{\underline{c}^\sigma\} & \text{“Arithmetic and (8)”} \\ = & e^{\epsilon N E_d(\underline{b},\underline{b}')} K^\star(b')\{c\} \ , & \text{“(9) for } b', c\text{”} \end{array}$$

as required.

3.1 Application to Text Documents

Recall the bag-of-words

$$b_2 \quad := \quad \langle \text{Chief}^1, \text{speaks}^1, \text{media}^1, \text{Illinois}^1 \rangle \ ,$$

and assume we are provided with a mechanism K satisfying the standard $\epsilon d_{\mathcal{X}}$-privacy property (7) for individual words. As in Theorem 1 we can create a mechanism K^* by applying K independently to each word in the bag, so that, for example the probability of outputting $b_3 = \langle \text{Chef}^1, \text{breaks}^1, \text{cooking}^1, \text{record}^1 \rangle$ is determined by (9):

$$K^\star(b_2)(\{b_3\}) \quad = \quad \sum_\sigma \prod_{1 \leq i \leq 4} K(b_{2i})\{\underline{b_3}_i{}^\sigma\} \ .$$

By Theorem 1, $K^\star$ satisfies $4\epsilon E_{d_\mathcal{S}}$-privacy. Recalling (3) that $E_{d_\mathcal{S}}(b_1, b_2) = 2.816$, we deduce that if $\epsilon \sim 1/16$ then the output distributions $K^\star(b_1)$ and $K^\star(b_2)$ would differ by the multiplier $e^{2.816 \times 4/16} \sim 2.02$; but if $\epsilon \sim 1/32$ those distributions differ by only 1.42. In the latter case it means that the outputs of $K^\star$ on b_1 and b_2 are almost indistinguishable.

The parameter ϵ depends on the randomness implemented in the basic mechanism K; we investigate that further in Sect. 4.

3.2 Properties of Earth Mover's Privacy

In machine learning a number of “distance measures” are used in classification or clustering tasks, and in this section we explore some properties of privacy when we vary the underlying metrics of an Earth Mover's metric used to classify complex objects.

Let $v, v' \in \mathbb{R}^n$ be real-valued n-dimensional vectors. We use the following (well-known) metrics. Recall in our applications we have looked at bags-of-words, where the words themselves are represented as n-dimensional vectors.[9]

1. Euclidean: $\|v-v'\| := \sqrt{\sum_{1\leq i\leq n}(v_i - v'_i)^2}$
2. Manhattan: $\lfloor v-v' \rfloor := \sum_{1\leq i\leq n} |v_i - v'_i|$

Note that the Euclidean and Manhattan distances determine pseudometrics on words as defined at Definition 2 and proved at Lemma 1.

Lemma 2. *If $d_{\mathcal{X}} \leq d_{\mathcal{X}'}$ (point-wise), then $E_{d_{\mathcal{X}}} \leq E_{d_{\mathcal{X}'}}$ (point-wise).*

Proof. Trivial, by contradiction. If $d_{\mathcal{X}} \leq d_{\mathcal{X}'}$ and $F_{ij}, F^{\star}_{ij}$ are the minimal flow matrices for $E_{d_{\mathcal{X}}}, E_{d_{\mathcal{X}'}}$ respectively, then $F^{\star}_{ij}$ is a (strictly smaller) minimal solution for $E_{d_{\mathcal{X}}}$ which contradicts the minimality of F_{ij}.

Corollary 1. *If $d_{\mathcal{X}} \leq d_{\mathcal{X}'}$ (point-wise), then $E_{d_{\mathcal{X}}}$-privacy implies $E_{d_{\mathcal{X}'}}$-privacy.*

This shows that, for example, $E_{\|\cdot\|}$-privacy implies $E_{\lfloor\cdot\rfloor}$-privacy, and indeed any distance measure d which exceeds the Euclidean distance then $E_{\|\cdot\|}$-privacy implies E_d-privacy.

We end this section by noting that Definition 4 satisfies *post-processing*; i.e. that privacy does not decrease under post processing. We write $K; K'$ for the composition of mechanisms $K, K' : \mathbb{B}\mathcal{X} \to \mathbb{D}(\mathbb{B}\mathcal{X})$, defined:

$$(K; K')(b)(Z) := \sum_{b' : \mathbb{B}\mathcal{X}} K(b)(\{b'\}) \times K'(b')(Z) \ . \tag{10}$$

Lemma 3 *[Post processing]. If $K, K' : \mathbb{B}\mathcal{X} \to \mathbb{D}(\mathbb{B}\mathcal{X})$ and K is $\epsilon E_{d_{\mathcal{X}}}$-private for (pseudo)metric d on $\mathcal{X}$ then $K; K'$ is $\epsilon E_{d_{\mathcal{X}}}$-private.*

3D plot Contour diagram

Fig. 2. Laplace density function Lap^2_{ϵ} in $\mathbb{R}^2$

[9] As we shall see, in the machine learning analysis *documents* are represented as bags of n-dimensional vectors (word embeddings), where each bag contains N such vectors.

4 Earth Mover's Privacy for Bags of Vectors in $\mathbb{R}^n$

In Theorem 1 we have shown how to promote a privacy mechanism on components to $E_{d_{\mathcal{X}}}$-privacy on a bag of those components. In this section we show how to implement a privacy mechanism satisfying (7), when the components are represented by high dimensional vectors in $\mathbb{R}^n$ and the underlying metric is taken Euclidean on $\mathbb{R}^n$, which we denote by $\|\cdot\|$.

We begin by summarising the basic probabilistic tools we need. A *probability density function* (PDF) over some domain $\mathcal{D}$ is a function $\phi : \mathcal{D} \to [0, 1]$ whose value $\phi(z)$ gives the "relative likelihood" of z. The probability density function is used to compute the probability of an outcome "$z \in A$", for some region $A \subseteq \mathcal{D}$ as follows:

$$\int_A \phi(x)\ dx\ . \tag{11}$$

In differential privacy, a popular density function used for implementing mechanisms is the *Laplacian*, defined next.

