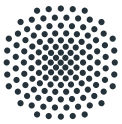


Universität Stuttgart

Differential Privacy for Sequential and Directional Data

Von der Fakultät 5 (Informatik, Elektrotechnik und Informationstechnik)
der Universität Stuttgart zur Erlangung der Würde eines Doktors
der Naturwissenschaften (Dr. rer. nat.) genehmigte Abhandlung

Vorgelegt von

Benjamin Weggenmann

aus Ulm

Hauptberichter: Prof. Dr. Ralf Küsters

Mitberichter: Prof. Dr. Florian Kerschbaum, University of Waterloo, Kanada
Prof. Dr. George Danezis, University College London, UK

Tag der mündlichen Prüfung: 4. Oktober 2023

Institut für Informationssicherheit (SEC) der Universität Stuttgart

2023

Acknowledgements

First and foremost, I am deeply grateful to my PhD advisor Florian Kerschbaum for accepting me as PhD student to begin with, and subsequently, for engaging in countless fruitful discussions as well as for providing guidance and encouragement throughout this endeavor. I also want to express my sincere gratitude to Ralf Küsters for taking me in as external PhD student and for the collaboration across institution boundaries.

Life as PhD student would be only half the fun without fellow (PhD) students and researchers; therefore, I want to extend big thanks to my colleagues at SAP Security Research, namely Andreas, Anselme, Benny, Daniel, Florian, Jonas, and Martin, as well as to my students with whom I worked on various topics. In particular, I want to thank Valentin, Jonas, Linda, Michael, and Justus for the inspiring discussions and great collaborations. Also, I would like to thank my current and former managers, Mathias Kohler and Detlef Plümper, for providing the opportunity of pursuing doctoral studies within SAP.

Last but not least, I want to thank my friends and family, for all the love, support, encouragement, and patience. I would not be who or where I am today without you.

Contents

List of Figures

List of Tables

List of Algorithms

List of Abbreviations

AAE adversarial autoencoder 9, 11, 55, 108, 116, 119, 168

BoW Bag-of-Words 8, 13, 43, 47, 48, 75, 76, 78–80, 87, 90, 100, 104, 109, 110, 112, 167, 170

CDF cumulative distribution function 16, 123, 130, 133, 135, 137, 138, 140, 147, 151, 164

CNN convolutional neural network 41, 53, 58

CRF conditional random field 51, 58

DLP data leakage prevention 35, 42, 43

DP differential privacy xix, xxi, 1, 3, 6–13, 15, 17–25, 28, 29, 42, 46–49, 62–65, 67–69, 71, 72, 75, 77–85, 95–97, 99–101, 104–108, 113, 115, 116, 119, 121–123, 130, 131, 135, 141, 147, 153, 155–158, 160–163, 167–173

DP-SGD differentially private stochastic gradient descent 12, 18, 119

ELBO evidence lower bound 101, 103

GAN generative adversarial network 41, 46, 52, 54, 60, 61, 66

GDPR General Data Protection Regulation 4–6

GI geo-indistinguishability 7, 11, 27, 67, 68, 122, 162

GMM Gaussian mixture model 39, 41, 55

GRU gated recurrent unit 63, 108, 109, 172

HIPAA Health Insurance Portability and Accountability Act 3–5, 32–34, 50, 62, 96

HMM hidden Markov model 39, 41, 58

i.i.d. independent and identically distributed 23–25

IR information retrieval 75–77, 79

KL Kullback–Leibler 21, 101, 103

LBS location-based service 1, 3, 8, 158

LLM large language model 12

LSTM long short-term memory 52, 172

MCC Matthews correlation coefficient 112, 115

ML machine learning 18, 34, 35, 38, 48, 51, 119

NER named entity recognition 34, 38, 50, 51, 53

NLP natural language processing 51

OCR optical character recognition 39, 40

PCA principal component analysis 66

PDF probability density function 16, 23–26, 28, 124, 142, 151

PHI protected health information 3, 4, 50, 62

PII personally identifiable information 2, 4, 8, 13, 31, 32, 34, 50, 51, 76, 94, 96, 172

PL Planar Laplace 7, 27, 48, 62, 65, 67, 144, 162, 171

RDP Rényi differential privacy 20–22, 26, 27, 72, 101, 107

RNN recurrent neural network 41, 51, 108, 109, 172

SGD stochastic gradient descent 103

SVM support vector machine 51, 89, 90, 94, 111, 112

tf term frequency 9, 51, 75, 77, 79–81, 87, 89, 90, 97, 167

tf–idf term frequency–inverse document frequency 77

tf–idf vector term frequency–inverse document frequency vector 80, 81, 89

VAE variational autoencoder xix, 9, 11, 13, 60, 100–102, 104–107, 116, 119, 168

VMF von Mises–Fisher 10, 11, 13, 121, 123, 127, 130–134, 138–140, 146, 147, 151, 153, 155–161, 163, 164, 168

WL Wrapped Laplace 123, 141–144, 155, 157–160

Abstract

This dissertation is concerned with mechanisms to protect the privacy of individuals in special types of data that are sequential or directional in nature. Importantly, sequential data includes human language which is commonly conveyed as text or speech (i.e., a sequence of words, symbols, or speech sounds), whereas directional data includes natural examples such as geographic locations and periodic time specifications. In many cases, such data may expose sensitive information that violate the privacy of individuals or even reveal their identity. *Differential privacy* (DP) is a formal notion of privacy based on randomness that allows quantifying and limiting information disclosure about individuals. While many DP mechanisms exist for structured data such as scalars or numerical vectors, we found a lack of suitable mechanisms for sequential and directional data: For instance, at the time of starting this dissertation, we found no existing DP mechanisms for textual data, and existing mechanisms for geolocations assumed only planar coordinates.

To fill these gaps, we aim at constructing novel privacy mechanisms for sequential and directional data and assessing their DP properties. Specifically, we develop methods to obfuscate text as an example of sequential data which either produce differentially private text representations or human-readable texts. Moreover, we introduce directional privacy, a special variant of DP for directional data along with two suitable directional privacy mechanisms that intrinsically respect the directional nature of the data to be obfuscated. We evaluate our proposed methods in realistic use cases to assess their performance regarding protection of privacy and preservation of utility in the obfuscated data. The results show that our methods for text effectively reduce re-identification risks of authorship attribution attacks while maintaining high utility for topic or sentiment analysis tasks. Furthermore, our directional mechanisms typically require fewer data to achieve a certain level of utility than standard privacy mechanisms adapted to directional data.

To our best knowledge, our work contributes the first DP mechanism for text and also has inspired other mechanisms that work on a word-level. Moreover, we are the first to exploit synergies between variational autoencoders and the Gaussian mechanism to achieve DP for human-readable text—an approach that is likely extensible to other domains of sequential data. Lastly, our work on directional privacy further provides theoretical contributions to directional statistics including a novel sampling algorithm for the Purkayastha distribution.

Abstract

Kurzzusammenfassung

Diese Dissertation beschäftigt sich mit Mechanismen zum Schutz der Privatsphäre von Personen in speziellen Arten von Daten, welche sequenzielle oder richtungsbezogene Eigenschaften haben. Ein bedeutsames Beispiel für sequenzielle Daten ist die menschliche Sprache, die üblicherweise als Text oder gesprochene Sprache (d.h., als Folge von Wörtern, Symbolen oder Sprachlauten) übermittelt wird, während zu den gerichteten Daten natürliche Beispiele wie geografische Orts- und periodische Zeitangaben gehören. In vielen Fällen können solche Daten sensible Informationen enthalten, die die Privatsphäre von Personen verletzen oder sogar ihre Identität preisgeben. *Differential Privacy* (DP) ist ein formaler Ansatz zum Schutz der Privatsphäre, der auf Zufälligkeit basiert und es ermöglicht, die Offenlegung von Informationen über Einzelpersonen zu quantifizieren und zu begrenzen. Während es für strukturierte Daten wie Skalare oder numerische Vektoren viele geeignete DP-Mechanismen gibt, haben wir festgestellt, dass es an geeigneten Mechanismen für sequenzielle und richtungsbezogene Daten mangelt: Zu Beginn dieser Dissertation konnten wir beispielsweise keine bestehenden DP-Mechanismen für Textdaten finden, während DP-Mechanismen für geografische Daten ein flaches Koordinatensystem voraussetzten.

Um diese Lücken zu schließen, setzen wir uns zum Ziel, neuartige Datenschutzmechanismen für sequenzielle und richtungsbezogene Daten zu entwickeln sowie deren DP-Eigenschaften zu bestimmen. Insbesondere entwickeln wir Methoden zur Verschleierung von Text als Spezialfall von sequenziellen Daten, die entweder kodierte Textrepräsentationen oder lesbare Fließtexte erzeugen. Darüber hinaus führen wir *Directional Privacy* als Spezialfall von DP für richtungsbezogene Daten ein, und präsentieren zwei geeignete Mechanismen, die den richtungsbezogenen Charakter der zu verschleiernden Daten berücksichtigen. Wir evaluieren unsere vorgeschlagenen Methoden in realistischen Anwendungsfällen, um ihre Eignung zum Schutz der Privatsphäre und des Erhalts des Nutzwerts der verschleierten Daten zu bewerten. Die Ergebnisse zeigen, dass unsere Mechanismen für Text die Risiken von Reidentifizierungs-Angriffen auf die Autoren der Texte effektiv reduzieren und gleichzeitig einen hohen Nutzenwert für Themen- oder Sentimentanalyse beibehalten. Darüber hinaus benötigen unsere richtungsbezogenen Mechanismen in der Regel weniger Daten, um eine bestimmte Güte an Genauigkeit zu erreichen, als normale, einfach an richtungsweisende Daten angepasste DP-Mechanismen.

Nach unserem besten Wissen stellt diese Arbeit den ersten DP-Mechanismus für Textdaten überhaupt vor und hat auch andere Mechanismen, die auf Wort-Ebene arbeiten, inspiriert. Weiterhin nutzen wir als erste Synergien zwischen *Variational Autoencoder* und dem *Gauß-Mechanismus*, um DP für lesbaren Text zu erreichen – ein Ansatz, der potenziell auch auf andere sequenzielle Datentypen erweiterbar ist. Schließlich liefert unsere Arbeit zu *Directional Privacy* weitere theoretische Beiträge zur gerichteten Statistik, einschließlich eines neuartigen Sampling-Algorithmus für die Purkayastha-Verteilung.

Chapter 1

Introduction

In recent years, large-scale collection and processing of data have become important drivers for the digital economy: For instance, users can share reviews of businesses and services through various online platforms, and location-based services (LBS) collect and analyze the geographic location of the users or their mobile devices. In these examples, and in many other cases, the data involved have a special structure: Reviews consist of texts, i.e., *sequences* of words or characters representing human language, and geolocations are *directional* in nature. However, text and location data are often privacy-sensitive, so adequate methods to protect the privacy of the users are required to encourage them to actually share their data.

Unfortunately, at the time of beginning this dissertation, there were no or only inadequate methods that offer formal privacy guarantees, more specifically differential privacy (DP), to protect text and geolocation data. Therefore, the goal of this work is to develop novel obfuscation methods with formal privacy guarantees, namely DP, for sequential and directional types of data, such as text and geolocations, respectively.

In this chapter, we first motivate our research from an application, user, and legal perspective. Next, we define the research problem and objectives, summarize our main contributions, and give an overview of related publications. Lastly, we outline the structure of the dissertation.

1.1 Motivation

The Internet has paved the way for many online platforms and services that allow individuals to interact in various ways, e.g., by checking in at venues such as bars and restaurants, and by sharing their opinions about various products and services. In many scenarios, this involves certain types of data that are sequential or directional in nature: For instance, human language, our primary means of communication, makes up a major proportion of online communication and digital media, including instant messages, online

reviews, tweets and comments, etc. Human language is typically conveyed as text or speech, i.e., a *sequence* of either written symbols (e.g., words or characters) or spoken sounds constituting phonemes and words. Furthermore, crowd-sourced data from mobile or wearable devices often includes the geographic location where and the time when the data was recorded; such types of spatial and (periodic) temporal data can be regarded as instances of *directional* data. Some social media platforms like Twitter allow users to post comments in form of short texts (e.g., so-called *tweets*) that include the time and optionally also their location [443], thus representing a combining sequential and directional data.

The resulting collection and processing of massive amounts of user data have become important drivers for the digital economy: The availability of crowd-sourced feedback on review platforms (e.g., Google Maps, Yelp, Glassdoor, IMDb) and shops (e.g., Amazon, eBay) not only helps prospective customers, patients, or employees to make informed decisions about their next buys, visits, or to evaluate their next potential employers, but also allows the business owners to analyze the provided feedback to gain insights into how their products, services, or brand image can be improved. Moreover, crowd-sourced location data drive *location-based marketing and analytics* platforms such as Foursquare, and allow *mapping services* such as Google Maps to estimate traffic and navigate users to their destinations on efficient routes, and to create daily "busyness" histograms that indicate popular visit times at places like stores or restaurants [170, 283], from which users can estimate how busy a location is during different times of the day. Lastly, the platforms and service providers themselves benefit from the collected data, e.g., by enabling them to provide (or rather harass) users with targeted advertising.

1.1.1 Privacy Matters

While the large-scale collection and processing of data from online platforms and services drives innovation and provides substantial value for both businesses and users, the data collected in such scenarios is often privacy-sensitive, which we discuss in the following.

Threats and Risks. Sequential data like text and speech often include personally identifiable information (PII) as described in Section 1.1.2, such as full names or addresses. Traditional sanitization approaches often work by removing those parts, or replacing them with pseudonyms (cf. the de-identification approaches we discuss as part of Section 3.3.4). Also, many online platforms allow users to post their reviews "anonymously", i.e., under a pseudonym without *directly* revealing their identity. However, the absence of explicitly identifying information is generally not enough to provide true anonymity: In many scenarios, users can be re-identified based on metadata or the data itself, e.g., by linkage

attacks [92, 93, 207, 318, 364] or through various identifiers contained in sequential data (cf. Section 3.2).

This is particularly critical for textual data which makes up a major proportion of online content posted by users and which represents a rich source of information: For instance, users can be identified based on their pseudonymized search logs [34], and even their *writing style* alone may be sufficient to de-anonymize them through modern authorship attribution techniques [127, 364, 376, 409, 422]. This may entail undesirable risks and consequences, ranging from, e.g., retaliatory actions and legal disputes for publicly criticizing businesses on Yelp or Google Maps [195], disclosure of patient identities and their sensitive personal information [57], over sanctions from the employer to potential lawsuits in the millions for critical reviews on sites like Glassdoor [323]. Users may hence feel reluctant to provide their honest feedback for fear of retaliation [371, 398], which also concerns internal surveys [414].

Also, in many cases, directional data conveys particularly sensitive information, as illustrated by recent news about location tracking on smartphones or fitness trackers [187, 437]. Personal locations are suspect to various attacks, cf. the survey by Krumm [243], in particular when combined with temporal information as shown by Primault et al. [353] or Pyrgelis et al. [356]. In fact, sufficiently accurate location information such as addresses are classified as protected health information (PHI) according to the Health Insurance Portability and Accountability Act (HIPAA) [327, 435] as described in the subsequent Section 1.1.2.1.

Consequences. As a result, users may be reluctant to share their opinions, comments, and feedback in surveys and online platforms, or their whereabouts during the course of the day to LBS. To convince them otherwise, it is thus necessary to develop methods that protect the anonymity of the users while preserving the quality and content of the original data, and ideally meet formal privacy guarantees in the form of DP [117], which is widely regarded as the gold standard of privacy protection.

1.1.2 Legal Aspects

Privacy laws that govern the collection and handling of personal data have been enacted in various legislations. In the following, we briefly discuss some relevant ones which also motivate the need for data anonymization methods as presented in this dissertation.

1.1.2.1 Health Insurance Portability and Accountability Act (HIPAA)

The Health Insurance Portability and Accountability Act (HIPAA) is a US law passed in 1996 governing the protection and handling of sensitive patient data [435, 447]. Specifically, in the context of health care, health records and medical documents containing PII that can identify an individual (e.g., a patient or their relatives) are referred to as *PHI* whose use is strictly limited by the HIPAA Privacy Rule [327]. However, the HIPAA Privacy Rule does not restrict the use or disclosure of *de-identified* health data which is considered as not individually identifiable and hence no longer regarded as PHI.

De-Identified Data. De-identification according to HIPAA may be achieved following "Expert Determination" or the "Safe Harbor" method: "Safe Harbor" defines a set of 18 identifiers that are considered PII pertaining to the individual, their relatives, employer, and household members, which must be removed from a de-identified document. These identifiers include, for instance, names, addresses/locations ("geographic subdivisions smaller than state"), phone and fax numbers, email and IP addresses, photos, biometric identifiers, etc. Since health records often consist of unstructured text, the "Safe Harbor" method inspired several software solutions to detect and mask (i.e., redact or replace) the 18 HIPAA identifiers in unstructured text. We discuss such de-identification methods that only change parts of a text in more detail in Section 3.3.4.1.

In contrast to existing de-identification methods, the obfuscation approaches presented in Chapters 5 and 6 of this dissertation not only change individual terms but transform entire texts to provide formal (differential) privacy guarantees and also defend against authorship attribution attacks. We discuss related defense techniques for text as well as audio and visual data in Section 3.3. Lastly, our work on *directional privacy* in Chapter 7 specifically addresses geographic locations as one type of HIPAA identifier. Therefore, the methods proposed in this dissertation could also be applied to medical use cases.

1.1.2.2 General Data Protection Regulation (GDPR)

The General Data Protection Regulation (GDPR) is a privacy and security law passed by the European Union (EU) in 2016 and put into effect on May 25, 2018 [125]. It covers the collection and processing of personal data related to people in the EU by organizations *anywhere*, even outside the EU. More concretely, the GDPR specifies (emphasis ours) that

> '*personal data*' means any information relating to an identified or identifiable natural person ('*data subject*'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier

> such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person [125, Article 4].

Collection and processing of personal data are limited to "specified, explicit and legitimate purposes" [125, Article 5]. Moreover, it requires a legal basis according to Article 6; for instance, this is provided if "the data subject has given consent to the processing of his or her personal data for one or more specific purposes" [125, Article 6].

Pseudonymous Data. The GDPR encourages *pseudonymization* of personal data to reduce privacy risks to the data subject [277], which in turn may allow processing of personal data beyond the original purposes and facilitate its use, e.g., for scientific, historical, or statistical purposes [125, Article 89]. In terms of the Regulation,

> '*pseudonymisation*' means the processing of personal data in such a manner that the personal data can no longer be attributed to a specific data subject without the use of additional information, provided that such additional information is kept separately and is subject to technical and organisational measures to ensure that the personal data are not attributed to an identified or identifiable natural person [125, Article 4].

Pseudonymization thus "reduces (but does not eliminate) compliance obligations", potentially "enabling a wider range of lawful productive uses of data" [49]. However, note that in general, *pseudonymized data still is considered personal data* and hence subject to the Regulation: "Personal data which have undergone pseudonymisation, which could be attributed to a natural person by the use of additional information should be considered to be information on an identifiable natural person" [125, Recital 26].

Differences to HIPAA. While the HIPAA and GDPR both share similar concepts and goals about secure and private handling of personal data, in general the GDPR goes beyond HIPAA in several aspects: First, the GDPR covers not only health data, but any kind of personally sensitive information. Second, the GDPR is stricter regarding the use of pseudonymized data: Both de-identification according to HIPAA and pseudonymization under the GDPR do not rule out re-identification of the data subject. However, while de-identified health data is no longer restricted under the HIPAA Privacy Rule, pseudonymized data is generally still subject to the GDPR [125, Recital 26] as discussed in the preceding paragraph.

Re-Identification. The main risk is that it may still be possible to *re-identify* the data subject of pseudonymized data, even after direct and indirect identifiers have been removed:

> To determine whether a natural person is identifiable, account should be taken of all the means reasonably likely to be used, such as singling out, either by the controller or by another person to identify the natural person directly or indirectly. To ascertain whether means are reasonably likely to be used to identify the natural person, account should be taken of all objective factors, such as the costs of and the amount of time required for identification, taking into consideration the available technology at the time of the processing and technological developments [125, Recital 26].

How to re-identify data is often not immediately obvious, but it could be possible, for instance, by linking it to external data (cf., e.g., [34, 317, 430]). In Section 3.2, we discuss specific examples of re-identification methods for sequential data, including *authorship attribution* for text.

Anonymous Data. Unlike with pseudonymous data, re-identification of the data subject pertaining to *anonymous data* must *not* be (reasonably) possible, so data anonymization of personal data is an irreversible process. In that context, the GDPR states the following:

> The principles of data protection should therefore *not apply to anonymous information*, namely information which does not relate to an identified or identifiable natural person or to personal data rendered anonymous in such a manner that the data subject is not or no longer identifiable. This Regulation does not therefore concern the processing of such anonymous information, including for statistical or research purposes [125, Recital 26].

The benefit of truly anonymized data therefore is that it is no longer subject to the GDPR, thus unlocking its potential for other uses. To this end, the goal of this dissertation is to develop methods for sequential and directional data that achieve proper data anonymization with formal privacy guarantees and prevent re-identification methods such as authorship attribution while maintaining good utility for legitimate purposes.

1.2 Research Problem

To protect the privacy of individuals while maintaining data-driven business models, the concept of *differential privacy (DP)* by Dwork et al. [117] presents the current state-of-the-art for quantifying and limiting information disclosure about individuals. DP mechanisms

have been proposed for various settings and data types, e.g., the standard Laplace [117] and Gaussian mechanisms [116], as well as the Planar Laplace (PL) mechanism [24], which are defined in Euclidean spaces. As such, they are readily applicable to structured data such as numbers or vectors, including, for instance, *planar* locations.

However, it is challenging to apply DP to sequential data such as text or speech, which comes in varying lengths and with different ways to express the same idea. Similarly, while post-processing, such as clipping or wrapping, can be applied to adapt common Euclidean-space DP mechanisms to spherical domains, none of them intrinsically respects the directional nature of the underlying data such as geolocations. In fact, at the time of starting the research for this dissertation, to the best of our knowledge, there were no DP mechanisms for text, and geo-indistinguishability by Andrés et al. [24] considered geolocations only as planar coordinates. Consequently, existing approaches are insufficient or inadequate to provide formal privacy guarantees for special data types, such as text and speech or geolocations and periodic time specifications, that are sequential or directional in nature.

1.3 Research Objectives

Given the lack of suitable DP mechanisms for text as an important example of sequential data, and the lack of suitable DP mechanisms for directional data that respect the underlying directional structure, the overall objective in this dissertation is to devise methods that provide provable privacy guarantees (e.g., DP) for these special types of data. More specifically, we aim at achieving the following research objectives:

RO1 Design novel DP mechanisms to obfuscate text as an illustrative example of sequential data.

RO2 Evaluate the performance of the proposed DP mechanisms for text in realistic scenarios, in particular how well they protect against authorship attribution attacks.

RO3 Design specialized DP mechanisms for directional data that intrinsically respect the directional nature of the data.

RO4 Evaluate the performance of the proposed DP mechanisms for directional data in realistic scenarios.

1.3.1 Limitations

Narrowed Focus on Specific Domains. Due to the vast number of sequential and directional data types, we need to narrow down the scope of this dissertation to specific, illustrative examples:

- For sequential data, we focus our research on textual data. Text is an important example that represents human language, our primary means of communication, and is used to send messages, share opinions, or post comments on social media sites and many other online platforms. However, to provide a sense of connection among other types of sequential data, we also review related work on attacks and defenses for audio and visual data along with text in Chapter 3.
- For directional data, we focus on geolocations and periodic time specifications. These are natural examples of directional data that frequently occur in our daily lives and are typically used in LBS and online mapping platforms.

While we are aware of many other interesting instances of sequential data (e.g., sensor readings or time series in general) and high-dimensional examples of directional data (e.g., gene expression vectors [103]), we regard other types of sequential and directional data as out of scope for this dissertation and leave them for future research (see Section 8.4).

Focus on Identifying Information (Identifiers). Apart from the identity of the originator of the data, other privacy-sensitive information could be inferred from sequential data: For instance, the writing style of a text or characteristic movement patterns in motion sensor data may also reveal the age or gender of an individual. However, since the identity of an individual is the most specific inference an attacker could make about the individual, and since legal regulations specifically protect PII (cf. Section 1.1.2), we also decided to focus on identifying information (identifiers) in our research in this dissertation.

1.4 Contributions

This section provides an overview of our contributions to knowledge in the field of DP for sequential and directional data, and, in particular, towards the research objectives stated in Section 1.3.

1.4.1 The First Differentially Private Mechanism for Text

In Chapter 5, we present *SynTF*, a novel DP method to compute *private* Bag-of-Words (BoW) *representations* (cf. Sections 3.3.2 and 3.3.3) for textual data. It works by randomly

replacing words from an input text with similar words using the Exponential mechanism by McSherry and Talwar [294] (cf. Section 2.3.4) and counting the resulting terms in a term frequency (tf) vector, which can be used as a feature vector for common information retrieval and text mining tasks such as text classification. To the best of our knowledge, SynTF represents the first published DP mechanism for textual data, as confirmed in the survey by Zhao and Chen [494]. Moreover, we argue that it also pioneered the line of research we call *word-level DP*, whose methods have in common that they replace words or tokens in a text independently in a differentially private manner. Such methods may be readily extended to mechanisms for entire texts by iterating over the texts word-by-word (cf. Section 3.3.3.1), however, they do have some limitations which we discuss in Section 3.3.5.1.

On the theoretical side, in Section 5.2.4, we prove the ϵ-DP properties of SynTF. Furthermore, we derive a heuristic argument that the privacy loss ϵ of the Exponential mechanism grows logarithmically in the size of the (discrete) output space if the result should provide a minimum level of utility. We experimentally verify our method on a corpus of newsgroups postings in a scenario where a benign analyst wants to infer the topic from the texts, whereas a malicious attacker tries to identify their author (Section 5.3). To better prevent authorship attribution, we introduce the *bigram overlap* as an additional technique that influences the choice of substituted words. The results indicate that our method has a much stronger impact on authorship attribution than on the topic inference task, whereas *scrubbing methods* (cf. Section 3.3.4.1) that only mask privacy-sensitive terms provide only insufficient protection.

1.4.2 Differential Privacy Mechanisms for Coherent, Human-Readable Text Obfuscation

In Chapter 6, we approach major limitations of word-level DP methods (cf. Section 3.3.5.1), with a novel text obfuscation approach that applies DP to full sentences instead of individual words: The fundamental method, DP-VAE, consists of a variational autoencoder (VAE) architecture that first encodes the input sentences to continuous, probabilistic latent representations following a Gaussian distribution. By imposing two constraints on the parametrization of the Gaussian distributions, we are able to exploit synergies with the Gaussian mechanism, resulting in differentially private latent samples which the decoder finally transforms into diverse and coherent, human-readable output texts. Furthermore, we propose an extension of DP-VAE to a differentially private adversarial autoencoder (*DP-AAE*) by integrating adversarial learning to disentangle the latent representations into a privacy-sensitive author/style vector and a privacy-insensitive content vector. This separation enables further improvements of the privacy-utility trade-off in a favorable

direction by applying stronger noise to the author vector.

We perform an extensive evaluation involving hyperparameter optimization to compare our DP-VAE and DP-AAE models against two non-DP baselines in a scenario with online reviews whose authors wish to remain anonymous (Section 6.5). The results indicate that DP-AAE outperformed all other methods and effectively reduces re-identification risks against authorship attribution attacks while producing readable sentences and preserving the content of the texts.

1.4.3 Novel Differential Privacy Definition and Mechanisms for Directional Data

In Chapter 7, we address the lack of suitable DP methods for the important class of directional data: First, we introduce a new notion of *directional privacy* based on the surface distance on the sphere. We then devise two conforming mechanisms based on the spherical von Mises–Fisher (VMF) and Purkayastha distributions that intrinsically suit directional data and prove that they fulfill directional as well as pure differential privacy. Furthermore, as a theoretical contribution, we derive various statistical properties such as expected distances, related densities and cumulative distribution functions for the underlying distributions. These results allow us to (i) show that adopted standard mechanisms based on wrapping can behave worse than the uniform distribution, and (ii) develop a novel sampling algorithm for the Purkayastha distribution for which to our best knowledge, no designated sampling method had been published before.

Moreover, we perform several analyses and experiments on real data to evaluate our directional privacy mechanisms: Specifically, we demonstrate their applicability to important applications, such as privately collecting mobility data in the local model, where the data collector cannot or may not be trusted by the users. Importantly, our results show an advantage of our directional privacy mechanisms over standard privacy mechanisms adapted to directional data, since our directional mechanisms typically required fewer data to achieve a certain level of utility (i.e., they have a lower sample complexity, cf. Section 8.3). We also demonstrate that for some directional statistics such as the circular mean, the local model can achieve a sample complexity as low as in the central model, making it preferable since it also does not require a trusted aggregator.

1.5 Integral and Related Publications

Integral Publications Contributing to this Dissertation. The main contributions presented in Chapters 5 to 7 of this dissertation are based on the following key publications:

Chapter 5 Benjamin Weggenmann and Florian Kerschbaum [465, 466]: "SynTF: Synthetic and Differentially Private Term Frequency Vectors for Privacy-Preserving Text Mining". In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval* (*SIGIR '18*).

Florian Kerschbaum suggested the topic of text anonymization with DP guarantees. The author developed the core idea of randomly replacing words using the Exponential mechanism and enhanced its rating function with the bigram overlap to further protect against authorship attribution attacks. Moreover, he implemented the code and conducted the experiments.

Chapter 6 Benjamin Weggenmann, Valentin Rublack, Michael Andrejczuk, Justus Mattern, and Florian Kerschbaum [468]: "DP-VAE: Human-Readable Text Anonymization for Online Reviews with Differentially Private Variational Autoencoders". In *Proceedings of the ACM Web Conference 2022* (*WWW '22*).

The author of this dissertation developed the basic idea of utilizing a VAE for text obfuscation by exploiting synergies with the Gaussian mechanism to achieve DP, as well as the theory behind the necessary DP constraints. The project was realized with the invaluable support and commitment of three students who were supervised by the author: Valentin Rublack suggested using the adversarial autoencoder (AAE) framework with disentangled representations [213] as code base and started with the implementation. Michael Andrejczuk and Justus Mattern subsequently improved the implementation. Justus Mattern started the evaluation, which was further enhanced and finalized by the author.

Chapter 7 Benjamin Weggenmann and Florian Kerschbaum [467]: "Differential Privacy for Directional Data". In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security* (*CCS '21*).

The author observed the limitation of geo-indistinguishability to planar locations [24], and devised the idea of utilizing tools from directional statistics as a basis for directional privacy. Specifically, the author investigated the VMF and Purkayastha (hyper)spherical distributions to create corresponding directional privacy mechanisms and proved their DP properties. For the Purkayastha distribution, the author developed a novel approximate inversion sampling algorithm based on a similar algorithm for the VMF distribution [249]. Lastly, the author implemented the code and performed the experiments.

For all these publications, Florian Kerschbaum contributed with his expertise in fruitful

discussions and by providing guidance for the experiments and for structuring the respective papers for submission.

Other Publications Related to this Dissertation. In the course of his dissertation, the author was also involved in co-authoring the following two publications, which are related but do not constitute major contributions to this dissertation:

- Justus Mattern, Benjamin Weggenmann, and Florian Kerschbaum [285, 286]: "The Limits of Word Level Differential Privacy". In *Findings of the Association for Computational Linguistics: NAACL 2022* (*Findings 2022*).

  The author observed theoretical limitations of word-level DP in the course of his dissertation (i.e., during the work on SynTF and DP-VAE); we discuss them briefly in Section 3.3.5.1. Justus Mattern suggested expanding on these observations by verifying them with additional experiments and proposed an alternative obfuscation method that works by fine-tuning a large language model (LLM) for paraphrasing. We achieve DP by sampling from the softmax layer with temperature, which can be interpreted as an instance of the Exponential mechanism (similar to Bo et al. [46]).

- Justus Mattern, Zhijing Jin, Benjamin Weggenmann, Bernhard Schölkopf, and Mrinmaya Sachan [284]: "Differentially Private Language Models for Secure Data Sharing". In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing* (*EMNLP 2022*).

  The main idea, fine-tuning a LLM with differentially private stochastic gradient descent (DP-SGD) as to be able to generate synthetic texts by prompting, as well as its implementation and evaluation are due to Justus Mattern. The author supported the theoretical foundation of the paper regarding DP and accounting methods for DP-SGD which was used to fine-tune the pre-trained LLM.

1.6 Structure of this Dissertation

In Chapter 1 (this chapter), we motivated the overall topic and state the research problem of why protecting privacy is important for sequential and directional data. Based on this, we formulated the research objectives and discussed limitations to important instances of sequential and directional data. Lastly, we provided an overview of our key contributions and listed related publications that were completed in the course of this dissertation.

In Chapter 2, we introduce necessary preliminaries on differential privacy. They provide an overarching theoretical foundation for the approaches proposed in this dissertation.

In Chapter 3, we discuss related work. While our focus will be on text as an example for sequential data, we also consider other types of sequential data, namely audio and visual data, and discuss various attacks on identifying information contained therein, justifying why sequential data often represents PII that needs protection.

In Chapter 4, we describe the overall methodology by which we approach the goals of this dissertation, i.e., the development of new DP mechanisms for sequential and directional data, proving their DP properties, and evaluating their performance.

In Chapter 5, we present *SynTF*, which to our best knowledge is the first method to anonymize textual data with DP guarantees. It works by randomly substituting individual words using the Exponential mechanism, thus producing a differentially private BoW representation.

In Chapter 6, we shift our focus to human-readable text obfuscation with DP guarantees. In short, we achieve this goal using a novel *differentially private variational autoencoder* (DP-VAE) architecture that we apply to entire sentences instead of individual words. We extend our method with adversarial training to disentangle the latent representations, which allows us to further improve the privacy-utility trade-off.

In Chapter 7, we address directional data, for which we propose a new notion of *directional privacy*. Based on the VMF and Purkayastha distributions, we design two novel DP mechanisms for directional data that intrinsically respect the spherical nature of directional data.

Lastly, we conclude this dissertation in Chapter 8. In particular, we summarize our main contributions as well as their impact and point out challenges we faced regarding the local model as well as potential directions for future research.

Chapter 2

A Primer on Differential Privacy

In this chapter, we provide an introduction to *differential privacy (DP)*, a formal notion of privacy which is currently considered the state of the art for quantifying and limiting information disclosure about individuals. It has first been introduced by Dwork et al. [117] in 2006 and forms a core element that is fundamental to all concepts proposed throughout this dissertation (i.e., in Chapters 5 to 7).

As a statistical concept, DP heavily relies on randomness. Therefore, we first revisit some definitions and notation from probability theory in Section 2.1 before we formally introduce DP and its privacy models in Section 2.2. For a broader introduction and further details on DP, we refer the reader to the books by Dwork and Roth [116] or Li et al. [259].

2.1 Probability Distributions

Given a measurable space $(\Omega, \mathcal{A})$ with an event set Ω and a σ-algebra $\mathcal{A}$ on Ω, a *probability distribution* (or *measure*) on $(\Omega, \mathcal{A})$ is a normed and σ-additive function $\mathsf{P} : \mathcal{A} \to [0, 1]$. We denote by $\mathscr{P}_\Omega$ the set of all *probability distributions* (or *measures*) on Ω. Unless stated otherwise, we commonly employ the Borel σ-algebra $\mathcal{A} = \sigma(\Omega)$ on Ω. Together, a measurable space $(\Omega, \mathcal{A})$ with a corresponding probability distribution $\mathsf{P} : \mathcal{A} \to [0, 1]$ forms a *probability space* $(\Omega, \mathcal{A}, \mathsf{P})$.

A measurable map $\mathbf{X} : (\Omega, \mathcal{A}, \mathsf{P}) \to (\Omega', \mathcal{A}')$ from a probability space to a measurable space $(\Omega', \mathcal{A}')$ is called a *random variable* on $(\Omega', \mathcal{A}')$. The probability of an event $S \in \mathcal{A}'$ (i.e., a measurable subset $S \subset \Omega'$) is denoted by $\Pr[\mathbf{X} \in S]$. We write $\mathbf{X} \sim \mathsf{P}_\mathbf{X}$ to indicate that the random variable $\mathbf{X}$ follows a certain distribution $\mathsf{P}_\mathbf{X}$ on $(\Omega', \mathcal{A}')$, in which case it holds that $\Pr[\mathbf{X} \in S] = \mathsf{P}_\mathbf{X}[S] = \mathsf{P}\left[\mathbf{X}^{-1}(S)\right]$ for any event $S \in \mathcal{A}'$. We denote by $\mathscr{R}_{\Omega'}$ the set of all random variables on Ω'.

When working with a random variable $\mathbf{X} : (\Omega, \mathcal{A}, \mathsf{P}) \to (\Omega', \mathcal{A}')$ with associated probability distribution $\mathsf{P}_\mathbf{X}$ on the image $(\Omega', \mathcal{A}')$ of $\mathbf{X}$, we can often avoid specifying the underlying probability space $(\Omega, \mathcal{A}, \mathsf{P})$. Moreover, if there is no ambiguity, we may omit

the name of the random variable $\mathbf{X}$ in the subscript and directly talk about the probability distribution P on the image $(\Omega', \mathcal{A}')$ of $\mathbf{X} \sim \mathsf{P}$. When talking about probability distributions or random variables on sets $\mathcal{X}$ or $\mathcal{Z}$, we implicitly assume that the Borel σ-algebra is used to form the corresponding measure space, unless stated otherwise.

A distribution P on $(\Omega, \mathcal{A})$ is typically specified by its *probability density function (PDF)*, which we denote by $\mathsf{P}[\omega]$ for $\omega \in \Omega$ by slight reuse of notation. For univariate distributions on $\Omega \subseteq \mathbb{R}$, we also denote the *cumulative distribution function (CDF)* at $Z \in \mathbb{R}$ by $\mathsf{P}[\omega \leq Z]$, shorthand for $\mathsf{P}[\{\omega \in \Omega : \omega \leq Z\}]$.

Definition 2.1. The *support of a probability distribution* P with values in $\mathcal{X}$ and probability density function (PDF) $\mathsf{P}[x]$ is

$$\operatorname{supp} \mathsf{P} := \{x \in \mathcal{X} : \mathsf{P}[x] > 0\}.$$

The *support of a random variable* $\mathbf{X}$ induced by a probability distribution $\mathsf{P}_{\mathbf{X}}$ is defined accordingly as $\operatorname{supp} \mathbf{X} = \operatorname{supp} \mathsf{P}_{\mathbf{X}}$.

We often consider families of distributions parametrized by one or more parameters, such as μ or ϵ, which we append in parentheses as in $\mathsf{P}(\mu, \epsilon)[\cdot]$, or simply $\mathsf{P}(\mu, \epsilon)$.

Definition 2.2 (Randomized mechanism)**.** Let $\mathcal{X}$ and $\mathcal{Z}$ be two sets, and let $\mathscr{R}_{\mathcal{Z}}$ be the set of random variables on $\mathcal{Z}$. A *randomized mechanism* from $\mathcal{X}$ to $\mathcal{Z}$ is a probabilistic function $\mathcal{M} : \mathcal{X} \to \mathscr{R}_{\mathcal{Z}}$ that assigns a random variable on $\mathcal{Z}$ to each input $x \in \mathcal{X}$. $\mathcal{M}$ can be specified through a parametrized family of distributions $\mathsf{M}(x)$ on $\mathcal{Z}$ so that $\mathcal{M}(x) \sim \mathsf{M}(x)$ for any $x \in \mathcal{X}$; we then say $\mathcal{M}$ is the *mechanism induced by* M and write $\mathcal{M} \sim \mathsf{M}$ in short. From an algorithmic point of view, we *run* an instance of a randomized mechanism $\mathcal{M}$ on a given input x by *sampling* a realization z of the random variable $\mathcal{M}(x)$. We write this as $z \leftsquigarrow \mathcal{M}(x)$.

Interpreting the output of a randomized mechanism $\mathcal{M}$ as (unnamed) random variable allows us to reason about its probabilities using common notation, e.g., by writing $\Pr[\mathcal{M}(x) \in S]$ to denote the probability that the mechanism $\mathcal{M}$ produces a result in the set $S \subset \mathcal{Z}$. Moreover, we can extend Definition 2.1 and define the *support of a randomized mechanism* $\mathcal{M} : \mathcal{X} \to \mathscr{R}_{\mathcal{Z}}$ as the union

$$\operatorname{supp} \mathcal{M} := \bigcup_{x \in \mathcal{X}} \operatorname{supp} \mathcal{M}(x).$$

In some situations, it may be convenient to employ an alternative notation that puts more emphasis on the underlying probability distributions as used by Chatzikokolakis et al.

[65] and Andrés et al. [24]: For a mechanism $\mathcal{M} \sim \mathsf{M}$ that is induced by a parametrized distribution $\mathsf{M}(x)$, we can thus express the probability $\Pr[\mathcal{M}(x) \in S]$ directly as $\mathsf{M}(x)[S]$.

2.2 Differential Privacy

Differential privacy (DP) is a formal notion of privacy that is currently considered the state of the art for quantifying and limiting information disclosure about individuals. It has been introduced by Dwork et al. [117] in 2006 under the name *ϵ-indistinguishability* with the goal of giving semantic privacy by quantifying the risk of an individual that results from participation in data collection. To that end, it uses randomized mechanisms as introduced in Definition 2.2 that incorporate carefully calibrated noise in order to obtain probabilistic outputs that hide the impact of individuals in the mechanism result. Notably, DP is a property of the randomized mechanism itself and not of the released data.

In the following two Sections 2.2.1 and 2.2.2, we provide the necessary preliminaries on DP together with its common privacy models as required in the dissertation. Furthermore, we also discuss an important generalization based on metrics in Section 2.2.3. For a broader introduction and details, we refer the reader to the books by Dwork and Roth [116] or Li et al. [259].

2.2.1 The Central Model

In the original, *central* or *global model* of DP, we assume the original data is collected by a trusted curator and stored in a central database, usually with one record per individual. If we consider *adjacent* databases that differ by at most one record (i.e., one individual's data), a differentially private query on both databases should yield matching results with similar probabilities, i.e., answers that are probabilistically *indistinguishable*. This is achieved via random mechanisms on the universe of datasets $\mathcal{X} = \mathcal{D}$ that return noisy query results, thus masking the impact of each individual.

Definition 2.3 (Differential privacy)**.** Let $\epsilon > 0$ be a privacy parameter, and $0 \leq \delta \leq 1$. A randomized mechanism $\mathcal{M} : \mathcal{X} \to \mathscr{R}_{\mathcal{Z}}$ fulfills (ϵ, δ)-*differential privacy* if for any pair of adjacent inputs $x, x' \in \mathcal{X}$, and all sets of possible outputs $Z \subset \operatorname{supp} \mathcal{M}$,

$$\Pr[\mathcal{M}(x) \in Z] \leq e^{\epsilon} \cdot \Pr[\mathcal{M}(x') \in Z] + \delta.$$

Specifically, we distinguish *pure ϵ-DP* with $\delta = 0$ as special case from the general case $\delta > 0$ which is also referred to as *approximate DP*.

The parameter ϵ is also called the *privacy budget* and provides a worst-case measure for the "amount of privacy" that is spent or lost when running a mechanism once and publishing the result. The second parameter δ in approximate (ϵ, δ)-DP can be understood as a *residual probability* in the sense that $\mathcal{M}$ may violate ϵ-DP "with probability up to δ" [309]. As a rule of thumb, δ should hence be negligible in the size of the input dataset; note, however, that this creates an implicit dependency on the anticipated size of the inputs that will be fed into the mechanism, which may pose an unknown risk that is not assessable in advance (cf. the discussion by McSherry [293]).

Such violations in (ϵ, δ)-DP may come in various manifestations: In the worst case, sometimes referred to as "catastrophic failure", the secret, i.e., whether a record is included in the dataset or not, is revealed completely. However, it may also be possible that the privacy guarantee degrades "gracefully" so that weaker ϵ_i-DP is fulfilled with probability $1 - \delta_i$ for gradually increasing privacy budgets $\epsilon_i > \epsilon_{i-1}$ and decreasing residual probabilities $\delta_i < \delta_{i-1}$ (that sum up to 1) for $i = 1, 2, \ldots$, where $\epsilon_0 = \epsilon$ and $\delta_0 = \delta$.

Definition 2.4 (Privacy loss)**.** The *privacy loss* of a pure ϵ-DP mechanism $\mathcal{M}$ is the quantity

$$\ell(\mathcal{M}) := \sup_{x \sim x'} \sup_{Z \subset \operatorname{supp} \mathcal{M}} \ln \frac{\Pr[\mathcal{M}(x) \in Z]}{\Pr[\mathcal{M}(x') \in Z]}$$

where we interpret $0/0 = 0$.

Note that by definition, the privacy budget ϵ is an upper bound for the privacy loss $\ell(\mathcal{M})$ of an ϵ-differentially private mechanism $\mathcal{M}$; therefore, any random mechanism $\mathcal{M}$ with finite privacy loss $\ell(\mathcal{M})$ also fulfills $\ell(\mathcal{M})$-DP, i.e., we can prove that a randomized mechanism fulfills pure DP by bounding its privacy loss.

Examples of Use. Some prime examples for central DP include counting and histogram queries, where calibrated Laplace noise is added to the counts or histogram bins [116]. We discuss the corresponding *Laplace mechanism* in Section 2.3.1. Another prominent application is the training of neural networks which typically requires training data sourced from many users. To this end, an approach commonly referred to as DP-SGD has been proposed [5, 37, 417] which perturbs the gradient updates to protect the training data of the resulting machine learning (ML) models. More examples of DP mechanisms in the context of ML can be found in the surveys by Ha et al. [173] and Ouadrhiri and Abdelhadi [335]. For a recent overview of various real-world deployments of DP mechanisms in both the central and the local model, we refer to Desfontaines [100].

2.2.2 The Local Model

The central model's need for a trusted curator who has access to all the collected, original data can constitute a severe limitation in some scenarios, e.g., if the curator in fact cannot be or simply is not trusted by the users. To solve this, we can use DP in the *local model*, which has first been introduced by Evfimievski et al. [126] under the name "amplification", and then more formally in the context of DP by Kasiviswanathan et al. [227] and Duchi et al. [114]: In the local model, the data is obfuscated locally at the data source, before it is collected for further processing or storage in a central database. In this way, the local model does not require a trusted third party and hence provides a stronger privacy model than central DP.

While the fundamental Definition 2.3 of DP remains still valid in the local model, the change from central DP is formally expressed in the definition of *adjacency*: The local model makes the strong assumption that *any* two inputs are adjacent, which often makes it difficult to achieve a satisfying privacy-utility trade-off. This often results in the need for larger collections of data [227, 472] or larger privacy budgets ϵ than in the central model as countermeasures to achieve satisfying utility [111].

Examples of Use. An early example of a DP mechanism that works in the local model is *randomized response* proposed by Warner [464] in 1965 to conduct privacy-preserving surveys, where each survey participant either provides a truthful or a random answer depending on the flip of an (unbiased) coin. Meanwhile, local DP has been prominently deployed in the industry by several large corporations, including, for instance,

- Google's RAPPOR to collect statistics in their Chrome browser [123, 134],
- Apple, who privately learn unknown words to improve word suggestions when typing [108],
- Microsoft, who privately collect telemetry data in Windows [109],
- and SAP, who support perturbation of numerical data in their database solution [23].

For more on the local model, we recommend the tutorial by Bebensee [38] or the survey by Yang et al. [485].

2.2.3 Generalization with Metrics

A limitation with central and local DP is that the indistinguishability is achieved between two adjacent inputs regardless of their actual values. This can be particularly problematic

in the local model, where each user might just submit one single record, in which case a DP mechanism with small privacy parameter ϵ would enforce all submitted records to be indistinguishable, thus rendering the collected data essentially useless. To the same end, Chatzikokolakis et al. [65] argue that in some scenarios, the level of (in)distinguishability between two inputs or databases as enforced by a privacy mechanism should depend on the values themselves instead of the number of differing records as in the central model. They hence propose a generalized notion of *privacy on metric spaces*, which extends to input domains beyond databases, where a conforming mechanism run on *nearby* inputs x, x' still has *similar* output probabilities:

Definition 2.5 (Metric privacy)**.** Let $\epsilon > 0$ be a privacy parameter. On a metric space $(\mathcal{X}, d)$, a mechanism $\mathcal{M}$ satisfies ϵd*-privacy* if for all $x, x' \in \mathcal{X}$ and all $Z \subset \operatorname{supp} \mathcal{M}$,

$$\mathcal{M}(x)[Z] \leq \exp(\epsilon \cdot d(x, x')) \cdot \mathcal{M}(x')[Z].$$

In other words, the level of indistinguishability of any two points x, x' is bounded by ϵ times their distance. Andrés et al. [24] provide another interpretation: If we consider an arbitrary but fixed distance $r > 0$, any two points with $d(x, x') \leq r$ achieve a level of indistinguishability at most ϵr; hence, an ϵd-private mechanism $\mathcal{M}$ achieves a *privacy level* $\ell = \epsilon r$ within a *protection radius* r.

2.2.3.1 Adjacency and Connection with Central and Local Differential Privacy

Note that we recover the original notion of central ϵ-DP (cf. Definition 2.3) on the space of databases $\mathcal{X} = \mathcal{D}$ if we use the *record-level edit distance* $d_{\pm 1}$, since datasets $x, x' \in \mathcal{D}$ are adjacent (i.e., they differ by at most one record) if and only if $d_{\pm 1}(x, x') \leq 1$. Similarly, we recover local ϵ-DP as extreme case of metric privacy with

$$d(x, x') := \begin{cases} 0 & \text{if } x = x', \\ 1 & \text{if } x \neq x'. \end{cases}$$

This motivates the following formal and broader definition of *adjacency*:

Definition 2.6 (Adjacency)**.** In a metric space $(\mathcal{X}, d)$, we say that two inputs $x, x' \in \mathcal{X}$ are *adjacent* (with respect to the metric d) if $d(x, x') \leq 1$. We write this as $x \sim_d x'$ (or $x \sim x'$ if d is understood from the context).

2.2.4 Rényi Differential Privacy

Mironov [309] has proposed Rényi differential privacy (RDP) as a generalization of DP

that allows more accurate tracking of the privacy loss. It is based on the Rényi divergence, which is defined as follows:

Definition 2.7 (Rényi divergence)**.** For probability distributions P and Q both defined over some set $\mathcal{Z}$, the *Rényi divergence of order* $\alpha > 1$ is

$$D_\alpha(\mathsf{P} \parallel \mathsf{Q}) := \frac{1}{\alpha - 1} \log \mathbf{E}_{\mathbf{z} \sim \mathsf{Q}} \left(\frac{\mathsf{P}[\mathbf{z}]}{\mathsf{Q}[\mathbf{z}]} \right)^\alpha .$$

By continuity, we can extend the Rényi divergence to $\alpha \in \{1, \infty\}$: For $\alpha = 1$, we have

$$D_1(\mathsf{P} \parallel \mathsf{Q}) := \lim_{\alpha \to 1} D_\alpha(\mathsf{P} \parallel \mathsf{Q}) = D_{\mathrm{KL}}(\mathsf{P} \parallel \mathsf{Q}),$$

which corresponds to the Kullback–Leibler (KL) divergence. Similarly, for $\alpha \to \infty$, we obtain

$$D_\infty(\mathsf{P} \parallel \mathsf{Q}) := \lim_{\alpha \to \infty} D_\alpha(\mathsf{P} \parallel \mathsf{Q}) = \sup_{\mathbf{z} \in \operatorname{supp} \mathsf{Q}} \log \frac{\mathsf{P}[\mathbf{z}]}{\mathsf{Q}[\mathbf{z}]} = \sup_{Z \subset \operatorname{supp} \mathsf{Q}} \log \frac{\mathsf{P}[Z]}{\mathsf{Q}[Z]} \tag{2.1}$$

The Rényi divergence can be extended to randomized mechanisms by means of their inducing distributions:

Definition 2.8 (Rényi divergence of a randomized mechanism)**.** Let $\mathcal{M} : \mathcal{D} \to \mathscr{R}_\mathcal{Z}$ be a randomized mechanism that is induced by a parametrized distribution M, i.e., $\mathcal{M}(D) \sim \mathsf{M}(D)$ for all $D \in \mathcal{D}$. We define the Rényi divergence of $\mathcal{M}$ between inputs $D, D' \in \mathcal{D}$ as

$$D_\alpha(\mathcal{M}(D) \parallel \mathcal{M}(D')) := D_\alpha(\mathsf{M}(D) \parallel \mathsf{M}(D')).$$

Definition 2.9 ((α, ϵ)-Rényi differential privacy)**.** A randomized mechanism $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ provides *ϵ-RDP of order α*, or *(α, ϵ)-RDP* for short, if for any adjacent D, D' it holds that $D_\alpha(\mathcal{M}(D) \parallel \mathcal{M}(D')) \leq \epsilon$.

2.2.4.1 Relation to Pure and Approximate Differential Privacy

The original variants of DP can be reobtained from Rényi differential privacy (RDP) as special cases. The first case relates RDP of order ∞ to *pure* ϵ-DP, which according to Mironov [309, page 3] are equivalent:

Proposition 2.10 (Equivalence of (∞, ϵ)-RDP and pure ϵ-DP)**.** *A randomized mechanism $\mathcal{M}$ fulfills ϵ-DP if and only if $D_\infty(\mathcal{M}(D) \parallel \mathcal{M}(D')) \leq \epsilon$.*

Proof of the "if" part. First, note that for $\alpha \to \infty$, the Rényi divergence of a randomized mechanism $\mathcal{M}$ between two adjacent inputs D, D' can be rewritten in terms of Eq. (2.1) as

$$D_\infty(\mathcal{M}(D) \parallel \mathcal{M}(D')) = \sup_{\mathbf{z} \in \operatorname{supp} \mathcal{M}} \log \frac{\mathsf{M}(D)[\mathbf{z}]}{\mathsf{M}(D')[\mathbf{z}]}.$$

If the supremum exists with value ϵ, we can rearrange the right-hand side to $\mathsf{M}(D)[\mathbf{z}] \leq e^\epsilon \mathsf{M}(D')[\mathbf{z}]$ for any $\mathbf{z} \in \mathcal{Z}$. By integrating $\mathbf{z}$ over a given set $Z \subset \mathcal{Z}$, this is just another way to express ϵ-DP (Definition 2.3 with $\delta = 0$), so a randomized mechanism $\mathcal{M}$ is ϵ-DP if $D_\infty(\mathcal{M}(D) \parallel \mathcal{M}(D')) \leq \epsilon$. □

The second and more general case allows us to translate RDP of finite order $\alpha < \infty$ to *approximate* (ϵ', δ)-DP using the following result [309, Proposition 3]:

Proposition 2.11 (From (α, ϵ)-RDP to approximate DP)**.** *If $\mathcal{M}$ is an (α, ϵ)-RDP mechanism, it also satisfies (ϵ', δ)-DP for any $0 < \delta < 1$, where*

$$\epsilon' = \epsilon + \frac{\log 1/\delta}{\alpha - 1}.$$

2.3 Some Fundamental Differential Privacy Mechanisms

After having introduced the formal concepts of DP, we finally introduce some fundamental mechanisms that actually realize DP in its various variants.

Additive Noise Mechanisms. A major class of DP mechanisms are *additive noise mechanisms* which add random noise to the result $f(\boldsymbol{x})$ of a (typically numerical, i.e., $\mathcal{Z} = \mathbb{R}^n$) query $f : \mathcal{X} \to \mathbb{R}^n$ in the central model. Note that we can straightforwardly employ those mechanisms as local DP mechanisms by choosing the query to be the identity function, $f \equiv \mathrm{id}$.

Tightly coupled with additive noise mechanisms is the *sensitivity* of the underlying query: For an additive noise mechanism that answers a numerical query f to be differentially private, the idea is that the introduced noise should cover any difference of f between any pair of adjacent inputs. This intuition is covered in the following definition:

Definition 2.12 (Sensitivity of query functions)**.** Let $f : \mathcal{X} \to \mathbb{R}^n$ be a query function. The *(global) sensitivity* of f is the largest possible distance of f on two adjacent inputs,

$$\Delta f := \max_{\boldsymbol{x} \sim \boldsymbol{x}'} \|f(\boldsymbol{x}) - f(\boldsymbol{x}')\|.$$

The exact norm used normally depends on the actual mechanism: More precisely, in case of

- $\|\cdot\|_1$, we write $\Delta_1 f$ for the L^1 or *Manhattan sensitivity*,
- $\|\cdot\|_2$, we write $\Delta_2 f$ for the L^2 or *Euclidean sensitivity*,

etc. If the query function f is understood from the context, we may shorten the notation to Δ_1, Δ_2, etc., and in case the norm is understood as well even shorter to Δ. Note that we assume that the maximum is well-defined and finite for the query functions f that we consider.

2.3.1 The Laplace Mechanism

The Laplace mechanism is the first pure ϵ-differentially private mechanism that was proposed by Dwork et al. [117]. As hinted at by its name, it is based on the Laplace distribution:

Definition 2.13 (Laplace distribution)**.** The *Laplace distribution* $\mathsf{Lap}(\mu, b)$ with mean $\mu \in \mathbb{R}$ and scale $b > 0$ is given by its PDF

$$\mathsf{Lap}(\mu, b)[x] = \frac{1}{2b} \exp\left(-\frac{|x - \mu|}{b}\right).$$

To declare a Laplace random variable L, we write $L \sim \mathsf{Lap}(\mu, b)$. If $\mu = 0$, we may omit μ and simply write $\mathsf{Lap}(b)$.

As an additive noise mechanism, the Laplace mechanism applies noise from a Laplace distribution to the output of a given query function:

Example 2.14 (Laplace mechanism)**.** Let $f : \mathcal{X} \to \mathbb{R}^n$ be a query function with L^1 sensitivity $\Delta_1 = \Delta_1 f$, and let $\epsilon > 0$ be a privacy parameter. For an input $\boldsymbol{x} \in \mathcal{X}$, we define the *Laplace mechanism of f at $\boldsymbol{x}$* as

$$\mathcal{L}_{\epsilon,f}(\boldsymbol{x}) := f(\boldsymbol{x}) + (L_1, \ldots, L_n),$$

where $L_i \sim \mathsf{Lap}(0, \Delta_1/\epsilon)$ are independent and identically distributed (i.i.d.) Laplace random variables centered at $\mu = 0$ with scale parameter $b = \Delta_1/\epsilon$ for all $i = 1, \ldots, n$. With this parameterization, the Laplace mechanism $\mathcal{L}_{\epsilon,f}$ fulfills ϵ-DP, as shown by Dwork et al. [117, Proposition 1].

Note that despite being an additive noise mechanism, we could entirely skip the additions and describe the Laplace mechanism directly in terms of random variables

$$\mathcal{L}_{\epsilon,f}(\boldsymbol{x}) = (L'_1, \ldots, L'_n),$$

where $L'_i \sim \mathsf{Lap}(\mu_i, \Delta_1 f/\epsilon)$ are i.i.d. Laplace random variables whose location parameters equal the coordinates of the query result $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_n) := f(\boldsymbol{x})$.

The tuple $(L'_1, \ldots, L'_n)$ can be interpreted as multivariate generalization of the Laplace distribution: Since it consists of n i.i.d. Laplace variables, its joint PDF amounts to

$$\begin{aligned}\mathsf{Lap}\left(\boldsymbol{\mu}, \frac{\Delta_1 f}{\epsilon}\right)[\boldsymbol{x}] &= \prod_{i=1}^{n} \frac{\epsilon}{2\Delta_1 f} \exp\left(-\frac{\epsilon|x_i - \mu_i|}{\Delta_1 f}\right) \\ &= \left(\frac{\epsilon}{2\Delta_1 f}\right)^n \exp\left(-\epsilon \frac{\|\boldsymbol{x} - \boldsymbol{\mu}\|_1}{\Delta_1 f}\right).\end{aligned} \tag{2.2}$$

As we can see, the exponent fits Definition 2.3 of ϵ-DP quite nicely, which is used in the proof to show it fulfills ϵ-DP; furthermore, the L^1 norm $\|\boldsymbol{x} - \boldsymbol{\mu}\|_1$ motivates the use of the L^1 sensitivity for the Laplace mechanism.

2.3.2 The Gaussian Mechanism

Another important additive noise mechanism is the Gaussian mechanism [116] which is based on the normal (Gaussian) distribution:

Definition 2.15 (Univariate normal distribution)**.** The univariate *normal* (or *Gaussian*) *distribution* $\mathcal{N}(\mu, \sigma^2)$ with mean location μ and variance σ^2 (or standard deviation σ) is given by its PDF

$$\mathcal{N}(\mu, \sigma^2)[x] = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\left(\frac{x - \mu}{\sigma}\right)^2\right).$$

As an additive noise mechanism, the Gaussian mechanism works analogously to the Laplace mechanism and applies Gaussian noise to each coordinate of a query result:

Definition 2.16 ((Isotropic) Gaussian mechanism)**.** Let $f : \mathcal{X} \to \mathbb{R}^n$ be a query function with finite L^2 sensitivity $\Delta_2 f$. The *(isotropic) Gaussian mechanism* $\mathcal{G}_{\sigma,f}$ with standard deviation $\sigma > 0$ is a randomized function

$$\mathcal{G}_{\sigma,f}(\boldsymbol{x}) := f(\boldsymbol{x}) + (N_1, \ldots, N_n)$$

where $N_i \sim \mathcal{N}(0, \sigma^2)$.

As with the Laplace mechanism, we can skip the additions and describe the Gaussian mechanism directly in terms of Gaussians $N_i \sim \mathcal{N}(\mu_i, \sigma^2)$ whose mean values $\mu_i := f(\boldsymbol{x})_i$ are defined by the query result, so

$$\mathcal{G}_{\sigma,f}(\boldsymbol{x}) = (N'_1, \ldots, N'_n).$$

The tuple $(N'_1, \ldots, N'_n)$ is the Cartesian product of n i.i.d. univariate Gaussians that share the same variance σ^2, which is also called an *isotropic multivariate Gaussian*:

Definition 2.17 (Isotropic normal distribution)**.** The n-dimensional *isotropic* (or *spherical*) *normal distribution* with mean location $\boldsymbol{\mu} \in \mathbb{R}^n$ and covariance matrix $\sigma^2\mathbf{I}$ is defined by the PDF

$$\mathcal{N}(\boldsymbol{\mu}, \sigma^2\mathbf{I})[\boldsymbol{x}] = \left(\sqrt{2\pi}\sigma\right)^{-n} \exp\left(-\frac{1}{2}\frac{\|\boldsymbol{x} - \boldsymbol{\mu}\|_2^2}{\sigma^2}\right).$$

Therefore, the Gaussian mechanism can be described more succinctly by an isotropic Gaussian whose mean is defined by the query result $\boldsymbol{\mu} := f(\boldsymbol{x})$:

$$\mathcal{G}_{\sigma,f}(\boldsymbol{x}) = (N'_1, \ldots, N'_n) \sim \mathcal{N}(f(\boldsymbol{x}), \sigma^2\mathbf{I})$$

We still need to explain the choice of σ, the standard deviation parameter of the Gaussian mechanism, which is determined subject to achieving a certain level of privacy. While the Laplace mechanism provides pure ϵ-DP, the Gaussian mechanism provides approximate (ϵ, δ)-DP:

Theorem 2.18 (Classical Gaussian mechanism [116, Theorem A.1])**.** *Let $f : \mathcal{X} \to \mathbb{R}^n$ be a query function whose L^2 sensitivity is $\Delta_2 = \Delta_2 f$, and let $\epsilon \in (0, 1)$. For $c^2 > 2\ln(1.25/\delta)$, the Gaussian mechanism $\mathcal{G}_{\sigma,f}$ with $\sigma \geq c\Delta_2/\epsilon$ is (ϵ, δ)-differentially private.*

Note that this "classical" theorem for the isotropic Gaussian mechanism requires $\epsilon < 1$ to determine values of the standard deviation σ that guarantee that (ϵ, δ)-DP is fulfilled. However, it may be possible to find (i) even tighter and (ii) more general bounds that extend to $\epsilon \geq 1$ because the estimation for σ in Theorem 2.18 is not optimal: In fact, it *is* possible to achieve (ϵ, δ)-DP for arbitrary $\epsilon > 0$ using the *Analytical Gaussian mechanism* by Balle and Wang [32] which employs a more optimal method to find suitable values for σ, even if $\epsilon \geq 1$. Moreover, it is possible to analyze the Gaussian mechanism in terms of Rényi DP [309] which also does not have this limitation, and then convert (α, ϵ)-RDP back to (ϵ, δ)-DP via Proposition 2.11 ([309, Proposition 3]):

2.3.2.1 Rényi Differential Privacy of the Gaussian Mechanism

In the following, we consider the general form of the Gaussian mechanism based on a multivariate Gaussian distribution with a positive definite, but otherwise unconstrained, covariance matrix:

Definition 2.19 (Multivariate normal distribution)**.** The multivariate *normal* (or *Gaussian*) *distribution* $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with mean vector $\boldsymbol{\mu} \in \mathbb{R}^n$ and positive definite covariance matrix

$\Sigma \in \mathbb{R}^{n \times x}$ is given by its PDF

$$\mathcal{N}(\mu, \Sigma)[x] = \det(2\pi\Sigma)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(x-\mu)^\intercal \Sigma^{-1}(x-\mu)\right).$$

The multivariate Gaussian mechanism uses a slightly more complex sensitivity which is based on the Mahalanobis norm:

Definition 2.20 (Mahalanobis norm)**.** Let $\Sigma \in \mathbb{R}^{n \times x}$ be a positive definite covariance matrix. The Mahalanobis norm corresponding to Σ is

$$\|z\|_\Sigma := \sqrt{z^\intercal \Sigma^{-1} z}.$$

Accordingly, the *multivariate Gaussian mechanism* is defined as follows:

Theorem 2.21 (RDP of multivariate Gaussian mechanism)**.** *Given a positive definite covariance matrix $\Sigma \in \mathbb{R}^{n \times n}$ and a query function $f : \mathcal{X} \to \mathbb{R}^n$ with* Mahalanobis sensitivity

$$\Delta_\Sigma = \Delta_\Sigma f := \max_{x \sim x'} \|f(x) - f(x')\|_\Sigma,$$

the multivariate Gaussian mechanism *is given by*

$$\mathcal{G}_{\Sigma,f}(x) \sim \mathcal{N}(f(x), \Sigma)$$

and satisfies

$$\left(\alpha, \frac{\alpha}{2}\Delta_\Sigma^2\right)\text{-RDP}.$$

Proof. According to Gil et al. [159, Table 2], the Rényi divergence of two Gaussians with means $\mu = f(x)$, $\mu' = f(x')$ and same covariance simplifies to

$$D_\alpha\left(\mathcal{N}(\mu, \Sigma) \,\middle\|\, \mathcal{N}(\mu', \Sigma)\right) = \frac{\alpha}{2}\left\|\mu - \mu'\right\|_\Sigma^2,$$

where $\|\cdot\|_\Sigma$ is the Mahalanobis norm corresponding to Σ, cf. Definition 2.20. The result follows, since $\left\|\mu - \mu'\right\|_\Sigma \leq \Delta_\Sigma$ by definition of the Mahalanobis sensitivity. □

In practice, we often only use the *isotropic* Gaussian mechanism whose covariance matrices $\Sigma = \sigma^2\mathbf{I}$ are isotropic. In this case, the Mahalanobis norm reduces to a scaled L^2 norm

$$\|\cdot\|_{\sigma^2\mathbf{I}} = \frac{1}{\sigma}\|\cdot\|_2,$$

and we obtain

Corollary 2.22 (RDP of the isotropic Gaussian mechanism)**.** *Given a query function $f : \mathcal{X} \to \mathbb{R}^n$ with L^2 sensitivity Δ_2, the isotropic Gaussian mechanism $\mathcal{G}_{\sigma,f} \equiv \mathcal{G}_{\sigma^2\mathbf{I},f}$ satisfies*

$$\left(\alpha, \frac{\alpha\Delta_2^2}{2\sigma^2}\right)\text{-RDP}.$$

Notably, this coincides with the RDP curve of the *univariate* Gaussian mechanism that is commonly treated in the literature [309, Section VI.C].

2.3.3 The Planar Laplace Mechanism

An example of a mechanism for *metric privacy* (Definition 2.5) is the Planar Laplace (PL) mechanism. It was first introduced by Chatzikokolakis et al. [65] to achieve d_2-privacy on the two-dimensional Euclidean plane $\mathcal{X} = \mathbb{R}^2$, and subsequently used by Andrés et al. [24] to achieve geo-indistinguishability, a specialization of d_2-privacy for location data. We use the following generalization to an arbitrary number of dimensions by Koufogiannis et al. [240], which was employed, e.g., to obfuscate high-dimensional gene expression vectors [29].

Definition 2.23 (Planar Laplace mechanism)**.** The n-dimensional *PL mechanism* is defined by the density

$$\mathsf{PL}(\boldsymbol{\mu}, \epsilon)[\mathbf{x}] = \frac{\epsilon^n \Gamma(\frac{n}{2}+1)}{\pi^{\frac{n}{2}}\Gamma(n+1)} \exp(-\epsilon\|\mathbf{x}-\boldsymbol{\mu}\|).$$

While the PL mechanism is a generalization of the univariate Laplace mechanism (Example 2.14 with $n = 1$), it is worth mentioning that it is different from the multivariate distribution obtained by drawing $n > 1$ independent Laplace samples as in Eq. (2.2).

Planar Laplace Sampling Procedure. Since the PL distribution is a location-scale distribution, we can draw a noise vector $\mathbf{v} \hookleftarrow \mathsf{PL}(\mathbf{0}, \epsilon)$ from the centered distribution and then translate $\boldsymbol{x}$ by $\mathbf{v}$ instead of sampling from $\mathsf{PL}(\boldsymbol{x}, \epsilon)$ directly. Note that this implies that the PL mechanism, too, can be regarded as an additive noise mechanism. Moreover, if we factor $\mathbf{v} = r \cdot \mathbf{u}$ with $r > 0$ and a unit vector $\mathbf{u}$, it can be shown that the noise magnitude r follows a gamma distribution $\mathsf{Gamma}(n, 1/\epsilon)$ with shape n and scale $1/\epsilon$, whereas $\mathbf{u}$ follows the uniform distribution $\mathsf{Uni}(\mathbb{S}^{n-1})$ on the unit sphere $\mathbb{S}^{n-1}$. Algorithm 1 illustrates the resulting sampling procedure.

Algorithm 1: Planar Laplace sampling procedure.

Input: $\boldsymbol{x} \in \mathbb{R}^n$, privacy parameter $\epsilon > 0$
Output: $\boldsymbol{z} \in \mathbb{R}^n$ with $\boldsymbol{z} \sim \mathsf{PL}(\boldsymbol{x}, \epsilon)$

1 $r \leftsquigarrow \mathsf{Gamma}(n, 1/\epsilon)$; // noise magnitude
2 $\boldsymbol{u} \leftsquigarrow \mathsf{Uni}(\mathbb{S}^{n-1})$; // noise direction
3 $\boldsymbol{z} \leftarrow \boldsymbol{x} + r\boldsymbol{u}$; // translate input vector

2.3.4 The Exponential Mechanism

An important and versatile example of a DP mechanism that does *not* belong to the additive noise family is the Exponential mechanism by McSherry and Talwar [294]. It applies to both numerical and categorical data and requires a "measure of suitability" for each possible pair of inputs and outputs:

Definition 2.24 (Rating/quality function)**.** A *rating* or *quality function* from $\mathcal{X}$ to $\mathcal{Z}$ is a function $\rho : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$, where the value $\rho(x, z)$ is the *rating* or *quality* for an output z given input x.

In case of the Exponential mechanism, the *sensitivity* is defined as the largest possible difference of the rating function ρ given two adjacent inputs, over all possible output values:

$$\Delta\rho := \max_{z \in \mathcal{Z}} \max_{x_1 \sim x_2} \big(\rho(x_1, z) - \rho(x_2, z)\big)$$

For a given input, the Exponential mechanism randomly yields an output value with probability proportional to the exponentiated rating function (times the privacy parameter ϵ):

Definition 2.25 (Exponential mechanism)**.** Let $\epsilon > 0$ be a privacy parameter, and let $\rho : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ be a rating function with sensitivity $\Delta = \Delta\rho$. Then the Exponential mechanism is defined as family of random variables $\mathcal{E}_{\epsilon,\rho}(x)$ for each $x \in \mathcal{X}$ whose PDF is given by

$$\Pr\big[\mathcal{E}_{\epsilon,\rho}(x) = z\big] = \frac{\exp\left(\frac{\epsilon}{2\Delta}\rho(x, z)\right)}{\int_{z'} \exp\left(\frac{\epsilon}{2\Delta}\rho(x, z')\right) \mathrm{d}z'}.$$

Note that a discrete version of the Exponential mechanism for countable $\mathcal{Z}$ is obtained by replacing the integral with a sum:

$$\Pr\big[\mathcal{E}_{\epsilon,\rho}(x) = z\big] = \frac{\exp\left(\frac{\epsilon}{2\Delta}\rho(x, z)\right)}{\sum_{z'} \exp\left(\frac{\epsilon}{2\Delta}\rho(x, z')\right)}.$$

The Exponential mechanism with privacy parameter ϵ fulfills ϵ-DP as shown by McSherry and Talwar [294, Theorem 6],

Chapter 3

Related Work

In this chapter, we discuss related work pertaining to protecting privacy-sensitive information in sequential and directional data. Our main focus is on existing approaches that provide differential privacy, however, we also consider "classical" solutions without such formal privacy guarantees. Besides text as an important instance of sequential data, we also include audio (speech) and visual data as further sequential domains. Moreover, we discuss relevant identification attacks that explain why sequential data often represents PII that needs protection and that motivated the development of both existing and our own defense techniques in the first place.

3.1 Concepts

In this section, we present some concepts that are useful to categorize attacks and defenses on sequential data, as well as sensitive, identifying information contained therein.

Types of Sequential Data. Sequential data occurs in various domains or formats. Our main focus will be on textual data since it provided the use cases for the defensive approaches presented in this dissertation. However, for illustrative purposes, we also consider audio and visual data as related types of sequential data where our approaches may also be applicable.

Textual Data. Text is an abstract representation of *written language*. Various writing systems have been developed to record and exchange *human* or *natural language*, which is used as means of communication between humans to express and convey thoughts and ideas. Moreover and more recently, *computer languages* have been formally defined as means to program computers and thus control their operation, where the corresponding source code is typically written as human-readable text.

Note that in this section, we consider raw *plain text* as stored on a computer (e.g., encoded as a sequence of ASCII or UTF-8 characters) for both human and computer languages. On a higher level, plain texts may be interpreted as a sequence of syllables or words instead of characters. Apart from plain text, natural languages can be expressed in various other modalities including auditory sounds, i.e., speech or spoken language, visual symbols and gestures such as handwriting and sign language, as well as tactile writing systems such as Braille.

Audio Data. We consider audio recordings as a temporal sequence of samples. In particular, audio recordings may represent *spoken language* (or *speech*) which naturally is a means of human communication.

Visual Data. Digital photos and images can be regarded as a spatial sequence of pixels in two dimensions, and videos in turn as a temporal sequence of individual images (called *frames* in this context). Images potentially contain *written language*, e.g., handwriting, printed text, or in the form of license plates (stamped and printed). They can also show persons, their faces, fingerprints, and many other potentially privacy-sensitive depictions.

Types of Identifiers. Privacy-sensitive information can come in many flavors. To limit the scope of this section, we primarily focus on the identity of an individual as the type of sensitive information that is most specific and hence most worthy of protection, since other sensitive attributes, such as gender or ethnicity, are linked to their identity.

Named Identifiers. Named identifiers are PII terms such as a person's name, their addresses, phone or credit card number, etc. from which a person could be identified directly or indirectly. Vehicle identifiers such as a car's license plate numbers may indirectly identify its owner. Other examples are given by the 18 HIPAA identifiers [327]. Named identifiers are typically stated explicitly as part in a piece of text or speech, or depicted visually in an image or video.

Biometric Identifiers. Biometric identifiers are derived from features that are intrinsically determined by human characteristics of an individual. We commonly distinguish *physiological* characteristics which are related to the composition of an individual's body, such as fingerprints, facial images, or the appearance of the iris, and *behavioral* characteristics which are related to the behavior of a person, including their movement (gait, signature, etc.), sound of their voice, among others. For a recent overview of biometric recognition techniques, we refer to the survey by Minaee et al. [307]. Some biometric

identifiers, including finger- and voiceprints, are also specifically included in the list of 18 HIPAA identifiers [327].

"Technometric" Identifiers. In a way analogous to biometric identifiers, we may think of particular technical characteristics of a device that uniquely identify that device, e.g., quirks in digital camera sensors, as "technometric" identifiers. Indirectly, such identifiers may also lead to the identification of the owner of the device, e.g., through device serial numbers recorded by the device vendor or manufacturer. In fact, device identifiers and serial numbers are also listed as one of the 18 HIPAA identifiers [327]. To stay within the scope of this dissertation, we do not explore specific attacks or defenses for this kind of identification in this chapter.

Locality of Identifiers. Sensitive information that represents identifiers or allows inferring sensitive attributes may be contained in sequential data in different levels of pervasiveness:

- At one end of the spectrum, the sensitive information is *confined* locally to one or few segments (e.g., terms, utterances, or pixels for text, speech, or image, respectively) of the sequence. For instance, named identifiers like names or addresses are typically represented by one or few terms in a text, which would correspond to a few utterances in spoken language.

- At the other end of the spectrum, the sensitive information is *pervasive*, i.e., distributed across many or even all segments of the sequence. For instance, voice characteristics of the speaker in a speech recording cover virtually all parts of the recording, apart maybe from short speech pauses.

The pervasiveness of sensitive information is an important consideration for the defensive measures aimed at protecting that information: On the one hand, if the sensitive information is confined locally and has little overlap with utility-critical information, then simple approaches that mask or redact the few relevant segments in the sequence may be suitable to protect the sensitive information, provided that the remaining parts convey enough information to maintain utility. On the other hand, if the sensitive information pervades large parts of the sequence that possibly also contain utility-critical information, then the defensive methods must modify all affected parts to hide the sensitive information, all while preserving enough of the utility-critical information.

3.2 Attacks

In this section, we will review several types of attacks aiming at detecting, extracting, and inferring sensitive information from sequential types of data, namely, text, audio, and visual data.

3.2.1 Attacks on Textual Data

Text can contain several forms of sensitive information that may be of interest to attackers: On the one hand, sensitive information can be explicitly stated in a text in the form of named identifiers, such as names, addresses, or phone numbers. On the other hand, the style of the text itself may reveal sensitive information about its author or even their identity. This section thus reviews attacks aiming at both types of sensitive information.

3.2.1.1 Named Identifiers

Identifiers and Secrets in Human Language. Due to its abstracted nature, written language is a prominent instance of sequential data where sensitive information is typically represented in a very explicit and "pure" form that can be easily detected and extracted. For instance, full names, physical and email addresses, phone and credit card numbers of individuals all are examples of explicit sensitive or personally identifiable information (PII), where the need to protect such identifiers is also reflected in legal standards (cf. Section 1.1.2.1, the 18 HIPAA identifiers).

While this information is plainly readable in plain text, automated methods to find and extract sensitive information have been investigated based on information theory [393] or more recently based on ML techniques [171, 270, 477]. Typically, the task of detecting privacy-sensitive terms in a text can be regarded as a form of *named entity recognition (NER)* (cf., e.g., [257, 312]), whose techniques often form the foundation of adapted methods that additionally or specifically recognize sensitive data [337]. A simpler task that is sometimes studied is detecting whether entire sentences or documents *contain* privacy-sensitive information [299, 322]. Such detection methods are often used in specialized de-identification (or "scrubbing") tools (cf. Section 3.3.4.1) that aim at removing sensitive identifiers that are protected under specific legal requirements, such as the 18 HIPAA identifiers in the medical and health sector.

Identifiers and Secrets in Computer Language. Apart from human language, text is also used as means to represent computer language, for instance, in the form of source code, configuration files, and even compiled binaries which may contain secrets such as

API keys or user credentials [300]. This can be especially problematic if the source code is publicly accessible, for instance, when hosted on open source repositories like GitHub where a simple search with the correct pattern or regular expression would yield results with cryptographic secrets, e.g., SSH keys or credentials for Amazon Web Services (AWS) [162, 239, 245, 365].

As with human language, work has been conducted to automatically detect such secrets with ML and related techniques [105, 267, 387] in addition to rule- and pattern-based approaches. While such detection techniques can be exploited by a malicious attacker to find and then extract secrets [106, 107], they also have legitimate uses, for instance, in data leakage prevention (DLP) solutions as discussed in Section 3.3.1.

3.2.1.2 Linguistic Writing Style

Authorship Attribution for Written Language. *Authorship attribution* is a set of methods concerned with attributing authorship of anonymous or disputed documents to their respective authors. Such methods usually make use of stylistic features to identify or discriminate authors, which is why those methods are also referred to as *stylometry*.

A classic example is given by the *Federalist Papers* [179], a collection of 77 essays published between 1787 and 1788 by three authors—Hamilton, Jay, and Madison—under the pseudonym "Publius". Until the last century, 12 of the papers had disputed authorship between Madison and Hamilton. In 1964, Mosteller and Wallace [315] identified certain stylistic keywords which they could use to discriminate the writing styles of Madison and Hamilton. Based on the frequency of these keywords and Bayesian statistics, they could correctly attribute the authorship of the disputed papers, confirming earlier work of Adair [9].

Significance in the Digital Age. The significance of the problem in the context of information technology was pointed out by Rao and Rohatgi [364] already in the year 2000: They take a computer science based approach at the example of newsgroup postings and are able to cluster and link their authors based on the usage of function words[1]. Importantly, they observe that there is "a significant amount of identifying information about the source that leaks from the contents of web traffic itself", and hence argue that "hiding explicit identity information is not sufficient to guarantee privacy."

Meanwhile, more sophisticated methods have evolved that utilize the power of computers: Earlier methods relied on statistics, rule-based algorithms, and classical ML with

[1] Function words typically have little lexical meaning and express grammatical relationships. Examples include articles (*a*, *the*), pronouns (*I*, *you*, ...), prepositions (*and*, *or*, ...), etc.; cf. https://en.wikipedia.org/wiki/Function_word.

engineered features such as the frequency of characters and certain words, n-gram overlap [78] and word sequence similarity [86], which are used, e.g., in the JGAAP [217] and JStylo [290] frameworks. For an overview of such early methods, we refer to the surveys by Stamatatos [422] and Jockers and Witten [211].

Particular sets of features that work well in uniquely characterizing individual authors have been proposed, e.g., by Abbasi and Chen [6] and Li et al. [256]. Such feature sets are sometimes referred to as *writeprints*, inspired by the use of fingerprints in forensics: Extracted feature values from texts with known authors form their writeprints, which then may be compared against features extracted from texts with unknown authorship.

Modern methods also employ deep learning techniques such as word- and character-level CNNs [376, 386, 409] or the attention mechanism that is prominently used in Transformer-based models [36, 128]. More recent techniques and developments are discussed in the surveys by Neal et al. [320], Swain et al. [428], and Tyo et al. [444]. Another extensive source of publications, events, and datasets related to authorship attribution and related problems is *PAN*, which is described as "a series of scientific events and shared tasks on digital text forensics and stylometry" (according to their website[2]). PAN stands for *Plagiarism Analysis, Authorship Identification, and Near-Duplicate Detection* and has its origins in a workshop held at the ACM SIGIR 2007 conference [423, 424].

Short Texts. The increasing popularity of instant messaging services as well as social media and microblogging sites in turn also spurred interest to research authorship attribution for short texts such as instant messages or short social media postings. An early study on authorship attribution for short texts was conducted by Sanderson and Guenter [394], and later studies by Bhargava et al. [44], Schwartz et al. [403] investigated authorship attribution on *tweets*, which are short messages limited to 140 characters that are posted on the social media site Twitter.

Eder [118] investigate the minimal length of texts that is required for reliable authorship attribution results. Moreover, short tweets and instant messages not only differ in length but also in language and style from longer documents: They may be less formal, less grammatically correct, and involve special "online" slang or text-based symbols such as emojis, whose frequency can be used as an additional stylistic feature [44]. As with other authorship attribution methods for longer texts, most early methods for short texts relied on handcrafted features such as character and word frequencies, n-grams, function word usage, etc. Recent works, however, also employ deep learning without manual feature engineering [436].

[2] https://pan.webis.de/

Use in Forensics. As a form of biometric identification, authorship attribution plays an important role in cyber forensics: Bhargava et al. [44] and Rocha et al. [379] consider authorship attribution for short texts since criminals may rely on online networks as convenient means of (presumedly) anonymous communication. Early works by de Vel et al. [94, 95] and later Iqbal et al. [202] investigate authorship attribution for emails. For a more extensive discussion of authorship attribution in the forensic setting, we refer to the book by Iqbal et al. [203].

Authorship Attribution for Computer Languages. Authorship attribution techniques have been successfully applied to source code [20, 55, 147, 148] as well as compiled binaries [59, 384]. Legitimate applications include, for instance, identification of malware authors [19] in the context of digital forensics.

Authorship Attribution through Linkage Attacks. Individuals can also be identified from their writings through *linkage attacks* [317, 318]: A famous example is the case of Thelma Arnold [34], an AOL user, whose history of search queries was released together with those of over 650,000 other users in 2006. The search logs were pseudonymized by associating the queries with their users through a unique numerical user identifier instead of their actual username. After some investigation into the search queries, the *New York Times* eventually learned enough information about user 4417749, so they could re-identify her as Thelma Arnold, a 62-year-old widow from Lilburn, a city in Georgia, USA.

Authorship Segmentation. For compilations of text, such as books, magazines, or newspapers, we could ask if there are different parts or segments of the text that vary in style due to having been written by different authors. *Authorship segmentation* accordingly aims at the detection of such style changes and has been investigated by Graham et al. [167] and more specifically for source code by Dauber et al. [91].

3.2.2 Attacks on Audio Data

As medium for spoken language, audio can convey named identifiers like text as medium for written language. Moreover, the characteristics of the human voice are a biometric identifier that can be used to identify the speaker. In this section, we hence consider attacks on audio data aiming at those kinds of sensitive information.

3.2.2.1 Named Identifiers

Identifiers and Secrets in Spoken Language. Spoken language, or speech, may also convey sensitive information: For instance, in the medical sector audio recordings such as physician-patient conversations, medical dictations, or patient phone calls may be subject to legal regulations [135]. Moreover, Quiroz et al. [361] remarked that "obtaining and sharing medical data presents a major obstacle due to privacy issues and the sensitive nature of the data". In a more general context, this also applies to call center recordings which may contain privacy-sensitive information [223] as well.

As with text, privacy-sensitive information in audio recordings may be detected via NER. In fact, NER for audio is traditionally performed in a two-stage process which first transcribes the audio to text through automated speech recognition (ASR) and then extracts the entities using text-based NER on the transcriptions [39, 61, 136], thus reducing the problem to detecting identifiers in textual data (cf. Section 3.2.1.1).

Recently, more direct *end-to-end* (*E2E*) approaches to NER have been proposed with experimental results indicating a similar or better performance than their two-staged counterparts. They work by integrating the two stages into one, for instance through a single ML model taking speech as input and directly producing entity labels with their associated values [248] or entity tags along with the transcribed characters [71, 157, 158, 482].

3.2.2.2 Voice

Audio recordings of human speech may not only contain explicitly named identifiers: Individual characteristics of the recorded voices allows recognition of the speakers; as such, the recorded voice acts as a *behavioral biometric identifier*. Moreover, many other privacy-sensitive attributes can be inferred from the speaker's voice, for instance, gender or ethnic origin, cf. the survey by Kröger et al. [242].

Speaker Recognition. As is common with biometric recognition approaches, speaker recognition can be classified into *speaker identification* whose aim is to identify a speaker among a set of candidates, or *speaker verification* whose aim is to validate whether a given voice sample originates from a specific speaker.

Typically, vocal features are extracted from utterances of a known speaker during a learning or enrollment stage and later compared to the vocal features of the speaker to be identified or verified. We also distinguish *text-dependent* and *-independent* methods: The former require the spoken utterances to be the same during the learning and recognition phases; the latter allow recognition of the speaker based on arbitrary utterances. Prominent vocal feature representations include *d-vectors* by Variani et al. [449] for text-dependent

methods, as well as *i-vectors* by Dehak et al. [96] and *x-vectors* by Snyder et al. [415, 416] for text-independent methods. In particular, x-vectors have also become popular as means to speaker obfuscation, as we discuss in Section 3.3.4.4.

Early speaker recognition methods were typically based on spectral properties of the voice and signal processing techniques such as filtering or vector quantization (VQ), cf. the surveys by Furui [153], Reynolds [374]. For instance, Kersta [228] describes an early automated speaker recognition system based on the spectrograms from recordings of certain cue words. The method was called "voiceprint identification" due to its similarity to identification based on fingerprints. Therefore, voice recognition also plays an important role in forensics [60].

Later methods also relied on probabilistic modeling; for instance, hidden Markov models (HMMs) in Zheng and Yuan [495], or Gaussian mixture models (GMMs) in Reynolds and Rose [372] as well as Reynolds [373]. Also, neural networks were used early on, e.g., by Oglesby and Mason [328] as well as Venayagamoorthy et al. [451]. More recently, also deep neural networks have been used, cf. the surveys by Irum and Salman [204] as well as Bai and Zhang [31]. For a broad overview of various methods and recent progress, we refer to the surveys by Ahmed and Hassan [13], Mohd Hanifa et al. [311], and Kabir et al. [218].

3.2.3 Attacks on Visual Data

Images and videos may also contain privacy-sensitive information such as identifiable faces of individuals or legible license plates on vehicles. This affects services such as Google Street View [164] and various recordings, e.g., of video conferences [220] or surgical procedures [411] where the identity of the participants or medical staff members should be protected, respectively. Moreover, Kosinski [238] showed that facial images of individuals can expose their political orientation.

3.2.3.1 Named Identifiers

Identifiers and Secrets in Written Language. Still and moving images may contain various forms of written language: They may show handwritten or printed texts in books, letters, and other documents, as well as public advertisements on billboards, street signs, and license plates on cars, among many others. Generally, we may assume that optical character recognition (OCR) techniques (see, e.g., [72, 301, 425]) can be applied to such depictions to extract the corresponding plain text; in those cases, we can utilize the methods described in Section 3.2.1 to detect and extract named identifiers (cf. Section 3.2.1.1) or to identify authors from their writing style (cf. Section 3.2.1.2).

Automated License Plate Recognition. An illustrative example of written identifiers is license plates which are used for the unique identification of vehicles, and indirectly, their owners. Automated license plate recognition (ALPR) systems typically work by first detecting the license plates in an image, followed by OCR to extract the license plate numbers from the identified image region depicting the license plate. For an overview of methods, we refer to the surveys by Anagnostopoulos et al. [22], Du et al. [113], and Shashirangana et al. [407].

3.2.3.2 Handwriting Style

Handwriting constitutes a behavioral biometric identifier by which its writers can be recognized. Strictly speaking, we must distinguish between an author who conceives and formulates a text and a writer who notes down the text in (hand-)written form: In most cases, they may be one and the same person, but the former (e.g., a doctor who dictates a text) and the latter (e.g., an assistant taking notes) could very well be different individuals. Therefore, the linguistic style actually stems from the creative thought process of the author who puts his thoughts into linguistic expressions, whereas the characteristics of a piece of (hand)writing, as opposed to its wording, arise from the inherent patterns in the movements of the writer's hand.

Handwriting Identification and Verification. Formally, we can distinguish *verification* and *identification of handwriting*: The former validates that a piece of handwriting originates from a specific writer, whereas the latter singles out one specific writer from a larger set of candidates that is the most likely one to have written the sample. In the following, we treat both handwriting verification and identification together for simplicity. Also, note that *handwriting recognition* typically refers to the extraction of text that was written, *independently* of its writer.

While plain text authorship attribution methods such as those described in Section 3.2.1.2 solely rely on stylistic characteristics of the text to identify its author, handwritten text offers additional clues to identify its writer. Based on the available kinds of additional features, handwriting identification (and verification) methods commonly follow one of the following two major approaches [201]:

- *Static* or *offline* methods rely on a static depiction of the finalized handwriting, from which visual features can be extracted (optionally in addition to the plain text, using OCR or related technologies).
- *Dynamic* or *online* methods also observe the writing process (e.g., when someone

is signing a document) from which even more biometric features such as pressure, speed, acceleration, etc. can be recorded over time.

Note that the extra features from dynamic approaches usually result in time series, i.e., another form of sequential data. However, since this section is concerned with visual data, we focus on the static approaches.

Early methods for handwriting identification and verification are discussed in the surveys by Plamondon and Lorette [347] and later by Leclerc and Plamondon [253]; notably, some of these methods already employed various kinds of simple neural networks [253, Table 2]. Other methods also utilize statistical models such as HMMs or GMMs [400, 401], as well as various dissimilarity measures and distance statistics [222, 489].

With technological progress, deep learning became more and more feasible and hence was also applied to the identification of handwriting: Convolutional neural networks (CNNs) were used first [143, 426, 476, 486] and were quickly followed by recurrent neural networks (RNNs) [185, 492]. Another approach by Wang and Jia [460] utilized generative adversarial networks (GANs). Recently, Koepf et al. [235] and Zhang [491] have proposed models based on Transformer networks [450]. A hybrid model that incorporates both visual features and stylistic features as utilized by plain text authorship attribution methods (cf. Section 3.2.1.2) has been investigated by Slaughter [413], where the combination of features implies the assumption that writer and author are the same person. For a recent overview, we refer to the surveys by Diaz et al. [104] and Hafemann et al. [176].

3.2.3.3 Facial Images

Face Detection (and Extraction). Automated methods to detect faces in images and videos have been studied extensively and a wide range of approaches has been proposed. Due to the vast amount of publications, we refer to the surveys by Hjelmås and Low [190], Zafeiriou et al. [488], Zhang and Zhang [490] as well as Kumar et al. [244] for an overview of past and recent methods. Detecting faces (and potentially other objects) in videos is often reduced to detecting faces in the individual frames of the video; additionally, face tracking techniques may be used as the position of a face will typically not move much between subsequent frames. Face detection typically also localizes faces within a larger image, i.e., it also yields the locations of any detected face instead of only indicating *whether some face* has been detected. Therefore, once the boundaries of a face have been determined, it can easily be extracted.

Face Recognition. Face recognition covers face identification ("Whose face is it?") and verification ("Does the face really look like yours?"), which are popularly used for biometric

authentication. Detection also forms the foundation of many face recognition methods, where the detected face is extracted based on its bounding box and then the actual recognition algorithm only considers the extracted facial image. Again due to the vast amount of works in this area, we refer to the surveys by Zhao et al. [493] and Jafri and Arabnia [205] for earlier methods, by Kasar et al. [225] for works based on machine learning, and by Masi et al. [281] as well as Wang and Deng [459] who give an overview of more recent methods based on deep learning techniques.

3.3 Defenses

In this section, we look at various defense techniques that prevent or mitigate attacks such as those discussed in Section 3.2. We roughly distinguish three types of defenses, namely data leakage prevention (DLP), private representations, as well as de-identification including masking and obfuscation techniques. Equally important, we also discuss approaches that provide differential privacy guarantees where applicable.

3.3.1 Data Leakage Prevention

Detection methods for privacy-sensitive data such as those discussed in Section 3.2 are often used by attackers to extract private and/or sensitive information. However, the same detection methods can also be utilized as a preventive measure to protect sensitive data: Data leakage prevention (DLP)[3] solutions detect unauthorized attempts to copy or transfer sensitive data, and thus prevent intentional or unintentional data breaches and data ex-filtration attempts (cf. Ullah et al. [445]), for instance, by raising alerts or blocking access to the data. Prominent examples of DLP solutions include *Google Cloud DLP* [77, 163] and *Nightfall AI* [3, 333].

Domain-Specific Solutions. Some DLP solutions such as *TruffleHog* [440, 441] or *Credential Digger* [267, 395] specialize on source code. They can help mitigate the attack by preventing unintentional leakage of secrets, for instance, by warning software developers and open-source contributors about potential secrets in their source code and in commits to code repositories [412].

[3] DLP is sometimes also infelicitously called data *loss* prevention, although we normally understand by *data loss* data becoming inadvertently destroyed or inaccessible in one way or another, where mitigation strategies typically involve backups or other forms of redundancy. These are, however, not in the scope of this dissertation.

DLP and Sanitization. Some advanced DLP solutions not only offer detection, but also sanitization of sensitive information by redacting, masking or replacing it to obtain sanitized or de-identified data, for instance, using obfuscation methods such as those discussed in Section 3.3.4. This may be useful when warning and blocking are inadequate or insufficient, for instance, when sharing or further processing of the (then sanitized) data is desired. Solutions focusing on obfuscation of explicit sensitive information include Microsoft's *Presidio* [2, 304], *Private AI* [434], and *Gretel.ai* [1].

For a broader overview of DLP and related approaches, we refer to the surveys by Alneyadi et al. [16], Shabtai et al. [404], and Kužina et al. [250].

3.3.2 Private Representations

Private representations of sequential data are often used with the intent of data sharing: Sharing the original data could leak privacy-sensitive information, whether explicitly in the form of named identifiers or implicitly, e.g., through some form of biometric attribute (cf. Section 3.2). Instead, if we encoded the original data into private representation vectors so that they contain only insensitive information, we could pass those instead without the privacy implications. A benign analyst then could use the private representations for further analysis tasks or to train machine learning models.

Representations can be obtained in various ways:

- The output of suitable intermediate layers in a (deep) neural network; we often refer to such outputs as *embedding vectors* (e.g., word embeddings).
- A BoW or term-frequency vector representation of text.
- The output of algorithmic transformations such as signal-processing techniques (e.g., the frequency spectrum of audio signals obtained via fast Fourier transform).
- The bitstream of data compression algorithms could also be seen as a form of representation (although again of sequential nature).

The remaining question is how to make these representations *private* so that they do not leak the same sensitive information as in the original data that we want to protect. In this section, we will look into several approaches to achieve this goal.

3.3.2.1 Private Text Representations

Various methods have been proposed to obtain private representations for textual data. Most methods directly obtain private text representations, e.g., by learning and sanitizing

the representations simultaneously, whereas other methods rely on existing, non-private representations and then sanitize them to make them private.

Learning Representations via Adversarial Learning. Coavoux et al. [80] consider a scenario for topic and sentiment classification of texts. Their classifiers consist of a text-to-vector encoder followed by a feed forward network for the actual classification into the utility (i.e., topic or sentiment) labels. Assuming an attacker who has access to the latent representations in the encoder output, they find that the representation vectors still leak private information about sensitive attributes such as age and gender of the authors of the texts, despite the classifier being only trained to predict the sentiment or topic. Moreover, they note that correlations between sensitive information and utility labels leads to a trade-off between privacy and utility, since one needs to sacrifice utility for privacy. To mitigate leakage of sensitive information from learned representations, they propose *adversarial learning* [155, 156] to suppress such sensitive information in the latent representations: An adversarial classifier that aims at predicting sensitive information from the latent vectors is added to the network, while the encoder network is optimized to fool the adversarial classifier and only encode information relevant for the utility task.

Elazar and Goldberg [119] confirm that adversarial learning reduces leakage of protected information from private text representations; however, they point out that a fair amount of sensitive information still remains and can be extracted from the private representations. Moreover, their experiments show that in order to estimate the attack performance accurately, the attack classifier must be retrained again after training the encoder that produces the private representations. Therefore, an *adaptive adversary* model is preferable over a static one as it provides more meaningful results.

Adversarial representation learning has also been employed by Li et al. [263] to protect more than one sensitive attribute. Friedrich et al. [151] train an automated de-identification system that removes PHI from medical records based on shared private representations.

Learning Representations via Reinforcement Learning. As an alternative to adversarial learning, Mosallanezhad et al. [314] investigate reinforcement learning using a reward function that includes attack and utility scores as feedback. They argue that their reinforcement approach allows better control of the trade-off between privacy and utility.

Sanitizing Representations via Linear Projections. Projection methods provide a way to obtain private representations from existing (potentially non-private) representations. A key advantage of such post hoc methods is that they work without retraining the embedding models.

In the privacy-related context of fairness and bias in pre-trained word embeddings, Bolukbasi et al. [47] analyze the geometry of the biased terms (here: gender) in the embedding space and are able to identify a low-dimensional subspace that captures this bias to a large extent. Accordingly, they propose a linear projection on the orthogonal subspace as a means to de-bias the embeddings. Subsequently, Ravfogel et al. [367] propose an automated approach called *Iterative Nullspace Projection*: It works by iteratively re-training linear classifiers for the protected attributes and projecting the data onto the nullspace (kernel) of each classifier's weight matrix, thus rendering the classifiers' decision boundaries ineffective. Haghighatkhah et al. [177] in turn propose two automated methods, *Mean Projection* and *Tukey Median Projection*, that work with a single projection only. Their experiments indicate that single projection methods can offer comparable protection but with fewer side effects on the overall space compared to Iterative Nullspace Projection.

A major limitation of methods based on linear projection is that they only protect the linear separability of the sensitive attributes; that is, other, non-linear classifiers may still be able to discriminate those attributes from the projected data successfully.

3.3.2.2 Private Audio Representations

Learning Representations via Adversarial Learning. To the best of our knowledge, Srivastava et al. [418] were the first to propose adversarial training as means to protect the identity of speakers in learned speech representations—although, as remarked by the authors, adversarial training had already found its way to automated speech recognition systems in previous works (see also Wali et al. [455]), e.g., to improve recognition performance. They demonstrated that adversarial training works well to protect the speakers' identity in a *closed-set* experiment where the speakers are known at training time. However, they found that the protection does not generalize well to *open-set* scenarios where the identity of a suspected speaker has to be protected in a larger crowd that includes speakers potentially unknown at training time.

Aloufi et al. [18] and Noé et al. [325] later independently proposed frameworks for learning privacy-preserving, disentangled latent representations that allow users to specify preferences as to what tasks may be performed and what attributes should be protected in the obtained representations.

3.3.2.3 Private Visual Representations

Learning Representations via Adversarial Learning. Feutry et al. [139] apply adversarial training [156, 161] to learn private representations from images. They demonstrate their approach in two scenarios with images of handwritten digits and facial expressions,

where the representations are learned in such a way as to allow classification of the digits or facial expressions, while hiding the identity of the writer or person whose face is shown, respectively. Pittaluga et al. [346] also learn private image representations using adversarial training, using various combinations of sensitive and desired utility attributes to be inhibited and sustained, respectively. As additional means to sustain utility, their approach also pursues a generic objective to maintain variance in the encoder output.

Li et al. [255] observed that non-private representations (e.g., extracted from a neural network layer) still may contain enough information that allows an attacker to apply representation inversion methods [274] to restore the original image. The attacker could thus learn sensitive information such as the identity of a depicted person. The authors hence propose an adversarial training framework called *DeepObfuscator* consisting of a classifier for the intended classification task, an adversary classifier simulating an attacker who aims to infer sensitive attributes, as well as an adversary "reconstructor" that simulates such an inversion attack by aiming to recover the original image. In their experiments, they compare their DeepObfuscator method against a baseline which simply adds Gaussian noise to each representation (corresponding to the Gaussian mechanism from DP). They find that simply adding Gaussian noise to the representations (or raw image) is less effective in protecting sensitive attributes or preventing reconstruction than the adversarial DeepObfuscator method while having a stronger impact on the intended classification tasks. A similar approach with adversarial representation learning is presented by Xiao et al. [475]: They also reconstruct the input image as in Li et al. [255], but the adversary attempts a model inversion attack following Fredrikson et al. [149, 150] to reconstruct the input.

Martinsson et al. [280] propose *PCGAN*, a two-step GAN architecture consisting of a filter and generator component, each with its own discriminator. Their model not only learns to suppress sensitive information in the representations (as often done when using GANs for privacy protection) through the filter component, but also to replace sensitive attributes by randomly chosen synthetic attributes in the generator component. They employ a *maximum entropy strategy* to optimize the filter as to maximize the uncertainty of the filter discriminator, which has been proposed by Roy and Boddeti [385] and shown to outperform the more traditional minimal log-likelihood strategy which would mislead the discriminator to consistently guess wrong values[4].