Definition 5. *Let $n \geq 0$ be an integer $\epsilon > 0$ be a real, and $v \in \mathbb{R}^n$. We define the Laplacian probability density function in n-dimensions:*

$$Lap^n_\epsilon(v) \quad := \quad c^\epsilon_n \times e^{-\epsilon\|v\|}\ ,$$

where $\|v\| = \sqrt{(v_1^2 + \cdots + v_n^2)}$, and c^ϵ_n is a real-valued constant satisfying the integral equation $1 = \int \ldots \int_{\mathbb{R}^n} Lap^n_\epsilon(v) dv_1 \ldots dv_n$.

When $n = 1$, we can compute $c^\epsilon_1 = \epsilon/2$, and when $n = 2$, we have that $c^\epsilon_2 = \epsilon^2/2\pi$.

In privacy mechanisms, probability density functions are used to produce a "noisy" version of the released data. The benefit of the Laplace distribution is that, besides creating randomness, the likelihood that the released value is different from the true value decreases exponentially. This implies that the utility of the data release is high, whilst at the same time masking its actual value. In Fig. 2 the probability density function $Lap^2_\epsilon(v)$ depicts this situation, where we see that the highest relative likelihood of a randomly selected point on the plane being close to the origin, with the chance of choosing more distant points diminishing rapidly. Once we are able to select a vector v' in $\mathbb{R}^n$ according to Lap^n_ϵ, we can "add noise" to any given vector v as $v+v'$, so that the true value v is highly likely to be perturbed only a small amount.

In order to use the Laplacian in Definition 5, we need to implement it. Andrés et al. [4] exhibited a mechanism for $Lap^2_\epsilon(v)$, and here we show how to extend that idea to the general case. The main idea of the construction for $Lap^2_\epsilon(v)$ uses the fact that any vector on the plane can be represented by spherical coordinates (r, θ), so that the probability of selecting a vector distance no more than r from the origin can be achieved by selecting r and θ independently. In order to obtain a distribution which overall is equivalent to $Lap^2_\epsilon(v)$, Andrés et al. computed that r must be selected according to a well-known distribution called the "Lambert W" function, and θ is selected uniformly over the unit circle. In our generalisation

to $Lap^n_\epsilon(v)$, we observe that the same idea is valid [6]. Observe first that every vector in $\mathbb{R}^n$ can be expressed as a pair (r, p), where r is the distance from the origin, and p is a point in B^n, the unit *hypersphere* in $\mathbb{R}^n$. Now selecting vectors according to $Lap^n_\epsilon(v)$ can be achieved by independently selecting r and p, but this time r must be selected according to the *Gamma distribution*, and p must be selected uniformly over B^n. We set out the details next.

Definition 6. *The* Gamma distribution *of (integer) shape n and scale $\delta > 0$ is determined by the probability density function:*

$$Gam^n_\delta(r) \quad := \quad \frac{r^{n-1}e^{-r/\delta}}{\delta^n(n-1!)} \ . \tag{12}$$

Definition 7. *The uniform distribution over the surface of the unit hypersphere B^n is determined by the probability density function:*

$$Uniform^n(v) \quad := \quad \frac{\Gamma(\frac{n}{2})}{n\pi^{n/2}} \ \textit{if} \ \ v \in B^n \ \ \textit{else} \ \ 0 \ , \tag{13}$$

where $B^n := \{v \in \mathbb{R}^n \mid \|v\| = 1\}$, and $\Gamma(\alpha) := \int_0^\infty x^{\alpha-1}e^{-x}\ dx$ is the "Gamma function".

With Definitions 6 and 7 we are able to provide an implementation of a mechanism which produces noisy vectors around a given vector in $\mathbb{R}^n$ according to the Laplacian distribution in Definition 5. The first task is to show that our decomposition of Lap^n_ϵ is correct.

Lemma 4. *The n-dimensional Laplacian $Lap^n_\epsilon(v)$ can be realised by selecting vectors represented as (r, p), where r is selected according to $Gam^n_{1/\epsilon}(r)$ and p is selected independently according to $Uniform^n(p)$.*

Proof (Sketch). The proof follows by changing variables to spherical coordinates and then showing that $\int_A Lap^n_\epsilon(v)\ dv$ can be expressed as the product of independent selections of r and p.

We use a spherical-coordinate representation of v as:

$$r := \|v\| \ , \ \textit{and}$$
$$v_1 := r\cos\theta_1 \ , \ v_2 := r\sin\theta_1\cos\theta_2 \ , \ldots v_n := r\sin\theta_1\sin\theta_2\ldots \ , \sin\theta_{n-2}\sin\theta_{n-1} \ .$$

Next we assume for simplicity that A is a hypersphere of radius R; with that we can reason:

$$\begin{aligned}
& \int_A Lap^n_\epsilon(v)\ dv \\
= \quad & \text{"Definition 5; } A \text{ is a hypersphere"} \\
& \int_{\|v\|\le R} c^\epsilon_n \times e^{-\epsilon|v|}\ dv \\
= \quad & \text{"}\|v\| = \sqrt{v_1^2 + \cdots + v_n^2}\text{"} \\
& \int_{\|v\|\le R} c^\epsilon_n \times e^{-\epsilon\sqrt{v_1^2+\cdots+v_n^2}}\ dv \\
= \quad & \text{"Change of variables to spherical coordinates; see below (14)"} \\
& \int_{r\le R}\int_{A_\theta} c^\epsilon_n \times e^{-\epsilon r}\,\tfrac{\partial(z_1,z_2,\ldots,z_n)}{\partial(r,\theta_1,\ldots,\theta_{n-1})}\ dr d\theta_1\ldots d\theta_{n-1} \\
= \quad & \text{"See below (14)"} \\
& \int_{r\le R}\int_{A_\theta} c^\epsilon_n \times e^{-\epsilon r} r^{n-1}\sin^{n-2}\theta_1\sin^{n-3}\theta_2\ldots\sin^2\theta_{n-3}\sin\theta_{n-2}\ dr d\theta_1\ldots d\theta_{n-1} \ .
\end{aligned}$$

Now rearranging we can see that this becomes a product of two integrals. The first $\int_{r \leq R} e^{-\epsilon r} r^{n-1}$ is over the radius, and is proportional to the integral of the Gamma distribution Definition 6; and the second is an integral over the angular coordinates and is proportional to the surface of the unit hypersphere, and corresponds to the PDF at (7). Finally, for the "see below's" we are using the "Jacobian":

$$\frac{\partial(z_1, z_2, \ldots, z_n)}{\partial(r, \theta_1, \ldots, \theta_{n-1})} = r^{n-1} \sin^{n-2} \theta_1 \sin^{n-3} \theta_2 \ldots \tag{14}$$

(For full details, see our complete paper [15].)