Learning Representations via Contrastive Learning. Osia et al. [334] modify existing neural networks by embedding an autoencoder between two intermediate layers, whose

[4]This leaves some information: E.g., if a binary classifier always guesses wrong, the adversary can easily get the correct answer by negating its output.

latent variable is then used as private representation. To hide sensitive attributes, they introduce an additional contrastive loss in the training objective that is based on the latent representations. Its goal is to reduce the distance between representations that correspond to inputs with *different* sensitive labels. They also propose adding noise as an extra measure for further privacy gains, however, no formal DP guarantees are given in the paper. Importantly, they observe that noise is more detrimental to fine-grained information and thus suggest to only inject noise when the granularity of the sensitive information (e.g., identity) is finer than the granularity of the target variable (e.g., facial expression) that is intended to be predicted.

Sanitizing Representations via Linear Projections. Xu et al. [479] use linear projections to remove feature components that lie in the nullspace of linear predictors of desired information (utility). Their approach is thus complementary to the projection-based methods for fairness in text embeddings discussed in Section 3.3.2.1 which determines the directions to remove based on the *sensitive* information that shall be suppressed. They evaluate their methods on various datasets, including datasets from the image and music domain, and could likely be adapted to various other domains, including text or audio.

Sanitizing Representations via Adversarial Learning. Morales et al. [313] assume a pre-trained embedding network and propose to append a multilayer projection network (*SensitiveNets*) that is trained to sanitize the representations and hide sensitive attributes while still allowing desired utility tasks (here: face verification). The network is adversarially trained layer-by-layer with a triplet loss so the layers in the projection network suppress the sensitive information.

3.3.3 Differentially Private Representations

Most fundamental DP mechanisms are only directly applicable to numerical data such as scalars or vectors, which is one reason why applying DP mechanisms to unstructured data is difficult. A possible solution that avoids devising complex mechanisms for complex, unstructured data is encoding it into vector representations, which are amenable to the many existing numerical DP mechanisms.

3.3.3.1 Differentially Private Text Representations

To the best of our knowledge, the first approach that produces purely differentially private text representations is our own, SynTF [465, 466], which we describe in detail in Chapter 5. It transforms texts into a term-frequency or Bag-of-Words (BoW) representation, using

the Exponential mechanism by McSherry and Talwar [294] to randomly substitute words with similar words, e.g., as determined by the cosine similarity of their word embeddings. Following our approach with BoW representations, Fernandes et al. [137, 138] propose a novel variant of metric privacy [65] (cf. Section 2.2.3) called *Earth Mover's Privacy*, which is based on a special case of the Earth Mover's Distance applied to "bags" (i.e., sets) of word embedding vectors. They also substitute each word, but by perturbing the original word embedding vector with a multivariate variant of the Planar Laplace (PL) mechanism (also see Section 2.3.3) and replacing it with the word pertaining to the nearest neighbor.

Xu et al. [480] propose a novel perturbation mechanism for (word) embeddings based on the Mahalanobis norm which better respects the distribution of words in the embedding space: Their *Mahalanobis mechanism* applies elliptical noise instead of isotropic noise that is produced by the usual Laplace or Gaussian mechanisms.

Lyu et al. [269] propose a locally differentially private protocol based on differentially private representations for learning from crowd-sourced textual content: Each user maps their texts to a sequence of (word or token) embeddings, encodes the sequence of embeddings to a fixed-length binary vector, where each bit is perturbed. The randomized vectors of all users are then transferred to a potentially untrusted server, who can then train a ML model (e.g., a classifier) based on the collected vectors.

Feyisetan and Kasiviswanathan [140] take existing embedding vectors and project them to a lower-dimensional space using a random projection. They then apply the Planar Laplace mechanism (Definition 2.23) to the projected vectors to achieve metric privacy[5] [65] (see Section 2.2.3), an extension of DP.

From Private Representations to Entire Texts. This line of research also inspired various differentially private obfuscation methods for entire texts (e.g., [141, 142]): Any DP mechanism for individual tokens (e.g., words as in SynTF) or their embeddings naturally extends to a DP mechanism for texts (regarded as sequences of tokens) by iteratively applying the DP mechanism to each token in the sequence and replacing it with the mechanism result. (In case of embeddings, we find the word whose embedding is the nearest neighbor of the perturbed embedding.) We discuss those methods and their limitations in more detail in Section 3.3.5.1.

In the context of pre-trained language models such as BERT [101], Qu et al. [360] investigate the effects of DP obfuscation of the model's input texts (viewed as a sequence of tokens or embeddings) at various stages of the model on the model's performance for various tasks: DP is achieved for a model input by applying the PL mechanism either to the

[5]The authors of the paper refer to metric privacy as Lipschitz privacy.

global sequence embedding (i.e., BERT's [CLS] token), individually to all the sequence's token embeddings, or at the text level following Feyisetan et al. [142] by replacing each token with the token closest to the perturbed token embedding via nearest neighbor search. The authors further analyze how well the model can adapt to each type of privatized input, and how they can improve the model's ability to handle privatized inputs by privacy-adaptive pre-training.

Faulty Approaches. Beigi et al. [40, 41] and Alnasser et al. [15] propose an approach based on a (non-variational) autoencoder that is trained to reconstruct the input texts: They perturb the latent vectors using the Laplace mechanism, but instead of releasing the reconstructed texts based on the noisy latent vectors, they release the noisy latent vectors directly as obfuscated text representations. Unfortunately, Habernal [175] identified a faulty Laplace inversion sampling procedure in their method, which therefore violates DP. Moreover, they implicitly use L^∞ clipping (due to element-wise tanh being used as activation function in the encoder output) whereas the Laplace mechanism is based on the L^1 sensitivity; this is suboptimal as the L^1 sensitivity will then grow with the dimensionality d of the latent space instead of being constant.

Another issue was found by Maheshwari et al. [275] in the differentially private representations of Plant et al. [348] and Lyu et al. [268]: Their approach adds Laplace noise to each entry of the encoded representations; however, they incorrectly assume a sensitivity $\Delta = 1$ by normalizing the encoded vectors *entry-wise* to the range $[0, 1]$, which in fact yields a much larger sensitivity of $\Delta = D$, where D is the dimension of the vector. The reported privacy losses ϵ are therefore substantially larger by a factor of D. Maheshwari et al. [275] in turn propose a corrected version by normalizing with the L^1 norm of the representation vector and also incorporate adversarial learning to improve fairness and protect other sensitive attributes.

3.3.3.2 Differentially Private Audio and Visual Representations

We are currently not aware of published methods that aim at producing differentially private audio or image representations as their *final* result. However, differentially private representations certainly are used as means to the end of obfuscating audio and visual content, as we discuss in Sections 3.3.4.4 and 3.3.4.5.

3.3.4 Data Obfuscation

In this section, we discuss data obfuscation methods that protect sensitive information and preserve the format of the data, i.e., an obfuscated text, audio recording, or image, is

again represented as text, audio recording, or image respectively.

Differentiation from Private Representations. Private representations are encoded representations of the original data, e.g., as numerical vector, but where sensitive information is obfuscated and hard to (ideally) impossible to extract (cf. Section 3.3.2). They are suitable in scenarios where the representation vectors are processed automatically, e.g., by a computer. However, in case the sanitized data should still be interpretable by humans, it is preferable that the sanitized result remains in the same format as the original data: For instance, when sanitizing a piece of text, we would expect the output to be again human-readable text instead of some form of vector representation.

Obfuscation through Private Representations. Data obfuscation methods for sequential data that preserve the original format of the data often rely on an encoder-decoder (or, in terms of signal processing: analysis-synthesis) architecture to transform the data so that (privacy)-sensitive information is suppressed while other (insensitive) content is preserved. On a high level, autoencoders consist of an encoder component that encodes the input to some intermediate, *latent* representation as well as a decoder (or generator) component that aims at reconstructing the original input from the encoded representation. Consequently, a natural approach for obfuscating sequential data is to apply the techniques for private representations, such as those presented in Section 3.3.2, to the latent representations in the autoencoder model so that the decoder produces sanitized output again in a human-interpretable format.

3.3.4.1 De-Identification of Textual Data

De-identification, *masking*, or *scrubbing* methods provide a way to remove or mask privacy-sensitive or personally identifiable information (PII) from (typically unstructured) documents. They are often motivated by the healthcare and medical sectors and focus on identifying and removing particular types of personal information such as protected health information (PHI), a list of 18 identifiers as specified in the US Health Insurance Portability and Accountability Act (HIPAA) [435, 447]. As such, they can generally be considered as specialized variants of named entity recognition (NER) focusing on privacy-sensitive data.

Earlier methods include the "Scrub System" [429], the PhysioNet "deid" software package [321], or the "MITRE Identification Scrubber Toolkit" (MIST) [7]. They typically work with lists of names and identifiers, regular expressions, and simple heuristics to identify and remove pieces of text that constitute PII.

More recent methods also employ modern ML and natural language processing (NLP) techniques to find (and replace) PII in unstructured data, such as, for instance, conditional random fields (CRFs) [25, 184], support vector machines (SVMs) [410], decision trees [432], RNNs [98, 230], or deep neural networks including Transformers [214, 316, 471]. Many solutions are in fact *hybrid systems* that use combinations of various techniques [265, 266]; for instance, Microsoft's "Presidio" [304] is a customizable framework that incorporates, among others, pattern-based rules such as regular expressions, NER based on various libraries such as spaCy [193], and other deep learning methods.

3.3.4.2 Authorship Obfuscation

Manual Methods and Machine Translation. Rao and Rohatgi [364] examine newsgroups postings and identify the authors from the body of the text by analyzing the frequency of *function words*, i.e., words such as articles or pronouns with little or no lexical, but grammatical meaning that expresses relationships between parts of a sentence. They suggest to either use automated "round-trip" *machine translation* to a foreign language and back, or to *educate authors* who want to write anonymous documents about authorship attribution attacks, so they can intentionally hide their writing style.

Brennan and Greenstadt [52] and Brennan et al. [51] consider *adversarial writing* by authors to intentionally hide their writing style or imitate other authors, and automated "round-trip" *machine translation* [51], both in line with the countermeasures proposed by Rao and Rohatgi [364]. Their evaluation indicates that machine translation is insufficient to protect against authorship attribution. This has been confirmed by Caliskan and Greenstadt [58] who observed that even *multiple rounds* of machine translations do not prevent authorship attribution. Furthermore, Afroz et al. [11] show that *deceptive writing* by an author trying to imitate another or to obfuscate his own writing style can still be detected with high accuracy.

Kacmarcik and Gamon [219] present an automated approach based on decision trees that informs authors about the most revealing terms regarding their identities so that they can manually revise the document. In their evaluation, the manual edits are simulated by adjusting the tf vector of a document by moving its feature values closer to those of other writers, as to prevent the classifier from identifying the correct author. While the countermeasure is effective against the evaluated SVMs with up to 70 features, the more sophisticated *unmasking* approach by Koppel and Schler [236] and Koppel et al. [237] is still able to distinguish the actual author from others. Kacmarcik and Gamon [219] in turn propose a "deep obfuscation" variant of their method which uses more iterations to make unmasking harder; however, this quickly becomes cumbersome as it requires the users to

make more and more manual changes to their documents. Their results suggest that it is *insufficient to change only small parts* of a text to successfully mitigate authorship attribution attacks.

"Anonymouth" by McDonald et al. [290] is based on JStylo and uses clustering of two reference sets with the author's and foreign sample texts to propose manual changes that have to be made to the document to prevent authorship attribution. The process must be repeated until the attack is mitigated sufficiently. The results indicate that both methods are successful in preventing authorship attribution attacks in theory. However, the authors of Anonymouth [290] observed that while users were able to implement the suggested changes for very small feature sets with only 9 features, they felt overstrained by the amount of changes needed for the more advanced "WritePrints (Limited)" feature set (which we also used in our experiments in Chapter 5). This is in line with the earlier observation by Kacmarcik and Gamon [219] that for a deep level of obfuscation, one would have to consider more and more features and make corresponding changes to the document, thus increasing the complexity for the user. In practice, such manual methods seem cumbersome for the user if a deep level of obfuscation shall be reached.

Deep Learning. Motivated by recent improvements in machine translation, Keswani et al. [229] re-evaluated back-translation based on state-of-the-art machine translation services (Google Translate, Bing Translate, Yandex). The authors conclude that translation "seems a worthy attempt"; however, as noted by Potthast et al. [350], the authorship verification was evaluated only in an *obfuscation-unaware* manner, i.e., in a static attacker model.

Shetty et al. [408] propose a cyclic GAN network [161] with long short-term memory (LSTM) encoder-decoder blocks [191] that learns to imitate the style of a target class while preserving the semantic content of the input. Similarly, Emmery et al. [121] propose an encoder-decoder architecture with LSTM cells in combination with a gradient reversal layer (GRL) [155] on the latent context vector to obtain style-invariant sentence embeddings, which are thus decoded into neutral rewrites of the input texts.

Feature-Engineering. Karadzhov et al. [224] propose an automated approach that modifies a text to appear "mediocre" subject to certain text metrics that are commonly used as significant features for authorship attribution. The modifications are taken from a set of transformation rules with the additional goal of preserving the meaning of the text. Similarly, Romanov et al. [381] develop a system for Russian text that smoothens out certain features that are significant for authorship attribution towards average values. The output text is generated by a Transformer-based model from the modified features.

Bevendorff et al. [43] propose using the Jensen-Shannon distance between the character trigram distributions as a metric to measure stylistic distance between texts. Based on this metric, they use a heuristic search to find a series of paraphrase operations that obfuscate a given text and reach a desired obfuscation distance.

Genetic Algorithms. Mahmood et al. [276] propose MUTANT-X, a genetic algorithm that anonymizes text by making word replacements aiming at lowering the attribution probability but preserving the text's semantics. The evaluation is performed with a fixed authorship attribution classifier, corresponding only to a static attacker model.

3.3.4.3 De-Identification of Audio Data

Cohn et al. [82] and Baril et al. [35] consider the de-identification of audio recordings. They reduce the core problem to textual data by breaking it down to automated speech recognition (ASR) followed by NER on the transcribed text, and aligning the text with the original audio recording. The segments tagged as sensitive entities are then redacted in the recording.

A slightly different scenario is addressed by Cohen-Hadria et al. [81] who consider the obfuscation of voices of people in *urban sound recordings*. This approach can be regarded as analogous to the blurring of license plates and faces for privacy reasons in photographs of urban surroundings as, for example, in Google Street View [164] (see Section 3.3.4.5).

3.3.4.4 Voice Sanitization

Voice Conversion. Early works by Pellom and Hansen [341] and later by Matrouf et al. [282] as well as Bonastre et al. [48] investigate automated voice transformation based on spectral properties of the voice signals (cut into smaller frames) to modify the voice of an imposter to be falsely recognized as a specific target speaker by the speaker recognition system. With the designated goal of de-identifying speech recordings, Jin et al. [209, 210] conducted benchmarks of various voice transformation methods. Their results show that the methods were successful in fooling two tested speaker identification systems into accepting the de-identified recordings, where the best method also prevented human listeners from recognizing the transformed voices of people well-known to the listeners.

Bahmaninezhad et al. [30] employ a CNN encoder-decoder architecture to map the intermediate, spectral and excitation features from the source speaker to an average of known target speakers. Another approach by Prajapati et al. [351] utilize a CycleGAN architecture as well as timescale modification to convert the voice characteristics from the source speaker to a pseudo target speaker.

Qian et al. [357] developed a mobile app called *VoiceMask* that protects both speaker identity using voice conversion technology and sensitive content (named identifiers) by replacing selected, sensitive keywords with safe surrogate terms using a differentially private substitution mechanism. In another study, Qian et al. [359] study various ways to sanitize speech that modify both voice and content, involving voice conversion and synthesis (from transcripts) as means for voice obfuscation, as well as content sanitization that substitutes sensitive key terms.

Adversarial Learning to Suppress Sensitive Information. Ericsson et al. [122] propose a model to protect sensitive attributes (gender) of the speakers from speech recordings. It first encodes the speech input to a spectrogram as an intermediate representation from which they filter out sensitive information by adapting a two-step GAN architecture [280] to the spectrogram domain and then transforms the sanitized spectrogram back into speech.

Aloufi et al. [17] employ a CycleGAN architecture [497] to suppress sensitive information that reflects emotions of the speech signal in the extracted features, and then re-synthesize a neutral speech signal that is free of emotion.

Voice Distortion. Patino et al. [338, 339] propose a simple voice anonymization approach relying on basic signal processing techniques using the McAdams coefficient [289]. It is also used as one of two baselines at the VoicePrivacy challenge [439]. Kai et al. [221] propose a lightweight pseudonymization (reversible) and anonymization (irreversible) framework for speech. Their method is based on signal processing techniques with few parameters that allow efficient processing, where they use data-driven hyperparameter optimization to optimize the signal processing parameters.

Methods Based on x-vectors. In the context of speaker recognition, Snyder et al. [416] proposed *x-vectors*, which are fixed-dimensional embeddings that can serve as robust speaker representations. They are extracted from a deep neural network trained to discriminate speakers and have since gained popularity as speaker representation and also as a means to speaker obfuscation as we will see in the following.

Fang et al. [133] propose an x-vector based speaker anonymization system. It utilizes a speaker-independent automated speech recognition system to extract content features representing the spoken words, as well as a pre-trained x-vector model to encode the speaker identity. To obfuscate the speaker identity, they change the x-vector to a combination of external speakers' x-vectors and re-synthesize the speech signal from the obfuscated

x-vector and unmodified content features. Their method also serves as one of two baselines in the *VoicePrivacy challenge* [439].

Champion et al. [64] build on the VoicePrivacy baseline based on x-vectors by Fang et al. [133]. They propose better content features by training a deep encoder-decoder network for automatic speech recognition. Moreover, to prevent speaker information leaking into the content features, they apply adversarial training to the feature representations to suppress the undesired information leakage. They substitute the speaker's x-vector with a random target speaker's x-vector to obfuscate the speaker identity.

A similar approach is presented by Perero-Codosero et al. [343] who also follow the x-vector baseline [133], but they use an *AAE* to transform the original x-vector into a sanitized x-vector, removing sensitive information such as the speaker's identity, gender, and/or accent from the autoencoder's latent vector through adversarial training.

Turner et al. [442] observe that the VoicePrivacy baseline [133] produces pseudo speaker voices that sound much more similar to each other than real speakers. They argue that this is due to the averaging of multiple x-vectors from the pool of x-vectors to define the target x-vector, which smoothens out details and individual voice characteristics that are present in unmodified x-vectors. They hence propose a GMM to generate target x-vectors that better capture the diversity of voice characteristics of real speakers.

The method by Mawalim et al. [288] is also based on x-vectors as well as clustering and singular value decomposition (SVD): They first apply gender-dependent clustering to the pool of x-vectors and choose the centroid of the furthest cluster as target x-vectors. Next, they decompose the target x-vector using SVD and truncate the number of singular values to obtain a more general speaker representation. As further measures for obfuscation, they modify the fundamental frequency (F_0) and stretch the speech duration.

Miao et al. [302, 303] investigate language-independent speaker anonymization. They rely on the Transformer-based HuBERT model [194] which they fine-tune to learn content representations in a self-supervised manner. They also propose an updated model to derive speaker representations similar to x- or d-vectors [349, 416, 449] which they substitute to that of another speaker to achieve voice anonymization.

Benchmark Studies. Srivastava et al. [420] evaluate the effectiveness of voice conversion methods to protect the identities of the speakers. They raise concerns that previous studies often assume a weak attacker who does not take into account that the speech data has been obfuscated. Therefore, they consider *informed* attackers (corresponding to an adaptive attacker model) who aim at identifying the speaker through various linkage attacks that *are* aware of the inner workings of the conversion methods. They conclude that depending on the attacker's level of knowledge, the evaluated methods were unable to sufficiently

hide the speaker identities from the obfuscated data.

In two later studies, Srivastava et al. [419, 421] analyze the impact of different pseudo-speaker selection strategies on the x-vector VoicePrivacy baseline [133]. Again, they consider different attacker models in their evaluation, showing that an ignorant (i.e., static) attacker model overestimates privacy and that an informed (adaptive) attacker model provides a more substantial privacy assessment. Those observations agree with our own findings that it is important to rely on an adaptive attacker model to properly assess the effectiveness of the protection mechanism.

3.3.4.5 De-Identification and Obfuscation of Visual Data

In this section, we look at methods to de-identify or obfuscate visual data. While we treat de-identification and obfuscation separately for text and speech, we merge both approaches for visual data since both approaches may be applicable to the same kind of visual identifiers.

Visual identifiers such as depicted faces or license plates may be confined locally to a *region of interest* (ROI) that is small relative to the entire picture, e.g., in street-level imagery where a person is shown getting out of their car. For this reason, many approaches for protecting visual identifiers first apply methods as discussed in Section 3.2.3 to detect the sensitive areas or ROIs in a larger image, before feeding this region to the actual protection mechanism. In this case, simple methods such as masking or blurring of the sensitive regions may be sufficient to de-identify the image, *if* the remaining parts that are left unchanged still convey enough useful information (utility). However, if we zoom in, say on the person's face, we gradually transition from a street photo towards an image that is filled out entirely by the face. In that case, masking or blurring the entire sensitive area would destroy most or all information in the image; therefore, more sophisticated obfuscation techniques are preferable that only change the privacy-sensitive, identifying characteristics, while leaving non-sensitive attributes of the face (e.g., expression or viewing direction) unchanged. In summary, depending on the context, for visual data we may find that either simple masking techniques or more sophisticated obfuscation techniques are better suited to protect the same kind of visual identifier, hence we discuss both approaches jointly in this section.

For comparison, recall that in text and speech, named identifiers, like names and addresses, typically are locally confined to a few words only, whereas biometric identifiers, like writing style and voice characteristics, typically pervade long parts of the sequence. Therefore, for those linguistic types of data, it makes sense to distinguish between de-identification for locally confined, named identifiers which can be achieved with simple

techniques such as masking, and more sophisticated obfuscation techniques for pervasive, biometric identifiers.

Redaction (Masking, Black Bars). Redaction may be useful in scenarios where sensitive information is locally confined to one or few smaller areas that can be masked completely without affecting the utility of the image: For instance, Orekondy et al. [332] propose an automated, segmentation-based framework for detecting and redacting various kinds of sensitive information in images, e.g., as posted on social media platforms: These may, for instance, depict persons, their faces, ID cards, or named identifiers in depicted written texts (cf. Section 3.2.1.1) such as license plates, phone numbers, etc.

Issues with Redaction of Facial Images. In their evaluation, Newton et al. [324] examined how face recognition is affected by masking the eyes or eyes and nose with rectangular or T-shaped black bars, respectively. While those classical methods may suffice to prevent humans from identifying the masked faces, the results indicate that the used face recognition system was able to successfully recognize all masked images with 100% accuracy by simply retraining on faces that were masked in the same manner. Similarly, pixelization barely had any impact on face recognition performance.

Preston et al. [352] study the effectiveness of covering various parts of the face to a varying degree in a medical context. They conclude that covering facial features with black bars or boxes does not prevent re-identification by human viewers and note that covering large parts of a face comes with a reduction in utility. As an alternative, they recommend seeking patients' consent for the publication/sharing of patient images if anonymity cannot be guaranteed. Similar findings were obtained in an earlier study by Clover et al. [79]. Overall, simply covering faces with black bars has been discouraged in the medical sector [21]: For instance, the *Uniform requirements for manuscripts submitted to biomedical journals* states specifically that "masking the eye region in photographs of patients is inadequate protection of anonymity" [4, Section II. E.]. Unfortunately, a recent survey by Roguljić et al. [380] has found that some medical journals still publish identifiable patient images where only the eye regions are covered or blurred, and sometimes even no de-identification is used at all. Moreover, they found that the call for obtaining consent (cf. [42]) is not always followed.

Blurring and Pixelization. Martínez-Ponte et al. [279] propose to protect faces in videos by detecting facial features and encoding the identified regions in a low quality, resulting in a blurry appearance that disguises the facial features. Frome et al. [152] present a system that automatically detects and blurs faces and license plates in street-level photography on

Google Street View in particular. Similarly, *YouTube* has introduced a feature to automatically detect and blur faces as well as other potentially privacy-sensitive objects in videos [83, 84]. Ilia et al. [200] propose face blurring as means to mitigate privacy leakage by preventing unauthorized users from recognizing peoples' faces on social media images.

Issues with Blurring and Pixelization of Depicted Persons. In their study, Lander et al. [252] found that while pixelization and blurring can lower the chance or confidence of recognition, pixelized and blurred faces often remain recognizable, particularly if they are familiar to the human viewer. More drastically, Demanet et al. [97] conclude that "masking just the face leads to an unacceptably high degree of recognition, independent of which level of pixel[iz]ation was used". They also demonstrate that non-facial cues, such as clothing, body, and hair in particular also facilitate identification, and hence should also be masked for proper de-identification. Their findings are supported by Oh et al. [329], who build a *Faceless Recognition System* based on CRFs which demonstrates that it is possible to recognize persons even if their faces have been blurred or masked entirely. Moreover, Li et al. [262] demonstrate that blurring and pixelizing are ineffective even when applied to the entire body of a person. They further conclude that removing a person or object and masking or replacing it entirely, e.g., with an avatar or an in-painted background, are much more effective than obscuring just the face, and preferable from a viewer's perspective.

Another automated approach by McPherson et al. [291] demonstrates that CNNs are able to accurately recognize faces, objects, as well as handwritten digits that have been obfuscated by pixelization or blurring. Yang et al. [484] conduct a benchmark study covering several deep neural face recognition networks and observe only slight accuracy drops when faces are blurred or masked.

Issues with Blurring and Pixelization of Depicted Text. Hill et al. [188] demonstrate that the original text can be recovered with high accuracy from blurred or pixelized text using HMMs. Hence, they conclude that blurring and pixelization are, in general, ineffective to protect sensitive information in depicted text. Meanwhile, several tools have become publicly available that help to automatically recover the plaintext: The approach based on HMMs [188] has been implemented by Schatz [397] and builds on an "unredaction" tool called *Depix* by Schipper [399]. Another tool, *Unredacter* by Petro [344, 345], uses a simple brute-force approach that enumerates through all letters, pixelizes them, and matches them with the pixelized text.

Approaches Based on k-Anonymity: The k-Same Family. Newton et al. [324] propose a new approach to de-identify facial images, *k-Same*, which is based on k-anonymity

[392, 431]. It works by finding the closest k faces (e.g., from a dataset or extracted from a video stream), computing their average, and replacing each of the k faces with their averaged version, thus making them indistinguishable. The authors propose two variations, k-Same-Pixel and k-Same-Eigen, where the average is computed based on pixels and eigenfaces, respectively. Subsequently, Gross et al. [168] proposed an improvement called *k-Same-select* that better respects certain facial attributes. It works by partitioning the reference images into sets according to the attributes (e.g., facial expression), and averaging only within the cluster corresponding to the attributes of the input image to be obfuscated.

k-Same with Generative Models. Generative models such as active appearance models (AAMs) [85] allow modeling faces based on a set of model parameters. Such generative models can also be used in the context of k-Same, by doing the averaging in the space of model parameters (i.e., a form of latent representation of the image) instead of averaging existing reference images. A first approach employing AAMs is *k-Same-M* by Gross et al. [169]: For each input image, the parameters of a pre-trained generative model are adjusted, so the generated image looks similar to the input image. Then k-Same averaging is applied to the model parameters before a face is regenerated from the averaged parameters.

In order to better preserve the individual look of faces as determined by certain utility attributes (e.g., gender, age, race), Du et al. [112] train attribute-specific generative models on an external dataset, and for each model, determine n *superfaces* by clustering the training images' representations into n clusters. To obfuscate a facial image, they first determine its utility attributes through pre-trained classifiers and find the image's closest representation according to the corresponding attribute-specific generative model. The input face is then substituted with the superface whose representation is nearest to the input image's representation. Similarly, Jourabloo et al. [216] also find the nearest k images that share the same utility attributes with the input, but instead of a constant average, they use a weighted average. The weights are determined using gradient descent where the objective is to preserve the attributes of the original facial image via attribute classifiers, but to obtain a different appearance as estimated through a face verification classifier.

Meden et al. [296] find the k closest faces according to representation extracted by a deep face recognition network. Combinations of the k feature representations are fed into a generative neural network to create a synthetic face with the possibility to preserve non-identity-related attributes. Expanding on that idea, Meden et al. [295, 297] later proposed a method called *k-Same-Net* with an additional clustering step on an additional (proxy) image dataset that is used to train the generative network.

Autoencoders. Nousi et al. [326] employ a deep autoencoder model which is trained to

de-identify facial images in its latent representation. In a supervised setting, they fine-tune the encoder to produce defined target latent representations based on desired attributes to be preserved, as well as undesirable or sensitive attributes to be suppressed. They also propose a method for unsupervised scenarios where no attribute labels are available, and where target latent representations are defined based on clustering.

Gong et al. [160] propose a twofold chained VAE architecture with shared weights that learns disentangled representations for identity-related and identity-independent information by first replacing and then restoring the original identity. The disentanglement allows the face identity to be obfuscated while the non-identity information is preserved from the input image.

Adversarial Examples. Szegedy et al. [433] observed that it is possible to fool classifiers by incorporating small, intentional perturbations in the input image that are (almost) imperceptible to the human eye. The perturbed images are called *adversarial examples* and have also been used to fool automated face recognition systems: For instance, Chatzikyriakidis et al. [69], Oh et al. [330], Sharif et al. [406], and Yang et al. [483] generate adversarial examples that preserve the perceived facial appearance but which are misclassified by the face recognition system, thus providing some form of anonymity against computerized identification. Oh et al. [330] further compare the effectiveness of various adversarial perturbation techniques to evade recognition, as well as possible countermeasures, such as blurring or translating the perturbed image by a random offset, to mitigate the attack.

Adversarial examples come with several limitations, particularly when it comes to proper anonymization (also) against human observers:

1. First and foremost, they can still be recognized by humans. This may, however, be intended or acceptable.
2. They usually only fool one or few specific targeted classifiers.
3. In general, they assume a white-box model (inner workings of the targeted classifier are known), but possible extensions to a black-box model may be possible [406].
4. Simple countermeasures, such as slightly blurring the image before feeding it into the targeted classifier may already mitigate the attack [330].

Adversarial Learning. Raval et al. [366] train a GAN [161] with a denoising autoencoder as generator to hide secrets (e.g., QR codes) in the input images. In their experiments, they verify the effectiveness against both a weak and a strong adversary who have either

no access or black box access to the obfuscation mechanism, i.e., the strong adversary can create its own perturbed instances of the original training images to adapt to the obfuscation. The results indicate that the method protects well against the weak adversary, but the strong adversary is still able to correctly identify images with secrets with a 75% accuracy, thus highlighting the need to consider an adaptive attacker model for a meaningful evaluation.

Motivated by the fact that people can also be identified based on, e.g., hair and clothing (cf. *Issues with Blurring and Pixelization* and, e.g., [97, 329]), Brkic et al. [53] replace the entire silhouette of a person with a synthetic full-body image generated using a GAN architecture [161]. Sun et al. [427] propose a head inpainting method for face obfuscation based on GANs which completes head regions depending on the context and therefore also works for people whose heads appear in various poses and against diverse backgrounds. Similarly, Hukkelås et al. [197] propose *DeepPrivacy*, a conditional GAN model [161, 310] with a U-net architecture [383] that is able to generate realistic facial images that seamlessly fit into the existing background. A successor, *DeepPrivacy2* [196], has been extended to a full-body anonymization framework.

The approach by Wu et al. [474] also utilizes GANs but applies adversarial training only to a discriminator to generate sharp and realistic faces. It employs a "face verificator" module that is pre-trained but frozen during training of the obfuscation network to guide the network to generate a face with a different appearance; additionally, they add a loss term to preserve the structural similarity index (SSIM) between the input and output faces.

AnonymousNet by Li and Lin [261] is able to de-identify facial images while preserving a range of facial attributes. It is based on a combination of several approaches including t-closeness [258], adversarial perturbations [433], as well as adversarial learning [161].

Videos. Videos are temporal sequences of individual images (so-called *frames*). Hence, we can apply obfuscation methods for images, often in conjunction with some form of tracking across subsequent frames of the regions/segments containing the sensitive information (e.g., faces or entire persons) to be obfuscated. For specific approaches to de-identification in videos, we refer to the works, e.g., by Agrawal and Narayanan [12], Gafni et al. [154], Letournel et al. [254], Ren et al. [370], Silas et al. [411], Wang et al. [457, 458], and Zhu et al. [496].

Surveys. A comprehensive study of anti-facial recognition techniques was conducted by Wenger et al. [470]. They break down facial recognition systems into several stages from image collection and processing over feature extraction to the creation and querying of a reference database, and consider possible countermeasures at each stage. Another

recent survey focusing on face biometrics has been compiled by Meden et al. [298]. Earlier surveys are provided by Padilla-López et al. [336], Ribaric and Pavesic [377], as well as Ribaric et al. [378].

3.3.4.6 Obfuscation for Directional Data

Directional data is often privacy-sensitive: For instance, a geolocation referring to a specific address can represent privacy-sensitive information. This is also reflected in legal frameworks such as the HIPAA [447], which classifies "geographic subdivisions smaller than a state, including street address, city, county, precinct, ZIP code, and their equivalent geocodes" [327] as PHI.

Many obfuscation techniques have been proposed to provide privacy for location-based services (LBS) and enable privacy-preserving data publishing (PPDP) for location data and trajectories. Due to the vast amount of literature, we refer to the surveys by Chatzikokolakis et al. [68], Primault et al. [354], and Jiang et al. [208], as well as by Fiore et al. [145] who focus on trajectory data.

3.3.5 Differentially Private Obfuscation

In this section, we consider obfuscation methods for sequential and directional data that additionally provide DP guarantees.

3.3.5.1 Differentially Private Mechanisms for Text

Word-Level Differential Privacy. In some cases, it is possible to interpret obfuscation schemes that redact or replace only individual parts of a text as *word-level DP*, where two texts are neighboring if they differ in a single word or term only [10]. The idea of word-level DP can be generalized to *token-level DP* for (discrete) sequences other than text, where two sequences are neighboring if they differ in a single token. As noted in Section 3.3.3.1, methods such as SynTF (Chapter 5) [465, 466] and related approaches [137, 138] that perturb words or tokens, or their embeddings, can be extended to entire texts by iterating over the entire sequence. In the following, we discuss such works that explicitly process entire sequences in order.

Feyisetan et al. [142] perturb word embeddings (GloVe [342]) directly using the PL mechanism, and find the word pertaining to the nearest neighbor as a substitute. Their approach fulfills metric privacy [65], where the underlying distance metric is defined between adjacent texts based on the sum of Euclidean distances between the word embeddings of both texts, and where two texts are adjacent if they contain the same

number of words. In a subsequent paper, Feyisetan et al. [141] employ hierarchical Poincaré embeddings in Hyperbolic space instead of embeddings in Euclidean space, which have been shown to better capture hierarchical relationships between words.

Yue et al. [487] propose SanText, which employs the Exponential mechanism similarly to our own SynTF method (cf. Chapter 5) to sample substitute tokens based on the token embedding distance. Its extension SanText$^+$ additionally makes a distinction between sensitive and insensitive vocabulary, and applies the Exponential mechanism only to sensitive words.

Limitations of Word-Level Differential Privacy. Word- or token-level DP has some limitations, and it is challenging to extend it to entire sequences or phrases: For instance, the privacy guarantees only hold for sequences of the same length, and in case a masking strategy is used following Adelani et al. [10], the guarantee is even further restricted to cover only sequences sharing the same structure anywhere but at the position of tokens deemed privacy-sensitive and considered to be obfuscated by the masking strategy. Moreover, due to the *group privacy* property of DP, when multiple tokens are replaced, the total privacy loss typically grows linearly in the number of replacements. From a linguistic perspective, the individual replacements often lead to texts that are incoherent and ungrammatical with a lack of fluency and variety.

For a more detailed discussion of the implications and limitations of word-level DP, we refer to the related work by the author and his co-authors [285].

Methods Based on Autoencoders. Bo et al. [46] propose an autoencoder architecture with gated recurrent unit (GRU) cells [75] and an autoregressive decoder which generates subsequent tokens conditioned on the encoded input and the previously generated tokens, thus making their model able to generate coherent, human-friendly text. At each step, the decoder models a score for each possible candidate token; however, instead of greedily choosing the token with the highest score, they sample the next output token using a two-set variant of the Exponential mechanism to achieve DP. Moreover, the model is trained using reinforcement learning with a reward function that encourages sampling of under-rated tokens to increase the variety of the generated text.

Faulty Differential Privacy Approaches. Krishna et al. [241] propose an autoencoder based differentially private text transformation method, short *ADePT*, where the Laplace or Gaussian mechanism is applied to the L^2-normalized latent representation vectors produced by the encoder. The problem with their implementation is that they calibrate

the noise according to the L^2 sensitivity, but use the Laplace mechanism which actually requires the L^1 sensitivity as pointed out by Habernal [174].

An issue with many methods that perturb the original word embeddings is that their nearest neighbor has a high chance to correspond again to the original word. Xu et al. [481] hence apply *Vickrey auction* [452] by also considering the second-nearest neighbor to the perturbed word embedding as to reduce the risk that sensitive words remain unchanged. However, note that the proposed Vickrey mechanisms [481, Algorithm 1 and 3] ensure that a word will *never* be replaced by itself, as the input word w_i is always excluded in the arg min operator that represents the nearest neighbor search. Since DP requires that any mechanism output that does occur for some input also occurs with non-zero probability for *any* input, we are not sure how the DP guarantee holds in this case.

3.3.5.2 Differentially Private Obfuscation for Audio Data

Qian et al. [358] propose a modified variant of their earlier *VoiceMask* framework [357] which applies DP to the reconstructed voice signal in a straightforward way by directly perturbing the samples in each frame using the Laplace mechanism. The authors then apply parallel composition [292, Theorem 4] across the frames in a recording to compute the overall privacy loss. Unfortunately, we cannot follow the argument why the frames would be uncorrelated and allow parallel composition, since speaker-dependent voice characteristics are consistent over time and manifest themselves in virtually every frame containing utterances.

Han et al. [180] adapt metric privacy [65] (see Section 2.2.3) to voice representations of speech data: For a given utterance, they extract a separate content representation and the x-vector [416] as voice representation. They then randomly choose an x-vector to substitute the extracted one using a variant of the Exponential mechanism. The speech signal is then synthesized from the unmodified content representation and the substituted x-vector. Note that this approach only applies DP to the x-vector that encodes the voice characteristics, but not to the content representation; therefore, this approach does not provide end-to-end DP guarantees that cover the whole speech obfuscation pipeline.

Shamsabadi et al. [405] also rely on x-vector substitution to obfuscate the speaker identity representations. However, they observe that speaker information may leak into content representations and hence propose to make the entire speech obfuscation pipeline differentially private by applying DP to content features as well: They use the Laplace mechanism in specially designed *pitch* and so-called *bottleneck* feature extractors, and randomly substitute x-vectors independently of the source utterance.

3.3.5.3 Differentially Private Obfuscation for Visual Data

Pixel-Based Mechanisms. A straightforward approach to differentially private image obfuscation works in pixel-space by perturbing the pixel values directly.

Fan [130] propose DP-Pix, a method which directly applies the Laplace mechanism (Example 2.14) to pixelized images. The pixelization is used to reduce the dimension of the image and works by grouping blocks of $b \times b$ pixels together, replacing each block with a pixel whose value is the average of the block's original pixels. Their DP guarantee holds for neighboring images that differ in up to m pixels. Subsequently, Fan [131] propose DP-Blur which extends on DP-Pix by applying a Gaussian blur filter to smoothen the pixelized image obtained by running DP-Pix on the input image.

Croft et al. [88] propose an alternative approach to DP-Pix [130] that directly obfuscates the raw image using the Exponential mechanism instead of the Laplace mechanism. As quality function, they use the structural similarity index measure (SSIM) [463] which estimates the perceived image similarity. However, this would involve enumerating all possible images in the entire output space; therefore, they reduce the complexity with a sliding window approach by moving over the image in strides of p pixels and applying the Exponential mechanism to the smaller $p \times p$ blocks individually, as well as quantizing the pixel intensities.

Mechanisms Based on Algorithmic Transformations. In an early work, Raafat et al. [362] apply DP to (facial) images by adding Laplace noise to the frequency components obtained by a fast Fourier transform (FFT) of the input images. They define a relaxed notion of adjacency for pairs of images if they differ in a block of $b \times b$ frequency components.

Another work by Fan [132] proposes DP-SVD which utilizes singular value decomposition (SVD) and achieves metric privacy [65] for images (or more precisely, for certain privacy-sensitive regions of interest) by applying the PL mechanism to the k largest singular values. It then reconstructs an obfuscated image based on the perturbed singular values. Both methods, DP-Pix and DP-SVD, are compared in a study by Reilly and Fan [368] together with two other differentially private methods, and demonstrated in an interactive framework (DP-Shield) by Saleem et al. [390].

Chamikara et al. [63] consider the problem of privacy-preserving face recognition, with the goal to "prevent leakage of the biometric features while identifying a person". Note that this objective appears to us as somewhat difficult to achieve in practice, since face recognition *is* based on biometric features. Their approach works by perturbing *eigenfaces*, i.e., eigenvectors of facial images that are used in the context of face recognition, using the differentially private Laplace mechanism. The eigenface perturbation mechanism is

applied both to the training data when learning a face recognition model and to the testing data when querying the trained model.

Mechanisms Based on Learned Representations. Croft et al. [87] synthesize obfuscated facial images from a generative neural network conditioned on several attributes such as identity, gender, and facial expression. They apply the Laplace mechanism to the identity representation at an intermediate layer of the network. Note that the model does *not* directly obfuscate existing images, but generates new synthetic facial images for identities that are known at training time, based on the attribute representations and the learned concepts from the training data.

Li and Clifton [260] obfuscate facial images by perturbing their representation in the latent space of a GAN model that produces synthetic images: For an input image, they find its latent vector representation in the latent space of a GAN model, i.e., a corresponding seed from which the GAN synthesizes an image close enough to the input. They then clip the representation to obtain a finite sensitivity and perturb the clipped representation using the Laplace mechanism. The perturbed vector is then fed as a seed into the GAN to re-synthesize an obfuscated version of the original input image.

In a later work, Croft et al. [89] employ a GAN encoder-decoder architecture to obfuscate the identity in facial images while preserving desired attributes such as pose or gender. They apply Laplace noise to the encoded image representations to achieve metric privacy [65]. Before the perturbation, an additional principal component analysis (PCA) step is applied to the encoded vectors to further reduce the dimensionality of the image representations, which leads to a lower sensitivity and hence a more favorable privacy-utility trade-off.

Tölle et al. [438] rely on conditional *invertible neural networks* (*INNs*) (see, e.g., [26, 27]). Obfuscation of an image works by forwarding it through the trained INN to obtain its latent representation, which is then clipped and perturbed using the Laplace mechanism. The obfuscated image is recovered by feeding the perturbed latent vector backward through the INN, optionally conditioned on desired attributes to change or preserve their manifestation in the reconstructed image.

Mechanisms Based on Disentangled Representations. Wen et al. [469] employ an autoencoder architecture with separate encoder networks to encode attribute and identity representations, respectively, together with a fusion network decoder that aims at reconstructing the input image from the representations. In the training phase, the model is trained to reconstruct the input image as well as to preserve its (disentangled) attribute and identity representations for the reconstructed image. Images are obfuscated by

feeding them through the trained network, where Laplace noise is added to the identity representations; the obfuscated image is then reconstructed from the unmodified attribute and the perturbed identity representations.

3.3.5.4 Differentially Private Obfuscation for Directional Data

DP has also been used for the obfuscation of individual locations as well as location traces. A main challenge with location traces is correlations, both spatial (e.g., between subsequent points) and temporal in nature (e.g., due to repeated daily commutes of a user). Due to the vast amount of works in this area, we refer to the surveys by Errounda and Liu [124] for a general overview of DP research for location and trajectory data, and by Miranda-Pascual et al. [308] for a more specific focus on trajectories. In the following, we discuss one particular set of methods that inspired our work on directional privacy in Chapter 7.

Geo-Indistinguishability. A popular line of research to obfuscate individual locations with DP guarantees is *geo-indistinguishability* by Andrés et al. [24]. More specifically, it fulfills a form of metric privacy [65] (cf. Section 2.2.3) adapted to planar location data. A related approach called (D, ϵ)-DP that protects locations within a protection radius D has been proposed by ElSalamouny and Gambs [120]. Common to these approaches is that they assume a flat coordinate system that corresponds to the two-dimensional Euclidean space, whose points could be specified by, e.g., Cartesian or polar coordinates. A popular mechanism to achieve geo-indistinguishability or (D, ϵ)-DP in Euclidean spaces is the PL mechanism [24, 65] which we discuss in Section 2.3.3.

Assuming a flat coordinate system may be sufficient in scenarios where locations are confined to smaller regions, such as cities. However, on a larger scale, approximating the curved surface of the globe through planar coordinates quickly becomes inaccurate. An example application that provides geo-indistinguishability to the web users is *Location Guard* by Chatzikokolakis et al. [66], a browser extension that obfuscates the users' locations. Since web users could be located anywhere on the globe, their implementation wraps the PL distribution around the sphere to respect the rather (approximately) spherical surface of the Earth. Our work on directional privacy [467] presented in Chapter 7 explores DP for spherical or *directional data* in more detail; in particular, we show that the wrapping technique is not optimal either.

Similar to word-level DP mechanisms for text (cf. Section 3.3.5.1), methods for individual locations, such as geo-indistinguishability, can be generalized to location traces in a straightforward manner by obfuscating each location in the sequence individually. A critical

review of geo-indistinguishability for location traces has been conducted by Primault et al. [353]: They found that obfuscated location data still comes with a high re-identification risk for points of interest that users have visited. Moreover, while a strong level of obfuscation provides better protection, it also degrades utility, so a good privacy-utility trade-off seems hard to achieve. Furthermore, they point out that most DP approaches for location traces only obfuscate locations, but not the corresponding timestamps; obfuscation of both locations and (periodic) time specifications is possible with our directional privacy mechanisms presented in Chapter 7.

3.4 Chapter Summary

Our review of related work showed some significant findings, which we summarize in the following.

Similarities Among Sequential Types of Data. Similar techniques are used in protection methods for text as well as audio and visual data: For de-identification of locally confined identifiers, a common approach is detection followed by masking or replacement of the sensitive segments. For obfuscation of pervasive identifiers, a common approach is to rely on private representations by first transforming (encoding) the data to a latent representation, then sanitizing this representation, and finally (approximately) inverse-transforming (decoding) the sanitized representation back to the original domain. In many cases, this is achieved by an encoder-decoder neural network architecture or signal processing together with adversarial learning, projections, or filtering to sanitize the representations.

Scope of the Defense Must Match the Identifier. When dealing with identifiers that are locally confined (e.g., named identifiers such as names or addresses in a longer piece of text), we can typically protect the identifier with local obfuscation methods such as simple masking approaches that preserve the remaining sections of the data and thus maintain utility. In the case of pervasive identifiers, however, local defenses would be insufficient to protect privacy, and masking the entire identifier would destroy utility; in this case, we hence require more sophisticated obfuscation methods that hide the sensitive information but preserves privacy-insensitive information for utility across the entire pervasive identifier.

Use Adaptive Attacker Model for Meaningful Results. While it is easy to fool a static (uninformed) adversary, in a realistic scenario, we should assume that the obfuscation

method is public (i.e., avoiding the *security/privacy through obscurity* anti-pattern), so the attacker can calibrate their attack to the obfuscation method, e.g., by creating their own instance of obfuscated training data. While we observed this in our own approaches proposed in this dissertation, this was also observed in related works on audio [419, 420, 421], image [366], and textual data [119].

Challenges of the Local Model. Noise in perturbed data affects both adverse and benign downstream tasks, hence it is often challenging to achieve a good privacy-utility trade-off. We made the same observation in our work in Chapters 5 and 6, where we proposed additional measures to push the privacy-utility trade-off in a favorable direction: For SynTF (Chapter 5), we introduce the *bigram overlap* in Section 5.3.1.2 which manipulates the scores of the Exponential mechanism's rating function to prefer differently spelled substitutions of terms. For DP-AAE (Chapter 6), we employ adversarial training to disentangle author- and content-specific information of the input sentences into separate representation vectors.

Some differentially private defenses discussed in this chapter also take additional measures, similar to our own approaches: SanText$^+$ [487] only applies the Exponential mechanism to a subset of sensitive tokens to mitigate utility loss. ER-AE [46] uses reinforcement learning to encourage sampling of under-rated but semantically similar tokens as a substitute for the original tokens. Adversarial training has been used to protect (i.e., suppress) sensitive information in differentially private representations for text [275], as well as in obfuscation mechanisms for images based on latent [89] or disentangled representations [469]. Many DP mechanisms for speech rely on separate content and speaker representations, such as speaker x-vectors [416], and obfuscate only the speaker representations while applying little to no obfuscation to the content representations [180, 405]. However, while methods with separate levels of obfuscation for privacy-sensitive and -insensitive components may achieve improved privacy-utility trade-offs in practice, this usually comes at the cost of degraded theoretical privacy guarantees.

Chapter 4

Methodology

As motivated in Section 1.1, our aim in this dissertation is to research and develop novel differential privacy mechanisms that are suitable for sequential and directional data, such as text and geolocations, respectively. We found that existing DP mechanisms typically work only with structured data such as numbers or numerical vectors and hence no or only few existing mechanisms were readily available for the specific data types we intend to handle; more specifically, no differentially private methods had been published for textual data, and popular approaches for geolocations assumed a flat coordinate system.

Therefore, in each of our studies presented in the following Chapters 5 to 7, we set our primary goal to construct new privacy mechanisms for sequential or directional data that provide DP guarantees, and our secondary goal to experimentally evaluate the new mechanisms' privacy-utility trade-offs in a realistic usage scenario.

Construction of New Privacy Mechanisms. On an abstract level, our primary goal consists of the following steps:

1. either adapt existing DP mechanisms or devise fundamentally new DP mechanisms for the specific use case or data type to be processed, and
2. formally prove the DP properties of the new mechanisms.

Note that for fundamentally new mechanisms, such as the directional privacy mechanisms presented in Chapter 7, we also need to provide a corresponding implementation to perform the experimental evaluation for our secondary goal.

Experimental Evaluation. Once a new mechanism has been devised, we proceed with our secondary goal which consists of evaluating its privacy-utility trade-off in a realistic usage scenario.

1. We define one or more utility tasks relating to a relevant use case that uses the targeted data type.

2. We establish the reference performance of the utility tasks based on the unobfuscated data.

3. We experimentally assess the performance of the proposed mechanism over a range of DP or RDP parameters (e.g., the privacy loss ϵ) based on the utility task on obfuscated data.

Where available, we consider existing DP obfuscation mechanisms that are suitable for the use case as baseline mechanisms and also assess their performance based on the utility tasks for further comparison.

Since sequential data covers a wide variety of domains, we chose to perform our experiments on textual data as an important and representative example domain. For directional data, we focus on spatio-temporal data (geolocations and periodic time specifications).

As we have described in Chapter 3, for sequential data, identifying information such as certain biometric identifiers can pervade large parts or the entire sequence. Therefore, in our experiments on textual data we also evaluate authorship attribution as an additional attack task on a biometric identifier to measure the performance of the proposed DP mechanisms as protective authorship obfuscation methods.

In case the attack and utility tasks require some form of training (e.g., since they are based on machine learning), we evaluate their success at achieving their designated goals on unseen test data. Additionally, we consider two levels of *adaptability* or *informedness* for the downstream tasks where applicable. We explain them first from the attacker's perspective:

The static (also: non-adaptive or uninformed) attacker may be given other original (i.e., unobfuscated) data samples in advance ("training data") to calibrate his attack, where we assume that the obfuscation method is unknown to the attacker.

The adaptive (also: informed) attacker may be given other data samples in advance ("training data") to calibrate his attack, where we assume that the obfuscation method is public.

Given some original data samples, an adaptive attacker hence is able to create his own instances of obfuscated samples by transforming the original samples using the public obfuscation method. This often results in a stronger attacker model, since the attack can be better calibrated to the characteristics of the obfuscated target data which might have easily fooled a static attacker. Therefore, an obfuscation mechanism that aims at providing reliable privacy guarantees in practice must *not* rely on the "security (*privacy*)

through obscurity" anti-pattern which assumes that the mechanism is unknown to the attacker. Similarly, we also evaluate a static and an adaptive setting for the utility task where the imaginary analyst who conducts the utility task is given only unchanged or already obfuscated training samples, respectively.

Chapter 5

SynTF: Synthetic and Differentially Private Term Frequency Vectors

One of the most prominent models to represent documents in many common text mining and information retrieval tasks is the *Bag-of-Words (BoW)* or *vector space model* where each document is represented as a vector, typically containing its term frequencies or related quantities. In this chapter, we therefore propose an automated text anonymization method called *SynTF* that produces synthetic term frequency (tf) vectors for the input documents that can be used in lieu of the original vectors. We evaluate our method on an exemplary text classification task and demonstrate that it only has a low impact on its accuracy. In contrast, we show that our method strongly affects authorship attribution techniques to the level that they become infeasible with a much stronger decline in accuracy. Other than previous authorship obfuscation methods, our approach is the first that fulfills DP and hence comes with formal privacy guarantees.

This chapter is based on the following publication [465] and its extended version [466]:

> Benjamin Weggenmann and Florian Kerschbaum: "SynTF: Synthetic and Differentially Private Term Frequency Vectors for Privacy-Preserving Text Mining". In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval* (*SIGIR '18*).

5.1 Introduction

For centuries, text has been used to convey information between human beings through books, letters, newspapers, and magazines. With the advent of the digital age, more and more textual data is being processed and analyzed by machines. Typical tasks include text classification, which is used in particular for spam filtering [388] and automated email routing [56], document retrieval [391], where indexed documents are retrieved and ranked

according to search queries, sentiment analysis [264], and a wide variety of other tasks in the information retrieval (IR) and text mining domains.

In many cases, it is desirable for an author that his writings stay anonymous. This could be the case if the textual data contains sensitive information about the author, for instance in search queries. Negative feedback from customer surveys might negatively impact business relations if the author or his company is known, and critical news or blog articles about a company (or government) might have severe (or fatal) consequences for the author of the article. In other areas, anonymity is required for compliance or legal reasons, e.g., in the selection of job candidates to eliminate discrimination. Furthermore, without anonymity people and data owners might feel reluctant to participate in surveys or to release their data. Offering anonymity might be a means to convince them to share their data in an anonymized form, which could then be used to perform evaluations and as training data for machine learning models.

Traditional sanitization approaches for free text include removing parts containing PII such as the author's name, or replacing it with a pseudonym (cf. Section 3.3.4.1). However, these methods are insufficient to protect the author's identity: As the famous Netflix de-anonymization attack [318] and other studies [92, 207, 364, 430] have shown, the *originator* of data can be *re-identified from the data itself*. We illustrate this in the case of the AOL search data release [34], where search queries of over 650,000 users were released for research purposes in 2006. The search logs were "anonymized" (in fact, only pseudonymized) by linking the queries to a numerical identifier instead of the actual username. After some investigation in the data, the New York Times eventually learned enough information about user 4417749, so they could re-identify her as Thelma Arnold, a 62-year-old widow from Lilburn, a city in Georgia.

Moreover, special *authorship attribution* methods allow attributing authorship of an anonymous or disputed document to its respective author. Such methods usually make use of stylistic features to identify or discriminate authors, as has been done with the statistic techniques in [315] to resolve the dispute of the Federalist Papers. Recently, more sophisticated methods have evolved that use statistical analysis and machine learning to tackle the problem; we refer to our discussion in Section 3.2.1.2 for an overview of methods. While these powerful methods are useful in the literary world and in forensics, they can often pose a threat to the privacy and integrity of authors of documents with potentially sensitive content.

Solution Approach. For many IR and text mining tasks including text classification, documents can be represented as Bags-of-Words (BoWs) or vectors in the vector space model [391]: To obtain a representation corresponding to the vector space and BoW

models, documents are transformed into feature vectors where each entry corresponds to a certain word in an underlying vocabulary. The process of this transformation is also called *vectorization*. Two common representations are *term frequency (tf)* vectors where each entry equals the number of occurrences of the corresponding term in the document, and *term frequency–inverse document frequency (tf–idf)* vectors which are derived from tf vectors by also taking the number of documents into account that contain the corresponding term. For more information, we refer to the IR book by Schütze et al. [402].

Since many IR and text mining algorithms rely on the vector space model, we propose a solution that targets this representation by producing synthetic tf vectors with DP guarantees which can be used as a substitute for the original tf vectors.

Contributions. More specifically, we make the following contributions:

- In Section 5.2, we propose *SynTF*, a differentially private method to compute anonymized, *synthetic tf vectors* for textual data that can be used as feature vectors for common IR and text mining tasks such as text classification.
- In Section 5.2.4, we give theoretical results on the DP properties of our method. We derive improved bounds for the privacy loss of our method and give a heuristic argument that DP on large (discrete) output spaces demands a large privacy loss if the result should fulfill a minimum usefulness requirement.
- In Section 5.3, we experimentally verify our method on a corpus of newsgroups postings: A benign, well-intended analyst wants to classify the documents into certain topics, whereas a malicious attacker tries to re-identify the author of these documents using authorship attribution techniques. The results show that our method has a much stronger impact on the attacker's than on the analyst's task.

Based on our motivation and results, we presume that the synthetic tf vectors (SynTF vectors) can be used in a multitude of text mining and IR tasks where the semantic similarity of documents is decisive. On the other hand, our method obliterates stylistic features that could otherwise reveal the identity and other privacy-sensitive information about the writer such as age or gender.

5.2 Synthetic Term Frequency Vectors

In this section, we first describe the intended usage scenario. We then take a closer look under the hood of authorship attribution techniques and derive the basic motivation

behind our SynTF method. Finally, we describe our method in detail and present its DP properties.

5.2.1 Usage Scenario

Consider a data processor that wishes to share sensitive training data for machine learning with a third-party analyst. Feature vectors are sufficient for most machine learning tasks since they are produced by the analyst in a preprocessing step anyway. Our method automatically creates anonymized feature vectors that can be shared with the analyst and that he can use in lieu of his own vectors.

In our present scenario, we are given a set of text documents such as email messages, job applications or survey results. The documents shall be analyzed by a (benign) third-party analyst, who wants to perform a typical text mining task such as text classification. Our aim is to prevent authorship attribution attacks as described above. Therefore, to protect the identity of the authors and prevent re-identification, we only provide the analyst with synthetic BoW feature vectors instead of the original documents. Email providers and search engines could share anonymized feature vectors of emails or (aggregated) search queries with advertising networks to provide personalized ads while protecting their users.

Attacker Model. The attacker is presented with a document of unknown authorship which has been written by one of several suspected authors. Her goal is to identify the document's actual author from the group of suspects. We assume that she has a set of similar reference documents from each suspect that she can use to help decide which suspect to assign the unknown document to.

We compare the attacker's capability to re-identify the authors on the original plaintexts as well as the anonymized feature vectors. We assume the attacker knows the dictionary, so she can convert the numbers in the feature vectors to a textual representation by repeating each word accordingly. This allows her to (partially) deduce more complex features beyond BoW, such as the *WritePrints* feature set which is often used in authorship attribution [6, 290]. As explained in the next Section 5.2.2, most of these features cannot be correctly inferred anymore, which is beneficial for our method as these are precisely the stylistic features (beyond BoW) that are exclusively exploited by our attacker.

5.2.2 Preventing Authorship Attribution

A popular feature set for authorship attribution has been described in the WritePrints method [6]. It includes the following types of stylistic features:

Lexical Counts of letters, digits, special characters, number of characters and words, etc.

Syntactic Frequency of function words, punctuation, parts of speech (POS) tags.

Structural Number and length of paragraphs and sentences, URLs or quoted content, etc.

Content Frequencies of words (BoW model).

Idiosyncratic Misspelled words.

For some features such as letters, words, digits, and POS tags, it also considers their bi- and trigrams, thus taking *order information* into account. These features have a strong capability to capture individual stylistic characteristics expressed by the writer of a text. For instance, one author might subconsciously prefer using the passive voice or past tense, so many verbs will end in an "ed" bigram, whereas another author might tend to use the present continuous or gerund which causes many "ing" trigrams.

Ordinary text mining and IR tasks such as classification typically only use content-level features which are often modeled and represented as tf vectors in the BoW model. Most of the stylistic features used for authorship attribution thus get *lost in vectorization*: In fact, the tf *vectors* by their very nature do not capture any structural information, and most syntactic features will be destroyed as well. Apart from the content (and idiosyncratic) features, however, we can still derive lexical features if the BoW vocabulary is known.

Since the attacker can still exploit the derived lexical features, we aim at disturbing them in a way that keeps the meaning or theme of a document intact, thus further allowing the classification task but impairing authorship attribution. Lexical features are mostly related to the spelling, therefore, our idea is to replace words in the input with words with similar meaning (synonyms) but different spelling to make the lexical features meaningless for the attacker. On the other hand, this will preserve the general theme of the text, so we hope that the impact is little on the classification task.

5.2.3 The SynTF Mechanism

Our goal is a differentially private anonymization method to derive synthetic feature vectors that keeps the theme of the represented document intact and at the same time prevents authorship attribution attacks. For performance and memory efficiency reasons, we require our method to preserve the sparseness in the tf vectors. Simply applying Laplace noise [117] or differentially private histogram publication methods [478] will fail this requirement, since they produce dense vectors. Our core idea is to take a word count entry for one term in the tf vector and probabilistically distribute it across all terms in the pre-defined vocabulary, using the Exponential mechanism (Definition 2.25) to achieve DP.

The probability of each term is determined according to its *similarity* with the original word. Word similarity can be expressed in various ways, which we will discuss later in Section 5.3.1.2 where we make concrete suggestions for suitable rating functions ρ.

DP presents a strong requirement for the method: Namely, *every* possible output must occur with non-zero possibility for *any* other input. This means that a statement on food preference can be processed to the same output as a conversation on politics, with non-zero probability. This has two implications: First, we must ensure that the probability of picking a term is always greater than zero, even for totally unrelated words. Second, it must be possible that two input texts of different lengths produce the same number of words in their resulting tf vectors. Therefore, we must also specify the output length. Note that this approach limits the number of entries that are changed from the original to the anonymized tf vector, so it keeps the sparseness of the resulting vector intact.

Algorithm Description. Since we work with textual documents, we adopt the vector space/BoW model where each document T is represented as feature vector $\mathbf{t} \in \mathbb{R}^L_{\geq 0}$ over some vocabulary $\mathcal{V}$ of size $|\mathcal{V}| = L$. The vocabulary could be derived, for instance, from a reference corpus of documents from a similar context as the target documents which shall be anonymized. We work in the *local model* (cf. Section 2.2.2) where each document is obfuscated independently, and hence assume $\mathcal{X} = \mathcal{Z} = \mathbb{R}^L_{\geq 0}$ for the input and output domains of our algorithm. This also implies that any two texts are considered as adjacent, which is the most strict and conservative way to define adjacency.

We will describe the SynTF approach for a single document T, but it is possible to anonymize an entire corpus simultaneously. The anonymization for a document T consists of two main phases:

Analysis We *vectorize* the document T to its feature vector $\mathbf{t} = (t_1, \ldots, t_L) \in \mathbb{R}^L_{\geq 0}$. Typically, $\mathbf{t}$ will be a tf or tf–idf vector over the underlying vocabulary $\mathcal{V}$. Next, we *normalize* $\mathbf{t}$ with respect to the L^1-norm to transform it into a *composition vector* $\boldsymbol{\theta}_\mathbf{t} := \mathbf{t}/\|\mathbf{t}\|_1$ whose entries can be interpreted as a probability distribution over $\mathcal{V}$.

Synthesis We repeatedly sample terms $v_1, \ldots, v_n$ from the distribution $\boldsymbol{\theta}_\mathbf{t}$ on $\mathcal{V}$. We run the Exponential mechanism on $\mathcal{V} \times \mathcal{V}$ for each v_i to pick a substitute output term $w_i \in \mathcal{V}$ with probability proportional to a *similarity rating* $\rho(v_i, w_i)$. Finally, we construct a synthetic tf vector $\mathbf{s} \in \mathbb{N}^L_{\geq 0}$ of length n by counting all the terms w_i.

Algorithm 2 illustrates the synthesis phase of our SynTF mechanism in pseudocode. For a discussion of suitable rating functions ρ, see Section 5.3.1.2. In our experiments, ρ will be bounded to $[0, 1]$, which implies that its sensitivity is $\Delta\rho \leq 1$.

Algorithm 2: SynTF term-frequency vector synthesis.

Input: Document composition vector $\boldsymbol{\theta}_{\mathbf{t}}$, desired output length n, privacy parameter $\epsilon > 0$, rating function $\rho : \mathcal{V} \times \mathcal{V} \to [0, 1]$.
Result: Synthetic tf vector $\mathbf{s} \in \mathbb{N}^{|\mathcal{V}|}$ with $|\mathbf{s}| = n$.

for $i \leftarrow 1$ **to** n **do** // produce output term-by-term
  $v_i \leftsquigarrow \mathrm{Cat}(\boldsymbol{\theta}_{\mathbf{t}})$; // sample word v_i
  $w_i \leftsquigarrow \mathcal{E}_{\epsilon,\rho}(v_i)$; // choose synonym for v_i
end
$\mathbf{s} \leftarrow \big(|\{i \in [1, n] : w_i = w\}|\big)_{w \in \mathcal{V}}$; // count synonyms

For completeness, we also state the following definition which is used in the code:

Definition 5.1 (Categorical distribution)**.** For an enumerable set $V = \{v_1, \ldots, v_k\}$ and associated probability vector $\mathbf{p} = (p_v)_{v \in V}$ with $\sum_{v \in V} p_v = 1$, the *categorical distribution*, denoted $\mathrm{Cat}(\mathbf{p})$, is defined on V through $\Pr[\mathrm{Cat}(\mathbf{p}) = v_i] = p_i$.

5.2.4 Differential Privacy Results

In this section, we will prove that our SynTF mechanism fulfills DP (Definition 2.3), which amounts to deriving an upper bound ϵ on its privacy loss.

We keep the previous notation where $\mathcal{V}$ is the vocabulary of size L, $\mathbf{t} = (t_1, \ldots, t_K)$ is the tf or tf–idf vector of the target document to be anonymized, and $\boldsymbol{\theta}_{\mathbf{t}} := \mathbf{t}/\|\mathbf{t}\|_1$ is the corresponding vector of probabilities. For each pair of words $v, w \in \mathcal{V}$, we have a similarity score $\rho(v, w) \in [0, 1]$. This score will be used in the Exponential mechanism, which outputs w on input v with probability

$$\pi_{v,w} := \Pr[\mathcal{E}_{\epsilon,\rho}(v) = w] = \frac{\exp\left(\frac{\epsilon}{2\Delta}\rho(v, w)\right)}{\sum_{w'} \exp\left(\frac{\epsilon}{2\Delta}\rho(v, w')\right)}.$$

Note that in the local model, we assume that *all* potential inputs are adjacent which is a very conservative interpretation of DP. This is used in the following lemma, which presents a (niched) counterpart to the known post-processing property [116, Proposition 2.1], to show that a convex combination of an ϵ-differentially private algorithm is again ϵ-differentially private.

Lemma 5.2 (Randomized Preprocessing)**.** *Given two independent randomized mechanisms $\mathcal{A} : \mathcal{X} \to \mathcal{R}(\mathcal{Y})$ and $\mathcal{B} : \mathcal{Y} \to \mathcal{R}(\mathcal{Z})$, we define their functional composition $\mathcal{B} \circ \mathcal{A} : \mathcal{X} \to \mathcal{R}(\mathcal{Z})$ as first sampling from $\mathcal{A}$ and using the resulting sample as input for $\mathcal{B}$. The composition $\mathcal{B} \circ \mathcal{A}$ is*

ϵ-differentially private provided that $\mathcal{B}$ is ϵ-differentially private where all *inputs $y, y' \in \mathcal{Y}$ to $\mathcal{B}$ are considered adjacent (that is, $d_{\mathcal{Y}}(y, y') \leq 1$).*

Proof. Define $\alpha_{x,y} := \Pr[\mathcal{A}(x) = y]$ and $\beta_{y,z} := \Pr[\mathcal{B}(y) = z]$ for all $x \in \mathcal{X}, y \in \mathcal{Y}, z \in \mathcal{Z}$. Then

$$\begin{aligned}\Pr[(\mathcal{B} \circ \mathcal{A})(x) = z] &= \Pr\left(\bigsqcup_{y \in \mathcal{Y}} [\mathcal{A}(x) = y] \wedge [\mathcal{B}(y) = z]\right) \\ &= \sum_{y \in \mathcal{Y}} \Pr[\mathcal{A}(x) = y] \cdot \Pr[\mathcal{B}(y) = z] \\ &= \sum_{y \in \mathcal{Y}} \alpha_{x,y} \cdot \beta_{y,z}.\end{aligned}$$

The first equality stems from enumerating, over $y \in \mathcal{Y}$, all possible ways to get output z on input x. The second equality is due to the fact that these possibilities are disjoint, and uses the independence between the two randomized mechanisms.

Fix any adjacent $x_1, x_2 \in \mathcal{X}$ and $z \in \mathcal{Z}$ and define the quantities $\hat{\beta}_z := \max_{y \in \mathcal{Y}} \beta_{y,z}$ and $\check{\beta}_z := \min_{y \in \mathcal{Y}} \beta_{y,z}$. Now

$$\begin{aligned}\frac{\Pr[(\mathcal{B} \circ \mathcal{A})(x_1) = z]}{\Pr[(\mathcal{B} \circ \mathcal{A})(x_2) = z]} &= \frac{\sum_y \alpha_{x_1,y} \beta_{y,z}}{\sum_y \alpha_{x_2,y} \beta_{y,z}} \\ &\leq \frac{\sum_y \alpha_{x_1,y} \hat{\beta}_z}{\sum_y \alpha_{x_2,y} \check{\beta}_z} = \frac{\hat{\beta}_z}{\check{\beta}_z} \leq e^{\epsilon},\end{aligned}$$

since the sums are convex combinations of $\beta_{y,z}$ and since both values of $y \in \mathcal{Y}$ that maximize/minimize $\beta_{y,z}$ are adjacent. □

Our main result is that Algorithm 2 is differentially private:

Theorem 5.3 (Differential Privacy of SynTF)**.** *Given a privacy parameter $\epsilon > 0$ and an output length $n \in \mathbb{N}$, our SynTF mechanism (Algorithm 2) fulfills ϵn-differential privacy.*

Proof. Each iteration (the body of the `for`-loop) consists of two steps: First, our algorithm samples one word v according to the probabilities in $\boldsymbol{\theta}_t$, which can be thought of as running a randomized mechanism $\mathcal{A}$ with the underlying categorical distribution. Second, it substitutes v with another word $w \in \mathcal{V}$ according to their similarity using the Exponential mechanism $\mathcal{E}_{\epsilon,\rho}$, which provides ϵ-DP. By the preceding Lemma 5.2, both steps combined are ϵ-differentially private. Since we iterate n times, the sequential composition theorem [116, theorem 3.16] yields ϵn-DP for the entire `for`-loop. Aggregating the synonym counts

is a simple post-processing step which keeps the privacy loss unchanged [116, Proposition 2.1], and we hence achieve ϵn-DP for Algorithm 2. □

5.2.4.1 Alternative Bound for the Exponential Mechanism

We can derive an alternative bound for the privacy loss of the Exponential mechanism by also considering the maximum change across *all outputs* for *fixed inputs* (in contrast to the sensitivity which tracks the maximum change across *adjacent inputs* for *fixed outputs*):

Theorem 5.4 (Alternative bound). *Let $\epsilon > 0$ be a privacy parameter and $\rho : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ be a rating function with sensitivity Δ where $|\mathcal{Z}| = L$. Let*

$$\bar{\Delta} := \max_{x \in \mathcal{X}} \max_{z, z' \in \mathcal{Z}} |\rho(x, z) - \rho(x, z')|.$$

Then the privacy loss $\ell(\mathcal{E}_{\epsilon,\rho})$ is bounded by $(\bar{\epsilon} + \ln \eta)$, where

$$\bar{\epsilon} = \epsilon \frac{\bar{\Delta}}{\Delta} \quad \textit{and}$$
$$\eta = \eta(\bar{\epsilon}, L) = \frac{e^{-\bar{\epsilon}/2} + L - 1}{e^{\bar{\epsilon}/2} + L - 1} < 1.$$

Proof. For any $x \in \mathcal{X}$ and $z \in \mathcal{Z}$, denote by

$$\pi_{x,z} := \Pr[\mathcal{E}_{\epsilon,\rho}(x) = z] = \frac{\exp\left(\frac{\epsilon}{2\Delta}\rho(x, z)\right)}{\sum_{z'} \exp\left(\frac{\epsilon}{2\Delta}\rho(x, z')\right)}$$

the probabilities that $\mathcal{E}_{\epsilon,\rho}$ outputs z on input x. Then for adjacent $x_1, x_2 \in \mathcal{X}$ and any fixed $z \in \mathcal{Z}$, we bound

$$\begin{aligned}
\frac{\pi_{x_1,z}}{\pi_{x_2,z}} &= \frac{\exp\left(\frac{\epsilon}{2\Delta}\rho(x_1, z)\right)}{\sum_{z'} \exp\left(\frac{\epsilon}{2\Delta}\rho(x_1, z')\right)} \cdot \left(\frac{\exp\left(\frac{\epsilon}{2\Delta}\rho(x_2, z)\right)}{\sum_{z'} \exp\left(\frac{\epsilon}{2\Delta}\rho(x_2, z')\right)} \right)^{-1} \\
&= \frac{\sum_{z'} \exp\left(\frac{\epsilon}{2\Delta}[\rho(x_2, z') - \rho(x_2, z)]\right)}{\sum_{z'} \exp\left(\frac{\epsilon}{2\Delta}[\rho(x_1, z') - \rho(x_1, z)]\right)} \\
&\leq \frac{1 + \sum_{z' \neq z} \exp\left(\frac{\epsilon}{2\Delta}[\overbrace{\rho(x_2, z') - \rho(x_2, z)}^{\leq \bar{\Delta}}]\right)}{1 + \sum_{z' \neq z} \exp\left(\frac{\epsilon}{2\Delta}[\underbrace{\rho(x_1, z') - \rho(x_1, z)}_{\geq -\bar{\Delta}}]\right)}
\end{aligned}$$

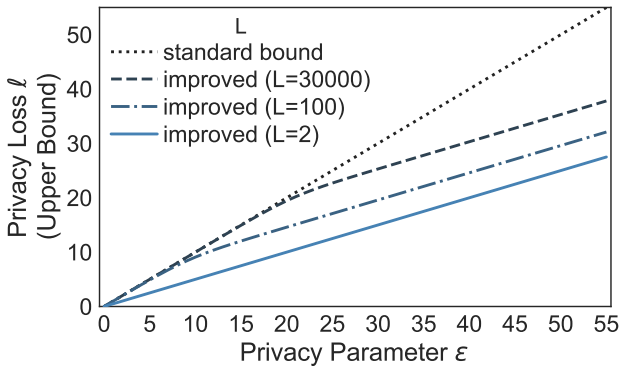


Figure 5.1: Standard and alternative upper bound $\epsilon + \ln \eta$ for the privacy loss $\ell(\mathcal{E}_{\epsilon,\rho})$ given different output space sizes L.

$$\leq \frac{1 + (L-1)\exp\left(\frac{\bar{\epsilon}}{2}\right)}{1 + (L-1)\exp\left(-\frac{\bar{\epsilon}}{2}\right)} = e^{\bar{\epsilon}} \cdot \underbrace{\frac{e^{-\bar{\epsilon}/2} + L - 1}{e^{\bar{\epsilon}/2} + L - 1}}_{=:\eta<1}.$$

The result follows by taking logarithms and observing that the numerator for η is strictly smaller than its denominator. □

Note that normally, we have $\bar{\Delta} > \Delta$ since the sensitivity Δ is restricted to *adjacent* inputs. The growth due to the factor $\bar{\Delta}/\Delta$ in $\bar{\epsilon} = \epsilon\bar{\Delta}/\Delta$ therefore typically exceeds the savings due to $\ln \eta < 0$, so the alternate bound $\bar{\epsilon} + \ln \eta$ would be *worse* than the original bound ϵ as derived in the standard DP proof for the Exponential mechanism [294]. However, if we consider all inputs as adjacent, and if ρ is symmetric in its arguments, then we will have $\bar{\Delta} = \Delta$ and $\bar{\epsilon} = \epsilon$, and thus the factor $\eta < 1$ will provide a real improvement over the original bound. This is the case in our algorithm:

Corollary 5.5 (Improved DP bound)**.** *Given a privacy parameter $\epsilon > 0$ and an output length $n \in \mathbb{N}$, our SynTF mechanism fulfills $\big((\epsilon + \ln \eta(\epsilon, L)) \cdot n\big)$-DP.* □

Proof. The proof is identical to that of Theorem 5.3 with the improvement that the Exponential mechanism provides $(\epsilon + \ln \eta(\epsilon, L))$-DP. □

We illustrate the effects of the factor $\eta(\epsilon, L)$ in Fig. 5.1: The original upper bound ϵ is the black-dotted line on top, the other lines show the improved upper bound $\epsilon + \ln \eta$ for different values of $L \in \{2, 100, 30\,000\}$. 30 000 is approximately the size of the vocabulary in some of our experiments. The effect of the improved bound increases with the privacy parameter ϵ, whereas large output spaces have a smoothing effect that dampens the improvement.

5.2.4.2 Tight Worst-Case Bounds

A major factor in the DP proof of Theorem 5.4 and Corollary 5.5 consists of bounding the privacy loss $\ell(\mathcal{E}_{\epsilon,\rho})$ for the Exponential mechanism used in Algorithm 2. This privacy loss is defined as the smallest upper bound for the logarithm of the fractions $\pi_{v_1,w}/\pi_{v_2,w}$, where $\pi_{v,w} \propto \exp\left(\frac{\epsilon}{2\Delta}\rho(v,w)\right)$ are the associated probabilities. The probabilities $\pi_{v,w}$ depend on the underlying vocabulary $\mathcal{V}$, the rating function ρ, and the privacy parameter ϵ, but do not take the documents $\mathbf{t}$ and $\mathbf{t}'$ into account. Therefore, we can compute the privacy loss

$$\ell(\mathcal{E}_{\epsilon,\rho}) = \max_{w\in\mathcal{V}} \ln \frac{\max_{v\in\mathcal{V}} \pi_{v,w}}{\min_{v\in\mathcal{V}} \pi_{v,w}} \tag{5.1}$$

in advance and independently of any documents to be anonymized once the parameters $\mathcal{V}$, ρ, and ϵ have been determined. Our SynTF method with privacy parameter ϵ and output length n thus in fact fulfills ℓn- instead of ϵn-DP where $\ell = \ell(\mathcal{E}_{\epsilon,\rho})$ is the privacy loss of the Exponential mechanism. This turns out to lead to huge gains in practice, reducing the privacy loss upper bound by almost 50% in our experiments (cf. Section 5.3.2).

To see that these bounds are tight, note that we can craft two input documents t_1 and t_2 that each consists of only a single word v_1 and v_2, respectively, where v_1 and v_2 are precisely those that maximize the fraction $\frac{\pi_{v_1,w}}{\pi_{v_2,w}}$ in Eq. (5.1) for the optimal $w \in \mathcal{V}$.

5.2.4.3 Relationship between Utility and Privacy Loss

We present the following theoretical results for the Exponential mechanism which suggest that in order to get "useful" outputs with a large output space, we need to choose a large privacy parameter ϵ in the order of $\ln|\mathcal{Z}|$, under the assumption that there are only few good outputs for each input.

Theorem 5.6 (Upper and Lower Bounds for Utility). *Let $\rho : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ be a rating function with $|\mathcal{Z}| \in \mathbb{N}$, and let Δ be the corresponding sensitivity. Take any fixed $x \in \mathcal{X}$ and denote by $\hat{\rho}_x$ and $\check{\rho}_x$ the maximum and minimum rating scores of any output for x, respectively. For a desired minimum rating $\tau \in [\check{\rho}_x, \hat{\rho}_x]$, split $\mathcal{Z}$ into $\mathcal{T} := \{z \in \mathcal{Z} : \rho(x,z) \geq \tau\}$ and $\overline{\mathcal{T}} := \mathcal{Z} \setminus \mathcal{T}$. Then the probability $\Pr\left[\mathcal{E}_{\epsilon,\rho}(x) \in \mathcal{T}\right]$ that the Exponential mechanism yields an element with score at least τ has lower and upper bounds*

$$\frac{|\mathcal{T}|}{|\mathcal{T}| + |\overline{\mathcal{T}}| \exp\left(-\frac{\epsilon c}{2\Delta}\right)} \leq \Pr\left[\mathcal{E}_{\epsilon,\rho}(x) \in \mathcal{T}\right] \leq \frac{|\mathcal{T}|}{|\mathcal{T}| + |\overline{\mathcal{T}}| \exp\left(-\frac{\epsilon \bar{\Delta}}{2\Delta}\right)},$$

where $c := \tau - \max_{z\in\overline{\mathcal{T}}} \rho(x,z)$ is the difference between τ and the next lower rating score, and $\bar{\Delta} := \hat{\rho}_x - \check{\rho}_x$.

Proof. For the lower bound, consider the inverse probability

$$\begin{aligned}\Pr\big[\mathcal{E}_{\epsilon,\rho}(x) \in \mathcal{T}\big]^{-1} &= \frac{\sum_{z\in\mathcal{Z}} \exp\left(\frac{\epsilon}{2\Delta}\rho(x,z)\right)}{\sum_{z\in\mathcal{T}} \exp\left(\frac{\epsilon}{2\Delta}\rho(x,z)\right)} \\ &= 1 + \frac{\sum_{z\in\overline{\mathcal{T}}} \exp\left(\frac{\epsilon}{2\Delta}\rho(x,z)\right)}{\sum_{z\in\mathcal{T}} \exp\left(\frac{\epsilon}{2\Delta}\rho(x,z)\right)} \\ &\leq 1 + \frac{|\overline{\mathcal{T}}|\exp\big(\frac{\epsilon}{2\Delta}(\tau - c)\big)}{|\mathcal{T}|\exp\big(\frac{\epsilon}{2\Delta}\tau\big)} \\ &\leq 1 + \frac{|\overline{\mathcal{T}}|}{|\mathcal{T}|}\exp\Big(-\frac{\epsilon c}{2\Delta}\Big).\end{aligned}$$

The upper bound is derived similarly. □

Solving for ϵ, these bounds lead to the following corollary:

Corollary 5.7 (Necessary and Sufficient Conditions on ϵ)**.** *Given a probability $p \in [0,1]$, with the notation from Theorem 5.6, we have the following necessary and sufficient conditions on ϵ for $\Pr[\mathcal{E}_{\epsilon,\rho}(x) \in \mathcal{T}] \geq p$:*

$$\epsilon \geq \begin{cases} \dfrac{2\Delta}{\bar{\Delta}} \ln\Big(\dfrac{p}{1-p} \cdot \dfrac{|\overline{\mathcal{T}}|}{|\mathcal{T}|}\Big) & \textit{(necessary condition)} \\ \dfrac{2\Delta}{c} \ln\Big(\dfrac{p}{1-p} \cdot \dfrac{|\overline{\mathcal{T}}|}{|\mathcal{T}|}\Big) & \textit{(sufficient condition)} \end{cases}$$

Note that for our SynTF algorithm, we have $\bar{\Delta} = \hat{\rho}_x - \check{\rho}_x \leq \Delta$. Hence for $p = 1/2$, the necessary condition becomes

$$\epsilon \geq 2\ln\left(\frac{p}{1-p} \cdot \frac{|\overline{\mathcal{T}}|}{|\mathcal{T}|}\right) = 2\ln\left(\frac{|\overline{\mathcal{T}}|}{|\mathcal{T}|}\right) = 2\ln\left(\frac{|\mathcal{Z}| - |\mathcal{T}|}{|\mathcal{T}|}\right).$$

Given a reasonable choice for the desired rating lower bound τ, the number $|\mathcal{T}|$ of "useful" outputs whose score is at least τ will be small. In the case of our SynTF mechanism, we can think of τ as a threshold for the rating function that distinguishes good alternatives for a given word from poor ones, and $|\mathcal{T}|$ would reflect the number of suitable substitutes (synonyms). If we assume $|\mathcal{T}|$ to be bounded by some constant, then $\epsilon \in \Omega(\ln|\mathcal{Z}|)$, that is, ϵ *needs* to grow logarithmically in the size of the output space $|\mathcal{Z}|$ in order to allow meaningful results.

5.3 Evaluation

In this section, we first describe our implementation of the SynTF mechanism along with associated parameters and our implementation choices. We then describe our experiment setup and report the evaluation results. Finally, we compare SynTF with a traditional information removal approach in the same experiment setup.

5.3.1 Algorithm Implementation and Parameters

We implemented a prototype of our SynTF algorithm in Python using the `SpaCy` package (`http://spacy.io/`) for text parsing functionality as well as the `numpy` and `SciPy` packages [215, 456] for (vector) computations. Besides the explicit parameters mentioned in Algorithm 2, there are various implementation-dependent parameters that influence SynTF in its different stages. We now describe these parameters and corresponding implementation choices.

5.3.1.1 Vocabulary and Vectorization

We build a custom vectorizer to extract the vocabulary from the training or a given reference corpus, and to subsequently transform documents to their BoW tf vectors. We can specify several special options: Firstly, we can choose, for each extracted word, to keep its spelling as-is, to change its morphology through lemmatization, or to convert it to lower case. Secondly, we can instruct the vectorizer to include additional terms that are similar or synonymous to the actually extracted words, as to provide a greater choice of candidates for replacing a word with a suitable synonym but hopefully with different spelling to disturb lexical authorship attribution features. Our implementation uses the synonyms provided by WordNet's synsets. We remove stop words and numbers by default.

5.3.1.2 Similarity Rating Function

We now describe the rating function $\rho(v, w)$ that expresses the suitability of a substitute term w for an input term v. One fundamental technique are *word vectors* or *embeddings* which are dense vector representations of words in a real vector space. They are commonly derived with the intention that similar words have embeddings in the vector space that are nearby. We can therefore compute the similarity between two words simply and efficiently as *cosine similarity* between their corresponding word vectors. Two recent models to derive word vectors that achieve high accuracy in word similarity and analogy benchmarks are "word2vec" [305, 306] and "GloVe" [342].

Eroding Stylistic Features with the Bigram Overlap. As we saw in Section 5.2.2, features such as the frequency of certain words and character n-grams often make an essential and decisive contribution to authorship attribution methods. Suppose we can choose a substitute for a given input term from a set of candidates with comparable similarity ratings. Then to best prevent the attack, it is beneficial to pick the candidate that differs most in spelling from the input in order to obscure our word and n-gram frequencies. We can achieve this by including the (normalized) Levenshtein or n-gram distance in the rating function for the terms. Note that care must be taken to weigh this appropriately – a too strong preference for differently-spelled substitutes will often pick completely different words that also have a different meaning from the original word, thus also negatively affecting the utility.

We have implemented the word similarity rating function as

$$\rho(v, w) := \cos(v, w) - sB(v, w),$$

where $\cos(v, w)$ is the cosine similarity between the corresponding GloVe [342] word vectors, and $B(v, w) \in [0, 1]$ is the *bigram overlap*, i.e., the proportion of matching letter bigrams in v and w. The scaling factor s determines if and how strongly the bigram overlap affects the rating. As an optimization, we precompute the word similarity ratings and probabilities for the Exponential mechanism for the entire vocabulary, which yields a significant performance boost.

5.3.2 Experiment Description

In this section, we describe the context and setup of our evaluation.

Dataset. We perform a series of experiments with our algorithm on the "20 newsgroups" dataset[1]. It comprises almost 19,000 postings from 20 different newsgroups, and comes with predefined train (60%) and test (40%) sets which we use throughout our experiments. For the text classification task, a label is provided for each message indicating the corresponding newsgroup. For the authorship attribution task, we extracted the "From" field in the header of each message and use it as the author identifier. Note that we strip header and footer data before performing the actual classification and identification tasks as to make them more realistic.

[1] http://qwone.com/~jason/20Newsgroups/

Table 5.1: Attack scenarios with minimum *per author* numbers for active groups and train/test messages in the dataset.

Scenario	Suspects	#Groups	#Train	#Test
Top 5/Any	Top 5	≥ 1	≥ 35	≥ 17
Top 10/Any	Top 10	≥ 1	≥ 28	≥ 9
Top 5/Multi	Top 5	≥ 2	≥ 29	≥ 9
Top 10/Multi	Top 10	≥ 2	≥ 21	≥ 8

Attack Scenarios. After filtering out missing and ambiguous identifiers, we count 5735 authors, but the majority provides insufficient training samples (below 20 for 5711 authors) for properly fitting a model. We therefore evaluate the attack only for the "top" authors with the largest number of messages in the dataset. Since the number of candidate suspects from which the correct author has to be determined also can influence the authorship attribution performance, we evaluate the attack for the *top 5* and *top 10* authors. Table 5.1 provides the number of train and test messages per author.

Another issue with the dataset is that some users are active in only a single newsgroup, in which case knowledge of authorship (attack) implies knowledge of the targeted newsgroup (utility). We therefore devise two subsets of authors:

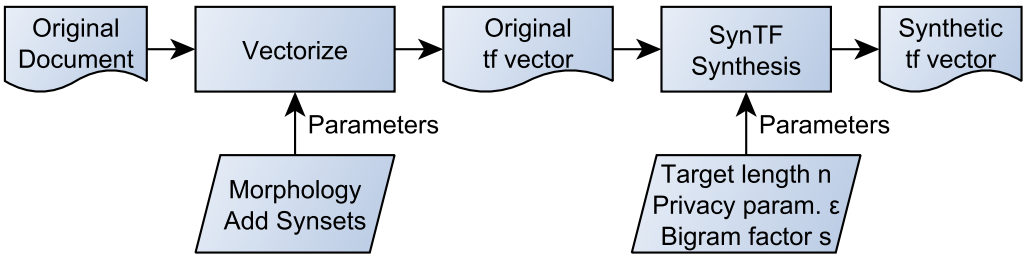
Any Each suspect author can have postings in any number (one or more) of newsgroups.

Multi Each author must be active in *at least two* different newsgroups.

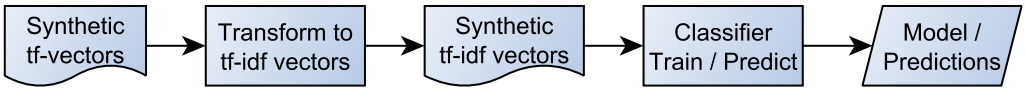
The idea of the "Multi" group is to reduce the similarity between the attacker's and analyst's tasks to allow a clearer distinction when evaluating the impact of our anonymization technique.

Processing Pipeline. All documents traverse a processing pipeline that can be broken down into three parts: For each document, the main SynTF pipeline (Fig. 5.2a) first produces a synthetic tf vector (cf. Section 5.2.3). It can be influenced by a number of parameters as described in Section 5.3.1. Next, the synthetic tf vectors traverse the analyst's text classification pipeline (Fig. 5.2b) and the attacker's authorship attribution pipeline (Fig. 5.2c) to measure the prediction performance for each task. In both cases, we evaluate a multinomial naïve Bayes classifier and a linear SVM. We perform 10 runs of the entire pipeline (anonymization + evaluation) for each combination of parameters to reduce fluctuations and get stable results.

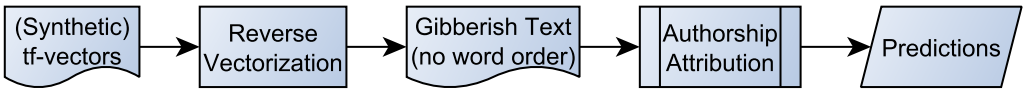
The analyst (cf. Fig. 5.2b) first transforms the tf vectors to tf–idf vectors which are commonly used in classification tasks. He then trains a classifier with the *training* subset of the dataset, and subsequently uses it to predict the newsgroups for the *test*



(a) The main SynTF processing pipeline, including parameters.



(b) Subsequent text classification pipeline.



(c) Subsequent authorship attribution pipeline.

Figure 5.2: Processing pipelines for the main SynTF mechanism and subsequent analyst and attacker tasks.

subset. We implement the classification in Python based on `scikit-learn` [340], using its `MultinomialNB` classifier with smoothing ($\alpha = 0.01$), and its `LinearSVC` classifier with default parameters ($C = 1$).

For the attack as depicted in Fig. 5.2c, we make use of the *JStylo* authorship attribution framework [290]. It supports several extended feature sets such as *WritePrints* [6]. WritePrints includes additional stylistic features (cf. Section 5.2.2) on top of the usual BoW that have to be extracted from full texts. However, since the attacker only gets synthetic tf vectors and not full texts, she first converts the numbers in the tf vectors to text by repeating each word accordingly, which allows at least partial deduction of WritePrints features ("reverse vectorization" in Fig. 5.2c).

Note that the full WritePrints feature set contains a virtually endless number of features and severely degrades performance (speed). Furthermore, the authors of [290] have shown that despite its title, a limited version of WritePrints even outperforms the full version in terms of accuracy, which we could confirm in own experiments. Therefore, we keep the default JStylo configuration with the *WritePrints (Limited)* feature set. JStylo builds on the Weka machine learning library. We use its `NaiveBayesMultinomial` classifier with Laplace smoothing and its `SMO` SVM classifier with linear kernel and $C = 1$ by default.

Table 5.2: Evaluated and optimal SynTF parameters.

Parameter	Values	Description
morphology	lemma	Lemmatize words.
	lower	Convert words to lower case.
	orth	Leave spelling unchanged.
synsets	true/false	Extend vocabulary with additional synonyms from WordNet.
s	0, 0.1, 0.2, 0.3, 0.4	Impact factor of letter bigram overlap on rating function ρ.
n	100, 150, 200	Length of output vector (words).
ϵ	35–55 (47.5), effectively 25.4	Privacy parameter (step size 2.5). Effective loss ℓ, cf. sec. 5.2.4.2.

Finding Optimal Parameters. We perform a grid search over the SynTF parameters listed in Table 5.2 to find "optimal" parameters in the sense that they should simultaneously strongly affect authorship attribution but mostly leave classification into newsgroups unaffected. As a metric to find these optimal settings we use the difference between the relative performance impacts on utility and attack: Given parameters $\mathbf{p}$, denote by $\beta_U(\mathbf{p})$ the relative performance of the analyst's classification task (measured as F_1 score), and similarly denote by $\beta_A(\mathbf{p})$ the relative performance of the attacker's task. Then the optimal parameters are $\mathbf{p} = \operatorname{argmax}_{\mathbf{p}}(\beta_U(\mathbf{p}) - \beta_A(\mathbf{p}))$. Since we want them to equally cover all four attack scenarios, we find optimal parameters that maximize the *minimum* difference $\beta_U(\mathbf{p}) - \beta_A(\mathbf{p})$ over all attack scenarios. Furthermore, we perform 10 runs of the anonymization–evaluation process for each combination of parameters to reduce fluctuations and get stable results.

5.3.3 Discussion of Results

After running the evaluation, we found the optimal parameters highlighted in Table 5.2 with privacy parameter $\epsilon = 47.5$. However, our tight bounds analysis (cf. Section 5.2.4.2) shows that the effective privacy loss $\ell(\mathcal{E}_{\epsilon,\rho}) \approx 25.4$ is only about half as large. Table 5.3 provides exemplary performance figures in the "Top 10/Any" scenario for both topic classification and authorship attribution.

Figure 5.3 depicts the relative performance between utility (green lines, left y-axis) and attack (red lines, right y-axis) in the different stages of SynTF. The bottom x-axis indicates the privacy parameter ϵ, with the corresponding effective privacy loss values $\ell(\mathcal{E}_{\epsilon,\rho})$ on the top. The dotted, dashed, and solid lines mark the utility and attack performances with the original (plaintext), vectorized, and synthetic data, respectively, where we used the

Table 5.3: Evaluation results (Top 10/Any).

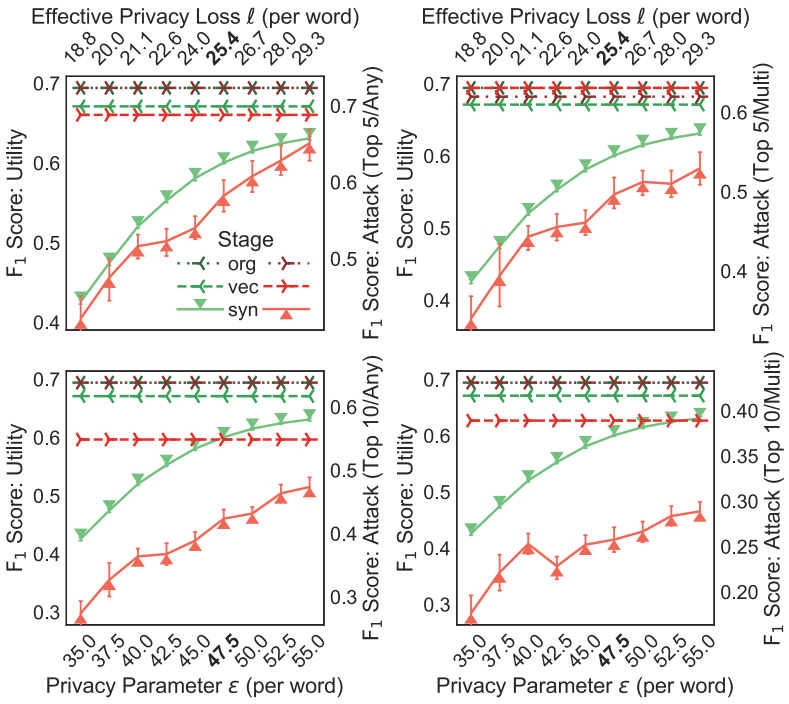
	Utility			Attack			Gain
Method	F_1	P	R	F_1	P	R	ΔF_1
none (original)	0.69	0.71	0.70	0.64	0.71	0.63	0.06
SynTF abs.	0.60	0.61	0.61	0.42	0.44	0.43	0.18
scrubadub abs.	0.64	0.65	0.65	0.57	0.63	0.57	0.06
SynTF rel.	87%	86%	87%	66%	61%	69%	20%
scrubadub rel.	92%	92%	92%	90%	88%	91%	02%

optimal parameters for vectorization and synthesis as mentioned above. A positive gap with the green above the red line shows how much the attack is more affected than utility.

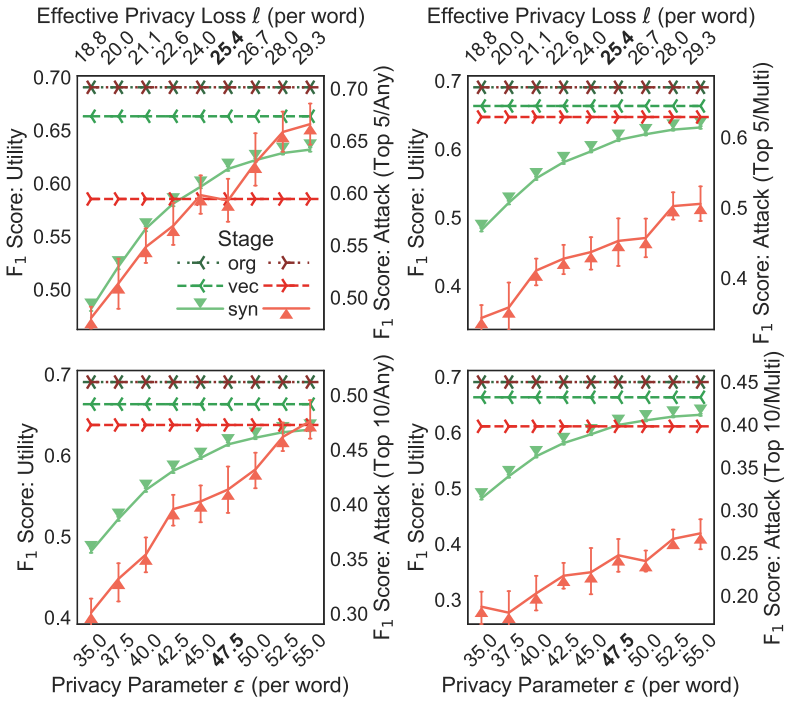
We observe that the vectorization already affects the attack more due to the loss of structural and syntactic features, except in one case (Top 5/Multi). Note that the size of the (positive) gap between the green and red lines indicate the analyst's gain over the attacker in terms of the relative performance of the corresponding stage of the anonymization. Obviously, both utility and attack suffer with a decreasing privacy parameter ϵ. However, in most cases, the gap between analyst and attacker is even higher than after vectorization, which indicates a growing advantage for the analyst. Furthermore, it shows that our SynTF mechanism successfully impairs authorship attribution while having only a mild effect on the classification task.

Impact of Attack Scenarios. Comparing the four scenarios with respect to the gap size, we make the following deductions: As expected, authorship attribution quickly becomes harder with an increasing number of suspect authors. Similarly, excluding authors who are active in only one newsgroup widens the gap, as we can see when going from the "Any" to the "Multi" scenarios. This indicates that our method is even more effective when the benign and malicious tasks are actually based on *distinct* problems.

Impact of Parameters from Table 5.2. A key factor in the success of our method is the letter bigram overlap B in the rating function ρ. Its effect of preferring synonyms with different spelling improves the capability of our method to prevent authorship attribution attacks. We illustrate this effect depending on the bigram overlap factor s in Fig. 5.4: Without bigram overlap ($s = 0$), the attacker has an advantage in all "Top 5" scenarios (red bars). Only when $s \geq 0.3$, we see a shift of power in favor of the analyst (green bars). In the "Top 10" scenarios, the analyst enjoys an advantage even without the bigram overlap, but we can roughly double his advantage if we choose the optimal value $s = 0.3$.



(a) Multinomial naïve Bayes.



(b) Linear SVM.

Figure 5.3: Relative performance of analyst (green) and attacker (red) in different attack scenarios and stages of the SynTF process (org: original data, vec: tf vectors, syn: SynTF vectors).

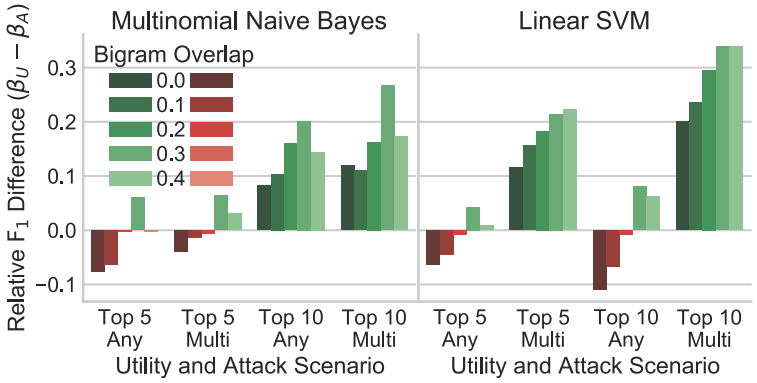


Figure 5.4: Impact of letter bigram overlap factor s.

Regarding morphology, observe that the use of upper and lower case letters is a stylistic feature that can pose a clue for authorship attribution but barely has any relevance for topic inference. Therefore, transforming all words to lowercase affects the attacker more than the analyst. Lemmatization strips off word endings and hence reduces the attacker's information on writing style further, but it also has an impact on classification since the meaning can change between a word and its lemma. Still, in terms of our definition of "optimal" parameters, using lemmatized words gave the best relative performance gain for the analyst, indicating that the lost word endings are more severe for the attack.

Other parameters are less insightful: Increasing the output length will help increase both tasks' performance, however, the gain becomes less for larger output lengths. Moreover, the inclusion of additional synonyms in the vocabulary did not provide any benefit.