We can now assemble the facts to demonstrate the n-Dimensional Laplacian.

Theorem 2 (n-Dimensional Laplacian). *Given $\epsilon > 0$ and $n \in \mathbb{Z}^+$, let $K : \mathbb{R}^n \to \mathbb{DR}^n$ be a mechanism that, given a vector $x \in \mathbb{R}^n$ outputs a noisy value as follows:*

$$x \xmapsto{K} x + x'$$

where x' is represented as (r, p) with $r \geq 0$, distributed according to $Gam^n_{1/\epsilon}(r)$ and $p \in B^n$ distributed according to $Uniform^n(p)$. Then K satisfies (7) from Theorem 1, i.e. K satisfies $\epsilon\|\cdot\|$-privacy where $\|\cdot\|$ is the Euclidean metric on $\mathbb{R}^n$.

Proof (Sketch). Let $z, y \in \mathbb{R}^n$. We need to show that for any (measurable) set $A \subseteq \mathbb{R}^n$ that:

$$K(z)(A)/K(y)(A) \quad \leq \quad e^{\epsilon\|z-y\|} \ . \tag{15}$$

However (15) follows provided that the probability densities of respectively $K(z)$ and $K(y)$ satisfy it. By Lemma 4 the probability density of $K(z)$, as a function of x is distributed as $Lap^n_\epsilon(z-x)$; and similarly for the probability density of $K(y)$. Hence we reason:

$$\begin{array}{lll} & Lap^n_\epsilon(z-x)/Lap^n_\epsilon(y-x) & \\ = & c^\epsilon_n \times e^{-\epsilon\|z-x\|} / c^\epsilon_n \times e^{-\epsilon\|y-x\|} & \text{“Definition 5”} \\ = & e^{-\epsilon\|z-x\|} \times e^{\epsilon\|y-x\|} & \text{“Arithmetic”} \\ \leq & e^{\epsilon\|z-y\|} \ , & \text{“Triangle inequality; } s \mapsto e^s \text{ is monotone”} \end{array}$$

as required.

Theorem 2 reduces the problem of adding Laplace noise to vectors in $\mathbb{R}^n$ to selecting a real value according to the Gamma distribution and an independent uniform selection of a unit vector. Several methods have been proposed for generating random variables according to the Gamma distribution [30] as well as for the uniform selection of vectors on the unit n-sphere [35]. The uniform selection of a unit vector has also been described in [35]; it avoids the transformation to spherical coordinates by selecting n random variables from the standard normal distribution to produce vector $v \in \mathbb{R}^n$, and then normalising to output $\frac{v}{|v|}$.

4.1 Earth Mover's Privacy in $\mathbb{BR}^n$

Using the n-dimensional Laplacian, we can now implement an algorithm for $\epsilon NE_{\|\cdot\|}$-privacy. Algorithm 1 takes a bag of n-dimensional vectors as input and applies the n-dimensional Laplacian mechanism described in Theorem 2 to each vector in the bag, producing a noisy bag of n-dimensional vectors as output. Corollary 2 summarises the privacy guarantee.

Algorithm 1. Earth Mover's Privacy Mechanism

Require: vector v, dimension n, epsilon ϵ
procedure GENERATENOISYVECTOR(v, n, ϵ)
  $r \leftarrow Gamma(n, \frac{1}{\epsilon})$
  $u \leftarrow \mathcal{U}(n)$
  return $v + ru$
end procedure

Require: bag X, dimension n, epsilon ϵ
procedure GENERATEPRIVATEBAG(X, n, ϵ)
  $Z \leftarrow ()$
  for all $x \in X$ **do**
    $z \leftarrow$ GENERATENOISYVECTOR(x, n, ϵ)
    add z to Z
  end for
  return Z
end procedure

Corollary 2. *Algorithm 1 satisfies $\epsilon NE_{\|\cdot\|}$-privacy, relative to any two bags in $\mathbb{BR}^n$ of size N.*

Proof. Follows from Theorems 1 and 2.

4.2 Utility Bounds

We prove a lower bound on the utility for this algorithm, which applies for high dimensional data representations. Given an output element x, we define Z to be the set of outputs within distance $\Delta > 0$ from x. Recall that the distance function is a measure of utility, therefore $Z = \{z \mid E_{\|\cdot\|}(x, z) \leq \Delta\}$ represents the set of vectors within utility Δ of x. Then we have the following:

Theorem 3. *Given an input bag b consisting of N n-dimensional vectors, the mechanism defined by Algorithm 1 outputs an element from $Z = \{z \mid E_{\|\cdot\|}(b, z) \leq \Delta\}$ with probability at least*

$$1 - e^{-\epsilon N \Delta} e_{n-1}(\epsilon N \Delta) \ ,$$

whenever $\epsilon N \Delta \leq n/e$. (Recall that $e_k(\alpha) = \sum_{0 \leq i \leq k} \frac{\alpha^i}{i!}$, the sum of the first $k+1$ terms in the series for e^α.)

Proof (Sketch). Let $\underline{b} \in (\mathbb{R}^n)^N$ *be a (fixed) vector representation of the bag* b*. For* $v \in (\mathbb{R}^n)^N$*, let* $v^\circ \in \mathbb{BR}^n$ *be the bag comprising the* N *components if* v*. Observe that* $NE_{\|\cdot\|}(b, v^\circ) \leq M_{\|\cdot\|}(\underline{b}, v)$*, and so*

$$Z_M = \{v \mid M_{\|\cdot\|}(\underline{b}, v) \leq N\Delta\} \quad \subseteq \quad \{v \mid E_{\|\cdot\|}(b, v^\circ) \leq \Delta\} = Z_E \ . \tag{16}$$

Thus the probability of outputting an element of Z *is the same as the probability of outputting* Z_E*, and by (16) that is at least the probability of outputting an element from* Z_M *by applying a standard* n*-dimensional Laplace mechanism to each of the components of* $\underline{b}$*. We can now compute:*

$$\begin{array}{ll} & \textit{Probability of outputting an element in } Z_E \\ \geq & \textit{“(16)”} \\ & \int \ldots \int_{v \in Z_M} \prod_{1 \leq i \leq N} Lap^n_\epsilon(\underline{b}_i - v_i) dv_1 \ldots dv_N \\ = & \textit{“Lemma 4”} \\ & \int \ldots \int_{v \in Z_M} \prod_{1 \leq i \leq N} c^\epsilon_n e^{-\epsilon \|\underline{b}_i - v_i\|} dv_1 \ldots dv_N \ . \end{array}$$

The result follows by completing the multiple integrals and applying some approximations, whilst observing that the variables in the integration are n*-dimensional vector valued. (The details appear in our complete paper [15].)*

We note that in our application word embeddings are typically mapped to vectors in $\mathbb{R}^{300}$, thus we would use $n \sim 300$ in Theorem 3.

5 Text Document Privacy

In this section we bring everything together, and present a privacy mechanism for text documents; we explore how it contributes to the author obfuscation task described above. Algorithm 2 describes the complete procedure for taking a document as a bag-of-words, and outputting a “noisy” bag-of-words. Depending on the setting of parameter ϵ, the output bag will be likely to be classified to be on a similar topic as the input.

Algorithm 2 uses a function *Vec* to turn the input document into a bag of word embeddings; next Algorithm 1 produces a noisy bag of word embeddings, and, in a final step the inverse Vec^{-1} is used to reconstruct an actual bag-of-words as output. In our implementation of Algorithm 2, described below, we compute $Vec^{-1}(x)$ to be the word w that minimises the Euclidean distance $\|z - Vec(w)\|$. The next result summarises the privacy guarantee for Algorithm 2.

Theorem 4. *Algorithm 2 satisfies* $\epsilon N E_{d_\mathcal{S}}$*-privacy, where* $d_\mathcal{S} = dist_{Vec}$*. That is to say: given input documents (bags)* b, b' *both of size* N*, and* c *a possible output bag, define the following quantities as follows:* $k := E_{\|\cdot\|}(Vec^\star(b), Vec^\star(b'))$*,* $pr(b, c)$ *and* $pr(b', c)$ *are the respective probabilities that* c *is output given the input was* b *or* b'*. Then:*

$$pr(b, c) \quad \leq \quad e^{\epsilon N k} \times pr(b', c) \ .$$

Algorithm 2. Document privacy mechanism

Require: Bag-of-words b, dimension n, epsilon ϵ, Word embedding $Vec : \mathcal{S} \to \mathbb{R}^n$

procedure GENERATENOISYBAGOFWORDS(b, n, ϵ, Vec)
  $X \leftarrow Vec^\star(b)$
  $Z \leftarrow$ GENERATEPRIVATEBAG(X, n, ϵ)
  return $(Vec^{-1})^\star(Z)$
end procedure

Note that $Vec^\star : \mathbb{B}\mathcal{S} \to \mathbb{B}\mathbb{R}^n$ applies Vec to each word in a bag b, and $(Vec^{-1})^\star : \mathbb{B}\mathbb{R}^n \to \mathbb{B}\mathcal{S}$ reverses this procedure as a post-processing step; this involves determining the word w that minimises the Euclidean distance $\|z - Vec(w)\|$ for each z in Z.

Proof. The result follows by appeal to Theorem 2 for privacy on the word embeddings; the step to apply Vec^{-1} to each vector is a post-processing step which by Lemma 3 preserves the privacy guarantee.

Although Theorem 4 utilises ideas from differential privacy, an interesting question to ask is how it contributes to the PAN@Clef author obfuscation task, which recall asked for mechanisms that preserve content but mask features that distinguish authorship. Algorithm 2 does indeed attempt to preserve content (to the extent that the topic can still be determined) but it does not directly "remove stylistic features".[10] So has it, in fact, disguised the author's characteristic style? To answer that question, we review Theorem 4 and interpret what it tells us in relation to author obfuscation.

The theorem implies that it is indeed possible to make the (probabilistic) output from two distinct documents b, b' almost indistinguishable by choosing ϵ to be extremely small in comparison with $N \times E_{\|\cdot\|}(Vec^\star(b), Vec^\star(b'))$. However, if $E_{\|\cdot\|}(Vec^\star(b), Vec^\star(b'))$ is very large – meaning that b and b' are on entirely different topics, then ϵ would need to be so tiny that the noisy output document would be highly unlikely to be on a topic remotely close to either b or b' (recall Lemma 3).

This observation is actually highlighting the fact that, in some circumstances, the topic itself is actually a feature that characterises author identity. (First-hand accounts of breaking the world record for highest and longest free fall jump would immediately narrow the field down to the title holder.) This means that *any* obfuscating mechanism would, as for Algorithm 2, only be able to obfuscate documents so as to disguise the author's identity if there are several authors who write on similar topics. And it is in that spirit, that we have made the first step towards a satisfactory obfuscating mechanism: provided that documents are similar in topic (i.e. are close when their embeddings are measured by $E_{\|\cdot\|}$) they can be obfuscated so that it is unlikely that the content is disturbed, but that the contributing authors cannot be determined easily.

[10] Although, as others have noted [53], the bag-of-words representation already removes many stylistic features. We note that our privacy guarantee does not depend on this side-effect.