SVM Anomaly. We observe one anomaly in the "Top 5/Any" scenario for the SVM. Apparently, vectorization already causes a drastic reduction of the attack performance. However, for $\epsilon \geq 45$, going from vectorized to synthetic vectors *increases* the attack performance. This is unexpected since the information *lost in vectorization* will not be restored by the synthesis process. Our current hypothesis is that the SVM might overfit on the vectorized training data, causing poor predictions on the vectorized test data, and the randomness in the synthesis step in turn acts as regularization.

5.3.4 Comparison with Scrubbing Methods

We run the open source *scrubadub* (`http://scrubadub.readthedocs.org/`) tool on the 20 newsgroups dataset to remove PII and evaluate the utility and attack performance in our scenarios. Figure 5.5 shows a comparison of the results with our SynTF method and optimal parameters. The results indicate that our method outperforms the scrubbing

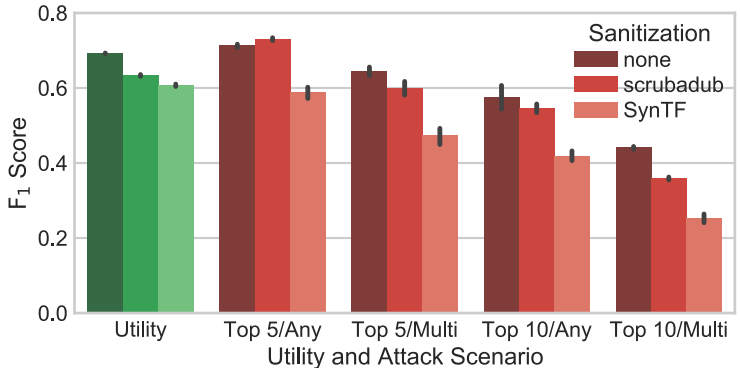


Figure 5.5: Comparing SynTF and traditional data removal.

technique in preventing the attack in all four attack scenarios, at a comparable level of utility. For instance, in the "Top 10/Any" scenario listed in Table 5.3, SynTF achieves an F_1 score of 0.60 for classification, where scrubadub is slightly better with 0.64, down from 0.69. For the attack, however, scrubadub drops from 0.64 to 0.57, whereas SynTF manages to more than triple the reduction and push the attacker's performance down to 0.42.

5.4 Comparison with Related Work

Authorship Obfuscation. To the best of our knowledge, most works on authorship obfuscation that appeared before SynTF proposed manual methods or machine translation as discussed in Section 3.3.4.2, also without providing any DP guarantees. On the face of it, such methods initially appeared to provide some protection against authorship attribution; however, manual methods are cumbersome to implement by the users [219, 290] and can still be detected with high accuracy [11], whereas machine translation has been shown to be ineffective [51, 58]. Furthermore, both approaches only prevent authorship attribution with respect to a *specific* reference corpus with other authors. While our SynTF method does not produce human-readable texts, it requires no manual changes to the documents, and its protection is independent of a reference corpus.

Moreover, we also evaluate our method in an *adaptive adversary* model where the attacker is able to recalibrate his authorship attribution attack to the obfuscation method, instead of the weaker static adversary model. While simple obfuscation methods may be sufficient to fool a static adversary, e.g., by always changing documents from one author to the style of another, this could be detected by an adversary that is able to adapt his attack to the deterministic changes made by the simple obfuscation technique.

De-Identification. De-identification (also called *masking* or *scrubbing*) methods as discussed in Section 3.3.4.1 aim at removing sensitive or personally identifiable information (PII) that is stated explicitly from textual documents. Prominent examples are provided by the 18 HIPAA identifiers [327] (cf. Section 1.1.2.1) and include, e.g., the name, address, or phone number of an individual.

While it is clear that this kind of information must be removed to protect the privacy of the subjects mentioned in the document, our experiments in Section 5.3.4 show that de-identification based on scrubbing provides no adequate protection for the document's author, although this is often critical, as in the case of complaint letters or patient records to protect the privacy of the treating physician. Moreover, we found that publications on these methods typically only evaluate their methods' ability to identify and remove all pieces of PII in the text (cf. the survey by Uzuner et al. [448]). We have not seen any evaluation on the impact of scrubbing on further processing with text mining techniques such as document classification, and more importantly, we have not found an evaluation of whether and to what extent these methods prevent authorship attribution techniques.

Differential Privacy. DP has been successfully applied to a wide range of problems from simple statistical functions to machine learning. The survey by Dwork [115] provides a good overview of some earlier results. It is commonly used to provide *aggregate* statistics, that is, multiple records are combined into one result. A good example is RAPPOR [123], which allows the collection of anonymized user statistics even over time.

However, releasing aggregate information only allows inferences on an entire population, whereas we want to classify each document individually. Releasing *individual* data with an ϵ comparable to aggregating mechanisms causes too much noise for individual records as it masks any difference (topic, sentiment, etc.) between two inputs and hence prevents any utility. The issue is well-known in the literature and has been observed, e.g., in the context of locations [24, 273], graphs [389], and recommender systems [272].

Approaches to mitigate this issue typically involve relaxing the privacy- or adjacency-definition [24, 65, 186]. Andrés et al. [24] circumvent the issue for location data by generalizing DP to metrics [65]. For graphs, Hay et al. [181] define two variants of DP, namely *node* and *edge* privacy, where two graphs are considered adjacent if they differ either in an entire node (including its edges) or in just a single edge, respectively. According to Kasiviswanathan et al. [226], most works focus on the strictly weaker edge privacy since it is harder to create node private algorithms providing good utility with a comparable privacy loss. For instance, Sala et al. [389] revert to edge privacy for sharing graphs and obtain usable results with $\epsilon = 100$ *per edge* (instead of *per node*). In comparison, our SynTF mechanism achieves a privacy loss of only 25.4 *per word* in the output (instead of *per*

document).

5.5 Chapter Summary

In this chapter, we have presented SynTF, a novel approach to produce anonymized, synthetic tf vectors which can be used in lieu of the original tf vectors in typical applications based on the vector space model. Our method produces sparse vectors which are favorable regarding performance and memory efficiency. We have proved that our method fulfills DP which currently serves as a "gold standard" for privacy definitions. Since our method anonymizes each text individually, it can be used locally at the data source to anonymize documents on-premise before collection, e.g., to obtain anonymized training data for machine learning or provide personalized ads based on anonymized emails or search queries.

Although our method requires a large ϵ to get reasonable utility, we provide evidence that this is necessary: First, we want to be able to analyze records independently of each other, thus the anonymization must *not* hide the influence of individual records in the result. Second, we have derived a necessary condition on the privacy parameter ϵ for the Exponential mechanism indicating that it must grow logarithmically in the size of the output space when high utility is required but only a limited number of "good" outputs are available. To further address the issue, we have derived alternative bounds on the privacy loss of the Exponential mechanism, which in our case provide a substantial reduction of almost 50%.

We have performed an extensive evaluation of SynTF on the 20 newsgroups dataset and analyzed the influence of different parameters. Our results indicate that it effectively prevents authorship attribution while facilitating tasks such as classification (utility). In contrast, our experiments show that traditional scrubbing methods are insufficient at preventing authorship attribution attacks.

Chapter 6

Differentially Private Variational Autoencoders

In this chapter, we tackle anonymization of textual data and propose an end-to-end differentially private variational autoencoder architecture. Unlike previous approaches that achieve DP on a per-word level through individual perturbations, our solution works at an abstract level by perturbing the latent vectors that provide a global summary of the input texts. Decoding an obfuscated latent vector thus not only allows our model to produce coherent, high-quality output text that is human-readable, but also results in strong anonymization due to the diversity of the produced data. We evaluate our approach on IMDb movie and Yelp business reviews, confirming its anonymization capabilities and preservation of the semantics and utility of the original sentences.

This chapter is based on the following publication [468]:

> Benjamin Weggenmann, Valentin Rublack, Michael Andrejczuk, Justus Mattern, and Florian Kerschbaum: "DP-VAE: Human-Readable Text Anonymization for Online Reviews with Differentially Private Variational Autoencoders". In *Proceedings of the ACM Web Conference 2022* (*WWW '22*).

6.1 Introduction

The Internet has paved the way for many online platforms allowing individuals to share their opinions about various products and services. The availability of such reviews not only helps prospective customers, patients, or employees to make informed decisions about their next buys, visits, or to evaluate their next potential employers, but also allows the business owners to analyze the provided feedback to gain insights into how their products, services, or brand image can be improved. To a large degree, such analyses can be automated, e.g., using sentiment analysis, text mining, and other text-based data science or machine learning techniques.

While many such platforms allow users to post their reviews "anonymously", i.e., under a pseudonym without *directly* revealing their identity, the absence of explicit identifiers is often not enough to provide true anonymity: In many scenarios, users can be re-identified based on metadata or the data itself, e.g., through linkage attacks [92, 93, 207, 318, 364]. This is particularly critical for textual data which represents a rich source of information: Users can be identified based on their pseudonymized search logs [34], and even their *writing style* alone may be sufficient to de-anonymize them through modern authorship attribution techniques [127, 364, 376, 409, 422] (cf. Section 3.2.1.2).

This may entail undesirable risks and consequences, ranging from, e.g., retaliatory actions and legal disputes for publicly criticizing businesses on Yelp or Google Maps [195], disclosure of patient identities and their sensitive personal information [57], over sanctions from the employer to potential lawsuits in the millions for critical reviews on sites like Glassdoor [323]. Users may hence feel reluctant to provide their honest feedback for fear of retaliation [371, 398], which also concerns internal surveys [414]. To convince them otherwise, it is thus necessary to develop methods that protect the anonymity of the reviewers while preserving the quality and content of the original data, and ideally meet formal privacy guarantees such as DP [117] which is widely regarded as the gold standard of privacy protection.

Problem. While many DP mechanisms are readily available for structured data such as numbers or vectors, it is not always easy to apply DP to unstructured data such as text, which comes in varying lengths and with different ways to express the same idea. Existing DP approaches for text work on a *per-word level* (cf. Section 3.3.5.1) [46, 138, 141, 142, 465, 466], thus obfuscating the original input while keeping it statistically relevant overall. However, each of these approaches has one or more undesired drawbacks: the output is a vector or BoW representation and thus not human-readable, the produced text is incoherent due to the words being perturbed individually, the DP guarantees only apply to texts of the same length, and/or the privacy loss ϵ grows linearly with the length of the output.

Proposed Solution. We propose a radical approach to completely rewrite a given sentence instead of randomizing words individually. Our approach relies on a VAE network [233, 375] where we constrain its probabilistic encoder to facilitate *differentially private latent sampling*. The noisy latent representations serve as a global blueprint for the decoder and thus allow our resulting *DP-VAE* architecture to reconstruct coherent outputs. Besides text, it can handle a variety of data formats, thus providing an end-to-end differentially private obfuscation mechanism for generating private synthetic data. In Section 6.3, we describe our core DP-VAE architecture and derive its privacy properties in

terms of local RDP [309].

To further push the privacy-utility trade-off in our favor, we augment our core method with adversarial training to disentangle the latent representations into separate content and style vectors [213], representing the semantics and author-specific attributes, respectively. We thus obtain a *differentially private adversarial autoencoder (DP-AAE)* which allows us to approach text anonymization as a style transfer task where we change the specific style of an author by applying DP selectively to the latent author representation while preserving the content representation. We describe a scenario about online reviews in Section 6.4, where we introduce two novel anonymization architectures based on our DP-VAE architecture and its adversarial extension with disentangled latent representations.

In Section 6.5, we conduct experiments on the IMDb movie and Yelp business reviews datasets, measuring the anonymization performance, the preservation of sentiment and semantics, as well as the language quality of the transformed texts. Our results demonstrate the effectiveness of our methods, which substantially mitigate the risks of reviewers being re-identified through authorship attribution attacks, while the sentiments of the reviews are preserved.

6.2 A Primer on Variational Autoencoders

In this section, we give a brief introduction to variational autoencoders (VAEs) as presented by Kingma and Welling [233]. For a more detailed explanation, we refer to the tutorials by Doersch [110] or Kingma and Welling [234].

VAEs model the actual data distribution $p(\mathbf{x})$ as a generative random process through a parametrized family of distributions $p_\theta(\mathbf{x})$,

$$p(\mathbf{x}) \approx p_\theta(\mathbf{x}) = \int_{\mathbf{z}} p_\theta(\mathbf{x}|\mathbf{z}) p_\theta(\mathbf{z}) \, \mathrm{d}\mathbf{z}, \tag{6.1}$$

i.e., $\mathbf{x}$ is generated conditionally with likelihood $p_\theta(\mathbf{x}|\mathbf{z})$ based on a latent variable $\mathbf{z}$, which in turn follows a prior distribution $p_\theta(\mathbf{z})$.

The goal of a VAE is to learn the parameters θ from the training data. To make training feasible, we need a *recognition model* $q_\phi(\mathbf{z}|\mathbf{x})$ to approximate the true but generally intractable posterior $p_\theta(\mathbf{z}|\mathbf{x})$. The training objective then is to jointly optimize both θ and ϕ to simultaneously maximize the log-likelihood of the data $\log(p_\theta(\mathbf{x}))$ and minimize the KL divergence to $q_\phi(\mathbf{z}|\mathbf{x})$ from $p_\theta(\mathbf{z}|\mathbf{x})$, which is achieved by maximizing the *evidence lower*

bound (ELBO)

$$\begin{aligned} L(\mathbf{x}) &= \log(p_\theta(\mathbf{x})) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z}|\mathbf{x})) \\ &= \mathbf{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\big[\log(p_\theta(\mathbf{x}|\mathbf{z}))\big] - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z})). \end{aligned} \tag{6.2}$$

6.2.1 Realization with Neural Networks

In an actual implementation of a VAE, the recognition and generative models are implemented through neural networks which derive the immediate parameters for the conditional distributions: Concretely, we use an *encoder network* $E_\phi(\mathbf{x})$ to parametrize the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$, and a *decoder* or *generator network* $G_\theta(\mathbf{z})$ for the parameters of the likelihood $p_\theta(\mathbf{x}|\mathbf{z})$. To know what immediate parameters are required, we must specify the used families of distributions.

In the following, we consider a common Gaussian VAE. We assume that the true (but generally intractable) posterior $p_\theta(\mathbf{z}|\mathbf{x})$ roughly follows a Gaussian with diagonal covariance. Accordingly, in the recognition model, we model the approximate posterior as

$$q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}\big(\mathbf{z}; \boldsymbol{\mu}, \mathrm{diag}\big(\boldsymbol{\sigma}^2\big)\big), \tag{6.3}$$

where the encoder determines the distribution parameters

$$\boldsymbol{\mu}, \boldsymbol{\sigma} = E_\phi(\mathbf{x}) := \big(\boldsymbol{\mu}(\mathbf{x}), \boldsymbol{\sigma}(\mathbf{x})\big) \tag{6.4}$$

from the input $\mathbf{x}$. All in all, we obtain $\mathbf{z} \sim \mathcal{N}\big(\boldsymbol{\mu}(\mathbf{x}), \mathrm{diag}\big(\boldsymbol{\sigma}^2(\mathbf{x})\big)\big)$.

In the generative model, the prior of the latent variable is a standard Gaussian that needs no further parametrization,

$$p_\theta(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I}). \tag{6.5}$$

In contrast, we parametrize the conditional likelihood $p_\theta(\mathbf{x}|\mathbf{z})$ which models the probability of the data $\mathbf{x}$ based on the latent variable $\mathbf{z}$ by a decoder network G_θ. Since we work with textual data, $\mathbf{x}$ is a sequence $\mathbf{x} = (\mathrm{x}_1, \ldots, \mathrm{x}_n)$ of discrete tokens (e.g., words, syllables, or characters) over some predefined vocabulary (or alphabet) $\mathcal{V}$. The chain rule allows us to factor the likelihood into conditional probabilities of the next word given the previous words,

$$p_\theta(\mathbf{x}|\mathbf{z}) = \prod_{t=1}^{n} p_\theta(\mathrm{x}_t \mid \mathbf{z}, \mathrm{x}_{<t}), \tag{6.6}$$

where the conditionals are modeled as categorical distributions

$$p_\theta(\mathrm{x}_t \mid \mathbf{z}, \mathrm{x}_{<t}) = \mathrm{Cat}(\mathrm{x}_t; \mathcal{V}, \mathbf{p}_t). \tag{6.7}$$

The probability vectors $\mathbf{p}_t$ over the vocabulary $\mathcal{V}$ are determined by the stateful decoder at each position $t = 1, \ldots, n$ as

$$\mathbf{p}_t = G_\theta(\mathbf{z}, \mathrm{x}_{<t}). \tag{6.8}$$

6.2.1.1 Training

To achieve the training objective, we use stochastic gradient descent (SGD) or a derived method with backpropagation to compute the gradients and minimize the negative ELBO from Eq. (6.2),

$$\begin{aligned} \mathcal{L}_{\mathrm{VAE}}(\mathbf{x}) &= -L(\mathbf{x}) \\ &= \mathbf{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\big[-\log(p_\theta(\mathbf{x}|\mathbf{z}))\big] && (6.9) \\ &\quad + D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z})). && (6.10) \end{aligned}$$

The second term (6.10) with the KL divergence is called the *KL loss*, which in our case of a Gaussian prior and an approximate posterior as in Eqs. (6.3) and (6.5) can be computed analytically as

$$\begin{aligned} \mathcal{L}_{\mathrm{KL}}(\mathbf{x}) &= D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z})) \\ &= \frac{1}{2} \sum_{i=1}^{L} \Big(\sigma_i^2 + \mu_i^2 - \log(\sigma_i^2) - 1 \Big). \end{aligned} \tag{6.11}$$

The first term (6.9) constitutes the *reconstruction loss*

$$\mathcal{L}_{\mathrm{rec}}(\mathbf{x}) = \mathbf{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\big[-\log(p_\theta(\mathbf{x}|\mathbf{z}))\big] \simeq -\log(p_\theta(\mathbf{x}|\mathbf{z})) \tag{6.12}$$

whose expectation is typically approximated with a simple Monte Carlo estimator using just a single latent sample $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})$. It hence corresponds to the *negative log likelihood (NLL)*, which in case of a sequence $\mathbf{x} = (\mathrm{x}_1, \ldots, \mathrm{x}_n)$ can be expanded as per Eq. (6.6) to

$$-\log p_\theta(\mathbf{x}|\mathbf{z}) = -\sum_{t=1}^{n} \log p_\theta(\mathrm{x}_t \mid \mathbf{z}, \mathrm{x}_{<t}), \tag{6.13}$$

which coincides with the *categorical cross-entropy loss* aggregated over the entire sequence.

One remaining problem with the sampling step in Eq. (6.12) is that it is non-differentiable,

but we need gradients for backpropagation. A way to obtain them is the popular *reparameterization trick* [233, 375] or implicit reparameterization [144] as more general alternative.

6.2.1.2 Inference

For inference, we run an input $\mathbf{x}$ though the encoder network E_ϕ to obtain parameters for $q_\phi(\mathbf{z}|\mathbf{x})$, from which we draw a sample $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})$. This latent representation $\mathbf{z}$ we then pass to the decoder network G_θ to reconstruct an output $\hat{\mathbf{x}} \sim p_\theta(\mathbf{x}|\mathbf{z})$. In the case of sequential data such as text, a full output sequence $\hat{\mathbf{x}} = (\hat{x}_1, \ldots, \hat{x}_m)$ can be constructed iteratively by sampling subsequent tokens $\hat{x}_t \sim \text{Cat}(\mathcal{V}, \mathbf{G}_\theta(\mathbf{z}, x_{<t}))$, cf. Eqs. (6.7) and (6.8), until a designated *end-of-sequence* token or maximum length is reached.

In practice, other decoding strategies may be used: For instance, *greedy search* simply takes the next token with the highest probability $\hat{x}_t = \arg\max_x \text{Cat}(x; \mathcal{V}, \mathbf{p}_t)$ (and thus degenerates into a deterministic algorithm), and *beam search* decodes multiple sequences simultaneously, keeping track of the most likely ones.

6.3 Differentially Private Inference through Variational Autoencoders

DP (cf. Section 2.2) is a notion of privacy based on *randomness*, i.e., any non-trivial DP mechanism must be non-deterministic. In the context of textual data, existing DP mechanisms typically operate on a word-by-word basis [46, 138, 141, 142, 465] by randomly determining each word in the output sequence. The common ground of these methods is to impose a probability distribution, locally at each position, over the words in the output space (or similarly, over the embedding space) and draw a word (or corresponding embedding) from that distribution. This has one or more drawbacks, for instance,

- the output is not in a human-readable form, e.g., a vector or BoW representation,
- the produced text is incoherent due to the words being perturbed independently,
- the DP guarantees only cover texts of equal length, and/or
- the required privacy budget in terms of ϵ grows linearly with the length of the output sequence.

Bowman et al. [50] explored VAEs (see Section 6.2) as means to map texts to *distributed* latent representations, providing global summaries of the input texts. The idea is that

"reasonable" sentences are encoded to representations that are close to the origin in the latent space (due to prior regularization), ensuring that nearby representations can also be decoded to coherent outputs. VAEs are non-deterministic by design, as they produce each latent representation by sampling from a posterior distribution based on the input. Remarkably, the posteriors commonly are Gaussian distributions, which are at the heart of the Gaussian mechanism [116] (cf. Definition 2.16 and Theorem 2.21) that is frequently used in the context of DP.

Unfortunately, despite its inherent randomness, a vanilla Gaussian VAE architecture is not sufficient to guarantee DP: On one hand, without additional restrictions, the latent space is *unbounded*, which prevents a finite sensitivity (cf. Definition 2.12) as required for DP. On the other hand, the variance of the posterior distribution is determined by the encoder from the input; therefore, the latent representation of an input with a tiny variance could degenerate to a tiny disc around its mean covering most of the probability mass, where another input with a large variance would have a distinctively smaller probability, which may again violate given DP guarantees.

6.3.1 Differentially Private Latent Sampling

In the following, we modify a vanilla VAE to exploit its random latent variable in a way to achieve (Rényi) DP in the latent representations.

Mean Bound. In a Gaussian VAE, the parameter $\boldsymbol{\mu} = \boldsymbol{\mu}(\mathbf{x})$ determines the mean location of the latent representations sampled from the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}\big(\mathbf{z}; \boldsymbol{\mu}, \operatorname{diag}(\boldsymbol{\sigma}^2)\big)$. Since we work in the local model (Section 2.2), any two input texts are considered adjacent. To obtain differentially private latent representations, we hence need a finite sensitivity $\Delta = \max_{\mathbf{x},\mathbf{x}'} \big(\|\boldsymbol{\mu}(\mathbf{x}) - \boldsymbol{\mu}(\mathbf{x}')\|\big)$. Normally, the mean locations are unbounded, so arbitrary inputs $\mathbf{x}, \mathbf{x}'$ could cause $\|\boldsymbol{\mu}(\mathbf{x}) - \boldsymbol{\mu}(\mathbf{x}')\|$ to become arbitrarily large. We hence propose a continuous mean bound in the latent space, namely, a vector-valued *radial hyperbolic tangent* map

$$\tanh^*(\boldsymbol{\mu}) := \tanh\big(\|\boldsymbol{\mu}\|\big) \frac{\boldsymbol{\mu}}{\|\boldsymbol{\mu}\|} \tag{6.14}$$

that contracts the vector $\boldsymbol{\mu}$ to lie inside a unit ball about the origin.

If necessary, we can resize the co-domain of $\tanh^*$ to a ball of radius $R > 0$ using $R \tanh^*(\boldsymbol{\mu})$; this is useful as the prior $p_\theta(\mathbf{z})$ regulates the latent representations to follow a standard Gaussian distribution, and by choosing, e.g., $R = 3$, the shrunk latent space would still cover over 99.7% of its probability mass (3σ rule). We thus obtain an approximate

(a) Gaussian VAE: unbounded μ and σ dependent on $\mathbf{x}$.

(b) DP-VAE: contracted mean μ^* and constant, isotropic $\sigma \equiv \sigma_{\text{fix}}$.

Figure 6.1: Sketch of latent space with posteriors $q_\phi(\mathbf{z} \mid \mathbf{x}_i) = \mathcal{N}\big(\mathbf{z}; \mu(\mathbf{x}_i), \operatorname{diag}\big(\sigma^2(\mathbf{x}_i)\big)\big)$ without and with DP constraints.

posterior $\mathcal{N}\big(\mu^*, \operatorname{diag}\big(\sigma^2\big)\big)$ with $\|\mu^*\|_2 \leq R$ by replacing the original mean $\mu = \mu(\mathbf{x})$ as determined by the encoder with the contracted mean

$$\mu^* = \mu^*(\mathbf{x}) := R \tanh^*\big(\mu(\mathbf{x})\big). \tag{6.15}$$

The effect of contracting the mean is shown in Fig. 6.1: In the original VAE, $\mu(\mathbf{x})$ is only *regulated* by the KL loss to be close to the origin, but there is no hard limit for that distance. This is illustrated by $\mu(\mathbf{x}_3)$ in Fig. 6.1a, which is outside the dashed disc. On the other hand, the contracted mean values $\mu^*(\mathbf{x}_i)$ in Fig. 6.1b all lie within a ball of radius R about the origin. Note though, that *samples* from the approximate posteriors may still end up further away.

Global Variance. A similar issue as with the mean arises if the standard deviations $\sigma(\mathbf{x})$ are determined by the encoder: While $\sigma(\mathbf{x}_1)$ could be relatively large for one input $\mathbf{x}_1$, we may bring $\sigma(\mathbf{x}_2)$ arbitrarily close to $\mathbf{0}$ for another input $\mathbf{x}_2$. Now if we consider the ratio of probabilities $p(Z \mid \mathbf{x}_2)/p(Z \mid \mathbf{x}_1)$ for some small event set of representations Z enclosing $\mu(\mathbf{x}_2)$ as illustrated in Fig. 6.1a by the yellow area, the numerator will be large, but the denominator very small. Due to the continuity of the encoder network, we may be able to change $\mathbf{x}_1$ and $\mathbf{x}_2$ as to increase the difference between $\sigma(\mathbf{x}_1)$ and $\sigma(\mathbf{x}_2)$ even further; therefore, the ratio cannot be bounded in general, violating the core idea of DP. To solve this, we no longer determine the standard deviation by the encoder as a function $\sigma(\mathbf{x})$, but specify a global $\sigma := \sigma_{\text{fix}}$ that is independent of the input $\mathbf{x}$.

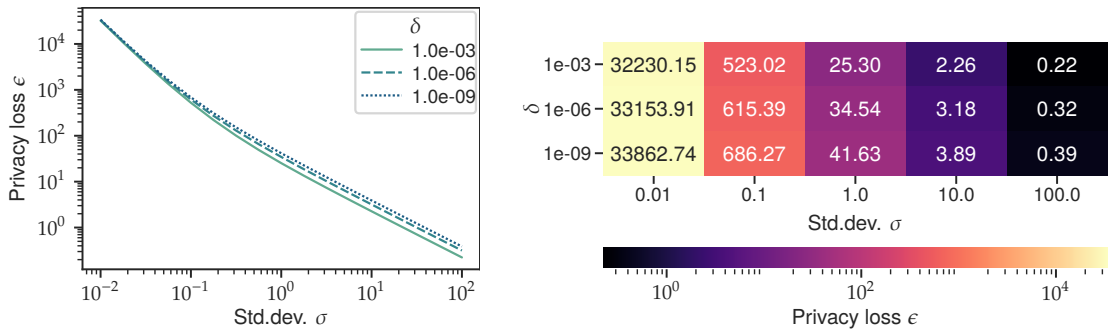


Figure 6.2: Privacy guarantee of isotropic Gaussian $\left(\alpha, \frac{\alpha\Delta_2^2}{2\sigma^2}\right)$-RDP mechanisms with sensitivity $\Delta_2 = 6$ in terms of (ϵ, δ)-DP, over a range of $\sigma \in [0.01, 100]$.

6.3.2 Differential Privacy Properties of the Constrained VAE

The modified VAE with mean bound and global variance as described in Section 6.3.1 ends up with a new probabilistic encoder that approximates the posterior distribution as

$$q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mathbf{z}; \boldsymbol{\mu}^*, \operatorname{diag}\left(\boldsymbol{\sigma}_{\text{fix}}^2\right)\right), \tag{6.16}$$

where $\boldsymbol{\mu}^* = R \tanh^*\left(\boldsymbol{\mu}(\mathbf{x})\right)$ as in Eq. (6.15) and $\boldsymbol{\sigma}_{\text{fix}}$ is a hyperparameter that is independent of the input $\mathbf{x}$. For simplicity, we assume an isotropic Gaussian with variances $\boldsymbol{\sigma}_{\text{fix}}^2 = (\sigma^2, \ldots, \sigma^2)$. According to Theorem 2.21, the modified probabilistic encoder achieves

$$\left(\alpha, \frac{\alpha\Delta_2^2}{2\sigma^2}\right)\text{-RDP} \tag{6.17}$$

with sensitivity $\Delta_2 = 2R$. By Proposition 2.11, we can translate this RDP curve to (ϵ, δ)-DP, with exemplary values using $R = 3$ shown in Fig. 6.2.

Another benefit of our differentially private latent sampling mechanism is that decoding acts as post-processing, which is known to preserve DP [116, Proposition 2.1] and RDP [309, p. 4] properties. Therefore, we can use any desired decoding strategy that improves the quality of the output data. For instance, in the case of text, this could be beam search, Top-K or Top-p sampling [129, 192], etc.

6.4 Anonymizing Online Reviews

We apply the proposed DP-VAE architecture from Section 6.3 to the task of anonymizing online reviews. Specifically, we consider the following scenario: An online platform wants to publish its users' anonymous reviews for interested readers as well as for businesses

and researchers to build sentiment analysis models. However, an attacker attempts to identify the authors of given reviews, for example, when these are critical of their business or product. The attacker suspects a limited pool of candidates to have written negative reviews and is thus able to train an authorship attribution model given a set of similar documents written by the candidates.

Our goal is to obfuscate the reviews to prevent the attacker from identifying the authors while preserving their utility for a sentiment classification model, keeping them semantically as close to the original as possible and of high quality in terms of language. To that end, we test variations of two different architectures, which we describe in detail in the following sections.

6.4.1 End-to-End Differentially Private VAE

Our first approach applies our DP constraints from Section 6.3.1 to the overall latent variable $\mathbf{z}$ in a Gaussian VAE model for text (e.g., as in [50]) to achieve differentially private latent sampling of $\mathbf{z}$.

The encoder E_ϕ predicts the parameters of the Gaussian posterior $q_\phi(\mathbf{z}|\mathbf{x})$ from which the sentence representation $\mathbf{z}$ is sampled. Specifically, it consists of a bidirectional multi-layer RNN with GRUs [76] that encodes a sequence of word embeddings $\mathbf{x} = (x_1, \ldots, x_n)$. The concatenated hidden states from both directions of the last layer are passed to a modified feed-forward network that infers the *contracted* mean $\boldsymbol{\mu}^* = R \tanh^*(\boldsymbol{\mu}(\mathbf{x}))$ as in Eq. (6.15), while $\sigma := \sigma_{\text{fix}}$ is fixed (cf. Section 6.3.1). The generator G_θ also consists of a RNN with GRUs. It iteratively decodes the latent vector $\mathbf{z}$ to a sentence $\hat{\mathbf{x}} = (\hat{x}_1, \ldots, \hat{x}_m)$ by mapping the output of its final layer to a word from the vocabulary through a softmax function. The overall loss $\mathcal{L}_{\text{VAE}}(\mathbf{x})$ then is the sum of the reconstruction loss and the KL loss.

6.4.2 Disentangled Latent Representations

Our second approach follows John et al. [213] and incorporates auxiliary losses to disentangle the latent representation into two variables, $\mathbf{z}_c$ and $\mathbf{z}_a$, representing content- and author-specific information of the input text, respectively. Our idea with this *adversarial autoencoder (AAE)* is to leave the content embedding $\mathbf{z}_c$ unchanged, but obfuscate the author embedding $\mathbf{z}_a$ so that the decoder produces an output that preserves the semantics of the input while protecting the author information: To that end, we either stick to [213] and set $\mathbf{z}_a$ to its average encoding, or we perturb the author embedding $\mathbf{z}_a$ using DP latent sampling (Section 6.3.1). As we cannot prove that there is no leakage of author-related information into $\mathbf{z}_c$, this approach does not provably fulfill end-to-end DP; however, we hypothesize that it may lead to better privacy-utility trade-offs.

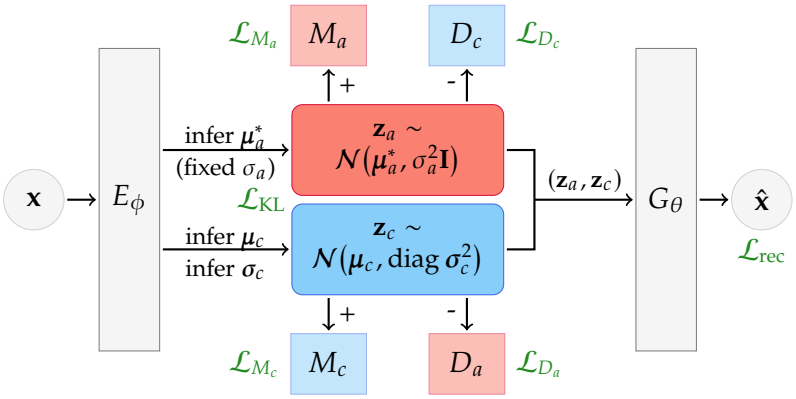


Figure 6.3: Sketch of our disentangled latent space approach.

Like our first approach in Section 6.4.1, the architecture consists of a GRU-RNN based encoder and generator. A feed-forward network predicts the parameters μ_c, σ_c characterizing the Gaussian posterior $q_\phi(\mathbf{z}_c|\mathbf{x})$ from which the content embedding $\mathbf{z}_c$ is sampled, whereas $\mu_a^*, \sigma_{a,\text{fix}}$ characterizing the Gaussian posterior $q_\phi(\mathbf{z}_a|\mathbf{x})$ for the author embedding $\mathbf{z}_a$ obey the privacy constraints described in Section 6.3. Alternatively [213], during inference, we can set $\mathbf{z}_a$ to the average of all author representations generated within one batch to let the generator decode an obfuscated author representation.

6.4.2.1 Motivating Losses

To ensure that only $\mathbf{z}_c$ contains semantic information and only $\mathbf{z}_a$ contains author-specific information, we apply auxiliary losses in addition to $\mathcal{L}_{\text{VAE}}$. Following Romanov et al. [382], we differentiate between losses motivating $\mathbf{z}_c$ and $\mathbf{z}_a$ to store content- and author-related information, respectively, and discriminating losses penalizing $\mathbf{z}_c$ and $\mathbf{z}_a$ for storing author- and content-related information, respectively (cf. Fig. 6.3).

Content Motivator. To preserve semantics in the content embedding, we define a vocabulary $\mathcal{V}$ of content words and train a content motivating network M_c to predict the Bag-of-Words (BoW) distribution p_b of the input sentence as proposed by John et al. [213],

$$\hat{p}_b = M_c(\mathbf{z}_c) = \text{softmax}(W_{M_c} * \mathbf{z}_c + b_{M_c}).$$

The content motivating loss $\mathcal{L}_{M_c}$ then is the cross-entropy loss

$$\mathcal{L}_{M_c} = -\sum_{w \in \mathcal{V}} p_b(w) \cdot \log\big(\hat{p}_b(w)\big).$$

Author Motivator. The author motivating network M_a aims to predict the one-hot encoded author p_a from the author vector $\mathbf{z}_a$,

$$\hat{p}_a = M_a(\mathbf{z}_a) = \text{softmax}(W_{M_a} * \mathbf{z}_a + b_{M_a}).$$

We again employ the cross-entropy loss as author motivating loss

$$\mathcal{L}_{M_a} = -\sum_{u \in \mathcal{A}} p_a(u) \cdot \log\big(\hat{p}_a(u)\big),$$

where $\mathcal{A}$ denotes the set of authors in our dataset.

6.4.2.2 Discriminating Losses

Our content and author discriminating losses $\mathcal{L}_{D_c}$ and $\mathcal{L}_{D_a}$ work analogous to the motivating losses: The content discriminator D_c is trained to predict the BoW distribution from the latent author representation $\mathbf{z}_a$ and our author discriminator D_a is trained to predict the author of a given sentence from $\mathbf{z}_c$. Other than the motivating networks M_c and M_a, D_c and D_a are two-layer networks, as we hypothesize that the prediction tasks are harder for the discriminators than for the motivators.

6.4.2.3 Training Objective

Our overall training objective is a min-max objective. For each batch in our training dataset, we first minimize the discriminators' losses with respect to their weights and consequently minimize the autoencoder's overall loss $\mathcal{L}_{\text{ovr}}$ while keeping the discriminators' weights fixed. $\mathcal{L}_{\text{ovr}}$ is defined as

$$\mathcal{L}_{\text{ovr}} = \mathcal{L}_{\text{VAE}} + \lambda_{M_c}\mathcal{L}_{M_c} + \lambda_{M_a}\mathcal{L}_{M_a} - \lambda_{D_c}\mathcal{L}_{D_c} - \lambda_{D_a}\mathcal{L}_{D_a} \tag{6.18}$$

where the weights $\lambda_{M_c}, \lambda_{M_a}, \lambda_{D_c}, \lambda_{D_a}$ are hyperparameters.

6.5 Evaluation

We conduct several experiments to test our proposed architectures. In this section, we describe the experiments' setup and conduction including the used datasets, evaluation metrics, and their results.

6.5.1 Datasets

We use two publicly available datasets in our evaluation:

IMDb Movie Reviews. The IMDb movie review dataset [271] contains 100,000 movie reviews from 62 users with a rating label on a scale of one to ten. We reduce the size of the dataset to 10,000 by only keeping reviews from ten authors, and simplify the sentiment label by treating every rating below 5 as "negative" and every rating of 5 or higher as "positive", resulting in a binary classification task.

Yelp Product Reviews. The Yelp dataset[1] consists of over 6 million user reviews of businesses such as bars and restaurants with ratings on a scale of one to five as well as pseudonymous author labels. We only keep data from the ten users with the most reviews, hereby reducing the dataset to the size of 15,729 reviews. We simplify the rating labels by treating every rating of one to three as "negative" and the rest as "positive", thus creating a binary classification task.

6.5.2 Evaluation Metrics

We evaluate the proposed architectures in terms of four aspects: First, we investigate how well the transformed texts are anonymized, as measured by the effectiveness of mitigating authorship attribution. Second, we evaluate the utility of the produced texts from the perspective of a third party wanting to analyze the sentiment of customer reviews, measuring how well the outputs reflect the sentiment of the original reviews using sentiment classifiers. Third, we assess the semantic similarity between the output texts and their originals. Lastly, we also evaluate the readability of the texts.

6.5.2.1 Privacy (Authorship Obfuscation)

We evaluate our models' anonymization effectiveness through authorship attribution classifiers predicting the writers from the transformed texts. We employ two models, a shallow SVM classifier predicting the author from a word uni- and bigram frequency vector, as well as a classifier based on the BERT [101] language model which has been shown to perform well for authorship attribution [128]. Specifically, we fine-tune BERT's last three layers with two additional dense layers to predict an author label given BERT's classification token [CLS].

[1] https://www.yelp.com/dataset/

Attacker Models. Assuming that an attacker only has access to labeled original reviews as training data, we train *static* models on the original texts and compare their authorship attribution performance on both the original and the transferred texts. However, in a realistic scenario, the obfuscation method may be public (no "security/privacy by obscurity"), so the attacker could simply produce labeled *obfuscated* training data himself. Therefore, we also train and evaluate *adaptive* models on the transformed sentences.

Classification Metrics. Besides measuring the *accuracy* of our attribution models, we compute the *Matthews correlation coefficient (MCC)* [165, 287] to account for any imbalances in the datasets (cf. Chicco and Jurman [73]). MCC scores range from -1 to 1, with 0 indicating uninformed or random guesses, -1 indicating intentionally avoiding correct choices, and +1 indicating correct choices only. Notably, always predicting the majority label in an imbalanced dataset may result in misleadingly high accuracy scores whereas the MCC would stay around 0.

6.5.2.2 Utility (Sentiment Preservation)

We measure utility from the perspective of a company/researcher aiming to analyze the sentiment of user reviews while protecting the privacy of the consumer. Thus, similar to the evaluation of privacy, we train a BoW based SVM classifier and a BERT based classifier. Besides accuracy, we also measure the MCC of both static and adaptive models.

6.5.2.3 Semantic Similarity (Content Preservation)

As we want our transformed sentences to preserve the semantics of the original ones, we compute three metrics measuring the semantic similarity of these.

METEOR (ME). For comparison with other works, we include the established METEOR score [33]. Note that its scoring method relies on aligning (stemmed) word unigrams in the input and output sentences, but usage of n-grams also constitutes a stylometric feature that may be distinctive to certain writers. Therefore, METEOR may be misleading, since high stylometric similarity may also indicate a high chance of success for the authorship attribution attack.

Sentence-BERT (SB) and Universal Sentence Encoder (USE). We employ modern sentence embeddings based on Sentence-BERT [369] and the Universal Sentence Encoder [62] and compute the cosine similarity of the input and output sentences. Importantly, unlike metrics such as METEOR or word overlap, they do *not* focus on the exact wording

and spelling that is used, and thus provide a more abstract and robust measure of semantic similarity without the strong connection to stylometric similarity.

6.5.2.4 Readability / Language Quality

We measure the readability of our generated sentences by computing the perplexity (PPL) based on the log-likelihood of a GPT-2 language model [363].

6.5.3 Experiment Conduction

We train and evaluate four different architectures:

V-VAE: A vanilla Gaussian VAE architecture for text (similar to Bowman et al. [50]) as a baseline without DP constraints or auxiliary losses.

DP-VAE: Our DP-VAE architecture with DP constraints and formal privacy guarantees as explained in Section 6.4.1.

AVG-AAE: The adversarial autoencoder from Section 6.4.2 with disentangled representations, where the author representations in an inference batch are set to their average across the batch.

DP-AAE: The adversarial autoencoder from Section 6.4.2 with disentangled representations, where the author representations are obtained through differentially private sampling.

We implement all architectures in TensorFlow and use Optuna [14] to individually tune their hyperparameters, such as the weights of the loss functions or the hidden layer sizes of the encoder and decoder, over 40 trials before comparing them on a final unseen test set. In each trial, we train three models for 30 epochs on randomized shuffle splits of the training data, and for each trained model we transform three sets of validation data. Thus, we obtain a total of 9 sets of obfuscated reviews per trial, based on which we choose the best trial in terms of privacy as specified in Section 6.5.2.1, utility as specified in Section 6.5.2.2, and semantic similarity according to Sentence-BERT as specified in Section 6.5.2.3. The discriminator networks are optimized using RMSprop [189], while we use Adam [232] for the overall model. For both DP-constrained models DP-VAE and DP-AAE, we report results with mean bound $R := 3$ and the best found value for σ_{fix}. Moreover, we assess the impact of σ_{fix} on the privacy-utility trade-off by re-evaluating the best DP models with varying $\sigma_{\text{fix}} \in [0.01, 100]$.

Table 6.1: Evaluation results of author (A) and sentiment (S) classifiers based on a static / adaptive SVM or BERT model. (Text metrics: SB=Sentence-BERT, USE=Universal Sentence Encoder, ME=METEOR, PPL=GPT-2 perplexity. Best trade-off in *italics*.)

	Accuracy								MCC								Text metrics			
	SVM sta		SVM ada		BERT sta		BERT ada		SVM sta		SVM ada		BERT sta		BERT ada					
Model	A ↓	S ↑	A ↓	S ↑	A ↓	S ↑	A ↓	S ↑	A ↓	S ↑	A ↓	S ↑	A ↓	S ↑	A ↓	S ↑	SB ↑	USE ↑	ME ↑	PPL ↓
IMDb:																				
Original	0.90	0.89	0.90	0.89	0.87	0.86	0.87	0.86	0.89	0.47	0.89	0.47	0.85	0.58	0.85	0.58	1.0	1.0	1.0	157.9
V-VAE	0.77	0.87	0.77	0.87	0.59	0.79	0.73	0.87	0.75	0.37	0.74	0.34	0.56	0.41	0.71	0.45	*0.59*	*0.62*	*0.24*	213.3
DP-VAE	*0.14*	*0.86*	*0.24*	*0.86*	*0.13*	*0.80*	*0.28*	*0.86*	0.07	0.00	0.16	0.00	0.04	0.01	0.20	0.00	0.33	0.35	0.09	73.4
AVG-AAE	0.24	0.86	0.31	0.87	0.23	0.84	0.30	0.86	0.16	0.22	0.24	0.34	0.15	0.26	0.22	0.35	0.47	0.50	0.15	72.6
DP-AAE	0.17	0.86	0.29	0.88	0.18	0.77	0.29	0.86	*0.09*	*0.25*	*0.21*	*0.38*	*0.09*	*0.27*	*0.22*	*0.39*	0.44	0.44	0.14	*58.3*
Yelp:																				
Original	0.85	0.72	0.85	0.72	0.85	0.77	0.85	0.77	0.83	0.43	0.83	0.43	0.83	0.55	0.83	0.55	1.0	1.0	1.0	199.4
V-VAE	0.69	0.69	0.71	0.69	0.66	0.70	0.70	0.70	0.65	0.39	0.66	0.37	0.61	0.41	0.65	0.41	*0.59*	*0.57*	*0.22*	233.1
DP-VAE	0.20	0.52	0.30	0.54	0.16	0.52	0.33	0.51	0.08	0.02	0.15	0.07	0.05	0.03	0.21	0.02	0.12	0.12	0.05	*53.2*
AVG-AAE	*0.28*	*0.67*	0.41	0.68	*0.25*	*0.67*	0.40	0.67	*0.18*	*0.34*	0.31	0.35	*0.17*	*0.34*	0.31	0.34	0.52	0.50	0.15	152.1
DP-AAE	0.33	0.68	*0.42*	*0.70*	0.28	0.67	*0.41*	*0.69*	0.25	0.36	*0.32*	*0.39*	0.20	0.33	*0.31*	*0.38*	0.50	0.48	0.14	175.0

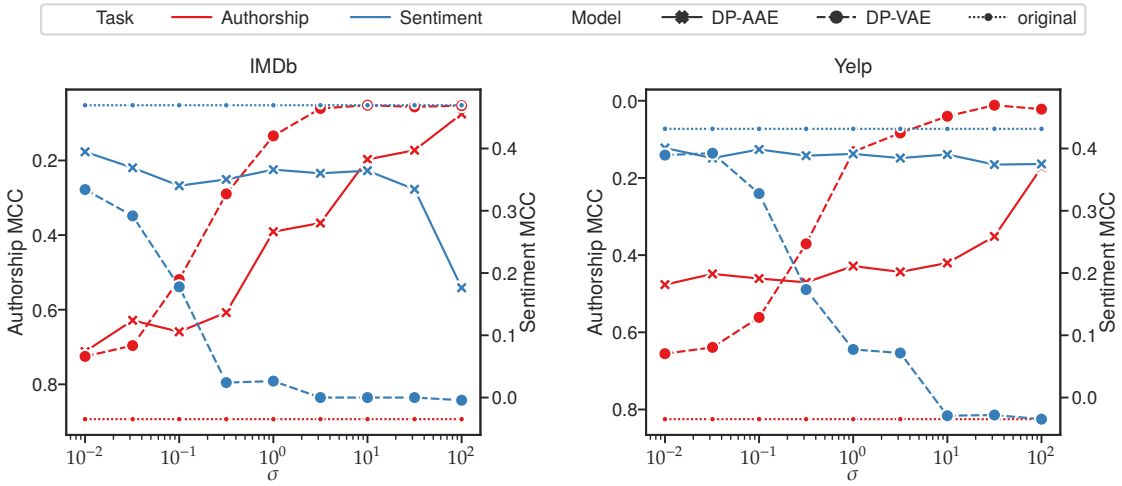


Figure 6.4: Privacy-utility trade-off for DP-VAE and DP-AAE over $\sigma_{\text{fix}} \in [10^{-2}, 10^{2}]$, measured as MCC of the adaptive SVM authorship (inverted y-axis left) and sentiment classifiers.

6.5.4 Results

In Table 6.1, we report our full results with the best privacy-utility trade-off scores (relative "author minus rating" classifier performance) highlighted in *italics*. Tables 6.2 and 6.3 show examples of generated sentences from the IMDb and Yelp datasets, respectively.

First off, the accuracy scores are typically much higher than their MCC counterparts, particularly for the sentiment classifiers. However, note that the final test split for IMDb is strongly imbalanced with over 80% "positive" sentiment labels, so even a classifier that always predicts "positive" would achieve accuracies over 0.8. We henceforth rely on MCC to discuss our results, as it provides a much more sound and meaningful metric even with imbalanced datasets. We also observe that an adaptive strategy is generally more powerful: While it appears relatively easy to fool a static author classifier trained on unobfuscated data, an attacker with an adaptive strategy has a much better chance of re-identifying the authors. Similar gains can be seen for the sentiment analysis task.

As expected, models relying on disentangled representations achieve better privacy–utility trade-offs than vanilla or DP-VAE. The latter two baselines present two opposing ends of a spectrum with V-VAE providing the best content preservation and the highest attack and utility scores at the same time, whereas DP-VAE with DP constraints on the entire latent space strongly affects both tasks with poor content preservation. Remarkably, perturbing the author representations through DP latent sampling in DP-AAE leads to slightly better results than averaging with AVG-AAE.

Fig. 6.4 visualizes the effect of varying the standard deviation parameter σ_{fix} and thus

the privacy loss ϵ due to its inverse correlation with the privacy loss ϵ as explained in Proposition 2.11. As expected, a higher σ_{fix} leads to higher privacy and lower utility scores. Since DP-VAE applies private sampling to the entire latent variable, the effect of varying σ_{fix} is stronger than for DP-AAE, which can be seen in the early drop of the sentiment scores in Fig. 6.4. The disentangling DP-AAE can maintain a good utility even with larger variances σ^2_{fix} while at the same time preventing the authorship attribution attack. Our DP models listed in Table 6.1 use a σ_{fix} of 1.0 for DP-VAE, 14.2 (IMDb) or 38.8 (Yelp) for DP-AAE, respectively. In terms of (ϵ, δ)-DP with, e.g., $\delta = 10^{-6}$, this corresponds to a privacy loss in the range $34.539 \gtrsim \epsilon \gtrsim 0.815$, respectively (cf. Fig. 6.2). The corresponding α values for translating the RDP curve to (ϵ, δ)-DP (cf. Proposition 2.11) are 1.876, 13.398, and 34.996, respectively.

6.6 Comparison with Related Work

Authorship Obfuscation. Countering authorship attribution has been of great interest within the research community, resulting in a wide range of different authorship obfuscation approaches as discussed in Section 3.3.4.2. Bo et al. [46] propose a combination of an autoencoder with DP where they sample subsequent words of the output using a variant of the Exponential mechanism. While their approach still samples the output word-by-word and hence is subject to some limitations of word-level DP (cf. Section 3.3.5.1), it does achieve some level of coherence by conditioning the next word on the latent representation of the input and the previous words.

The work in this chapter merges the concept of DP with generative models and adversarial training to propose a novel approach for text anonymization: To the best of our knowledge, we are the first who exploit the Gaussian noise in the latent space of a VAE and interpret it as Gaussian mechanism to achieve DP for full sentences instead of individual words, and are thus able to produce diverse and coherent, human-readable outputs. Optionally, we can augment our model with an AAE architecture to obtain disentangled author and content representations which can be obfuscated individually to improve the privacy-utility trade-off.

Besides text [46, 241], autoencoders and disentangled representations have been used to obfuscate different kinds of sequential data, such as facial images [160, 326, 469] or audio in several methods based on x-vectors [416] (cf. our discussion of related work in Section 3.3.4). Therefore, we argue that our DP-VAE and DP-AAE approaches are also applicable to other types of sequential data.

Table 6.2: Transformed IMDb reviews

Original	*DP-VAE*
this is an OK TV movie , i guess .	this is a b movie , but i guess .
i think it's supposed to be a black comedy .	i think it's supposed to be a drama .
it had some ironic moments , to be sure .	it has been more dangerous than anything else .
i did think the acting was pretty decent .	i didn't think the acting was pretty decent .
i really liked the soundtrack .	i really liked the music .
i thought it was very unique .	i thought it was very clever .
i felt it fit nicely withe movie .	i found it predictable .
it was nice to see Judd Nelson in a different kind of role .	it was nice to see edward g . <unk> in a small role .
i think he pulled it off well .	i just thought he pulls it off .
to me , that's a sign of a good actor .	for me , he is a sign of a good .
and like i said , the acting was decent all around .	and , i like the movie , it's just ok .
it was just the movie itself that didn't quite work for me .	it was the movie really made the movie to be .
DP-AAE	*AVG-AAE*
this is a ok movie .	this is a ok movie , but that's not a tv movie .
i think this is supposed to be a black comedy .	i don't think it's a comedy .
it has some good moments , but it doesn't really matter .	it has some good moments , but it has some good moments .
i think this is a decent cast .	i think the acting is decent .
i liked the soundtrack .	i give it a soundtrack .
i thought it was unique .	i thought it was unique .
i felt it fit the movie .	i felt it fit the movie .
it is a perfect role as a perfect role .	it was a nice job in the film .
i think it was pulled it .	i think it was .
the actor is a sign of a good actor .	he's a good actor , but that's not a bad actor
the acting is ok , but it isn't bad .	and then the film is great , but it's not great .
it doesn't work out of the work .	it was a good job in the film .

Table 6.3: Transformed Yelp reviews

Original	*DP-VAE*
It was a wonderful experience !	it was a nice !
The coffee was great too .	the service was good .
I will certainly be back !	i will be back !
Yum Yum Yum Yum Yum .	yum .
We got here on a Wednesday night and it was packed !	we got a friend and it was a good !
They told us it would be at least a 2 hour wait .	we were not a few minutes for me to be back .
It was worth it though !	it was so much !
The fries are very tasty as well .	the fries were very good and the fries were very good .
I will definitely come here again to dine in !	i will be back to go back !
DP-AAE	*AVG-AAE*
it was a great experience .	it was great !
the coffee was great .	the coffee was great .
i would certainly be back !	will i return !
yum yum ! !	yum yum . yum .
we got the night and it was packed .	got the lunch specials and it was pretty good !
they have a few minutes to order the next time .	they were told me to wait for 15 minutes .
it was worth it .	it was worth it !
the fries were tasty and tasty .	the fries were tasty and tasty .
i would come back for a game, but i would come back .	i would come back here again !

Differentially Private Optimization. DP-SGD [5, 37, 417] and derived methods [172], or objective perturbation [70], have become popular techniques to protect the *training data* of ML models by making the gradient updates or evaluation of the loss function differentially private. Our approach with DP-VAEs and -AAEs can be regarded as complementary since it protects the data during *inference*. In fact, both approaches could be used together to protect both training data and then use the final model to obfuscate other sensitive data.

6.7 Chapter Summary

In this chapter, we have proposed a novel approach for authorship obfuscation that rewrites full sentences with DP guarantees, obfuscating both the authors' style and sensitive identifiers while maintaining the meaning of the texts. At the heart of our approach called DP-VAE lies a VAE architecture which has been shown to learn continuous representations of sentences in its latent space, which we modify to provide DP. To our best knowledge, we are the first to exploit synergies between probabilistic (Gaussian) latent representations of a VAE and randomness of DP mechanisms to achieve differentially private obfuscation for full sentences instead of only individual words, thus able to produce diverse and coherent, human-readable outputs.

Moreover, we extend our approach to a differentially private adversarial autoencoder (DP-AAE) by integrating adversarial learning to disentangle the latent representations into a privacy-sensitive author/style vector and a privacy-insensitive content vector. This separation enables us to further improve the trade-off between privacy and utility in a favorable direction, here, by applying stronger noise to the privacy-sensitive style vector. Further extensions are possible, e.g., to protect other sensitive attributes.

We evaluate our methods in a scenario with online reviews whose authors wish to remain anonymous. The results show that our DP-AAE approach effectively reduces re-identification risks against authorship attribution attacks while preserving the content of the texts. Lastly, due to the wide applicability of VAEs to many types of data besides text (i.e., sequences of discrete tokens), such as images or time series (sequences of numerical data), we argue that our approach is likely adaptable to other privacy-sensitive scenarios whose evaluation we leave as future work.

Chapter 7

Differential Privacy for Directional Data

Directional data is an important class of data where the magnitudes of the data points are negligible. It naturally occurs in many real-world scenarios: For instance, *geographic locations* (approximately) lie on a sphere, and periodic data such as *time of day*, or *day of week* can be interpreted as points on a circle. Massive amounts of directional data are collected by location-based service platforms such as Google Maps or Foursquare, which depend on mobility data from users' smartphones or wearable devices to enable their analytics and marketing businesses. However, such data is often highly privacy-sensitive and hence demands measures to protect the privacy of the individuals whose data is collected and processed. In this chapter, we develop tailored DP solutions for directional data by combining directional statistics with DP: First, we introduce a novel variant of metric privacy [65] (cf. Section 2.2.3) for directional data called *directional privacy*. Next, we construct and analyze two suitable directional privacy mechanisms starting with the spherical von Mises–Fisher (VMF) distribution. As we verify experimentally, our novel privacy mechanisms achieve better privacy-utility trade-offs than adoptions of established DP mechanisms to directional data, especially in the medium to high privacy regime.

This chapter is based on the following publication [467]:

> Benjamin Weggenmann and Florian Kerschbaum: "Differential Privacy for Directional Data". In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security* (*CCS '21*).

7.1 Introduction

In recent years, large-scale collection and processing of directional data have become important drivers for the digital economy: For instance, crowd-sourced data from mobile or wearable devices often includes the geographic location where and the time when the data was recorded. Prominent applications include *location-based marketing and analytics*, as provided by platforms such as Foursquare, and the collection of check-in data by online

mapping services such as Google Maps which provide, e.g., daily "busyness" histograms of visit times at places like stores or restaurants, from which users can estimate how busy a location is during different times of the day.

While such techniques provide substantial value for businesses and drive innovation, the data collected in such scenarios is often privacy-sensitive, and users may be reluctant to share their whereabouts during the course of the day. In many cases, directional data conveys particularly sensitive information, as illustrated by recent news about location tracking on smartphones or fitness trackers [187, 437]. Personal locations are suspect to various attacks, cf. the survey by Krumm [243], in particular when combined with temporal information as shown by Primault et al. [353] or Pyrgelis et al. [356].

Problem. To protect the privacy of individuals while maintaining data-driven business models, the concept of *DP* by Dwork et al. [117] presents the current state-of-the-art for quantifying and limiting information disclosure about individuals. DP mechanisms have been proposed for various settings and data types, e.g., the standard Laplace mechanism [117] which extends infinitely on the real line, or the Planar Laplace mechanism by Andrés et al. [24] which is defined for *planar* locations. While post-processing, such as clipping or wrapping, can be applied to adapt these mechanisms to periodic domains, none of them intrinsically considers the potentially directional nature of the underlying data. In fact, adapted standard mechanisms based on wrapping can behave even worse than uniform noise, as we show in Section 7.4.3.1. We hence argue that specialized, *directional* privacy mechanisms are needed to provide superior privacy–utility trade-offs and investigate proper ways to provide DP *intrinsically* for directional data (cf. Section 7.3).

Inspired by the notion of geo-indistinguishability [24], a variant of metric privacy [65] for planar location data, we propose *directional privacy* as an adaptation to directional data. As a benefit, this notion allows relaxing the guarantees of pure DP to protect data within a given protection radius (i.e., surface distance or angle) $r > 0$ with a specified privacy level ℓ. By setting the protection radius $r = \Delta$ to the sensitivity, this also covers pure ϵ-DP. Relaxing the privacy guarantees to a smaller radius is very useful when working in the local model, e.g., when we want to protect spatial or temporal data that are close to each other, such as restaurants or other venues in densely populated areas, where pure DP would inject too much noise. We demonstrate this in our experiments in Section 7.4.4.2.

As we observe in Section 7.4.3, several directional statistics such as the circular mean benefit from our specialized mechanisms: At $\epsilon = 1.0$, we achieve a more than 4.8-fold reduction in the number of required survey responses over adapted baselines to reach an error below 0.1, so that the service provider needs to collect only ≈ 750 responses instead of over 3600. Conversely, given the same number of responses, our proposed

mechanisms achieve MAEs of only 0.407 and 0.321, which is less than half of 0.695 as for the Wrapped Laplace baseline. Strikingly, for such directional statistics, local DP can be as accurate as central DP and hence is the method of choice, since it does not require a trusted aggregator. Moreover, in Section 7.4.4, we observe that a wrapped Planar Laplace variant for geolocations yields larger errors for histograms than our proposed mechanisms in the critical range $10^{-1} \lesssim \epsilon \lesssim 10$.

Contributions. Our results concern theoretical aspects (Section 7.3) in the areas of privacy and directional statistics, as well as experiments (Section 7.4) to substantiate the theory and its applicability:

- As for privacy, in Section 7.3, we propose the notion of *directional privacy*, an adaptation of metric privacy [24, 65] for directional data based on the surface distance on the sphere. To realize this notion, we form the novel *von Mises–Fisher* and *Purkayastha privacy mechanisms* from the eponymous distributions and prove their (differential) privacy properties.
- We derive *analytical formula* in terms of confluent hypergeometric series for the *expected Euclidean distance* and the *cumulative distribution function (CDF) of the mixture density* of the Von Mises–Fisher distribution in Section 7.3.2, as well as *closed-form solutions* in terms of elementary functions for the *expected surface distance* and the *CDF of the angular density* of the Purkayastha distribution in Section 7.3.3. We use those formulas to compare our directional with traditional baseline mechanisms in Section 7.3.6 and assess their error.
- In Section 7.3.5, we make use of our closed-form solution for the angular CDF to build an *approximate inversion sampling method* for the Purkayastha distribution. To our best knowledge, this is the first published method for this distribution which has been deemed numerically hard to sample from in dimensions over 150 [90]. Our benchmarks show that it is applicable in up to tens of thousands of dimensions.
- We apply our proposed mechanisms in several real-world settings and compare their privacy–utility trade-offs: We consider the *periodic mean* in the central and local privacy models for time-of-day data in Section 7.4.3, as well as *histograms* of location and time-of-day data in the local model in Section 7.4.4. We also illustrate privately collecting check-in time *and* location data to create "busyness" histograms of popular visit times even if the data curator is untrusted.
- Finally, we perform supplementary *simulation experiments* in Section 7.4.2 to support the correctness of our derived formula for the expected distances and CDFs. Based

on the empirical expected distances, we also compare the *privacy-utility trade-off* for both mechanisms at a given privacy level.

7.2 Directional Statistics

This section introduces key concepts and results of directional statistics as required in this dissertation. For further information on the subject, we recommend the book by Mardia and Jupp [278].

To deal with directional distributions and describe their mathematical properties, we sometimes need special mathematical functions. Since not all readers may be familiar with them, we provide supplementary information about these special functions and their notation in Section 7.2.5. Readers familiar with the matter may skip that part or refer to it as needed.

7.2.1 The Unit Sphere

Directional statistics is an area of statistics that is concerned with directions, i.e., data points whose magnitudes can be neglected. Since directions are independent of magnitude, they can be identified by unit vectors, i.e., points on a unit sphere:

Definition 7.1. For $n \in \mathbb{N}$, the *unit* $(n-1)$*-sphere*

$$\mathbb{S}^{n-1} := \{x \in \mathbb{R}^n : \|x\|_2 = 1\}$$

is the set of unit vectors in n-dimensional Euclidean space. We write $r\mathbb{S}^{n-1}$ for the $(n-1)$-sphere of radius $r > 0$.

Fact 7.2. *The* surface area *of the unit sphere $\mathbb{S}^{n-1}$ is given by its $(n-1)$-dimensional volume*

$$S_{n-1} := \operatorname{vol}\left(\mathbb{S}^{n-1}\right) = 2 \cdot \pi^{\frac{n}{2}} \cdot \Gamma^{-1}\left(\frac{n}{2}\right).$$

For a sphere of radius r, we have $\operatorname{vol}(r\mathbb{S}^{n-1}) = S_{n-1} r^{n-1}$.

Example 7.3. The uniform distribution $\mathsf{Uni}(\mathbb{S}^{n-1})$ on $\mathbb{S}^{n-1}$ has a constant PDF

$$\mathsf{Uni}(\mathbb{S}^{n-1})[\mathbf{x}] \equiv S_{n-1}^{-1} = \frac{1}{2}\Gamma\left(\frac{n}{2}\right)\pi^{-\frac{n}{2}}.$$

7.2.2 Rotationally Symmetric Distributions

We consider unimodal distributions on $\mathbb{S}^{n-1}$ that are rotationally symmetric about the *mode* $\boldsymbol{\mu} \in \mathbb{S}^{n-1}$. The corresponding densities $\mathsf{P}[\mathbf{x}]$ depend on $\mathbf{x}$ only through the projection

$t = \boldsymbol{\mu}^\mathsf{T}\mathbf{x}$ of $\mathbf{x}$ on the *modal axis* through $\boldsymbol{\mu}$, so all points $\mathbf{x}$ with $\boldsymbol{\mu}^\mathsf{T}\mathbf{x} = t$ have constant density $\mathsf{P}[\mathbf{x}] = \bar{\mathsf{P}}[\boldsymbol{\mu}^\mathsf{T}\mathbf{x}] = \bar{\mathsf{P}}[t]$ for a corresponding *kernel function* $\bar{\mathsf{P}} : [-1, 1] \to \mathbb{R}_{\geq 0}$.

To sample from such distributions, it is helpful to consider *marginal distributions* that are easier to handle. A way to obtain them is through the so-called *tangent-normal decomposition* (cf. Fig. 7.1): A random vector $\mathbf{x} \in \mathbb{S}^{n-1}$ can be decomposed into two components along the mode $\boldsymbol{\mu}$ and along a tangential unit vector $\boldsymbol{\xi} \perp \boldsymbol{\mu}$ as

$$\mathbf{x} = t\boldsymbol{\mu} + \sqrt{1 - t^2}\boldsymbol{\xi}, \tag{7.1}$$

where $t = \boldsymbol{\mu}^\mathsf{T}\mathbf{x}$ is the length along $\boldsymbol{\mu}$ (marked in blue).

Mixture Density. Due to the rotational symmetry, $\boldsymbol{\xi} \in \mathbb{S}^{n-2} \perp \boldsymbol{\mu}$ is distributed uniformly on the subsphere orthogonal to $\boldsymbol{\mu}$ (green circle in Fig. 7.1). Following Ulrich [446], the length $t = \boldsymbol{\mu}^\mathsf{T}\mathbf{x}$ along $\boldsymbol{\mu}$ (marked in blue) is called the *mixture variable*. Its associated *mixture density*

$$\mathsf{PMix}[t] = \int_{\mathbf{x}:\boldsymbol{\mu}^\mathsf{T}\mathbf{x}=t} \mathsf{P}[\mathbf{x}]\,\mathrm{d}\mathbf{x}, \quad t \in [-1, 1],$$

can be evaluated as follows:

Lemma 7.4 (Mixture density)**.** *Given a rotationally symmetric distribution* P *with kernel function* $\bar{\mathsf{P}}[t]$*, we can express its* mixture density $\mathsf{PMix}[t]$ *in terms of the kernel function as*

$$\mathsf{PMix}[t] = S_{n-2} \cdot \left(1 - t^2\right)^{\frac{n-3}{2}} \cdot \bar{\mathsf{P}}[t], \quad t \in [-1, 1]. \tag{7.2}$$

Proof. Since $\mathsf{P}[\mathbf{x}] = \bar{\mathsf{P}}[t]$ for $t = \boldsymbol{\mu}^\mathsf{T}\mathbf{x}$ and t is fixed in the integral, we can pull out the kernel function and obtain

$$\mathsf{PMix}[t] = \int_{\mathbf{x}:\boldsymbol{\mu}^\mathsf{T}\mathbf{x}=t} \mathsf{P}[\mathbf{x}]\,\mathrm{d}\mathbf{x} = \bar{\mathsf{P}}[t] \int_{\mathbf{x}:\boldsymbol{\mu}^\mathsf{T}\mathbf{x}=t} 1\,\mathrm{d}\mathbf{x}.$$

To evaluate the remaining integral, first note that the points $\mathbf{x} \in \mathbb{S}^{n-1}$ with $\boldsymbol{\mu}^\mathsf{T}\mathbf{x} = t$ form an $(n-2)$-dimensional subsphere centered at $t\boldsymbol{\mu}$ with radius $\sqrt{1-t^2}$ that is orthogonal to $\boldsymbol{\mu}$. By Fact 7.2, its $(n-2)$-dimensional surface area is

$$\mathrm{vol}\left(\sqrt{1 - t^2}\mathbb{S}^{n-2}\right) = S_{n-2} \cdot \left(1 - t^2\right)^{\frac{n-2}{2}}.$$

The angle between the modal axis and the subsphere in terms of t is $\arccos(t)$; with respect to the differential $\mathrm{d}t$, the subsphere's width on the surface along $\boldsymbol{\mu}$ hence is

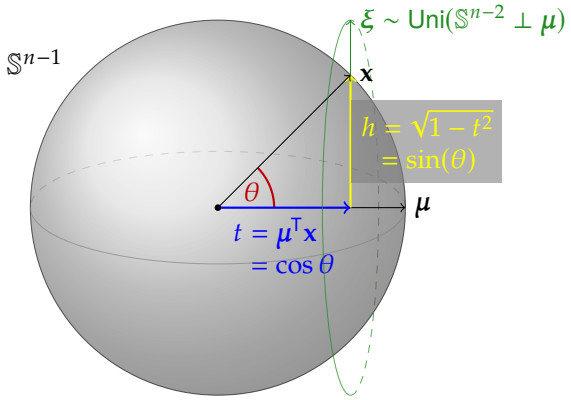


Figure 7.1: Tangent-normal decomposition of a random unit vector $\mathbf{x}$ into orthogonal components along the mode $\boldsymbol{\mu}$ and a tangential vector $\boldsymbol{\xi} \perp \boldsymbol{\mu}$ of lengths t and h, respectively.

$|\mathrm{d}/\mathrm{d}t \arccos(t)| = 1/\sqrt{1-t^2}$. Overall, the surface element amounts to $S_{n-2} \cdot \left(1-t^2\right)^{(n-3)/2}$, so we can express the mixture density in terms of the kernel function $\bar{\mathsf{P}}$ as

$$\mathsf{PMix}[t] = S_{n-2} \cdot \left(1-t^2\right)^{\frac{n-3}{2}} \cdot \bar{\mathsf{P}}[t]. \qquad \square$$

Angular Density. We obtain an alternative representation of the tangent-normal decomposition of $\mathbf{x} \in \mathbb{S}^{n-1}$ by substituting $t = \cos(\theta)$ in Eq. (7.1),

$$\mathbf{x} = \cos(\theta)\boldsymbol{\mu} + \sin(\theta)\boldsymbol{\xi}, \tag{7.3}$$

where $\theta = \arccos(\boldsymbol{\mu}^\mathsf{T}\mathbf{x})$ is the angle or arc length between $\mathbf{x}$ and the mode $\boldsymbol{\mu}$ (marked in red). The *angular density* of θ is as follows:

Corollary 7.5 (Angular density)**.** *Given a rotationally symmetric distribution* P *with kernel function* $\bar{\mathsf{P}}[t]$*, we can express its angular density* $\mathsf{PArc}[\theta]$ *for an angle* $\theta \in [0, \pi]$ *as*

$$\mathsf{PArc}[\theta] = S_{n-2} \sin^{n-2}(\theta) \cdot \bar{\mathsf{P}}[\cos(\theta)].$$

Proof. This follows from Lemma 7.4 by a change of variables $\theta = \arccos(\boldsymbol{\mu}^\mathsf{T}\mathbf{x}) = \arccos(t)$. $\square$

Importantly, the tangent-normal decomposition thus reduces the multivariate sampling problem $\mathbf{x} \sim \mathsf{P}$ to a univariate one, namely $t \sim \mathsf{PMix}$ or $\theta \sim \mathsf{PArc}$, plus a uniform one, $\boldsymbol{\xi} \sim \mathsf{Uni}(\mathbb{S}^{n-2} \perp \boldsymbol{\mu})$. This avoids the curse of dimensionality since the mixture or angular densities are one-dimensional, and uniform samples from a hypersphere are easily created by normalizing samples from a (multivariate) standard normal distribution.

7.2.3 The Von Mises–Fisher Distribution

The $(n-1)$-dimensional VMF distribution, named after von Mises [454] and Fisher [146], is a probability distribution on the unit hypersphere $\mathbb{S}^{n-1}$. Due to its popularity, it has been studied thoroughly, and proven sampling methods have been published previously [249, 446, 473] (see Section 7.3.4.1). Therefore, we use it as a starting point to construct a first novel privacy mechanism for directional data in Section 7.3.2.

Definition 7.6. The *VMF distribution* on $\mathbb{S}^{n-1}$ with *mean direction* $\boldsymbol{\mu} \in \mathbb{S}^{n-1}$ and *concentration* parameter $\kappa \geq 0$ is given by the density

$$\mathsf{VMF}(\boldsymbol{\mu}, \kappa)[\mathbf{x}] = C_{\mathsf{VMF}}(n, \epsilon) \cdot \exp\left(\kappa \cdot \boldsymbol{\mu}^{\mathsf{T}}\mathbf{x}\right).$$

If we set $\nu := \frac{n}{2} - 1$, the normalization factor amounts to

$$C_{\mathsf{VMF}}(n, \kappa) = \frac{\kappa^{\nu}}{(2\pi)^{\nu+1} I_{\nu}(\kappa)} = \frac{\Gamma(\nu+1)e^{\kappa}}{2 \cdot \pi^{\nu+1} M\left(\nu + \frac{1}{2}, 2\nu+1, 2\kappa\right)}.$$

The parameter κ characterizes how strongly the random vectors $\mathbf{x} \sim \mathsf{VMF}(\boldsymbol{\mu}, \kappa)$ are concentrated about the mean $\boldsymbol{\mu}$. If $\kappa > 0$, the distribution is unimodal and the mode matches $\boldsymbol{\mu}$. A VMF distribution with $\kappa = 0$ degenerates to the uniform distribution $\mathsf{Uni}(\mathbb{S}^{n-1})$.

7.2.4 The Purkayastha Distribution

Purkayastha [355] studied rotationally symmetric distributions on $\mathbb{S}^{n-1}$ for which the median direction is a maximum likelihood estimate of the location parameter. He proposed the following distribution that meets this criterion; in Section 7.3.3, we use it for a second mechanism for directional data.

Definition 7.7. The *Purkayastha distribution* on $\mathbb{S}^{n-1}$ with mean direction $\boldsymbol{\mu} \in \mathbb{S}^{n-1}$ and concentration parameter $\kappa \geq 0$ has density

$$\mathsf{Pur}(\boldsymbol{\mu}, \kappa)[\mathbf{x}] = C_{\mathsf{Pur}}(n, \kappa) \cdot \exp\Big(-\kappa \cdot \arccos(\boldsymbol{\mu}^{\mathsf{T}}\mathbf{x})\Big).$$

Its normalization factor is $C_{\mathsf{Pur}}(n, \kappa) = S_{n-2}^{-1} F_{n-2,-\kappa}^{-1}(\pi)$, where

$$F_{n-2,-\kappa}^{-1}(\pi) = \begin{cases} \dfrac{\kappa(\kappa^2+2^2)(\kappa^2+4^2)\cdots\left(\kappa^2+(n-2)^2\right)}{(n-2)!(1-e^{-\kappa\pi})} & \text{for even } n, \\ \dfrac{(\kappa^2+1^2)(\kappa^2+3^2)\cdots(\kappa^2+(n-2)^2)}{(n-2)!(1+e^{-\kappa\pi})} & \text{for odd } n \end{cases}$$

(cf. Lemma 7.23). Note that F also appears in the normalization constant of the angular and mixture densities in Section 7.3.3.1.

7.2.5 Special Functions and Notation

Directional distributions (as considered in Sections 7.2 and 7.3) often depend on certain special functions and their properties. We hereafter provide an overview of those functions used in this dissertation, and briefly summarize their key properties and relations. Further details can be found, for example, in Abramowitz and Stegun [8] or Gradshteyn and Ryzhik [166].

7.2.5.1 Gamma and Beta Functions

Definition 7.8. The *gamma function* (or *Euler integral of the second kind*) is defined for $z \in \mathbb{C}$ with real part $\Re(z) > 0$ as

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} \, \mathrm{d}t. \tag{7.4}$$

Important special values are $\Gamma(1) = 1$ and $\Gamma(\frac{1}{2}) = \sqrt{\pi}$. It has a functional relation $\Gamma(z+1) = z\Gamma(z)$, so $\Gamma(n+1) = n!$ for $n \in \mathbb{N}$. Therefore, Γ provides an extension of the *factorial* to complex numbers.

Definition 7.9. The *Pochhammer symbol* (or *rising factorial*) with k factors is defined as

$$(a)_k \coloneqq \frac{\Gamma(a+k)}{\Gamma(a)} = a(a+1)\cdots(a+k-1) \tag{7.5}$$

with the convention that $(a)_0 = 1$.

Definition 7.10. The *Beta function* (or *Euler integral of the first kind*) is defined for $x, y \in \mathbb{C}$ with real parts $\Re(x), \Re(y) > 0$ as

$$\mathrm{B}(x,y) = \int_0^1 t^{x-1}(1-t)^{y-1} \, \mathrm{d}t. \tag{7.6}$$

It is symmetric in its arguments. Particular relations are:

$$\mathrm{B}(x,y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)} \tag{7.7}$$

$$\mathrm{B}\left(\tfrac{1}{2}, x\right) = 2^{2x-1}\mathrm{B}(x,x) \tag{7.8}$$

7.2.5.2 Confluent Hypergeometric Series

Definition 7.11. *Kummer's (confluent hypergeometric) function*, denoted by $M(\alpha;\gamma;z)$ or ${}_1F_1(\alpha;\gamma;z)$, is a confluent hypergeometric series given by Kummer [246, 247] as

$$M(\alpha;\gamma;z) = {}_1F_1(\alpha;\gamma;z) = \sum_{k=0}^{\infty} \frac{(\alpha)_k}{(\gamma)_k} \frac{z^k}{k!}. \tag{7.9}$$

For $\Re\gamma > \Re\alpha > 0$, it has an integral representation

$$M(\alpha;\gamma;z) = \frac{\Gamma(\gamma)}{\Gamma(\alpha)\Gamma(\gamma-\alpha)} \int_0^1 t^{\alpha-1}(1-t)^{\gamma-\alpha-1} e^{zt}\,\mathrm{d}t. \tag{7.10}$$

Definition 7.12. The *modified Bessel function of the first kind* of order $\nu \in \mathbb{R}$ is given by the series

$$I_\nu(z) = \sum_{k=0}^{\infty} \frac{1}{k!\Gamma(k+\nu+1)} \left(\frac{x}{2}\right)^{2k+\nu}. \tag{7.11}$$

For $\Re(\nu) > -\frac{1}{2}$, it can be represented as integral, e.g.

$$I_\nu(z) = \frac{\left(\frac{z}{2}\right)^\nu}{\Gamma\left(\nu+\frac{1}{2}\right)\Gamma\left(\frac{1}{2}\right)} \int_{-1}^{1} \left(1-t^2\right)^{\nu-\frac{1}{2}} e^{\pm zt}\,\mathrm{d}t. \tag{7.12}$$

We can express $I_\nu(z)$ in terms of Kummer's function:

$$I_\nu(z) = \frac{e^{-z}}{\Gamma(\nu+1)} \left(\frac{z}{2}\right)^\nu M\left(\nu+\tfrac{1}{2}, 2\nu+1; 2z\right) \tag{7.13}$$

Humbert Series. Humbert [198, 199] introduced a set of seven hypergeometric double series that generalize Kummer's confluent hypergeometric series to two variables. One example we use is

Definition 7.13. The *Humbert series* Φ_1 is defined for $|x| < 1$ by a confluent hypergeometric series of two variables

$$\Phi_1(\alpha,\beta,\gamma;x,y) = \sum_{m,n=0}^{\infty} \frac{(\alpha)_{m+n}(\beta)_m}{(\gamma)_{m+n}} \frac{x^m y^n}{m!n!}. \tag{7.14}$$

For $\Re(\gamma) > \Re(\alpha) > 0$, it has an integral representation

$$\frac{\Gamma(\gamma)}{\Gamma(\alpha)\Gamma(\gamma-\alpha)} \int_0^1 t^{\alpha-1}(1-t)^{\gamma-\alpha-1}(1-xt)^{-\beta} e^{yt}\,\mathrm{d}t. \tag{7.15}$$

7.3 Directional Privacy Mechanisms

This section presents our main results. This comprises a novel notion of privacy for directional data as well as the conforming von Mises–Fisher and Purkayastha mechanisms. We derive certain marginal densities, expected values, and CDFs of the underlying distributions. These are important for assessing the average error, or sampling, as we show by constructing a novel Purkayastha sampling method. Moreover, we explain how the mechanism parameters depend on the desired privacy guarantees. Lastly, we describe adaptions of common privacy mechanisms to directional data as baselines.

7.3.1 Directional Privacy

Our goal is to define a variant of metric privacy [65] (Definition 2.5) for directions. To this end, we first need a suitable metric to measure distances between directions, i.e., angles on the sphere:

Definition 7.14. The *surface distance* between two points $\boldsymbol{x}, \boldsymbol{y} \in r\mathbb{S}^{n-1}$ is given by the arc length

$$d_\measuredangle(\boldsymbol{x}, \boldsymbol{y}) \coloneqq r \arccos(\boldsymbol{x}^\mathsf{T}\boldsymbol{y}).$$

On the unit sphere ($r = 1$), the surface distance $d_\measuredangle$ between two points is the enclosed angle (in radians) between them—together, $\mathbb{S}^{n-1}$ with $d_\measuredangle$ becomes a metric space for angles. We thus obtain

Definition 7.15 (Directional privacy)**.** Let $\epsilon > 0$. A mechanism $\mathcal{M}$ on $\mathbb{S}^{n-1}$ fulfills *$\epsilon d_\measuredangle$-privacy* if for all $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{S}^{n-1}$ and all $Z \subset \operatorname{supp} \mathcal{M}$,

$$\mathcal{M}(\boldsymbol{x})[Z] \leq \exp(\epsilon \cdot d_\measuredangle(\boldsymbol{x}, \boldsymbol{x}')) \cdot \mathcal{M}(\boldsymbol{x}')[Z].$$

Interpretation as Pure Differential Privacy. Following Chatzikokolakis et al. [65, Fact 5], ϵd-privacy on a space $\mathcal{Y}$ implies $\epsilon\Delta$-DP for a query function $f : \mathcal{D} \to \mathcal{Y}$ with d-sensitivity Δ on the universe of databases $\mathcal{D}$. We apply this fact specifically to *sphere-valued* functions with range $\mathcal{Y} \subseteq \mathbb{S}^{n-1}$ to obtain ϵ-DP:

Fact 7.16 (ϵd-privacy implies ϵ-DP)**.** *Let $f : \mathcal{D} \to \mathbb{S}^{n-1}$ be a query function, and let $\mathcal{M}_\epsilon$ be an ϵd-private mechanism on $\mathbb{S}^{n-1}$ with metric d. Then its d-*sensitivity *is $\Delta = \Delta_d f := \max_{x \sim_{\mathcal{D}} y} d(f(x), f(y))$, and the composition $\mathcal{M}_{\epsilon/\Delta} \circ f$ is ϵ-differentially private.*

7.3.2 Von Mises–Fisher Privacy Mechanism

The Laplace and Gaussian distributions are often used in Euclidean space, particularly as mechanisms to provide DP. Since the VMF distribution can be seen as a natural counterpart on the sphere, we propose it as a promising candidate to achieve DP for directional data:

Theorem 7.17 (ϵd_2-privacy of VMF mechanism)**.** *Let $\epsilon > 0$ be a privacy parameter. The* VMF *mechanism on $\mathbb{S}^{n-1}$ induced by $\boldsymbol{x} \mapsto \mathsf{VMF}(\boldsymbol{x}, \epsilon)$ for $\boldsymbol{x} \in \mathbb{S}^{n-1}$ fulfills ϵd_2-privacy.*

Proof. Let $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{S}^{n-1}$ be any fixed unit vectors, and take any fixed set $\mathbf{Z} \subseteq \mathbb{S}^{n-1}$. For any $\mathbf{z} \in \mathbf{Z}$, we have

$$\begin{aligned}\frac{\mathsf{VMF}(\boldsymbol{x}, \epsilon)[\mathbf{z}]}{\mathsf{VMF}(\boldsymbol{y}, \epsilon)[\mathbf{z}]} &= \frac{C_{\mathsf{VMF}} \cdot \exp\left(\epsilon \cdot \boldsymbol{x}^\mathsf{T}\mathbf{z}\right)}{C_{\mathsf{VMF}} \cdot \exp\left(\epsilon \cdot \boldsymbol{y}^\mathsf{T}\mathbf{z}\right)} \\ &= \exp\Big(\epsilon \cdot (\boldsymbol{x} - \boldsymbol{y})^\mathsf{T}\mathbf{z}\Big) \\ &\leq \exp(\epsilon \cdot \|\boldsymbol{x} - \boldsymbol{y}\|_2 \cdot \|\mathbf{z}\|_2) \\ &= \exp(\epsilon \cdot d_2(\boldsymbol{x}, \boldsymbol{y})).\end{aligned}$$

First, the normalization constants cancel out, and we can combine the exponents; the inequality is the Cauchy–Schwarz inequality; finally, note that $\|\mathbf{z}\|_2 = 1$. By integrating over $\mathbf{z} \in \mathbf{Z}$, we achieve ϵd_2-privacy. □

From there, we easily achieve directional privacy:

Corollary 7.18 ($\epsilon d_\measuredangle$-privacy of VMF mechanism)**.** *For any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{S}^{n-1}$, it holds that $d_2(\boldsymbol{x}, \boldsymbol{y}) \leq d_\measuredangle(\boldsymbol{x}, \boldsymbol{y})$, so the VMF mechanism fulfills $\epsilon d_\measuredangle$-privacy.*

By Fact 7.16, the VMF mechanism $\mathsf{VMF}(\boldsymbol{x}, \epsilon/\Delta)$ also provides ϵ-DP for sphere-valued functions $f : \mathcal{D} \to \mathbb{S}^{n-1}$ on the space of databases $\mathcal{D}$. Note that in this case, we can use the sensitivity Δ of f with respect to either $d_\measuredangle$ (by Corollary 7.18) or d_2 (by Theorem 7.17).

7.3.2.1 Von Mises–Fisher Marginal Densities

By Lemma 7.4 and Corollary 7.5, the mixture and angular densities of a VMF distribution are

$$\mathsf{VMFMix}[t] = C'_{\mathsf{VMF}} \cdot \left(1 - t^2\right)^{\frac{n-3}{2}} e^{\kappa t}, \tag{7.16}$$

$$\mathsf{VMFArc}[\theta] = C'_{\mathsf{VMF}} \cdot \sin^{n-2}(\theta)\, e^{\kappa \cos(\theta)}, \tag{7.17}$$

where the normalization factor amounts to

$$\begin{aligned} C'_{\mathsf{VMF}} &= C_{\mathsf{VMF}} \cdot S_{n-2} \\ &= \left(\frac{\kappa}{2}\right)^{\nu} \left(\Gamma\left(\tfrac{1}{2}\right)\Gamma\left(\tfrac{n-1}{2}\right) I_{\nu}(\kappa)\right)^{-1} \\ &= e^{\kappa} \cdot \mathrm{B}^{-1}\left(\tfrac{1}{2}, \tfrac{n-1}{2}\right) \cdot M^{-1}\left(\tfrac{n-1}{2}; n-1; 2\kappa\right). \end{aligned} \tag{7.18}$$

The mixture density is used in the rejection sampling scheme for the VMF distribution by Ulrich [446] and Wood [473], and is based on earlier work by Saw [396]. We use it next for the expected distance.

7.3.2.2 Expected Euclidean Distance

To assess the error induced by a mechanism, we can use statistical tools such as the expected value of an error measure based on the underlying distribution. Concretely, for a random vector $\mathbf{x} \sim \mathsf{VMF}(\boldsymbol{\mu}, \kappa)$, we provide an analytical expression for the expected L^2 distance to the mode $\boldsymbol{\mu}$:

Theorem 7.19. *The expected Euclidean distance between a random vector $\mathbf{x} \sim \mathsf{VMF}(\boldsymbol{\mu}, \kappa)$ and the mode $\boldsymbol{\mu}$ can be expressed as expected value over the mixture density. It evaluates to*

$$\begin{aligned} \mathbf{E}_{\mathbf{x}\sim\mathsf{VMF}}[d_2(\mathbf{x}, \boldsymbol{\mu})] &= \mathbf{E}_{t\sim\mathsf{VMFMix}}[\sqrt{2}\sqrt{1-t}] \\ &= \frac{\mathrm{B}\left(\frac{1}{2}, \frac{n}{2}\right)}{\mathrm{B}\left(\frac{1}{2}, n - \frac{1}{2}\right)} \frac{M\left(\frac{n-1}{2}; n - \frac{1}{2}; 2\kappa\right)}{M\left(\frac{n-1}{2}; n-1; 2\kappa\right)}. \end{aligned} \tag{7.19}$$

Proof. Because of the rotational symmetry, we can write $d_2(\boldsymbol{x}, \boldsymbol{\mu}) = \sqrt{2}\sqrt{1-t}$ in terms of the mixture variable t. Therefore,

$$\begin{aligned} \mathbf{E}_{\mathbf{x}\sim\mathsf{VMF}}[d_2(\mathbf{x}, \boldsymbol{\mu})] &= \mathbf{E}_{t\sim\mathsf{VMFMix}}[\sqrt{2}\sqrt{1-t}] \\ &= \int_{-1}^{1} \sqrt{2}\sqrt{1-t} \cdot \mathsf{VMFMix}[t]\, \mathrm{d}t \end{aligned}$$

$$= C'_{\mathsf{VMF}}\sqrt{2}\int_{-1}^{1} e^{\kappa t}\sqrt{1-t}(1-t^2)^{\frac{n-3}{2}}\,\mathrm{d}t$$
$$= C'_{\mathsf{VMF}}\sqrt{2}\int_{-1}^{1} e^{\kappa t}(1-t)^{\frac{n-2}{2}}(1+t)^{\frac{n-3}{2}}\,\mathrm{d}t.$$

Changing variables $t \mapsto \frac{t+1}{2}$ results in

$$= C'_{\mathsf{VMF}}2^{n-1}e^{-\kappa}\int_{0}^{1} e^{2\kappa t}(1-t)^{\frac{n-2}{2}}t^{\frac{n-3}{2}}\,\mathrm{d}t,$$

where we can express the integral as a Kummer function by Eq. (7.10), and then simplify with the normalization constant:

$$\begin{aligned}
&= C'_{\mathsf{VMF}}2^{n-1}e^{-\kappa}\mathrm{B}\left(\tfrac{n-1}{2},\tfrac{n}{2}\right)M\left(\tfrac{n-1}{2};n-\tfrac{1}{2};2\kappa\right)\\
&= 2^{n-1}\frac{\mathrm{B}\left(\frac{n-1}{2},\frac{n}{2}\right)}{\mathrm{B}\left(\frac{1}{2},\frac{n-1}{2}\right)}\frac{M\left(\frac{n-1}{2};n-\frac{1}{2};2\kappa\right)}{M\left(\frac{n-1}{2};n-1;2\kappa\right)}\\
&= \frac{\mathrm{B}\left(\frac{1}{2},\frac{n}{2}\right)}{\mathrm{B}\left(\frac{1}{2},n-\frac{1}{2}\right)}\frac{M\left(\frac{n-1}{2};n-\frac{1}{2};2\kappa\right)}{M\left(\frac{n-1}{2};n-1;2\kappa\right)}.
\end{aligned}$$

The last step follows by rewriting the fraction of Beta functions where we apply Eq. (7.8) to expand the numerator to $2^{1-n}\cdot\mathrm{B}\left(\frac{1}{2},\frac{n}{2}\right)$. □

7.3.2.3 Mixture CDF

Kurz and Hanebeck [249] provide analytical solutions for the CDF of the VMF *angular distribution* in the context of sampling. While their solution is an analytical, closed-form expression of elementary functions when n is odd, it involves an infinite series in terms of special functions for even n.

In the following, we present a concise, analytic solution for the CDF of the VMF mixture distribution in terms of confluent hypergeometric series covering both odd and even dimensions:

Theorem 7.20. *Setting* $\alpha \coloneqq \frac{n-1}{2}$ *and* $x \coloneqq \frac{T+1}{2}$, *the CDF of the VMF mixture distribution* $\mathsf{VMFMix}(n,\kappa)$ *at* $T \in [-1,1]$ *can be written as*

$$\mathsf{VMFMix}(n,\kappa)[t \le T] = \frac{x^\alpha}{\alpha}\frac{\Phi_1(\alpha,1-\alpha,1+\alpha;x,2\kappa x)}{\mathrm{B}(\alpha,\alpha)M(\alpha,2\alpha,2\kappa)}. \tag{7.20}$$

Proof. With $\alpha \coloneqq \frac{n-1}{2}$ and $x \coloneqq \frac{T+1}{2}$, we obtain

$$\begin{aligned}\mathsf{VMFMix}(n,\kappa)[t \leq T] &= \int_{-1}^{T} \mathsf{VMFMix}(n,\kappa)[t]\,\mathrm{d}t \\ &= C'_{\mathsf{VMF}} \int_{-1}^{T} \left(1-t^2\right)^{\frac{n-3}{2}} e^{\kappa t}\,\mathrm{d}t.\end{aligned}$$

Changing variables $t \mapsto \frac{t+1}{T+1}$ yields an integral that we can express as Humbert series according to Eq. (7.15), so we get

$$\begin{aligned}&= C'_{\mathsf{VMF}} 2^{n-2} x^{\frac{n-1}{2}} e^{-\kappa} \cdot \int_{0}^{x} e^{2\kappa x t} t^{\frac{n-3}{2}} (1-xt)^{\frac{n-3}{2}}\,\mathrm{d}t \\ &= \frac{x^\alpha}{\alpha} \frac{\Phi_1(\alpha, 1-\alpha, 1+\alpha; x, 2\kappa x)}{\mathrm{B}(\alpha,\alpha) M(\alpha, 2\alpha, 2\kappa)}.\end{aligned}$$

□

7.3.3 Purkayastha Privacy Mechanism

The VMF distribution enjoys wide popularity among spherical distributions, and provides differential as well as d_2- and $d_\measuredangle$-privacy as shown in the previous section. However, we also observe potential shortcomings, namely the probability decreases exponentially with the squared L^2 distance from the mode, i.e., the distance is measured on a straight line *through* the sphere. Instead, we would rather have it decrease exponentially with the *surface distance on* the sphere, i.e., with $\arccos(\boldsymbol{\mu}^\mathsf{T}\boldsymbol{x})$. It turns out that this is precisely the distribution in Definition 7.7 studied by Purkayastha [355]. We immediately obtain a corresponding *Purkayastha privacy mechanism* as follows:

Theorem 7.21 ($\epsilon d_\measuredangle$-privacy of Purkayastha mechanism)**.** *Let $\epsilon > 0$ be a privacy parameter. The* Purkayastha mechanism on $\mathbb{S}^{n-1}$ *induced by $\boldsymbol{x} \mapsto \mathsf{Pur}(\boldsymbol{x}, \epsilon)$ for $\boldsymbol{x} \in \mathbb{S}^{n-1}$ fulfills $\epsilon d_\measuredangle$-privacy.*

Proof. Let $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{S}^{n-1}$ be any fixed unit vectors, and take any fixed set $\mathbf{Z} \subseteq \mathbb{S}^{n-1}$. For any $\mathbf{z} \in \mathbf{Z}$, we have

$$\begin{aligned}\frac{\mathsf{Pur}(\boldsymbol{x},\epsilon)[\mathbf{z}]}{\mathsf{Pur}(\boldsymbol{y},\epsilon)[\mathbf{z}]} &= \frac{C_{\mathsf{Pur}} \cdot \exp\left(-\epsilon \cdot \arccos(\boldsymbol{x}^\mathsf{T}\mathbf{z})\right)}{C_{\mathsf{Pur}} \cdot \exp\left(-\epsilon \cdot \arccos(\boldsymbol{y}^\mathsf{T}\mathbf{z})\right)} \\ &= \exp\left(\epsilon \cdot \arccos(\boldsymbol{y}^\mathsf{T}\mathbf{z}) - \arccos(\boldsymbol{x}^\mathsf{T}\mathbf{z})\right) \\ &\leq \exp\left(\epsilon \cdot \arccos(\boldsymbol{x}^\mathsf{T}\boldsymbol{y})\right) \\ &= \exp\left(\epsilon \cdot d_\measuredangle(\boldsymbol{x},\boldsymbol{y})\right).\end{aligned}$$

First, the normalization constants cancel out, and we can combine the exponents; next, we apply the triangle inequality for the angular (arccos) distance. By integrating over $\mathbf{z} \in \mathbf{Z}$, we obtain $\epsilon d_\measuredangle$-privacy. □

By Fact 7.16, the Purkayastha mechanism $\mathsf{Pur}(\boldsymbol{x}, \epsilon/\Delta)$ also provides ϵ-DP for sphere-valued functions $f : \mathcal{D} \to \mathbb{S}^{n-1}$ with $d_\measuredangle$-sensitivity Δ on the space of databases $\mathcal{D}$.

7.3.3.1 Purkayastha Marginal Densities

By Lemma 7.4 and Corollary 7.5, we obtain the Purkayastha mixture and angular densities as

$$\mathsf{PurMix}[t] = C'_{\mathsf{Pur}} \cdot \left(1 - t^2\right)^{\frac{n-3}{2}} e^{-\kappa \arccos(t)}, \tag{7.21}$$

$$\mathsf{PurArc}[\theta] = C'_{\mathsf{Pur}} \cdot \sin^{n-2}(\theta)\, e^{-\kappa\theta}, \tag{7.22}$$

with normalization factor $C'_{\mathsf{Pur}} = C_{\mathsf{Pur}} \cdot S_{n-2} = F^{-1}_{n-2,-\kappa}(\pi)$.

Integrating the Angular Density. Having derived an expression for the angular density $\mathsf{PurArc}[\theta]$, we are interested in statistical properties such as its expected value to assess the average error, or the angular CDF $\mathsf{PurArc}[\theta \leq \vartheta]$ which is fundamental for the sampling algorithm we propose in Section 7.3.4.2.

The angular density is specified through a function $e^{ax} \sin^n x$, where $n \in \mathbb{N}$ and $a \in \mathbb{R}$. Gradshteyn and Ryzhik [166, 2.662] provide separate closed-form expressions for its antiderivative for even and odd n. We rewrite these expressions and provide the following unified solution which allows to efficiently evaluate such integrals:

Fact 7.22. *An antiderivative of $e^{ax} \sin^n x$ with $n \in \mathbb{N}$ and $a \in \mathbb{R}$ is*

$$E_{n,a}(x) \coloneqq e^{ax} \sum_{k=0}^{m} C_k \mathcal{T}_k(x), \tag{7.23}$$

where $m = \lfloor n/2 \rfloor$,

$$C_k = \frac{n!}{(n-2k)!} \prod_{\ell=0}^{k} \frac{1}{\left(a^2 + (n-2\ell)^2\right)}, \quad \textit{and}$$

$$\mathcal{T}_k(x) = \sin^{n-2k-1}(x)[a \sin(x) - (n-2k)\cos(x)].$$

In particular, the definite integral over $[0, r]$ *is given by*

$$F_{n,a}(r) \coloneqq \int_0^r e^{ax} \sin^n x \, \mathrm{d}x = E_{n,a}(r) - E_{n,a}(0). \tag{7.24}$$

A special case is the normalization factor $C'_{\mathsf{Pur}} = F^{-1}_{n-2,-\kappa}(\pi)$:

Lemma 7.23. *The integral* $F_{n,a}(\pi) = \int_0^\pi e^{ax} \sin^n x \, \mathrm{d}x$ *evaluates to*

$$F_{n,a}(\pi) = \begin{cases} \dfrac{n!(e^{a\pi} - 1)}{a(a^2 + 2^2)(a^2 + 4^2) \cdots (a^2 + n^2)} & \text{for even } n, \\ \dfrac{n!(e^{a\pi} + 1)}{(a^2 + 1^2)(a^2 + 3^2) \cdots (a^2 + n^2)} & \text{for odd } n. \end{cases} \tag{7.25}$$

Proof. Note that for any $\ell \in \mathbb{Z}$,

$$\mathcal{T}_k(\ell\pi) = \begin{cases} 0 & \text{for } k < m, \\ a & \text{for } k = m \text{ and even } n, \\ (-1)^{\ell+1} & \text{for } k = m \text{ and odd } n. \end{cases}$$

Therefore,

$$\begin{aligned} E_{n,a}(\ell\pi) &= e^{a\ell\pi} C_m \mathcal{T}_m(\ell\pi) \\ &= e^{a\ell\pi} C_m \begin{cases} a & \text{if } n \text{ is even}, \\ (-1)^{\ell+1} & \text{if } n \text{ is odd}, \end{cases} \end{aligned}$$

and we obtain as special case the normalization constant

$$F_{n,a}(\pi) = C_m \begin{cases} a(e^{a\pi} - 1) & \text{if } n \text{ is even}, \\ (e^{a\pi} + 1) & \text{if } n \text{ is odd}. \end{cases} \quad \square$$

7.3.3.2 Expected Surface Distance

We provide a closed-form solution for the expected angle of a Purkayastha random vector as follows:

Theorem 7.24. *The expected surface distance (or angle) between a random point* $\mathbf{x} \sim \mathsf{Pur}(\boldsymbol{\mu}, \kappa)$

and the mode $\boldsymbol{\mu} \in \mathbb{S}^{n-1}$ *can be expressed as expected angular density. It evaluates to*

$$\begin{aligned}\mathbf{E}_{\mathbf{x}\sim\mathsf{Pur}}[d_{\measuredangle}(\mathbf{x},\boldsymbol{\mu})] &= \mathbf{E}_{\theta\sim\mathsf{PurArc}}[\theta] \\ &= 2\kappa \sum_{\ell=1}^{m} A_\ell + \begin{cases} \dfrac{\pi}{1-e^{\kappa\pi}} - \dfrac{1}{\kappa} & \text{if } n \text{ is even,} \\ \dfrac{\pi}{1+e^{\kappa\pi}} & \text{if } n \text{ is odd,} \end{cases}\end{aligned} \tag{7.26}$$

where $A_\ell \coloneqq \left(\kappa^2 + (n-2\ell)^2\right)^{-1}$ *for* $1 \leq \ell \leq m \coloneqq \left\lfloor \frac{n}{2} \right\rfloor$.

Proof. The surface distance $\theta = d_{\measuredangle}(\mathbf{x},\boldsymbol{\mu})$ follows the angular distribution $\theta \sim \mathsf{PurArc}(n,\kappa)$. Therefore, we have

$$\begin{aligned}\mathbf{E}_{\mathbf{x}\sim\mathsf{Pur}}[d_{\measuredangle}(\mathbf{x},\boldsymbol{\mu})] &= \mathbf{E}_{\theta\sim\mathsf{PurArc}}[\theta] \\ &= C'_{\mathsf{Pur}} \int_0^\pi \theta e^{-\kappa\theta} \sin^{n-2}(\theta)\,\mathrm{d}\theta.\end{aligned}$$

Since $\theta e^{-\kappa\theta} = -\frac{\partial}{\partial\kappa} e^{-\kappa\theta}$, we can apply Leibniz' rule and Lemma 7.23:

$$\begin{aligned}&= -C'_{\mathsf{Pur}} \int_0^\pi \frac{\partial}{\partial\kappa} e^{-\kappa\theta} \sin^{n-2}(\theta)\,\mathrm{d}\theta \\ &= -C'_{\mathsf{Pur}} \frac{\partial}{\partial\kappa} \int_0^\pi e^{-\kappa\theta} \sin^{n-2}(\theta)\,\mathrm{d}\theta \\ &= \frac{\frac{\partial}{\partial\kappa} F_{n-2,-\kappa}(\pi)}{-F_{n-2,-\kappa}(\pi)}.\end{aligned} \tag{7.27}$$

The result follows from applying the generalized product rule. □

7.3.3.3 Angular CDF

We provide the following expression for the CDF of the angular distribution PurArc in terms of Eqs. (7.24) and (7.25):

Corollary 7.25. *The CDF of the Purkayastha angular distribution* $\mathsf{PurArc}(n,\kappa)$ *is*

$$\mathsf{PurArc}[\theta \leq \vartheta] = C'_{\mathsf{Pur}} \int_0^\vartheta e^{-\kappa\theta} \sin^{n-2}(\theta)\,\mathrm{d}\theta = \frac{F_{n-2,-\kappa}(\vartheta)}{F_{n-2,-\kappa}(\pi)}. \tag{7.28}$$

Note that this is a closed-form solution that can be efficiently evaluated in terms of finite sums $E_{n,a}(x)$ (Fact 7.22) and the formula for $F_{n,a}(\pi)$ (Lemma 7.23) for both odd and even n. This is crucial for the Purkayastha sampling method we develop in the next section.