We can see the importance of the "indistinguishability" property wrt. the PAN obfuscation task. In stylometry analysis the representation of words for eg. author classification is completely different to the word embeddings which have used for topic classification. State-of-the-art author attribution algorithms represent words as "character n-grams" [28] which have been found to capture stylistic clues such as systematic spelling errors. A *character 3-gram* for example represents a given word as the complete list of substrings of length 3. For example character 3-gram representations of "color" and "colour" are:

- "color" $\mapsto$ ⟦ "col", "olo", "lor" ⟧
- "colour" $\mapsto$ ⟦ "col", "olo", "lou", "our" ⟧

For author identification, any output from Algorithm 2 would then need to be further transformed to a bag of character n-grams, as a post processing step; by Lemma 3 this additional transformation preserves the privacy properties of Algorithm 2. We explore this experimentally in the next section.

6 Experimental Results

Document Set. The PAN@Clef tasks and other similar work have used a variety of types of text for author identification and author obfuscation. Our desiderata are that we have multiple authors writing on one topic (so as to minimise the ability of an author identification system to use topic-related cues) and to have more than one topic (so that we can evaluate utility in terms of accuracy of topic classification). Further, we would like to use data from a domain where there are potentially large quantities of text available, and where it is already annotated with author and topic.

Given these considerations, we chose "fan fiction" as our domain. Wikipedia defines *fan fiction* as follows: "Fan fiction ... is fiction about characters or settings from an original work of fiction, created by fans of that work rather than by its creator." This is also the domain that was used in the PAN@Clef 2018 author attribution challenge,[11] although for this work we scraped our own dataset. We chose one of the largest fan fiction sites and the two largest "fandoms" there;[12] these fandoms are our topics. We scraped the stories from these fandoms, the largest proportion of which are for use in training our topic classification model. We held out two subsets of size 20 and 50, evenly split between fandoms/topics, for the evaluation of our privacy mechanism.[13] We follow the evaluation framework of [28]: for each author we construct an known-author TEXT and an unknown-author SNIPPET that we have to match to an author on

[11] https://pan.webis.de/clef18/pan18-web/author-identification.html.

[12] https://www.fanfiction.net/book/, with the two largest fandoms being Harry Potter (797,000 stories) and Twilight (220,000 stories).

[13] Our Algorithm 2 is computationally quite expensive, because each word $w = Vec^{-1}(x)$ requires the calculation of Euclidean distance with respect to the whole vocabulary. We thus use relatively small evaluation sets, as we apply the algorithm to them for multiple values of ϵ.

the basis of the known-author texts. (See Appendix in our complete paper [15] for more detail.)

Word Embeddings. There are sets of word embeddings trained on large datasets that have been made publicly available. Most of these, however, are already normalised, which makes them unsuitable for our method. We therefore use the Google News word2vec embeddings as the only large-scale unnormalised embeddings available. (See Appendix in our complete paper [15] for more detail.)

Inference Mechanisms. We have two sorts of machine learning inference mechanisms: our adversary mechanism for author identification, and our utility-related mechanism for topic classification. For each of these, we can define inference mechanisms both within the same representational space or in a different representational space. As we noted above, in practice both author identification adversary and topic classification will use different representations, but examining same-representation inference mechanisms can give an insight into what is happening within that space.

Different-Representation Author Identification. For this we use the algorithm by [28]. This algorithm is widely used: it underpins two of the winners of PAN shared tasks [25,47]; is a common benchmark or starting point for other methods [19,39,44,46]; and is a standard inference attacker for the PAN shared task on authorship obfuscation.[14] It works by representing each text as a vector of space-separated character n-gram counts, and comparing repeatedly sampled subvectors of known-author texts and snippets using cosine similarity. We use as a starting point the code from a reproducibility study [40], but have modified it to improve efficiency. (See Appendix in our complete paper [15] for more details.)

Different-Representation Topic Classification. Here we choose fastText [7,22], a high-performing supervised machine learning classification system. It also works with word embeddings; these differ from word2vec in that they are derived from embeddings over character n-grams, learnt using the same skipgram model as word2vec. This means it is able to compute representations for words that do not appear in the training data, which is helpful when training with relatively small amounts of data; also useful when training with small amounts of data is the ability to start from pretrained embeddings trained on out-of-domain data that are then adapted to the in-domain (here, fan fiction) data. After training, the accuracy on a validation set we construct from the data is 93.7% (see [15] for details).

Same-Representation Author Identification. In the space of our word2vec embeddings, we can define an inference mechanism that for an unknown-author snippet chooses the closest known-author text by Euclidean distance.

[14] http://pan.webis.de/clef17/pan17-web/author-obfuscation.html.

Same-Representation Topic Classification. Similarly, we can define an inference mechanism that considers the topic classes of neighbours and predicts a class for the snippet based on that. This is essentially the standard k "Nearest Neighbours" technique (k-NN) [21], a non-parametric method that assigns the majority class of the k nearest neighbours. 1-NN corresponds to classification based on a Voronoi tesselation of the space, has low bias and high variance, and asymptotically has an error rate that is never more than twice the Bayes rate; higher values of k have a smoothing effect. Because of the nature of word embeddings, we would not expect this classification to be as accurate as the fastText classification above: in high-dimensional Euclidean space (as here), almost all points are approximately equidistant. Nevertheless, it can give an idea about how a snippet with varying levels of noise added is being shifted in Euclidean space with respect to other texts in the same topic. Here, we use $k = 5$. Same-representation author identification can then be viewed as 1-NN with author as class.

Table 1. Number of correct predictions of author/topic in the 20-author set (left) and 50-author set (right), using 1-NN for same-representation author identification (SRauth), 5-NN for same-representation topic classification (SRtopic), the Koppel algorithm for different-representation author identification (DRauth) and fastText for different-representation topic classification (DRtopic).

	20-author set			
ϵ	SRauth	SRtopic	DRauth	DRtopic
none	12	16	15	18
30	8	18	16	18
25	8	18	14	17
20	5	11	11	16
15	2	11	12	17
10	0	15	11	19

	50-author set			
ϵ	SRauth	SRtopic	DRauth	DRtopic
none	19	36	27	43
30	19	37	29	43
25	17	34	24	41
20	12	28	19	42
15	9	22	13	42
10	1	24	10	43

Results: Table 1 contains the results for both document sets, for the unmodified snippets ("none") or with the privacy mechanism of Algorithm 2 applied with various levels of ϵ: we give results for ϵ between 10 and 30, as at $\epsilon = 40$ the text does not change, while at $\epsilon = 1$ the text is unrecognisable. For the 20-author set, a random guess baseline would give 1 correct author prediction, and 10 correct topic predictions; for the 50-author set, these values are 1 and 25 respectively.

Performance on the unmodified snippets using different-representation inference mechanisms is quite good: author identification gets 15/20 correct for the 20-author set and 27/50 for the 50-author set; and topic classification 18/20 and 43/50 (comparable to the validation set accuracy, although slightly lower, which is to be expected given that the texts are much shorter). For various levels of ϵ, with our different-representation inference mechanisms we see broadly the behaviour we expected: the performance of author identification drops, while topic classification holds roughly constant. Author identification here does not drop to chance levels: we speculate that this is because (in spite of our choice

of dataset for this purpose) there are still some topic clues that the algorithm of [28] takes advantage of: one author of Harry Potter fan fiction might prefer to write about a particular character (e.g. Severus Snape), and as these character names are not in our word2vec vocabulary, they are not replaced by the privacy mechanism.

In our same-representation author identification, though, we do find performance starting relatively high (although not as high as the different-representation algorithm) and then dropping to (worse than) chance, which is the level we would expect for our privacy mechanism. The k-NN topic classification, however, shows some instability, which is probably an artefact of the problems it faces with high-dimensional Euclidean spaces. (Refer to our complete arXiv paper [15] for a sample of texts and nearest neighbours.)

7 Related Work

Author Obfuscation. The most similar work to ours is by Weggenmann and Kerschbaum [53] who also consider the author obfuscation problem but apply standard differential privacy using a Hamming distance of 1 between all documents. As with our approach, they consider the simplified utility requirement of topic preservation and use word embeddings to represent documents. Our approach differs in our use of the Earth Mover's metric to provide a strong utility measure for document similarity.

An early work in this area by Kacmarcik et al. [23] applies obfuscation by modifying the most important stylometric features of the text to reduce the effectiveness of author attribution. This approach was used in Anonymouth [36], a semi-automated tool that provides feedback to authors on which features to modify to effectively anonymise their texts. A similar approach was also followed by Karadhov et al. [24] as part of the PAN@Clef 2017 task.

Other approaches to author obfuscation, motivated by the PAN@Clef task, have focussed on the stronger utility requirement of semantic sensibility [5,8,34]. Privacy guarantees are therefore ad hoc and are designed to increase misclassification rates by the author attribution software used to test the mechanism.

Most recently there has been interest in training neural networks models which can protect author identity whilst preserving the semantics of the original document [14,48]. Other related deep learning methods aim to obscure other author attributes such as gender or age [10,32]. While these methods produce strong empirical results, they provide no formal privacy guarantees. Importantly, their goal also differs from the goal of our paper: they aim to obscure properties of authors in the *training set* (with the intention of the author-obscured learned representations being made available), while we assume that an adversary may have access to raw training data to construct an inference mechanism with full knowledge of author properties, and in this context aim to hide the properties of some other text external to the training set.

Machine Learning and Differential Privacy. Outside of author attribution, there is quite a body of work on introducing differential privacy to machine learning: [13] gives an overview of a classical machine learning setting; more recent deep learning approaches include [1,49]. However, these are generally applied in other domains such as image processing: text introduces additional complexity because of its discrete nature, in contrast to the continuous nature of neural networks. A recent exception is [37], which constructs a differentially private language model using a recurrent neural network; the goal here, as for instances above, is to hide properties of data items in the training set.

Generalised Differential Privacy. Also known as $d_{\mathcal{X}}$-privacy [9], this definition was originally motivated by the problem of geo-location privacy [4]. Despite its generality, $d_{\mathcal{X}}$-privacy has yet to find significant applications outside this domain; in particular, there have been no applications to text privacy.

Text Document Privacy. This typically refers to the sanitisation or redaction of documents either to protect the identity of individuals or to protect the confidentiality of their sensitive attributes. For example, a medical document may be modified to hide specifics in the medical history of a named patient. Similarly, a classified document may be redacted to protect the identity of an individual referred to in the text.

Most approaches to sanitisation or redaction rely on first identifying sensitive terms in the text, and then modifying (or deleting) only these terms to produce a sanitised document. Abril et al. [2] proposed this two-step approach, focussing on identification of terms using NLP techniques. Cumby and Ghani [11] proposed *k-confusability*, inspired by *k-anonymity* [50], to perturb sensitive terms in a document so that its (utility) class is confusable with at least k other classes. Their approach requires a complete dataset of similar documents for computing (mis)classification probabilities. Anandan et al. [3] proposed *t-plausibility* which generalises sensitive terms such that any document could have been generated from at least t other documents. Sánchez and Batet [45] proposed *C-sanitisation*, a model for both detection and protection of sensitive terms (C) using information theoretic guarantees. In particular, a *C-sanitised* document should contain no collection of terms which can be used to infer any of the sensitive terms.

Finally, there has been some work on noise-addition techniques in this area. Rodriguez-Garcia et al. [42] propose semantic noise, which perturbs sensitive terms in a document using a distance measure over the directed graph representing a predefined ontology.

Whilst these approaches have strong utility, our primary point of difference is our insistence on a differential privacy-based guarantee. This ensures that every output document could have been produced from any input document with some probability, giving the strongest possible notion of plausible-deniability.

8 Conclusions

We have shown how to combine representations of text documents with generalised differential privacy in order to implement a privacy mechanism for text documents. Unlike most other techniques for privacy in text processing, ours provides a guarantee in the style of differential privacy. Moreover we have demonstrated experimentally the trade off between utility and privacy.

This represents an important step towards the implementation of privacy mechanisms that could produce readable summaries of documents with a privacy guarantee. One way to achieve this goal would be to reconstruct readable documents from the bag-of-words output that our mechanism currently provides. A range of promising techniques for reconstructing readable texts from bag-of-words have already produced some good experimental results [20,52,54]. In future work we aim to explore how techniques such as these could be applied as a final post processing step for our mechanism.