7.3.4 Sampling Algorithms

In this section, we discuss concrete algorithms for our directional privacy mechanisms, i.e., to generate samples from the underlying distributions. For some general intuition on sampling rotationally symmetric distributions, we refer the reader to Section 7.2.2. Due to its popularity, the VMF distribution has been studied extensively, and proven sampling methods already have been published; two of them we describe in Section 7.3.4.1. In contrast, no methods have been published so far for the Purkayastha distribution. Therefore, in Section 7.3.4.2, we contribute the first sampling algorithm for the Purkayastha distribution.

7.3.4.1 Von Mises–Fisher Sampling Methods

To generate a point $\mathbf{x} \sim \mathsf{VMF}(\boldsymbol{\mu}, \kappa)$, we can employ the existing rejection scheme by Ulrich [446] and Wood [473]: Pursuant to Section 7.2.2, it involves two crucial steps: First, the tangent-normal decomposition $\mathbf{x} = t\boldsymbol{\mu} + \sqrt{1-t^2}\boldsymbol{\xi}$ in Eq. (7.1) reduces the *multivariate* sampling problem to a *univariate* one, namely sampling $t \leftsquigarrow \mathsf{VMFMix}(n, \kappa)$ from the mixture distribution, as well as a direction vector $\boldsymbol{\xi} \leftsquigarrow \mathsf{Uni}(\mathbb{S}^{n-2} \perp \boldsymbol{\mu})$. This avoids the curse of dimensionality since the mixture density is one-dimensional, and uniform samples from a hypersphere are easily created by normalizing samples from a (multivariate) standard normal distribution. Second, we need an efficient sampling algorithm for the reduced problem. A clever way to solve this is the *rejection method* [473, Algorithm VM*] for $\mathsf{VMFMix}(n, \kappa)$. Ulrich [446] showed that the acceptance ratio is *at least* $\approx 66\%$ for any parameters n and κ, resulting in a very efficient method even in high dimensions.

More recently, Kurz and Hanebeck [249] proposed another sampling algorithm for the VMF distribution that is best described as *approximate inversion method*. It works by substituting $t = \cos(\vartheta)$ in the tangent-normal decomposition as in Eq. (7.3) and constructing a sample $\mathbf{x} = \cos(\vartheta)\boldsymbol{\mu} + \sin(\vartheta)\boldsymbol{\xi}$. This reduces the problem to generating $\vartheta \leftsquigarrow \mathsf{VMFArc}(n, \kappa)$ from the univariate *angular* distribution. If the corresponding angular CDF $\mathsf{VMFArc}(n, \kappa)[\theta \leq \vartheta]$ was invertible analytically, a textbook version of the inversion method (see, e.g., [102]) could be used to sample ϑ. Kurz and Hanebeck solve this by *approximately* inverting the CDF: If we can efficiently compute the CDF $u = \mathsf{VMFArc}(n, \kappa)[\theta \leq \vartheta]$, we can approximate its inverse $\vartheta = \mathsf{VMFArc}(n, \kappa)^{-1}[u]$ numerically, e.g., by interval bisection, which is "guaranteed to converge up to machine precision" in a reasonable number of steps [249].

Unfortunately, their solution for $\mathsf{VMFArc}(n, \kappa)[\theta \leq \vartheta]$ is analytical only for odd n, while it contains an infinite series in terms of special functions for even n which we cannot evaluate efficiently. Therefore, this approach is only viable for VMF when n is odd, which is why we prefer the rejection scheme from the previous paragraph as it is fast and simple to use in general. However, we show next that this idea is useful for sampling the Purkayastha distribution.

7.3.4.2 Purkayastha Sampling Method

To our best knowledge, there is no published sampling method for the Purkayastha distribution. Cutting et al. [90] state that they generated samples for lower dimensions up to $n = 100$, but without specifying the exact method they used. Rather, they give the following explanation (emphasis ours):

> The Purkayastha distribution is *numerically hard to generate for dimensions larger than 150*, which is the only reason why the dimensions considered in this second simulation are smaller than in the first one.

Here, the "first" and "second simulation" refer to sampling from the VMF and Purkayastha distribution, respectively.

Algorithm 3: Approximate inversion method for the Purkayastha distribution.

Input: Dimension n, concentration parameter κ, max. no. of iterations $M \geq 1$, (optional: absolute tolerance δ_{abs})
Output: A sample $\vartheta \in [0, \pi]$ of $\mathsf{PurArc}(n, \kappa)$

$a \leftarrow 0; b \leftarrow \pi;$ // initial interval bounds
$u \leftsquigarrow \mathsf{Uni}(0, 1);$ // uniform sample
for $i \leftarrow 1$ **to** M **do**
  $\vartheta \leftarrow (a + b)/2;$
  $y \leftarrow \frac{F_{n-2,-\kappa}(\vartheta)}{F_{n-2,-\kappa}(\pi)};$ // evaluate $\mathsf{PurArc}[\theta \leq \vartheta]$
  if $|y - u| < \delta_{\text{abs}}$ **then** break; // (optional)
  if $y < u$ **then** $a \leftarrow \vartheta;$ // adjust lower,
  else if $y > u$ **then** $b \leftarrow \vartheta;$ // or upper bound
end
return ϑ

Approximate Inversion Purkayastha Sampling Algorithm. Recall that in Corollary 7.25, we have derived a solution for the angular CDF of the Purkayastha distribution,

$$\mathsf{PurArc}(n, \kappa)[\theta \leq \vartheta] = \frac{F_{n-2,-\kappa}(\vartheta)}{F_{n-2,-\kappa}(\pi)}.$$

While we are not aware of a way to directly compute its inverse to apply the inversion method [102] for $n > 2$, the solution itself is a finite closed-form expression that can be computed analytically. We hence propose an *approximate* inversion method, similar to the approach by Kurz and Hanebeck [249] for VMF in Section 7.3.4.1, to obtain a new Purkayastha sampling algorithm: Since we can efficiently compute the angular CDF $u = \mathsf{PurArc}(n, \kappa)[\theta \leq \vartheta]$ (cf. Corollary 7.25), we can approximate its inverse $\vartheta = \mathsf{PurArc}(n, \kappa)^{-1}[u]$ numerically.

We describe the core method to sample $\vartheta \leftsquigarrow \mathsf{PurArc}(n, \kappa)$ in Algorithm 3. Once we have a sample ϑ, we draw $\boldsymbol{\xi} \leftsquigarrow \mathsf{Uni}(\mathbb{S}^{n-2} \perp \boldsymbol{\mu})$ and as above use the tangent-normal decomposition Eq. (7.3) to construct

$$\mathbf{x} = \cos(\vartheta)\boldsymbol{\mu} + \sin(\vartheta)\boldsymbol{\xi} \sim \mathsf{Pur}(\boldsymbol{\mu}, \kappa).$$

Since our solution for the angular CDF is a closed-form expression with finitely many terms in *any* number of dimensions n, we argue that our approximate inversion method for the Purkayastha distribution is practical regardless of the parity of n.

Algorithm 3 can easily be vectorized to generate multiple samples at once, or parallelized to utilize multiple CPU cores. In fact, we benchmark our method in up to tens of thousands of dimensions (see Section 7.4.1), pushing beyond the status quo [90] by providing an efficient sampling algorithm in dimensions much larger than 150.

7.3.5 Choice of Parameters Based on Privacy Level

To actually run the proposed directional privacy mechanisms on a given input vector $\boldsymbol{x} \in \mathbb{S}^{n-1}$, we need to generate samples from $\mathsf{Pur}(\boldsymbol{x}, \kappa)$ or $\mathsf{VMF}(\boldsymbol{x}, \kappa)$ where the mode is given by the input $\boldsymbol{x}$ and the concentration parameter κ is defined through the privacy parameter ϵ. Having described sampling methods for both the VMF mechanism (cf. Section 7.3.4.1) and a novel sampling scheme for the Purkayastha mechanism (cf. Section 7.3.4.2), it remains to explain the exact choice of κ based on ϵ and the desired notion of privacy:

- Given a unit vector $\boldsymbol{x} \in \mathbb{S}^{n-1}$, in order to achieve *directional privacy* with privacy parameter ϵ, i.e. $\epsilon d_{\measuredangle}$-privacy (Definition 7.15), we simply need to set $\kappa = \epsilon$ and draw a sample $\mathbf{z} \leftsquigarrow \mathsf{Pur}(\boldsymbol{x}, \epsilon)$ or $\mathbf{z} \leftsquigarrow \mathsf{VMF}(\boldsymbol{x}, \epsilon)$ as shown in Corollary 7.18 and Theorem 7.21, respectively.

- *Metric privacy* (Definition 2.5) [65] and its variants can also be interpreted as providing

a privacy (or indistinguishability) level $\ell = \epsilon r$ to any two points $\boldsymbol{x}, \boldsymbol{x}'$ within a *protection radius* (or angle) $r > 0$, cf. [24]. In the case of directional privacy (Definition 7.15), this is achieved by sampling with $\kappa = \ell/r$. In other words, an (ℓ/r)-private mechanism achieves a privacy level ℓ within a protection radius r.

- As special case, when $\boldsymbol{x} = f(D)$ is the result of a (query) function $f : \mathcal{D} \to \mathbb{S}^{n-1}$, we achieve *pure ϵ-DP* by setting the protection radius $r := \Delta$ to the (worst-case) sensitivity of f, i.e., by sampling with a concentration parameter $\kappa = \epsilon/\Delta$ as per Fact 7.16. Thus, directional privacy allows relaxing pure DP by specifying a protection radius r smaller than the sensitivity Δ.

7.3.6 Circular and Spherical Baselines

For comparison, we consider the following adaptions of established standard privacy mechanisms to directional data. The first and second mechanisms, Clipped and Wrapped Laplace, are suitable for circular data ($n = 2$), whereas the third one, Polar Laplace, can be regarded as a variant of Wrapped Laplace for spherical data ($n = 3$).

7.3.6.1 Clipped Laplace

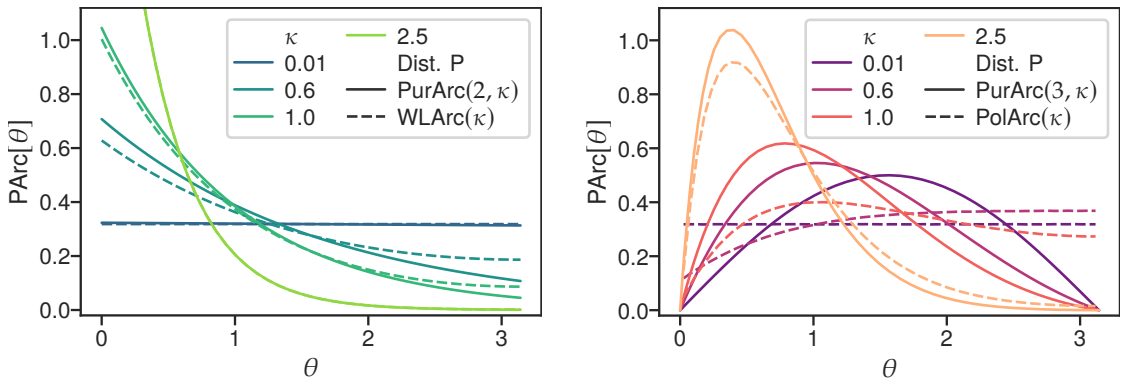
A straightforward application of the usual Laplace mechanism [117] with post-processing achieves DP on the circle by adding Laplace noise to a given angle, followed by clipping the result to an interval covering one full circle, say $[0, 2\pi)$ or $[-\pi, \pi)$. This method is simple, but clearly has drawbacks: For small ϵ, the major part of the probability mass will be outside the clipping range, creating a bias towards the angle at its boundaries. We therefore use it only in selected experiments.

7.3.6.2 Wrapped Laplace

Instead of clipping, we can add Laplace noise to the original angle α and wrap it around the circle by reducing the result modulo 2π. This results in a so-called (symmetric) *Wrapped Laplace (WL) distribution* with mean α. With the usual parametrization on the unit circle, the density of a WL distribution with zero mean and concentration parameter $\kappa \geq 0$ is (cf. [206])

$$\mathsf{WL}(\kappa)[\theta] = \frac{\kappa}{2}\left(\frac{e^{-\kappa\theta}}{1 - e^{-\kappa 2\pi}} + \frac{e^{\kappa\theta}}{e^{\kappa 2\pi} - 1}\right), \quad \theta \in [0, 2\pi). \tag{7.29}$$

Angular Density. In accordance with Corollary 7.5, the corresponding angular density WLArc is the density of points with the same angle from the mean, in any direction. That



(a) Angular PDFs on $\mathbb{S}^1$ (circular case, $n = 2$)

(b) Angular PDFs on $\mathbb{S}^2$ (spherical case, $n = 3$)

Figure 7.2: Comparison of angular densities of the Purkayastha distribution with (a) Wrapped Laplace and (b) Polar Laplace baselines (solid vs. dashed lines), respectively.

is, it identifies an angle $\theta \in [0, \pi)$ with its mirror image $2\pi - \theta$. By symmetry, it is just twice the density on the full circle:

$$\mathsf{WLArc}(\kappa)[\theta] = \kappa\left(\frac{e^{-\kappa\theta}}{1 - e^{-2\kappa\pi}} + \frac{e^{\kappa\theta}}{e^{2\kappa\pi} - 1}\right), \quad \theta \in [0, \pi). \tag{7.30}$$

While the Purkayastha angular density $\mathsf{PurArc}(2, \kappa)[\theta] \propto e^{-\kappa\theta}$ on $\mathbb{S}^1$ only has a single term $e^{-\kappa\theta}$, cf. Eq. (7.22), $\mathsf{WLArc}(\kappa)$ has an additional second term $e^{+\kappa\theta}$ that *increases* with the angle θ. The wrapping hence smoothens the distribution by moving probability mass away from the mode as illustrated in Fig. 7.2a. It hence provides less accuracy than Purkayastha at the same privacy level, thus motivating the need for specialized directional mechanisms.

Expected Angular Distance. Similarly to the derivation of the expected surface distance for the Purkayastha distribution from $\mathsf{PurArc}[\theta]$, we can derive the expected angular distance for the WL distribution from $\mathsf{WLArc}[\theta]$. The result is

$$\mathbf{E}_{\theta\sim\mathsf{WLArc}(\kappa)}[\theta] = \frac{1}{\kappa}\left(\frac{1}{1 + e^{-\kappa\pi}} - \frac{1}{1 + e^{\kappa\pi}}\right) = \frac{1}{\kappa}\frac{1 - e^{-\kappa\pi}}{1 + e^{-\kappa\pi}}. \tag{7.31}$$

For comparison, the expected angular distance of the circular Purkayastha distribution from Lemma 7.23 simplifies to

$$\mathbf{E}_{\theta\sim\mathsf{PurArc}(2,\kappa)}[\theta] = \frac{1}{\kappa} - \frac{\pi}{e^{\kappa\pi} - 1} = \frac{1}{\kappa} - \frac{\pi e^{-\kappa\pi}}{1 - e^{-\kappa\pi}}. \tag{7.32}$$

The formula for the expected angular distances allows us to analytically compare the average (angular) error induced by the distributions based on the concentration parameter

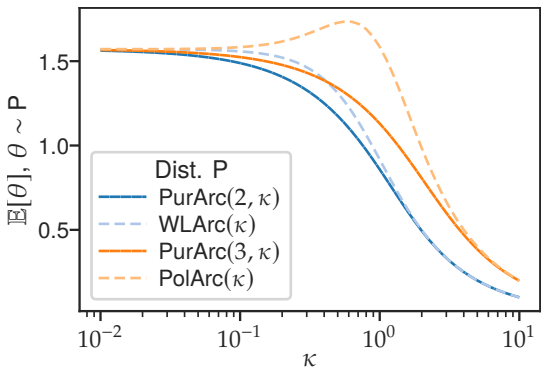


Figure 7.3: Comparison of expected angles between Purkayastha and Wrapped/Polar Laplace baselines (solid vs. dashed lines): The baselines show larger errors.

κ, which in turn depends on the privacy parameter ϵ (cf. Section 7.3.5):

Theorem 7.26. *For any value $\kappa > 0$, the WL distribution has a strictly larger expected angular distance than Purkayastha:*

$$\mathbf{E}_{\theta\sim\mathsf{WLArc}}[\theta] > \mathbf{E}_{\theta\sim\mathsf{PurArc}}[\theta] + \frac{\kappa\pi^2}{e^{2\kappa\pi}-1} > \mathbf{E}_{\theta\sim\mathsf{PurArc}}[\theta]$$

However, the expected angular distances of both distributions converge to the same limits as $\kappa \to 0, \infty$:

$$\lim_{\kappa\to 0} \mathbf{E}_{\theta\sim\mathsf{WLArc}}[\theta] = \lim_{\kappa\to 0} \mathbf{E}_{\theta\sim\mathsf{PurArc}}[\theta] = \frac{\pi}{2}$$
$$\lim_{\kappa\to \infty} \mathbf{E}_{\theta\sim\mathsf{WLArc}}[\theta] = \lim_{\kappa\to \infty} \mathbf{E}_{\theta\sim\mathsf{PurArc}}[\theta] = 0$$

Proof. We compute the exact difference between the expected values and apply the inequality $e^x \geq 1 + x$:

$$\begin{aligned}
\mathbf{E}_{\theta\sim\mathsf{WLArc}}[\theta] - \mathbf{E}_{\theta\sim\mathsf{PurArc}}[\theta] &= \frac{\pi}{e^{\kappa\pi}-1} - \frac{2}{\kappa(e^{\kappa\pi}+1)} \\
&= \frac{e^{\kappa\pi}(\kappa\pi-2)+\kappa\pi+2}{\kappa(e^{2\kappa\pi}-1)} \\
&\geq \frac{(1+\kappa\pi)(\kappa\pi-2)+\kappa\pi+2}{\kappa(e^{2\kappa\pi}-1)} \\
&= \frac{\kappa\pi^2}{e^{2\kappa\pi}-1} > 0.
\end{aligned}$$

□

The limits are trivial for $\kappa \to \infty$. For $\kappa \to 0$, they follow from l'Hôpital's rule.

Figure 7.3 shows expected angles of PurArc(2, κ) and WLArc(κ) (blue lines) for a range

of $\kappa \in [10^{-2}, 10]$. As we can see, the baseline has larger expected errors, which is in line with Theorem 7.26.

7.3.6.3 Polar Laplace

The *Planar Laplace (PL)* mechanism [24, 65] was originally invented in the context of protecting geolocation data. It can be considered as a two-dimensional variant of the standard Laplace mechanism that works in Cartesian coordinates by translating the initial starting point $\boldsymbol{x} \in \mathbb{R}^2$ by a certain distance r along a certain direction α. The distance r and direction α are polar coordinates obtained by sampling a random direction $\alpha \sim \mathsf{Uni}(0, 2\pi)$ and a *displacement radius* $r \sim \Gamma(2, 1/\epsilon)$ from a gamma distribution.

When applying the PL mechanism to spherical instead of Cartesian coordinates, we obtain the *Polar Laplace* mechanism[1] [66] that respects the curvature of the (roughly) spherical Earth: The initial point $\boldsymbol{x}$ is represented in spherical coordinates (e.g., latitude and longitude). We then draw a random sample of polar coordinates $(r, \alpha) \sim \Gamma(2, 1/\epsilon) \times \mathsf{Uni}(0, 2\pi)$ as with PL, and, as a post-processing step, solve the *direct geodesic problem*[2] to find the destination point $\mathbf{z}$ that is reached after traveling for a distance of r units in the direction specified by α. As with WL, we pass the starting point again every time a distance equal to the circumference of the sphere has been traversed; therefore, Polar Laplace can be regarded as a two-dimensional variant of the WL mechanism (cf. Section 7.3.6.2).

Angular Density and Expected Distance. In order to compare the Polar Laplace and Purkayastha mechanisms on the sphere $\mathbb{S}^2$, we again use their angular densities as auxiliary. We simulated 64M samples to approximate the angular density $\mathsf{PolArc}(\kappa)[\theta]$ and its expected value for θ. We compare it with the (exact) solutions for the three-dimensional Purkayastha angular density $\mathsf{PurArc}(3, \kappa)[\theta]$ and its expected value, as provided in Eq. (7.22) and Theorem 7.24.

Figure 7.2b shows the angular densities of the Purkayastha and Polar distributions. For all values of κ, $\mathsf{PurArc}(3, \kappa)[\theta]$ is higher near $\theta = 0$ and approaches 0 as $\theta \to \pi$, whereas $\mathsf{PolArc}(\kappa)[\pi]$ is strictly above 0. The expected angles of both spherical distributions are shown in Fig. 7.3 (orange lines) and approach 0 for $\kappa \to \pi$. As κ decreases from π to 0, $\mathbf{E}_{\theta \sim \mathsf{PurArc}}[\theta]$ steadily *rises* to $\frac{\pi}{2}$ and approaches the uniform distribution. In contrast, $\mathbf{E}_{\theta \sim \mathsf{PolArc}}[\theta]$ goes up to over 1.7 at $\kappa \approx 0.6$ (i.e., worse than the uniform distribution), and only then *falls* back to $\frac{\pi}{2}$, which is quite remarkable.

To explain this phenomenon, consider the expected displacement radius which amounts

[1] Implementation in `laplace.js` at https://github.com/chatziko/location-guard.
[2] Solution formula from https://www.movable-type.co.uk/scripts/latlong.html.

Figure 7.4: Sampling rates ($\times 10^3$) of the Purkayastha approximate inversion method (Algorithm 3, vectorized implementation) with various parameters.

to $\mathbf{E}_{r\sim\Gamma(2,1/\kappa)}[r] = \frac{2}{\kappa}$. For $\kappa \approx \frac{2}{\pi} \approx 0.637$, it is close to π, which is the farthest distance we can go from $\boldsymbol{x}$ to its antipodal point $-\boldsymbol{x}$ on $\mathbb{S}^2$; consequently, most random points will end up on the "wrong" hemisphere. This raises the expected angle $\mathbf{E}_{\theta\sim\mathsf{PolArc}}[\theta] \approx 1.733$ to over $\frac{\pi}{2}$ for such κ, indicating a *point of no return* where the distribution's mode reverses from $\boldsymbol{x}$ to $-\boldsymbol{x}$. Overall, these results indicate an advantage for Purkayastha over Polar Laplace, particularly for $\kappa \approx \frac{2}{\pi}$.

7.4 Experiments

In this section, we experimentally verify the proposed methods. We start by testing the efficiency of our novel Purkayastha sampling algorithm, which is crucial for the Purkayastha mechanism. We then apply our methods to real-world data: First, we analyze the impact of the privacy mechanisms on the circular mean and ranking statistics. Next, we consider temporal and spatial histograms from periodic times-of-day and geolocations on a spherical coordinate system. Finally, we compute "busyness" histograms indicating the activity or popularity of certain locations, such as stores or restaurants, over the course of a day, through a combined application of directional privacy mechanisms to both spatial *and* temporal check-in data.

Implementation. We use Python 3 for our experiments. Arithmetic and computations are based on `numpy` [331, 456] and `scipy` [453]. For confluent hypergeometric and special functions, we rely on the `mpmath` multi-precision library [212]. We implemented both sampling algorithms, Algorithm 3 for Purkayastha and the VMF rejection method by Ulrich [446] and Wood [473], with basic optimizations such as vectorization and JIT compilation via `Numba` [251].

7.4.1 Sampling Efficiency

To measure the efficiency of our proposed Purkayastha approximate inversion method, we run our implementation of Algorithm 3 with varying n and κ for at least 60 seconds and count the number of generated samples. Based on the counts and elapsed times, we compute the individual rate of samples per second. While single-threaded, our implementation uses vectorization to work on multiple samples simultaneously. The experiments were run in parallel on a 48-core Xeon Platinum 8259CL system with each instance corresponding to one parametrization (n, κ) of the PurArc distribution.

Results. Figure 7.4 shows the achieved sampling rate of our Purkayastha approximate inversion method in thousands of samples per second. We push beyond the status quo [90] by generating samples even in thousands of dimensions. Clearly, the rate decreases with the dimensionality n due to the increasing number of terms in Eq. (7.23) that is used to compute $\mathsf{PurArc}(n, \kappa)[\theta \leq \vartheta]$ (Corollary 7.25). Another factor is the concentration parameter κ: Larger values decrease the sampling rate first slightly, and then more pronounced for $\kappa \gtrsim 100$. However, with DP, we typically prefer low privacy losses ϵ that correspond to small values of κ (cf. Section 7.3.5)—and thus yield higher speeds.

Sampling rates of tens to over hundreds of thousands of samples per second clearly show that the Purkayastha approximate inversion method is practical in the low- to medium-dimensional setting. As the dimensionality n gets larger, however, the sampling rate decreases steadily until it will eventually become too low for the method to be practical. As this is an intrinsic issue with the method being based on a formula whose complexity increases with n, it leaves room for further research. Still, practical improvements to the current approach are possible, for instance by porting the Python code to a native language like C or parallelization on multiple cores. Lastly, we note that even fewer than hundreds of samples per second may be sufficient for many real-world applications, particularly in the local model where each participant perturbs just their own data (i.e., only few samples) prior to submitting it to a central server.

7.4.2 Empirical Verification through Simulation

The following experiments aim at verifying the analytic formula for the expected distances and CDFs of the VMF and Purkayastha distribution we derived in Section 7.3. We furthermore use them to compare the corresponding mechanisms' utility at a given privacy level.

7.4.2.1 Expected Distances

First, we want to check the correctness of the derived analytical formula Eq. (7.19) for the expected Euclidean distance $\mathbf{E}_{t\sim\mathsf{VMFMix}}\left[\sqrt{2}\sqrt{1-t}\right]$ of the VMF distribution as well as Eq. (7.26) for the surface distance $\mathbf{E}_{\theta\sim\mathsf{PurArc}}[\theta]$ of the Purkayastha distribution. To that end, we draw 1 million samples from each distribution $\mathsf{VMFMix}(n, \kappa)$ and $\mathsf{PurMix}(n, \kappa)$, and compute the empirical means of the corresponding Euclidean and angular distances. We chose $n \in \{2, 3, 25, 50, 100, 500\}$ and $\kappa \in \{10^k \mid -3 \leq k \leq 3\}$. We compare the thusly obtained empirical distances against the results given by the analytical formula we implemented in Python.

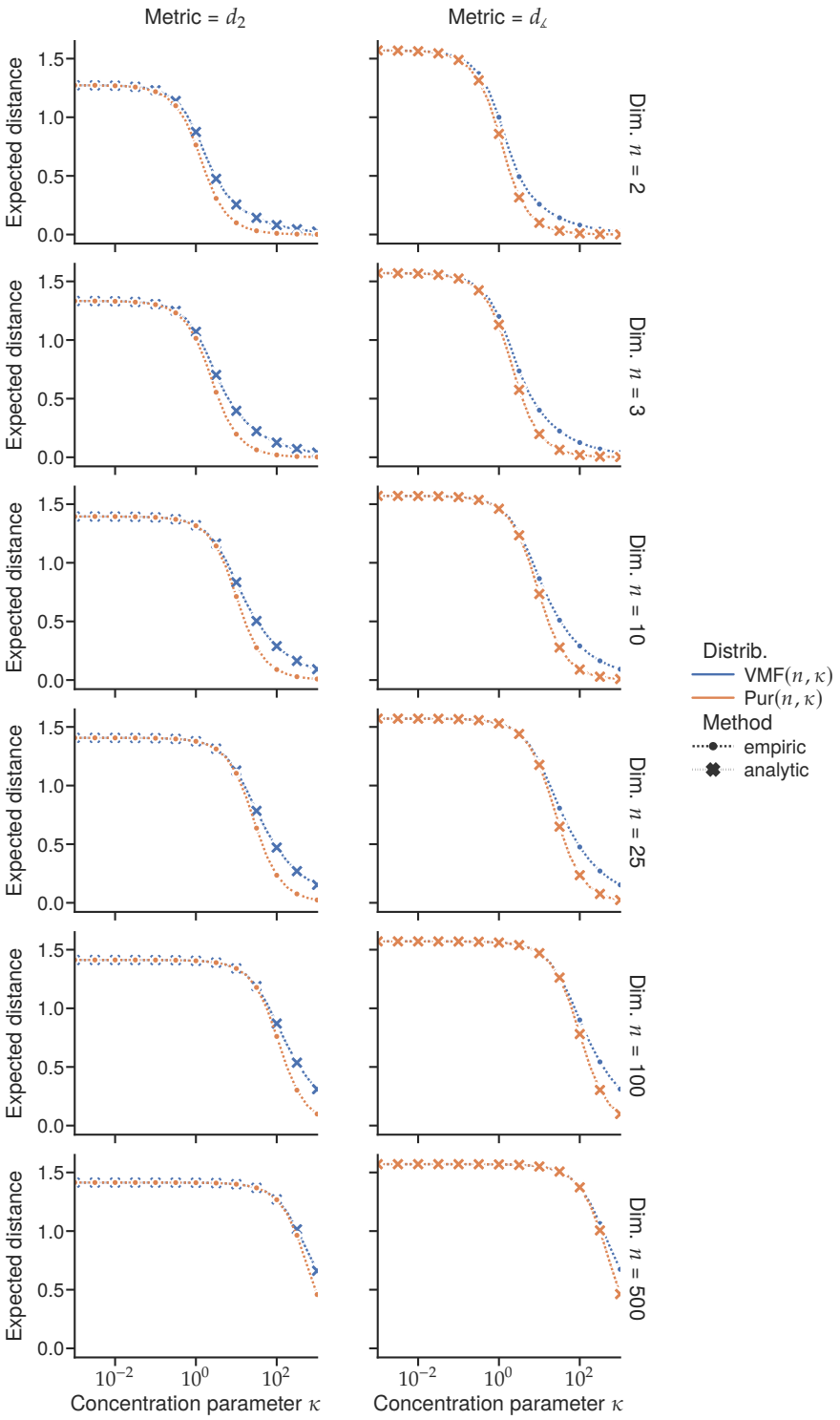


Figure 7.5: Expected d_2 and $d_\measuredangle$ distances for VMF and Purkayastha distributions in various settings. We obtained empirical averages (dotted lines) from 1M samples of each distribution and analytic solutions (X's) from Eqs. (7.19) and (7.26).

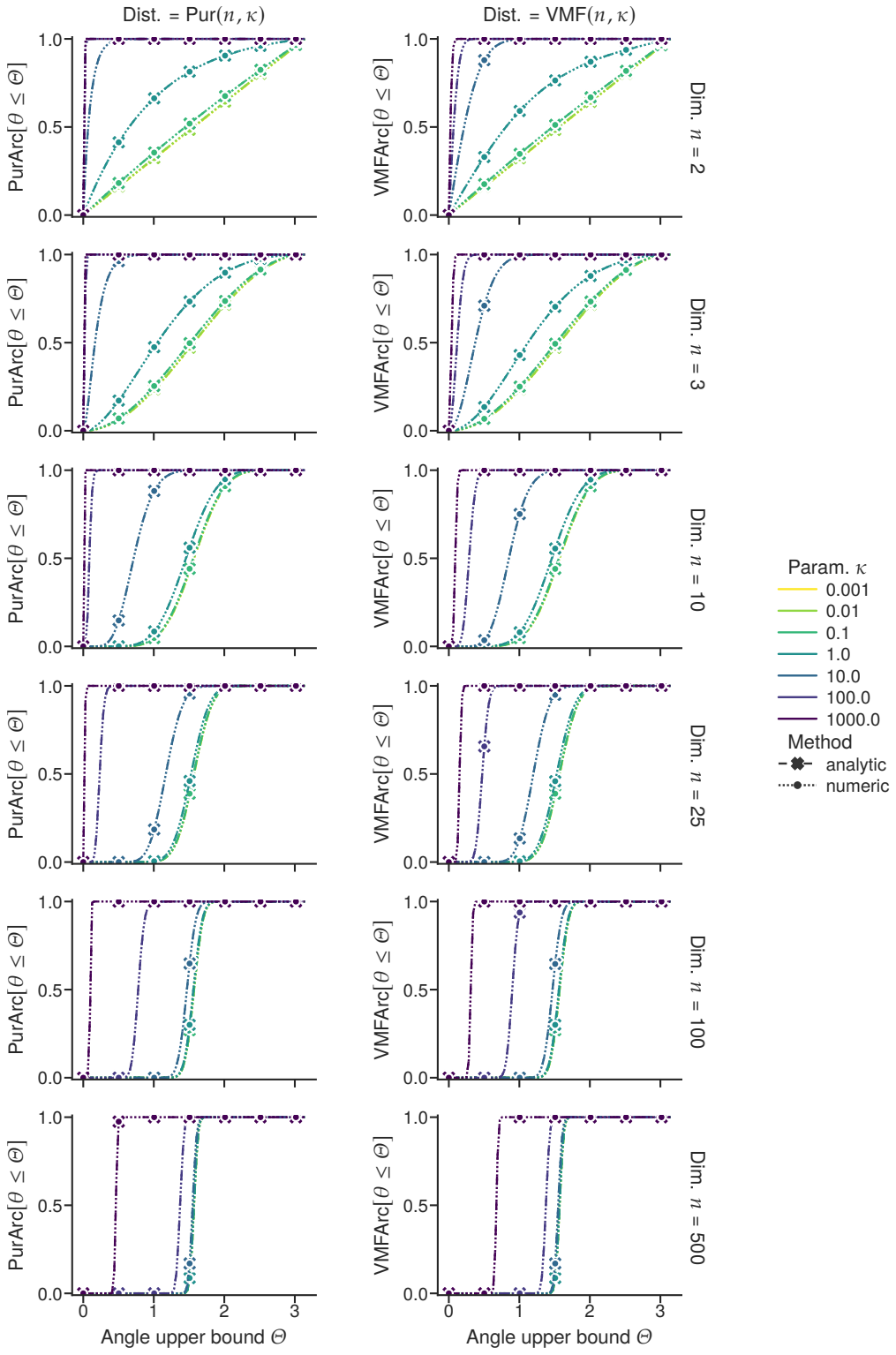


Figure 7.6: Angular CDFs of the VMF and Purkayastha distributions, obtained via numerical integration (dotted) of the PDFs and analytically (X's) via Eqs. (7.20) and (7.28).

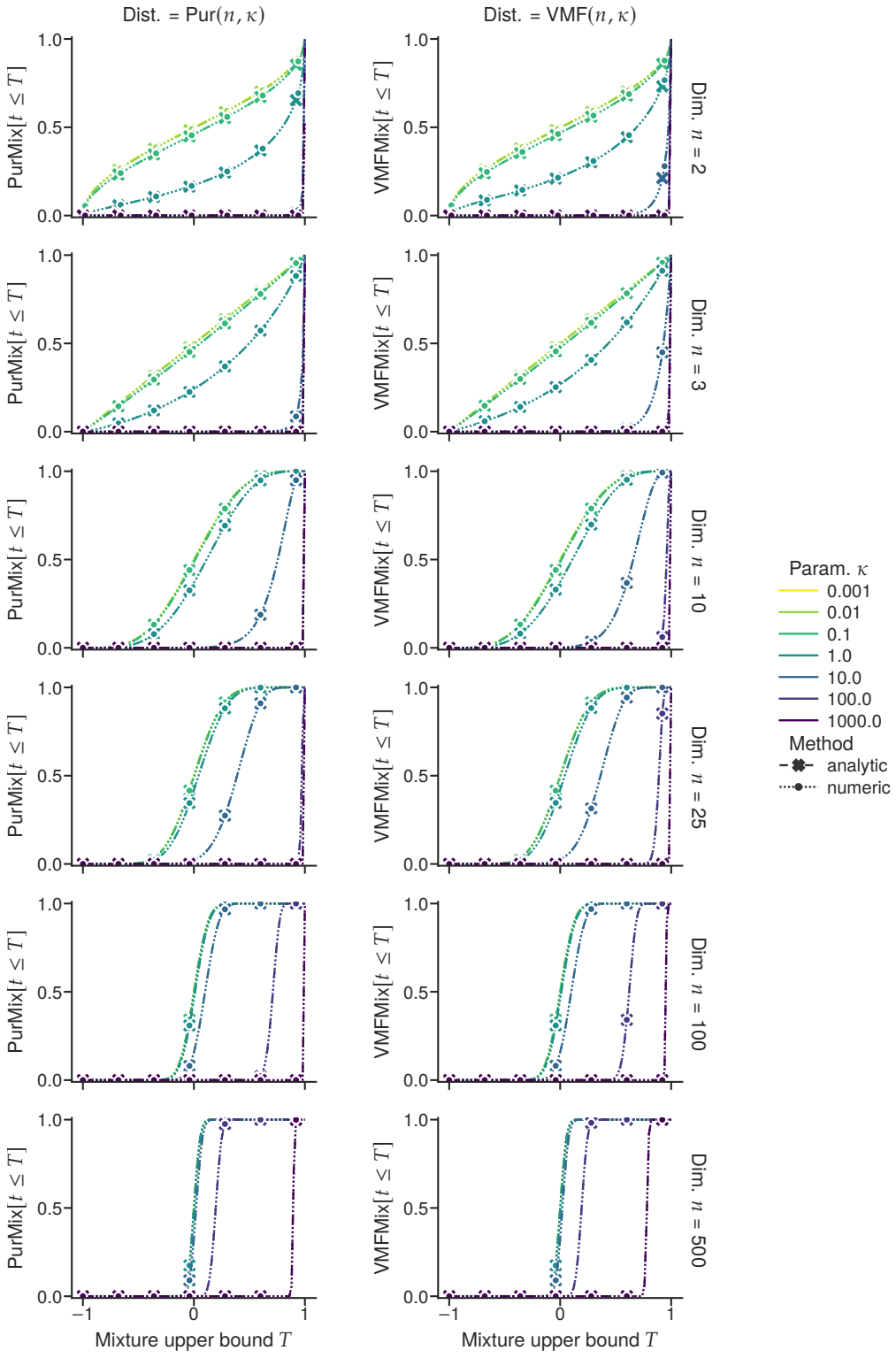


Figure 7.7: Mixture CDFs of the VMF and Purkayastha distributions, obtained via numerical integration (dotted) of the PDFs and analytically (X's) via Eqs. (7.20) and (7.28).

Results. As we can observe in Fig. 7.5, our analytical solution (marked by X's) precisely predicts the empirical distances, even in higher dimensions such as $n = 500$. Moreover, given the same concentration parameter κ, the Purkayastha distribution consistently has a lower expected distance (or error) which indicates that it has a generally more favorable privacy-utility trade-off.

7.4.2.2 Cumulative Distribution Functions

Second, we want to verify our formula for the CDFs $\mathsf{VMFMix}[t \leq T]$ given in Eq. (7.20) as well as $\mathsf{PurArc}[\theta \leq \vartheta]$ given in Eq. (7.28). As reference values, we numerically approximate the integrals over the probability density functions using the quadrature routines provided by `scipy`. We vary κ and n as in the previous experiment on expected distances.

Results. Note that we only derived analytical solutions for the *mixture* CDF for the VMF distribution and for the *angular* CDF for the Purkayastha distribution. Therefore, to give a complete picture, we show both the mixture and angular CDFs for both distributions by means of the transformations

$$\mathsf{VMFArc}[\theta \leq \vartheta] = 1 - \mathsf{VMFMix}[t \leq \cos(\vartheta)], \quad \text{and}$$
$$\mathsf{PurMix}[t \leq T] = 1 - \mathsf{PurArc}[\theta \leq \arccos(T)]$$

for better comparability. The results are presented in Fig. 7.6 for the angular CDFs and in Fig. 7.7 for the mixture CDFs. Again, we can observe that our analytical solutions (marked by X's) accurately predict the numerical approximations (dotted lines). While the CDFs of both distributions look similar for small κ, $\mathsf{PurArc}[\theta \leq \vartheta]$ grows more rapidly than $\mathsf{VMFArc}[\theta \leq \vartheta]$ as $\vartheta \to \pi$ for larger κ. Similar observations can be made for the mixture CDFs. Overall, this indicates a higher concentration near the mode and hence a better privacy-utility trade-off for the Purkayastha distribution.

7.4.3 Circular Mean on Periodic Data

The National Sleep Foundation (NSF) regularly conducts surveys of US citizens on their sleep habits including questions on their bed and wake times. Among the key reported figures in the surveys' findings are the average wake and bed times; these times-of-day are periodic on a 24-hour scale and hence provide a natural example of directional data that is suitable for directional privacy.

Scenario and Privacy Models. Suppose we work for a polling agency that wants to conduct a similar survey of sleeping habits, but with formal privacy guarantees as offered by differential privacy. The survey results with statistics such as average bed and wake times shall be made public or shared with another third party. We can distinguish two major approaches corresponding to the central and local privacy models introduced in Sections 2.2.1 and 2.2.2, respectively:

- In the *central model*, the survey participants trust the polling agency to handle their sensitive data confidentially. Hence, they faithfully report their unaltered answers to the agency. After the collection of all survey responses, the agency prepares the statistics from the original data and applies appropriate privacy mechanisms to sanitize the results, which can then be shared or made public.
- The *local model* can provide a suitable alternative if the survey participants do not trust the polling agency: Instead of providing faithful answers, the respondents first sanitize their answers themselves before reporting the altered responses back to the agency. From the collected obfuscated responses, the agency computes the desired statistics that can be publicized afterwards.

Circular Statistics. When taking the average or difference of periodic data, it is not sufficient to simply take the arithmetic mean or absolute distances. Instead, we must use periodic variants such as the *circular mean* which works by averaging the direction *vectors*, or the *circular distance* which takes the shortest path in any direction, clock- or counterclockwise, so two times differ by at most 12 hours.

Let $\boldsymbol{t} = (t_1 \ldots, t_N)$ be a sequence of real numbers. We write the usual *arithmetic mean* of $\boldsymbol{t}$ as $\varnothing(\boldsymbol{t}) = \varnothing(t_1, \ldots, t_N) \coloneqq \frac{1}{N}\sum_i t_i$. Now let us assume $\boldsymbol{t}$ is periodic with period (circumference of the circle) $p > 0$, i.e., each $t_i \in [0, p)$. Then the *circular mean* of $\boldsymbol{t}$ is

$$\varnothing_p(\boldsymbol{t}) = \varnothing_p(t_1, \ldots, t_N) \coloneqq \frac{p}{2\pi} \operatorname{arctan2}\left(\varnothing\left(\sin\frac{2\pi\boldsymbol{t}}{p}\right), \varnothing\left(\cos\frac{2\pi\boldsymbol{t}}{p}\right)\right).$$

The *circular difference* between p-periodic values s and t is

$$\delta_p(s,t) \coloneqq \begin{cases} p - (s-t) & \text{if } s - t > p/2, \\ p + (s-t) & \text{if } s - t < -p/2, \\ s - t & \text{otherwise,} \end{cases}$$

i.e., the signed difference equal to the positive clock- or negative counterclockwise arc length from t to s taking values in $[-p, p]$. The *circular distance* between s and t with values

in $[0, p]$ is

$$d_p(s,t) \coloneqq |\delta_p(s,t)|.$$

Note that the circular mean is highly sensitive to a change in the input: We can always construct a sequence $t_1 \ldots, t_N$ so that changing a single t_i will also cause the mean to point in the opposite direction. For instance, given a circle with period 2π, let $(t_1, t_2, t_3) = (\alpha, \pi - \alpha, \pm\frac{\pi}{2})$ for some small $\alpha > 0$. Therefore, we obtain the *same* sensitivity $\Delta_{\measuredangle} = \pi$ in the central *and* local model.

7.4.3.1 Local Model Advantage and Sample Complexity

Note that the central model normally has a lower sensitivity, so it injects less noise and hence is more accurate than the local model. However, in the case of the circular mean, we anticipate an *advantage for the local model*: Given a sufficient amount of noisy responses, the locally injected noise will gradually cancel out, resulting in more accurate statistics than in the central model. This is similar to the mean of i.i.d. Gaussians which has a lower variance than each Gaussian on its own. Moreover, the local model can be used in scenarios where the data curator cannot be or is not trusted by the participants.

To examine this effect in the case of the circular (and spherical) mean, we perform the following simulation experiment to determine the number of samples required to reach a certain accuracy: Let $t_{i,j} \leftsquigarrow \mathsf{P}$ denote i.i.d. samples from a circular distribution P with mean $\mu = 0$ ($1 \le i \le N$, $1 \le j \le R$). The number of samples required to maintain an average error below a given threshold $\tau \in [0, \pi]$ is

$$N_{\mathsf{P}}(\tau) \coloneqq \min\left\{ k \in \mathbb{N} : \frac{1}{R}\sum_{j=1}^{R} \varnothing_{2\pi}(t_{1,j}, \ldots, t_{i,j}) \le \tau \ \forall i \ge k \right\},$$

which we call the average *sample complexity of* P *at* τ over R runs.

Results. Figure 7.8 shows the sample complexity for the circular ($n = 2$, left) and spherical ($n = 3$, right) VMF, Purkayastha, as well as the Clipped and Wrapped ($n = 2$) or Polar ($n = 3$) Laplace mechanisms with sensitivity $\Delta = \pi$ over $R = 1000$ runs with $N = 10^8$ samples each. For large ϵ, Purkayastha and the Laplace baselines approximately require the same number of samples to reach a given threshold τ. For small ϵ, VMF has a slight advantage over Purkayastha since it can be used with the smaller d_2-sensitivity under pure DP. In this case, all directional mechanisms show a similar, gentle slope and clearly outperform the Laplace baselines, where CL performs worst. In fact, even 10^8 samples quickly become insufficient to reach the given thresholds for Polar Laplace in $n = 3$

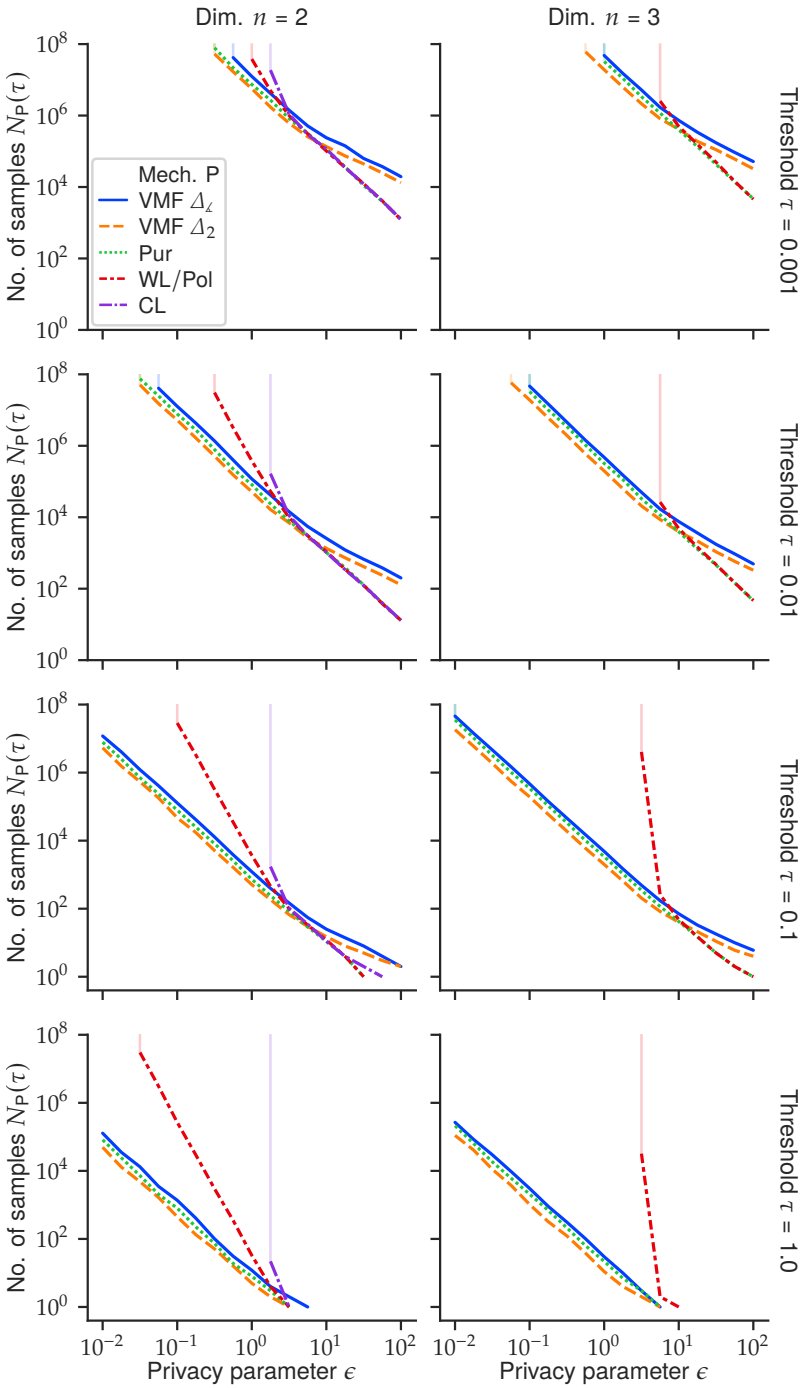


Figure 7.8: Sample complexity (mean over $R = 1000$ runs).

dimensions (thin vertical lines). This relates to Section 7.3.6.3, where we observe that Polar Laplace reaches an expected angle $1.733 > \frac{\pi}{2}$ for $\kappa \approx \frac{2}{\pi}$ (i.e., $\epsilon = \kappa\Delta_{\measuredangle} \approx 2$). The behavior of wrapped distributions thus causes utility *worse than a uniform distribution*.

For polls like the NSF survey, the impact is significant: To reach an error below 0.1 with $\epsilon = 1.0$, the service provider only needs to collect about 750 responses with Purkayastha instead of over 3600 with Wrapped Laplace, which represents an over 4.8-fold reduction. Conversely, given the same number of responses, our mechanisms achieve lower errors and higher accuracy as we see in the next Section 7.4.3.2. This makes the use of local DP practical under stricter conditions even with small privacy parameters $\epsilon \leq 10$.

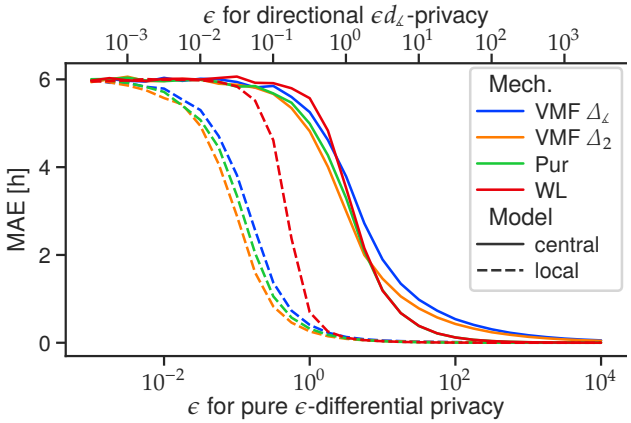
7.4.3.2 Average Wake Times and Ranking Statistics: Sleep Study

In the following experiments, we simulate a privacy-preserving survey in a real-world setting, in both the central and local privacy models. Note that in the central model, a savvy aggregator could choose to emulate the local model by first perturbing each collected value before aggregating them, *thus reaching the same level of accuracy as the local model*. However, to better illustrate the noise-cancelling effect in these experiments (i.e., throughout this very Section 7.4.3.2), we assume the central aggregator follows an ordinary (but naïve) approach that first aggregates the collected values and then applies noise only once.