References

1. Abadi, M., et al.: Deep learning with differential privacy. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, CCS 2016, pp. 308–318. ACM, New York (2016). https://doi.org/10.1145/2976749.2978318
2. Abril, D., Navarro-Arribas, G., Torra, V.: On the declassification of confidential documents. In: Torra, V., Narakawa, Y., Yin, J., Long, J. (eds.) MDAI 2011. LNCS (LNAI), vol. 6820, pp. 235–246. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22589-5_22
3. Anandan, B., Clifton, C., Jiang, W., Murugesan, M., Pastrana-Camacho, P., Si, L.: t-Plausibility: generalizing words to desensitize text. Trans. Data Priv. **5**(3), 505–534 (2012)
4. Andrés, M.E., Bordenabe, N.E., Chatzikokolakis, K., Palamidessi, C.: Geo-indistinguishability: differential privacy for location-based systems. In: Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security, pp. 901–914. ACM (2013)
5. Bakhteev, O., Khazov, A.: Author masking using sequence-to-sequence models—notebook for PAN at CLEF 2017. In: Cappellato, L., Ferro, N., Goeuriot, L., Mandl, T. (eds.) CLEF 2017 Evaluation Labs and Workshop – Working Notes Papers, Dublin, Ireland, 11–14 September. CEUR-WS.org, September 2017. http://ceur-ws.org/Vol-1866/
6. Boisbunon, A.: The class of multivariate spherically symmetric distributions. Université de Rouen, Technical report 5, 2012 (2012)
7. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information (2016). arXiv preprint: arXiv:1607.04606
8. Castro, D., Ortega, R., Muñoz, R.: Author masking by sentence transformation—notebook for PAN at CLEF 2017. In: Cappellato, L., Ferro, N., Goeuriot, L., Mandl, T. (eds.) CLEF 2017 Evaluation Labs and Workshop – Working Notes Papers, Dublin, Ireland, 11–14 September. CEUR-WS.org, September 2017. http://ceur-ws.org/Vol-1866/

9. Chatzikokolakis, K., Andrés, M.E., Bordenabe, N.E., Palamidessi, C.: Broadening the scope of differential privacy using metrics. In: De Cristofaro, E., Wright, M. (eds.) PETS 2013. LNCS, vol. 7981, pp. 82–102. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39077-7_5
10. Coavoux, M., Narayan, S., Cohen, S.B.: Privacy-preserving neural representations of text. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, pp. 1–10. Association for Computational Linguistics, October–November 2018. http://www.aclweb.org/anthology/D18-1001
11. Cumby, C., Ghani, R.: A machine learning based system for semi-automatically redacting documents. In: Proceedings of the Twenty-Third Conference on Innovative Applications of Artificial Intelligence (IAAI) (2011)
12. Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 265–284. Springer, Heidelberg (2006). https://doi.org/10.1007/11681878_14
13. Dwork, C., Roth, A., et al.: The algorithmic foundations of differential privacy. Found. Trends® Theor. Comput. Sci. **9**(3–4), 211–407 (2014)
14. Emmery, C., Manjavacas, E., Chrupała, G.: Style obfuscation by invariance (2018). arXiv preprint: arXiv:1805.07143
15. Fernandes, N., Dras, M., McIver, A.: Generalised differential privacy for text document processing. CoRR abs/1811.10256 (2018). http://arxiv.org/abs/1811.10256
16. Manuel, F., Pardo, R., Rosso, P., Potthast, M., Stein, B.: Overview of the 5th author profiling task at PAN 2017: gender and language variety identification in Twitter. In: Cappellato, L., Ferro, N., Goeuriot, L., Mandl, T. (eds.) Working Notes Papers of the CLEF 2017 Evaluation Labs. CEUR Workshop Proceedings, vol. 1866. CLEF and CEUR-WS.org, September 2017. http://ceur-ws.org/Vol-1866/
17. Global, T.: Native Language Identification (NLI) Establishes Nationality of Sony's Hackers as Russian. Technical report, Taia Global, Inc. (2014)
18. Grimmett, G., Stirzaker, D.: Probability and Random Processes, 2nd edn. Oxford Science Publications, Oxford (1992)
19. Halvani, O., Winter, C., Graner, L.: Authorship Verification based on Compression-Models. CoRR abs/1706.00516 (2017). http://arxiv.org/abs/1706.00516
20. Hasler, E., Stahlberg, F., Tomalin, M., de Gispert, A., Byrne, B.: A comparison of neural models for word ordering. In: Proceedings of the 10th International Conference on Natural Language Generation, pp. 208–212. Association for Computational Linguistics (2017). https://doi.org/10.18653/v1/W17-3531. http://aclweb.org/anthology/W17-3531
21. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. SSS, 2nd edn. Springer, New York (2009). https://doi.org/10.1007/978-0-387-84858-7
22. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification (2016). arXiv preprint: arXiv:1607.01759
23. Kacmarcik, G., Gamon, M.: Obfuscating document stylometry to preserve author anonymity. In: ACL, pp. 444–451 (2006)
24. Karadzhov, G., Mihaylova, T., Kiprov, Y., Georgiev, G., Koychev, I., Nakov, P.: The case for being average: a mediocrity approach to style masking and author obfuscation. In: Jones, G.J.F., et al. (eds.) CLEF 2017. LNCS, vol. 10456, pp. 173–185. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-65813-1_18

25. Khonji, M., Iraqi, Y.: A slightly-modified GI-based author-verifier with lots of features (ASGALF). In: Working Notes for CLEF 2014 Conference (2014). http://ceur-ws.org/Vol-1180/CLEF2014wn-Pan-KonijEt2014.pdf
26. König, D.: Theorie der endlichen und unendlichen Graphen. Akademische Verlags Gesellschaft, Leipzig (1936)
27. Koppel, M., Argamon, S., Shimoni, A.R.: Automatically categorizing written texts by author gender. Lit. Linguist. Comput. **17**(4), 401–412 (2002). https://doi.org/10.1093/llc/17.4.401
28. Koppel, M., Schler, J., Argamon, S.: Authorship attribution in the wild. Lang. Resour. Eval. **45**(1), 83–94 (2011)
29. Koppel, M., Winter, Y.: Determining if two documents are written by the same author. JASIST **65**(1), 178–187 (2014). https://doi.org/10.1002/asi.22954
30. Kroese, D.P., Taimre, T., Botev, Z.I.: Handbook of Monte Carlo Methods, vol. 706. Wiley, New York (2013)
31. Kusner, M.J., Sun, Y., Kolkin, N.I., Weinberger, K.Q.: From word embeddings to document distances. In: Proceedings of the 32nd International Conference on Machine Learning, pp. 957–966 (2015)
32. Li, Y., Baldwin, T., Cohn, T.: Towards robust and privacy-preserving text representations. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Short Papers, vol. 2, pp. 25–30. Association for Computational Linguistics (2018). http://aclweb.org/anthology/P18-2005
33. Malmasi, S., Dras, M.: Native language identification with classifier stacking and ensembles. Comput. Linguist. **44**(3), 403–446 (2018). https://doi.org/10.1162/coli_a_00323
34. Mansoorizadeh, M., Rahgooy, T., Aminiyan, M., Eskandari, M.: Author Obfuscation using WordNet and language models—notebook for PAN at CLEF 2016. In: Balog, K., Cappellato, L., Ferro, N., Macdonald, C. (eds.) CLEF 2016 Evaluation Labs and Workshop – Working Notes Papers, Évora, Portugal, 5–8 September. CEUR-WS.org, September 2016. http://ceur-ws.org/Vol-1609/
35. Marsaglia, G., et al.: Choosing a point from the surface of a sphere. Ann. Math. Stat. **43**(2), 645–646 (1972)
36. McDonald, A.W.E., Afroz, S., Caliskan, A., Stolerman, A., Greenstadt, R.: Use fewer instances of the letter "i": toward writing style anonymization. In: Fischer-Hübner, S., Wright, M. (eds.) PETS 2012. LNCS, vol. 7384, pp. 299–318. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31680-7_16
37. McMahan, H.B., Ramage, D., Talwar, K., Zhang, L.: Learning differentially private recurrent language models. In: International Conference on Learning Representations (2018). https://openreview.net/forum?id=BJ0hF1Z0b
38. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR abs/1301.3781 (2013). http://arxiv.org/abs/1301.3781
39. Potha, N., Stamatatos, E.: An improved *Impostors* method for authorship verification. In: Jones, G.J.F., Lawless, S., Gonzalo, J., Kelly, L., Goeuriot, L., Mandl, T., Cappellato, L., Ferro, N. (eds.) CLEF 2017. LNCS, vol. 10456, pp. 138–144. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-65813-1_14
40. Potthast, M., et al.: Who wrote the web? Revisiting influential author identification research applicable to information retrieval. In: Ferro, N., et al. (eds.) ECIR 2016. LNCS, vol. 9626, pp. 393–407. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-30671-1_29

41. Potthast, M., Rangel, F., Tschuggnall, M., Stamatatos, E., Rosso, P., Stein, B.: Overview of PAN'17: author identification, author profiling, and author obfuscation. In: Jones, G.J.F., et al. (eds.) CLEF 2017. LNCS, vol. 10456, pp. 275–290. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-65813-1_25
42. Rodriguez-Garcia, M., Batet, M., Sánchez, D.: Semantic noise: privacy-protection of nominal microdata through uncorrelated noise addition. In: 2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 1106–1113. IEEE (2015)
43. Rubner, Y., Tomasi, C., Guibas, L.J.: The earth mover's distance as a metric for image retrieval. Int. J. Comput. Vis. **40**(2), 99–121 (2000)
44. Ruder, S., Ghaffari, P., Breslin, J.G.: Character-level and Multi-channel Convolutional Neural Networks for Large-scale Authorship Attribution. CoRR abs/1609.06686 (2016). http://arxiv.org/abs/1609.06686
45. Sánchez, D., Batet, M.: C-sanitized: a privacy model for document redaction and sanitization. J. Assoc. Inf. Sci. Technol. **67**(1), 148–163 (2016)
46. Sapkota, U., Bethard, S., Montes, M., Solorio, T.: Not all character N-grams are created equal: a study in authorship attribution. In: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, pp. 93–102. Association for Computational Linguistics, May–June 2015. http://www.aclweb.org/anthology/N15-1010
47. Seidman, S.: Authorship verification using the imposters method. In: Working Notes for CLEF 2013 Conference (2013). http://ceur-ws.org/Vol-1179/CLEF2013wn-PAN-Seidman2013.pdf
48. Shetty, R., Schiele, B., Fritz, M.: A4NT: author attribute anonymity by adversarial training of neural machine translation. In: 27th USENIX Security Symposium, pp. 1633–1650. USENIX Association (2018)
49. Shokri, R., Shmatikov, V.: Privacy-preserving deep learning. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, CCS 2015, pp. 1310–1321. ACM, New York (2015). https://doi.org/10.1145/2810103.2813687
50. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertain. Fuzziness Knowl. Based Syst. **10**(5), 557–570 (2002)
51. Tetreault, J., Blanchard, D., Cahill, A.: A report on the first native language identification shared task. In: Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications, Atlanta, Georgia, pp. 48–57. Association for Computational Linguistics, June 2013. http://www.aclweb.org/anthology/W13-1706
52. Wan, S., Dras, M., Dale, R., Paris, C.: Improving grammaticality in statistical sentence generation: introducing a dependency spanning tree algorithm with an argument satisfaction model. In: Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009), pp. 852–860. Association for Computational Linguistics (2009). http://aclweb.org/anthology/E09-1097
53. Weggenmann, B., Kerschbaum, F.: SynTF: synthetic and differentially private term frequency vectors for privacy-preserving text mining (2018). arXiv preprint: arXiv:1805.00904
54. Zhang, Y., Clark, S.: Discriminative syntax-based word ordering for text generation. Comput. Linguist. **41**(3), 503–538 (2015). https://doi.org/10.1162/COLI_a_00229