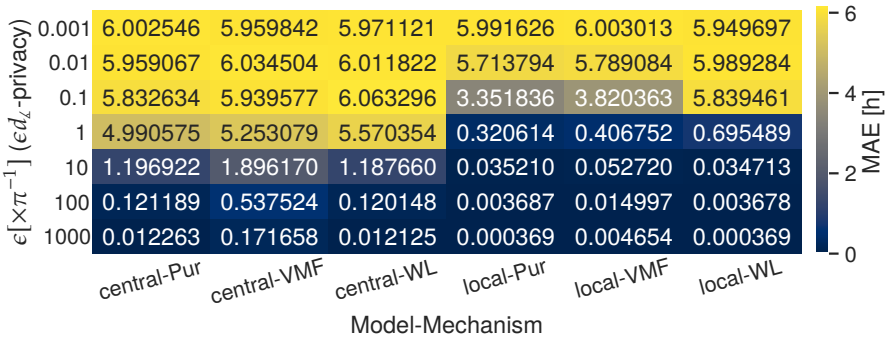
Dataset Description. We rely on the NSF's 2011 dataset [319], which includes a total of 1,508 survey responses. The questions include their bed and wake times, both on workdays and weekends. The respondents are divided into four age groups: Baby Boomers (46-64), Generation X (30-45), Y (19-29), and Z (13-18 years).

Sanitization Procedure and Parameters. To sanitize the times-of-day reported in the survey on a 24-hour scale, we need to express them as 2-dimensional unit vectors that we can use as mode of the VMF or Purkayastha distribution. This is easily achieved by assigning to each hour the corresponding angle (in radians) on a 24-hour clock, and then transforming these angles to coordinates via sine and cosine. Conversely, after perturbing the points with one of our new mechanisms, we transform the points back to the 24-hour scale using the inverse trigonometric *arc tangent* function.

For comparison, we also perturb the scalar data directly on the 24-hour scale by means of the standard Laplace mechanism [117]. Since Laplace noise can be arbitrarily positive or negative, we reduce the perturbed values modulo 24 to map the values back into the domain $[0, 24)$. In fact, this corresponds to a WL distribution on a 24-hour scale, as discussed in Section 7.3.6.1.



(a) MAE over various values of ϵ under directional and pure differential privacy (indicated by the top and bottom axis, respectively).



(b) Exemplary MAE values for various settings of the mechanisms (directional privacy; central and local model in cols. 1–3 and 4–6).

Figure 7.9: Comparison of the mean absolute error (MAE) between original and perturbed average wake times.

Let $\boldsymbol{t} = (t_1, \ldots, t_N)$ be the true times-of-day from the N participants. Depending on the privacy model, we proceed as follows: In the *central model*, we take the average $\bar{t} = \varnothing_{24}(\boldsymbol{t})$ of all truly reported times, and then perturb $\bar{t}$ using one of the privacy mechanisms. In the *local model*, we first perturb each participant's value t_i individually. Then, we compute the average from the perturbed values. The $d_\measuredangle$-sensitivity of the circular mean is $\Delta_\measuredangle = \pi$ radians, corresponding to 12 hours, *even in the central model* as changing a single input can cause the mean to flip to the opposite direction in the worst case. For VMF, we can also use the smaller d_2-sensitivity $\Delta_2 = 2$ corresponding to the diameter of the unit circle.

For each mechanism, VMF, Purkayastha, and Laplace, we vary the privacy parameter $\epsilon \in \{10^k \mid -4 \leq k \leq 3\}$ with step size $\Delta k = 0.2$. To stabilize the results, we repeat this procedure in each setting (privacy model, mechanism, and parameters) for $R = 10000$ runs, so we obtain a sequence $\tilde{\boldsymbol{t}} = (\tilde{t}_1, \ldots, \tilde{t}_R)$ of *anonymized* average times.

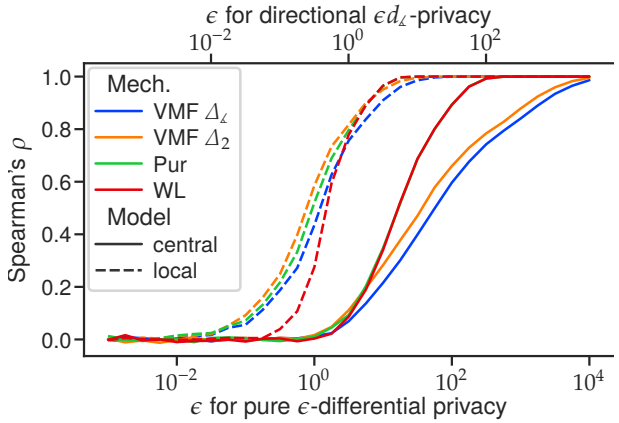


Figure 7.10: Comparison of Spearman's ρ across the four age groups, over ϵ under directional and pure differential privacy (indicated by the top and bottom axis, respectively).

Error Evaluation for Circular Mean. We take each anonymized time $\tilde{t}_i$, whose mean we denote by $\tilde{t} = \varnothing_{24}(\tilde{\boldsymbol{t}})$, as an estimate for the true average time $\bar{t}$. We want to estimate the error induced by the various privacy mechanisms on the average wake time compared to the original, unperturbed data. To this end, we chose the *mean absolute error (MAE)*, which is normally defined as $\frac{1}{R}\sum_{i=1}^{N}|\tilde{t}_i - \bar{t}|$. However, as noted earlier, we work with periodic data, so we must adapt the usual expression to its circular variant $\varnothing\big(d_p(\tilde{\boldsymbol{t}}, \bar{t})\big)$.

Figure 7.9a shows the MAE of the average wake time based on the original and perturbed values. In the local model (dashed lines), both directional privacy mechanisms clearly outperform WL across the entire range of privacy parameters ϵ. For *directional* privacy (top scale), Purkayastha shows the lowest errors due to its higher concentration at the mode. However, for *pure* DP (bottom scale), VMF can be employed with smaller d_2-sensitivity $\Delta_2 = 2 < \pi = \Delta_\measuredangle$ (orange line), which even outperforms Purkayastha in that case. In the central model (solid lines), WL and Purkayastha perform similarly well for large ϵ where VMF performs worst. However, in the strong privacy domain with small ϵ, WL is worst, with Purkayastha providing the best *directional* privacy guarantees and VMF with the reduced d_2-sensitivity yielding the best *differential* privacy guarantees for $\epsilon \lesssim 10^{0.25}$. Figure 7.9b lists exemplary MAE values specifically for *directional* privacy to support these observations with concrete numbers.

Strikingly, the local model outperforms the (naïve) central one in this experiment, which confirms what we anticipated in Section 7.4.3.1: The sensitivity of the circular mean is the same in both privacy models, where the locally injected noise gradually cancels out when many responses are averaged together, yielding lower errors. In both models, Purkayastha and VMF reach the lowest errors for a given directional and differential privacy parameter ϵ, respectively.

Ranking Statistics. In the context of the NSF's sleep study, one aspect is to compare the wake (or bed) times among different groups, and determine, e.g., who gets up first or goes to bed latest. Concretely, let us suppose we want to infer the order of wake-up times among the four age groups (Generation-X, -Y, -Z, and Baby Boomers) from the survey data. As a non-private baseline, we compute the average wake-up time for each group on the original dataset, and from there determine the ranking of the groups. We then simulate the survey being conducted in both the *central* and *local privacy models* as before, and determine the ranking of the age groups from the sanitized average wake-up times. To measure the impact of the privacy mechanisms on such statistics, we compute *Spearman's rank correlation coefficient* (also called *Spearman's* ρ) between the perturbed and original ranking of the four age groups.

Figure 7.10 shows Spearman's rank correlation coefficient ρ (averaged over all runs) for the different mechanisms over the parameter range of ϵ and both privacy models. As we can see, the observations on the rank correlation are in line with the observations on the mean absolute errors reported in the previous experiment.

In the *central model*, Purkayastha and Wrapped Laplace (WL) (overlapping green and red lines) achieve similar ρ values and both outperform VMF at virtually any given privacy level ϵ under both directional and differential privacy. However, in a small range of ϵ just below 1, Purkayastha shows a higher correlation than WL, and VMF with the d_2-sensitivity also overtakes WL under pure DP.

The *local model* generally shows a better privacy–utility trade-off than in the previous results. Notably, Purkayastha appears to reach the highest correlation values among the three mechanisms under directional privacy, at virtually any given privacy level, which is well observable for $10^{-3} \lesssim \epsilon \lesssim 1$. Under pure DP, the VMF mechanism with the d_2-sensitivity stands out again and achieves even higher correlation scores than Purkayastha.

7.4.4 Private Histograms for Spatio-Temporal Data

Histograms and heatmaps are practical tools to visualize and interpret empirical data, particularly in one or two dimensions.

Scenarios. Suppose a LBS, such as Google Maps or Foursquare, wants to use check-in data (e.g., from users' smartphones) to create daily histograms of popular visit times of businesses, such as stores or restaurants. This could allow other users to estimate how busy a location or area is during different times of the day, or provide store owners with insights on customer activity. The desired data is often privacy-sensitive, so users may

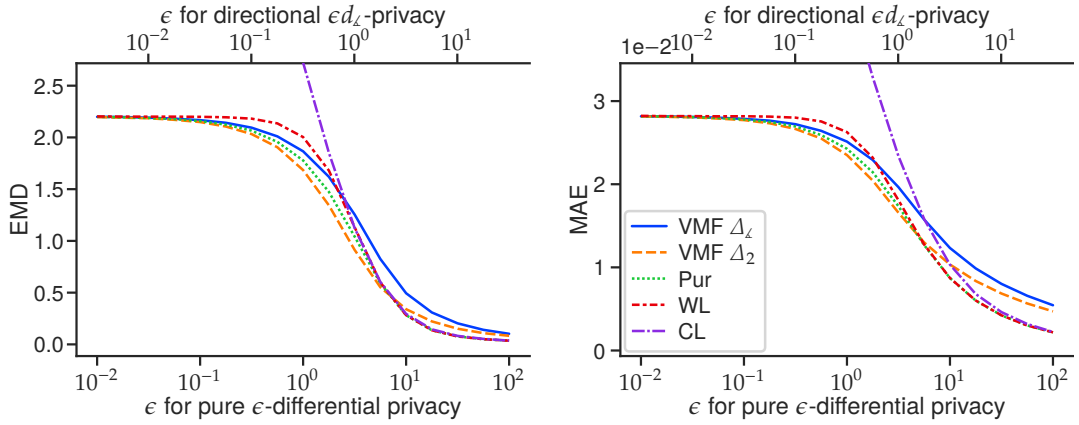


Figure 7.11: Comparison of Earth Mover's Distance (EMD) and mean absolute error (MAE) between histograms of original and perturbed check-in times from all check-ins at the top 100 locations.

distrust the data collector and be reluctant to share their whereabouts during the course of the day. To enable such use cases in a privacy-preserving way, we follow the local model and sanitize each user's data before it is collected and aggregated into histograms.

Dataset Description. We use the publicly available *Gowalla* dataset from [74]. Gowalla was a location-based social networking website where users could share their locations by checking in. It contains a total of 6,442,890 check-ins with their location and time recorded between Feb. 2009 and Oct. 2010.

7.4.4.1 Independent Analysis of Temporal and Spatial Data

We simulate data collection in the local model by perturbing the time-of-day and location of each check-in independently.

For the periodic *times-of-day*, we consider all check-ins at the top 100 locations. We follow a sanitization procedure as with the sleep data in Section 7.4.3.2 and use the VMF and Purkayastha mechanisms on $\mathbb{S}^1$, with Clipped (CL) and Wrapped Laplace (WL) as baselines (cf. Sections 7.3.2, 7.3.3 and 7.3.6). Similarly, to sanitize the *locations*, we take all check-ins from the top 100 users and represent them as unit vectors on $\mathbb{S}^2$. We then apply the appropriate VMF and Purkayastha mechanisms, with Polar Laplace (cf. Section 7.3.6.3) as baseline.

After gathering the perturbed data, we compute the following histograms: a *check-in time histogram* for each of the 100 locations with one bin for each hour of the day, and a *check-in location histogram* for each of the top 100 users with 90×180 bins, one for each pair

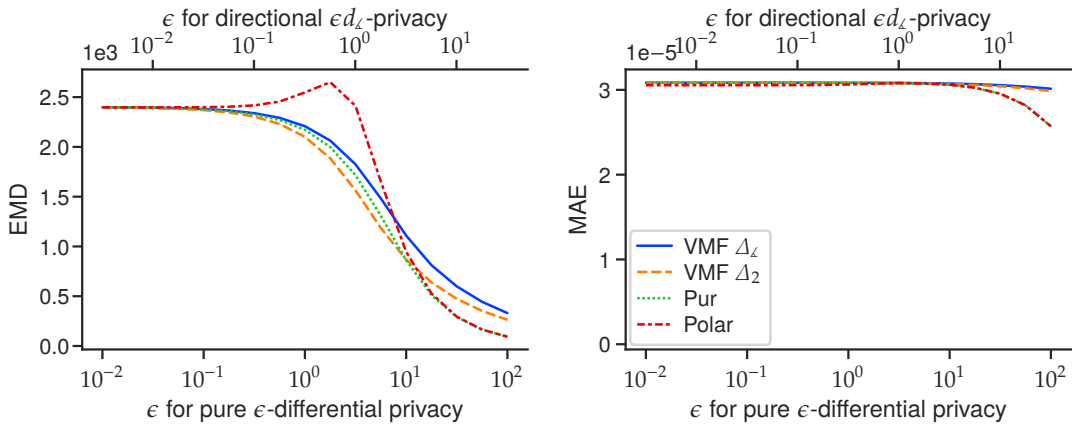


Figure 7.12: Comparison of Earth Mover's Distance (EMD) and mean absolute error (MAE) between histograms of original and perturbed check-in locations from all check-ins of the top 100 users.

of subsequent degrees of latitude and longitude. To stabilize the results, we repeat this procedure in every setting for 100 runs.

Error Metrics. As measures of error between the sanitized and original histograms, we again use the mean absolute error (MAE), as well as the Earth Mover's Distance (EMD) with a suitable distance matrix: For the distance between two check-in time histogram bins, we use their circular distance in hours. For 2D location histograms with latitude–longitude bins, we use the *great-circle distance*, i.e. the actual surface distance, between the geographic positions on the sphere corresponding to the bin centers. Unlike the MAE or MSE which look at the error of each histogram bin individually, the EMD so provides a measure of error that is *aware of the semantics* of the underlying data by considering how far off the target bin is from the original bin when counting a perturbed check-in location.

Results. Figure 7.11 shows the errors for the *check-in time* histograms. For large ϵ, both Wrapped and Clipped Laplace as well as Purkayastha show similar errors that are lower than VMF. For medium to small ϵ, our directional mechanisms gain an advantage over WL and CL with Purkayastha generally achieving the lowest errors under directional privacy, whereas VMF wins under pure DP when using the smaller d_2-sensitivity. In this case, CL performs worst with generally large MAE and EMD since virtually all counts will be in the first or last histogram bin.

Figure 7.12 shows the errors for the *check-in location* histograms. In terms of the MAE, VMF is worst while Purkayastha and Polar Laplace are almost indistinguishable. However, if we consider the EMD as a metric with spatial awareness, we recognize that the Polar

mechanism has a region with increased error for $10^{-1} \lesssim \epsilon \lesssim 10$, corresponding to the "bump" we describe in Section 7.3.6.3. Thus, in conclusion, the Purkayastha distribution shows the lowest errors for directional privacy, whereas VMF benefits from the reduced d_2-sensitivity under pure DP.

7.4.4.2 Combined Analysis of Spatio-Temporal Data: Location Busyness

The following experiment constitutes the *combined* application of directional privacy mechanisms to both *spatial and temporal* data. Our goal is to derive histograms of check-ins at the top 1000 locations from the Gowalla dataset over different times of day, where we perturb both the check-in times and locations using the Purkayastha mechanism, as well as Wrapped and Polar Laplace as baselines, respectively.

Using differential privacy in the local model is often challenging, since it injects too much noise and hence would make virtually all check-ins probabilistically indistinguishable in our scenario. This is especially problematic for locations, since some areas may be very densely populated with many bars and restaurants—so ideally, we would like to reduce the protection guarantees to reasonably smaller distances. This is an advantage of metric privacy and its variants such as directional privacy, as it allows relaxing the privacy guarantees to a defined protection radius.

Concretely, we sanitize all check-ins at the top 1000 locations by perturbing check-in times on a periodic 24h scale with a $\Delta_t \equiv 3$ hour protection radius, using the 2-dimensional Purkayastha and Wrapped Laplace mechanisms with temporal privacy levels $\ell_t \in \left[10^{-3}, 10^2\right]$, and corresponding check-in locations with a $\Delta_s \equiv 10$ meter protection radius, using the 3-dimensional Purkayastha and Polar Laplace mechanisms with spatial privacy levels $\ell_s \in \left[10^{-2}, 10^2\right]$. We perform 25 repetitions in each setting to obtain stabilized results. For each check-in, we use nearest neighbor search to assign the perturbed check-in coordinates to the nearest location, and aggregate all thusly obtained check-in times at each location into a 24-hour busyness histogram. Similarly, we obtain one daily histogram for each location based on the original, unperturbed data, which we use as a reference to compute error metrics for each anonymization run. As in previous experiments, we use the Earth Mover's Distance (EMD) as a metric to compare the mechanisms.

Results. Figure 7.13 shows the EMD over varying temporal and spatial privacy levels ℓ_t and ℓ_s. We clearly see an advantage for the Purkayastha mechanism over the baseline combination of Wrapped and Polar Laplace, which is largest for privacy levels $10^{-1} \lesssim \ell_t \lesssim 1$. We also observe that ℓ_s has a less pronounced discriminating effect, as an increase in ℓ_s generally reduces the error for both mechanisms, but slightly faster for Purkayastha

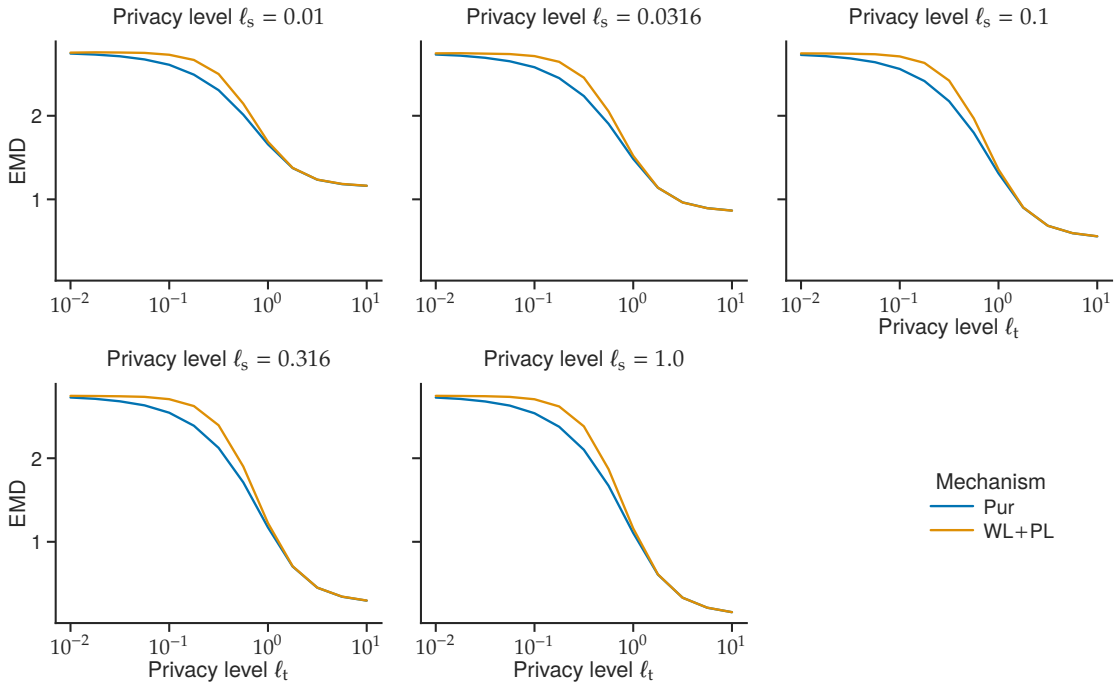


Figure 7.13: Earth Mover's Distance (EMD) between sanitized and original daily check-in activity histograms over temporal and spatial privacy levels ℓ_t (abscissa) and ℓ_s (columns) with protection radii $r_t \equiv 3\,\mathrm{h}$ and $r_s \equiv 10\,\mathrm{m}$, respectively.

than Laplace For a different perspective, see also Fig. 7.14 which shows the averaged EMD between daily busyness histograms with selected temporal privacy levels ℓ_t in the columns and continuous spatial privacy level ℓ_s in the abscissa. Figure 7.15 shows exemplary busyness histograms for four selected locations, where the check-in data have been sanitized with privacy levels $\ell_s = \ell_t \approx 0.316$. As we can see, Purkayastha is able to better preserve utility than the baseline mechanisms.

7.5 Comparison with Related Work

Various DP mechanisms have been proposed for particular types of data: In the context of location data, Andrés et al. [24] introduce the notion of *geo-indistinguishability* together with the suitable *PL mechanism*. However, their approach assumes a flat surface instead of a curved one, which restricts its usage to smaller areas where a planar approximation is acceptable. While it is possible to wrap the PL mechanism around the sphere, our experiments in Section 7.4.4 show that directional mechanisms provide superior utility at the same privacy level when considering global locations.

Data collection in the local model can be dated back to Warner [464], who proposed a

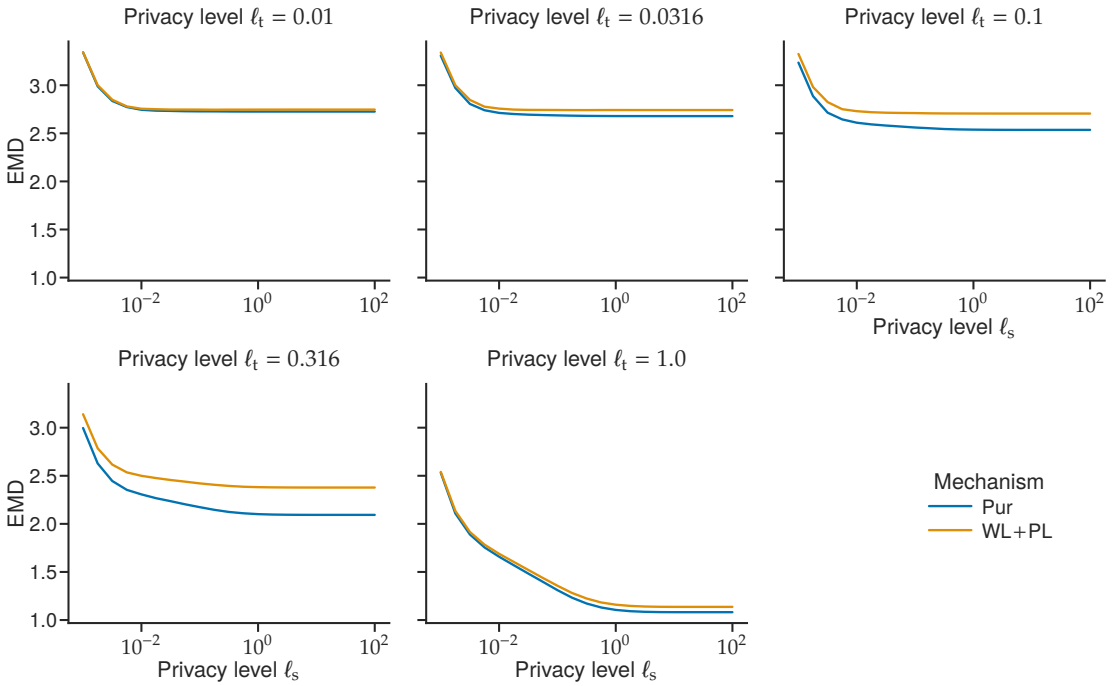


Figure 7.14: Earth Movers Distance (EMD) over spatial and temporal privacy levels ℓ_s (abscissa) and ℓ_t (columns) with protection radii $r_t \equiv 3$ h and $r_s \equiv 10$ m, respectively.

method to conduct surveys that allows the respondents to maintain privacy by randomizing their response. The goal is to eliminate evasive answer bias in cases where the individuals may prefer not to reply at all or to reply with incorrect answers to certain sensitive questions. Erlingsson et al. [123] present a modern variant called RAPPOR that privately collects statistics by hashing each user's sensitive value to a Bloom filter [45] and then applying randomized response to each bit in the filter array. Their method applies to discrete values, since hashing only slightly differing floating point values would drastically change their hashes. Kim et al. [231] employ RAPPOR to collect indoor positioning data based on a finite set of preinstalled indoor beacons.

Hay et al. [182, 183] evaluate existing DP histogram mechanisms in the *central* model. Compared to our approach with *local* sanitization, central DP mechanisms typically offer higher utility, but come at the expense of requiring a trusted data aggregator.

Wang et al. [462] propose a matrix-valued variant of the VMF distribution to achieve DP in the context of spectral graph analysis, i.e., computing eigenvalues and -vectors from graph adjacency matrices. While our proof for the VMF mechanism directly works with its probability density, they consider the *matrix Bingham-VMF* distribution with the Exponential Mechanism [294] as auxiliary, which penalizes the privacy guarantee by a factor of 2.

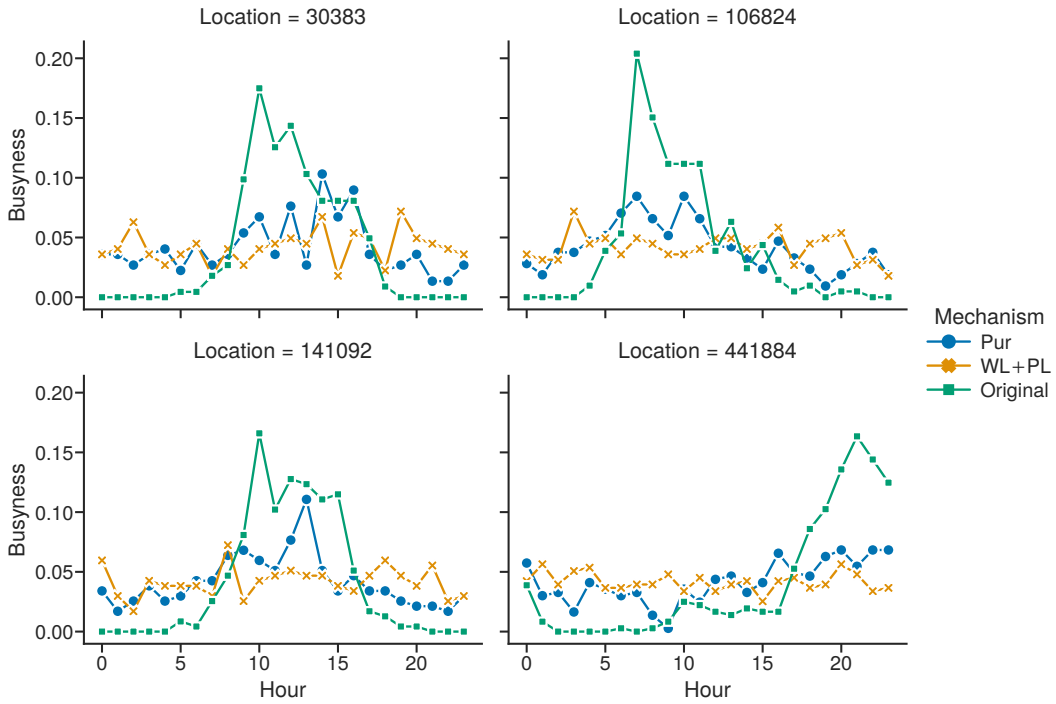


Figure 7.15: Daily check-in activity for a single sanitization run at exemplary locations, with directional privacy levels $\ell_s = \ell_t = 10^{-0.5} \approx 0.316$ as well as protection radii $r_s \equiv 10\,\mathrm{m}$ and $r_t \equiv 3\,\mathrm{h}$.

Kurz and Hanebeck [249] proposed an *approximate inversion method* for the VMF distribution as alternative to rejection sampling [446, 473], which also inspired our Purkayastha sampling scheme. The method relies on the angular CDF $\mathsf{VMFArc}[\theta \leq \vartheta]$, for which they provide an analytical solution for odd n. Unfortunately, for even n the solution contains an infinite series with special functions which we cannot evaluate efficiently. Moreover, as the number of terms grows with n, we suspect that the method may only be practical for VMF when the number of dimensions n is both odd *and* small. On the other hand, we argue that the method is still valuable and practical for the Purkayastha distribution, where our solution for $\mathsf{PurArc}[\theta \leq \vartheta]$ in Eq. (7.28) provides a closed-form expression with only finitely many terms, regardless of the parity of n. Our experiments in Section 7.4.1 confirm that our approach provides an effective sampling method that works in up to tens of thousands of dimensions, pushing beyond the status quo of 150 dimensions [90].

7.6 Chapter Summary

In this chapter, we have introduced a novel notion of *directional privacy* for the important class of directional data. To realize this notion, we have suggested the VMF and Purkayastha

mechanisms which are based on spherical distributions and intrinsically suit directional data. We have proved that they also conform to the notion of differential privacy, and derived other statistical properties such as expected distances, related densities and cumulative distribution functions. For the Purkayastha distribution, we have proposed a novel sampling algorithm where previously no method was published.

Moreover, we have performed several analyses and experiments on real data to show the applicability of our mechanisms and demonstrate their advantage over standard privacy mechanisms adapted to directional data: Importantly, we observed that the new directional mechanisms typically require fewer data to achieve a certain accuracy. For directional statistics such as the circular mean, we have demonstrated that the local model can achieve utility on par with the central model and hence is preferable since it also does not require a trusted aggregator. The facilitated use cases include important applications such as privately collecting mobility data in the local model, where the data collector cannot or may not be trusted by the users.

Further work could include finding other applications or domains for which specialized mechanisms yield improved privacy-utility trade-offs, as well as devising more efficient sampling routines for the underlying spherical distributions especially in high dimensions.

Chapter 8

Conclusion

In this chapter, we conclude this dissertation. First, we discuss the contributions and impact of our work and address the formulated research objectives. Moreover, we discuss additional challenges related to the local model. Lastly, we point out directions for future research that arise from the work in this dissertation.

8.1 Contributions and Impact

First Differential Privacy Mechanism for Text. In Chapter 5 we presented *SynTF*, a novel DP mechanism that produces differentially private BoW representations for texts. It works by randomly replacing words from an input text with similar words using the Exponential mechanism and counting the resulting terms in a tf vector. We introduced the *bigram overlap* as an additional technique that influences the choice of substituted words to further prevent authorship attribution. To the best of our knowledge, SynTF represents the first DP mechanism for text as confirmed in the survey by Zhao and Chen [494]; moreover, it also spurred a line of research we refer to as *word-level DP*, which works by replacing words in a text on a word-by-word basis in a differentially private manner (cf. Section 3.3.5.1). Note that most word-level methods provide DP guarantees only for texts of the same length, whereas SynTF provides DP to texts of varying lengths by fixing the number of terms in the resulting tf vector.

On the theoretical side, we proved the ϵ-DP properties of our method, and furthermore derived a heuristic argument that the privacy loss ϵ of the Exponential mechanism grows logarithmically in the size of the (discrete) output space if the result should provide a minimum level of utility. We experimentally verified our method on a corpus of newsgroup postings in a scenario where a benign analyst wants to infer the topic from the texts, whereas a malicious attacker tries to identify their author. The results showed that our method has a much stronger impact on authorship attribution than on the topic inference task, against which *scrubbing methods* that only mask sensitive named identifiers provide

only insufficient protection.

Human-Readable Text Obfuscation with Differential Privacy. In Chapter 6, we tackled major limitations of word-level DP methods (cf. Section 3.3.5.1), with a novel text obfuscation approach, *DP-VAE*, that applies DP to full sentences instead of individual words. DP-VAE employs a VAE architecture which encodes the input sentences to continuous, probabilistic latent representations following a Gaussian distribution. By imposing two constraints on the parametrization of the Gaussian distributions, we were able to exploit synergies with the Gaussian mechanism and achieve differentially private obfuscation that transforms full sentences into diverse and coherent, human-readable outputs. Furthermore, we extended our approach to a differentially private adversarial autoencoder (*DP-AAE*) by integrating adversarial learning to disentangle the latent representations into a privacy-sensitive author/style vector and a privacy-insensitive content vector. This separation allowed us to further improve the privacy-utility trade-off in a favorable direction by applying stronger noise to the author vector.

In Section 6.5, we performed an extensive evaluation involving hyperparameter optimization and compared our DP-VAE and DP-AAE models against two non-DP baselines in a scenario with online reviews whose authors wish to remain anonymous. The results showed that DP-AAE outperformed all other methods and effectively reduced re-identification risks of authorship attribution attacks while producing readable sentences and preserving the content of the texts. In addition, we observed that an adaptive attacker who could calibrate their authorship attribution attack to the obfuscation method showed much better chances of re-identifying the authors than a static attacker who trained author classifiers on unobfuscated data. Similarly, an adaptive strategy also indicated significant improvements for utility (here: sentiment analysis). In line with those results, several related works also found that an adaptive attacker model provides a more meaningful and realistic assessment of the protective performance of obfuscation methods, cf. our findings in Section 3.4. It is worth mentioning that we also used an adaptive attacker model in our SynTF experiments in Section 5.3.

Differential Privacy for Directional Data. In Chapter 7, we introduced the new notion of *directional privacy* for the important class of directional data based on the surface distance on the sphere. We devised two conforming mechanisms based on the spherical VMF and Purkayastha distributions that intrinsically suit directional data, for which we proved that they fulfill directional privacy and also differential privacy. Furthermore, as a theoretical contribution, we derived various statistical properties such as expected distances, related densities and cumulative distribution functions for the underlying distributions. Based

on these results, we (i) showed that adopted standard mechanisms based on wrapping can behave even worse than the uniform distribution, and (ii) described a novel sampling algorithm for the Purkayastha distribution where to our best knowledge, no designated sampling method had been published before.

Moreover, we performed several analyses and experiments on real data to show the advantage of our directional privacy mechanisms over standard privacy mechanisms adapted to directional data and to demonstrate their applicability to important applications, such as privately collecting mobility data in the local model, where the data collector cannot or may not be trusted by the users. Notably, we observed that our directional mechanisms typically required fewer data to achieve a certain level of utility (i.e., they have a lower sample complexity, cf. Section 8.3) than standard privacy mechanisms adapted to directional data. We also demonstrated that for some directional statistics such as the circular mean, the local model can achieve a sample complexity as low as in the central one, making it preferable since it also does not require a trusted aggregator.

Reassessment of Related Work. Lastly, in Chapter 3 we summarized related work on traditional and differentially private defenses for sequential and directional data, including more recent works that were developed in the meantime over the course of this dissertation, and also covering other types of sequential data besides text. Comparison with these works shows that the approaches we presented in this dissertation may be suitable for other types of sequential data, such as audio data, visual data, or trajectory data as a combination of the sequential and directional domains, as we outline in Section 8.4. It also supports the challenges and limitations we faced with our methods regarding the local model, for which we refer to Section 8.3.

8.2 Research Objectives

Having described our contributions in this dissertation, we can now address the research objectives that we formulated in Section 1.3, which we repeat here for convenience:

RO1 Design novel DP mechanisms to obfuscate text as an illustrative example of sequential data.

RO2 Evaluate the performance of the proposed DP mechanisms for text in realistic scenarios, in particular how well they protect against authorship attribution attacks.

RO3 Design specialized DP mechanisms for directional data that intrinsically respect the directional nature of the data.

RO4 Evaluate the performance of the proposed DP mechanisms for directional data in realistic scenarios.

Our work in Chapters 5 and 6 proposed and evaluated novel DP mechanisms for textual data that produce BoW representation vectors (SynTF) as well as readable text (DP-VAE and DP-AAE), respectively. Therefore, we achieved objectives **RO1** and **RO2**.

Similarly, our work in Chapter 7 introduced a new variant of metric privacy called *directional privacy* along with two novel DP mechanisms for directional data which we also evaluated in several scenarios involving spatial location and periodic temporal data. Therefore, we achieved **RO3** and **RO4**.

8.3 Challenges of the Local Model

Our directional privacy mechanisms from Chapter 7 can be used in both the central and the local model as demonstrated in Section 7.4 (e.g., for circular statistics). Moreover, in theory, it could make sense to deploy our DP mechanisms for text from Chapters 5 and 6 in the central model, for instance, to obfuscate text summaries generated from inputs sourced from multiple users. However, as we have demonstrated in our experiments, all DP mechanisms proposed in this dissertation can be deployed in realistic use cases conforming to the local model where the data is obfuscated individually, i.e., locally at the source. While the local model does have advantages, such as not requiring a trusted curator, it also comes with its challenges that we (and others) have faced, which we hence want to discuss in the following.

Noise Agnostic to Downstream Tasks. In the local model, the randomness applied to individual data values affects *any* downstream task, irrespective of whether the task is an attack that violates privacy or some benign analysis. Therefore, DP mechanisms to be deployed in the local model benefit from additional techniques that discriminate against malicious tasks (e.g., by suppressing privacy-sensitive information) and/or support benign tasks (e.g., by preserving privacy-insensitive, but utility-relevant information) to achieve good privacy-utility trade-offs.

To this end, for SynTF (Chapter 5), we proposed the *bigram overlap* in Section 5.3.1.2. It manipulates the scores of the Exponential mechanism's rating function to prefer surrogate words with spelling different to the original words, thus degrading important features for authorship attribution. For our best-performing model from Chapter 6, DP-AAE, we used adversarial learning to disentangle author- and content-specific information of the input sentences into separate representation vectors. This allowed us to improve

the privacy-utility trade-off over DP-VAE by applying different levels of obfuscation to the author and content representations independently of each other, so we could apply stronger obfuscation to the author information than to the content information.

Some related works (cf. Sections 3.3.3 and 3.3.5) take similar approaches: SanText$^+$ [487] only applies the Exponential mechanism to a subset of sensitive tokens to mitigate utility loss. ER-AE [46] uses reinforcement learning to encourage sampling of under-rated but semantically similar tokens as a substitute for the original tokens. Adversarial training has been used to protect (i.e., suppress) sensitive information in differentially private representations for text [275], as well as in obfuscation mechanisms for images based on latent [89] or disentangled representations [469]. Many DP mechanisms for speech rely on separate content and speaker representations, such as speaker x-vectors [416], and obfuscate only the speaker representations while applying little to no obfuscation to the content representations [180, 405]. However, while methods with separate levels of obfuscation for privacy-sensitive and -insensitive components may achieve improved privacy-utility trade-offs in practice, this usually comes at the cost of degraded theoretical privacy guarantees.

Lastly, we acknowledge the remaining challenge to incorporate such discrimination between "good" and "bad" tasks into our directional privacy mechanisms (Chapter 7) that obfuscate individual unit vectors (e.g, representing a GPS coordinate), since it is unclear how isotropic noise that treats each direction equally could achieve such discrimination. Some related works (cf. Section 3.3.5) point out potential approaches that could be adapted to directional privacy mechanisms: For the PL mechanism on the Euclidean plane, Chatzikokolakis et al. [67] propose a *Bayesian remapping* strategy that provides improved utility. In the context of text embeddings, Xu et al. [480] propose *elliptical noise* instead of isotropic noise that respects the geometry of the Euclidean embedding space, which they claim improves privacy at the same level of utility. We leave the task of exploring such approaches for directional data as an open challenge for future work.

High Error and Sample Complexity. A major issue with local DP mechanisms is that they perturb every data record of each user individually, thus injecting more noise than in the central model, leading to larger errors and therefore lower utility [485, Section 6.1]. To reduce the errors, applications that rely on local DP typically require more samples (i.e., they come with a high sample complexity; cf., e.g., [38, Section 2.1] and [461, Section 7]) and/or use a larger privacy loss parameter ϵ than in the central model [99] which increases the accuracy of each obfuscated sample but weakens the theoretical privacy guarantee. We have obtained a similar result in the context of SynTF (Chapter 5): In Section 5.2.4.3, we have derived necessary conditions on the privacy parameter ϵ for the Exponential

mechanism, stating that ϵ needs to grow logarithmically in the size of the output space in order to allow meaningful results with high utility.

Yang et al. [485, Section 6.1] provide a glimmer of hope and remark that one "principle of [local DP] getting accurate statistics is that the added positive and negative noises can be canceled out". In line with their remark, our directional privacy experiments on circular statistics in Section 7.4.3 show that the local model may reach the same level of accuracy as the central model for some DP mechanisms: When we average enough noisy samples for the circular mean, noise from directional privacy mechanisms cancels out in the local model, thus leading to improved accuracy.

8.4 Directions for Future Research

Utilize Transformers for Differentially Private Text Anonymization. We have already employed Transformer-based BERT [101] and Sentence-BERT [369] encoders to evaluate privacy as well as utility and content-preservation of the obfuscated texts in our DP-VAE and DP-AAE experiments in Section 6.5. However, the obfuscation models themselves still consist of "conventional" RNNs with GRUs [75]. While RNNs with GRU or LSTM cells [191] are well-suited for textual data, by now, they have been outperformed in many tasks by more recent architectures based on Transformers [450]. Therefore, we hypothesize that upgrading the encoder and decoder in our models to more recent architectures, such as BERT [101] for the encoder and GPT-2 [363] or GPT-3 [54] for the decoder, could further improve the quality of the obfuscated texts.

Evaluate Generalizability and Applicability to Other Domains. Our work on sequential data presented in Chapters 5 and 6 focuses on text as an example of a ubiquitous domain that often contains PII which may lead to the identification of individuals. For named identifiers, this information is typically confined locally, whereas biometric identifiers such as the writing style often pervade large parts of the sequence, and adequate protection mechanisms must be applied. In our survey of related work in Section 3.2, we found that PII may also be present in other types of sequential data, e.g., voice characteristics in speech or various visual identifiers in images and videos. Since similar defensive techniques have been used across these domains (cf. Section 3.3), it might be interesting to adapt and evaluate our proposed differentially private mechanisms in other domains besides text, such as speech or time series in general.

Similarly, there are other examples of directional data that may be worth investigating, particularly in higher dimensions: For instance, Dhillon and Sra [103] observed that gene

expression data also has directional characteristics; therefore, directional privacy and its mechanisms may also prove beneficial when the privacy of such data shall be protected.

Combine Approaches for Sequential and Directional Data. A particularly intriguing domain presents itself at the intersection of sequential and directional data, i.e., the focus areas of this dissertation: Trajectory data, such as location traces, are recorded as a series of individual directional data points at subsequent points in time, thus forming a temporal sequence of spatial data. Since spatio-temporal data occurs in a wide range of applications, techniques to analyze such data have evolved into their own field of *spatio-temporal data mining* [28, 178]. In some cases, such as the analysis of reoccurring patterns (e.g., daily or weekly commutes, sleeping habits as in Section 7.4.3.2, etc.), this may even involve periodic time specifications (time-of-day, day-of-week, etc.), which are another form of directional data. Likewise, also attackers can exploit spatio-temporal correlations as found in trajectory data: For instance, a study by de Montjoye et al. [92] revealed that "four spatio-temporal points are enough to uniquely identify 95% of the individuals" for location traces with spatial and temporal resolutions corresponding to the cellular network and one hour, respectively. Therefore, exploring ways to combine the individual approaches for sequential and directional data proposed in this dissertation (in particular, Chapters 6 and 7) into DP mechanisms to obfuscate spatio-temporal data, such as location traces, would provide an interesting extension of this work.

Bibliography

[1] Gretel.ai - The Developer Stack for Synthetic Data, . URL https://gretel.ai/. 43

[2] Microsoft Presidio, . URL https://microsoft.github.io/presidio/. 43

[3] Nightfall DLP Product Features, . URL https://nightfall.ai/features. 42

[4] Uniform requirements for manuscripts submitted to biomedical journals: Writing and editing for biomedical publication. *Journal of Pharmacology & Pharmacotherapeutics*, 1(1):42–58, 2010. ISSN 0976-500X. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3142758/. 57

[5] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, page 308–318, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450341394. doi: 10.1145/2976749.2978318. URL https://doi.org/10.1145/2976749.2978318. 18, 119

[6] A. Abbasi and H. Chen. Writeprints: A stylometric approach to identity-level identification and similarity detection in cyberspace. *ACM Transactions on Information Systems (TOIS)*, 26(2):7, 2008. URL http://dl.acm.org/citation.cfm?id=1344413. 36, 78, 90

[7] J. Aberdeen, S. Bayer, R. Yeniterzi, B. Wellner, C. Clark, D. Hanauer, B. Malin, and L. Hirschman. The MITRE Identification Scrubber Toolkit: Design, training, and assessment. *International Journal of Medical Informatics*, 79(12):849–859, Dec. 2010. ISSN 1386-5056. doi: 10.1016/j.ijmedinf.2010.09.007. URL https://www.sciencedirect.com/science/article/pii/S1386505610001681. 50

[8] M. Abramowitz and I. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, volume 55 of *Applied Mathematics*. National Bureau of Standards, 10th printing edition, 1972. 128

[9] D. Adair. The authorship of the disputed federalist papers. *The William and Mary Quarterly: A Magazine of Early American History*, pages 98–122, 1944. 35

[10] D. I. Adelani, A. Davody, T. Kleinbauer, and D. Klakow. Privacy Guarantees for De-Identifying Text Transformations. In *Interspeech 2020*, pages 4666–4670. ISCA, Oct. 2020. doi: 10.21437/Interspeech.2020-2208. URL https://www.isca-speech.org/archive/interspeech_2020/adelani20_interspeech.html. 62, 63

[11] S. Afroz, M. Brennan, and R. Greenstadt. Detecting Hoaxes, Frauds, and Deception in Writing Style Online. In *2012 IEEE Symposium on Security and Privacy*, pages 461–475, May 2012. doi: 10.1109/SP.2012.34. 51, 95

[12] P. Agrawal and P. J. Narayanan. Person De-Identification in Videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(3):299–310, Mar. 2011. ISSN 1558-2205. doi: 10.1109/TCSVT.2011.2105551. 61

[13] A. M. Ahmed and A. K. Hassan. Speaker Recognition Systems in the Last Decade – A Survey. *Engineering and Technology Journal*, 39(1B):30–40, Mar. 2021. ISSN 2412-0758. doi: 10.30684/etj.v39i1B.1589. URL https://etj.uotechnology.edu.iq/article_168149.html. 39

[14] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019. 113

[15] W. Alnasser, G. Beigi, and H. Liu. Privacy Preserving Text Representation Learning Using BERT. In R. Thomson, M. N. Hussain, C. Dancy, and A. Pyke, editors, *Social, Cultural, and Behavioral Modeling*, Lecture Notes in Computer Science, pages 91–100, Cham, 2021. Springer International Publishing. ISBN 978-3-030-80387-2. doi: 10.1007/978-3-030-80387-2_9. 49

[16] S. Alneyadi, E. Sithirasenan, and V. Muthukkumarasamy. A survey on data leakage prevention systems. *Journal of Network and Computer Applications*, 62:137–152, Feb. 2016. ISSN 1084-8045. doi: 10.1016/j.jnca.2016.01.008. URL https://www.sciencedirect.com/science/article/pii/S1084804516000102. 43

[17] R. Aloufi, H. Haddadi, and D. Boyle. Emotionless: Privacy-Preserving Speech Analysis for Voice Assistants. *Proceedings of the Privacy Preserving Machine Learning Workshop, ACM CCS'19*, Aug. 2019. URL http://arxiv.org/abs/1908.03632. 54

[18] R. Aloufi, H. Haddadi, and D. Boyle. Privacy-preserving Voice Analysis via Disentangled Representations. In *Proceedings of the 2020 ACM SIGSAC Conference on Cloud Computing Security Workshop*, CCSW'20, pages 1–14, New York, NY, USA, Nov. 2020. Association for Computing Machinery. ISBN 978-1-4503-8084-3. doi: 10.1145/3411495.3421355. URL https://doi.org/10.1145/3411495.3421355. 45

[19] S. Alrabaee, P. Shirani, M. Debbabi, and L. Wang. On the Feasibility of Malware Authorship Attribution. In F. Cuppens, L. Wang, N. Cuppens-Boulahia, N. Tawbi, and J. Garcia-Alfaro, editors, *Foundations and Practice of Security*, Lecture Notes in Computer Science, pages 256–272, Cham, 2017. Springer International Publishing. ISBN 978-3-319-51966-1. doi: 10.1007/978-3-319-51966-1_17. 37

[20] B. Alsulami, E. Dauber, R. Harang, S. Mancoridis, and R. Greenstadt. Source Code Authorship Attribution Using Long Short-Term Memory Based Networks. In S. N. Foley, D. Gollmann, and E. Snekkenes, editors, *Computer Security – ESORICS 2017*, Lecture Notes in Computer Science, pages 65–82, Cham, 2017. Springer International Publishing. ISBN 978-3-319-66402-6. doi: 10.1007/978-3-319-66402-6_6. 37

[21] amastyleinsider. Who Was That Masked Manual?, May 2011. URL https://amastyleinsider.com/2011/05/10/who-was-that-masked-manual/. 57

[22] C.-N. E. Anagnostopoulos, I. E. Anagnostopoulos, I. D. Psoroulas, V. Loumos, and E. Kayafas. License Plate Recognition From Still Images and Video Sequences: A Survey. *IEEE Transactions on Intelligent Transportation Systems*, 9(3):377–391, Sept. 2008. ISSN 1558-0016. doi: 10.1109/TITS.2008.922938. 40

[23] Andrea Kristen. Anonymization: Analyze sensitive data without compromising privacy, Nov. 2017. URL https://blogs.sap.com/2017/11/10/anonymization-analyze-sensitive-data-without-compromising-privacy/. 19

[24] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi. Geo-indistinguishability: Differential privacy for location-based systems. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*, CCS '13, pages 901–914, New York, NY, USA, Nov. 2013. Association for Computing Machinery. ISBN 978-1-4503-2477-9. doi: 10.1145/2508859.2516735. URL https://doi.org/10.1145/2508859.2516735. 7, 11, 17, 20, 27, 67, 96, 122, 123, 141, 144, 162

[25] E. Aramaki, T. Imai, K. Miyo, and K. Ohe. Automatic deidentification by using sentence features and label consistency. In *I2b2 Workshop on Challenges in Natural Language Processing for Clinical Data*, pages 10–11, Jan. 2006. 51

[26] L. Ardizzone, J. Kruse, C. Rother, and U. Köthe. Analyzing Inverse Problems with Invertible Neural Networks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=rJed6j0cKX. 66

[27] L. Ardizzone, C. Lüth, J. Kruse, C. Rother, and U. Köthe. Guided Image Generation with Conditional Invertible Neural Networks, July 2019. URL http://arxiv.org/abs/1907.02392. 66

[28] G. Atluri, A. Karpatne, and V. Kumar. Spatio-Temporal Data Mining: A Survey of Problems and Methods. *ACM Computing Surveys*, 51(4):1–41, July 2019. ISSN 0360-0300, 1557-7341. doi: 10.1145/3161602. URL https://dl.acm.org/doi/10.1145/3161602. 173

[29] M. Backes, P. Berrang, A. Hecksteden, M. Humbert, A. Keller, and T. Meyer. Privacy in epigenetics: Temporal linkability of MicroRNA expression profiles. In *USENIX Security Symposium*, pages 1223–1240, 2016. 27

[30] F. Bahmaninezhad, C. Zhang, and J. Hansen. Convolutional Neural Network Based Speaker De-Identification. In *The Speaker and Language Recognition Workshop (Odyssey 2018)*, pages 255–260. ISCA, June 2018. doi: 10.21437/Odyssey.2018-36. URL https://www.isca-speech.org/archive/odyssey_2018/bahmaninezhad18_odyssey.html. 53

[31] Z. Bai and X.-L. Zhang. Speaker Recognition Based on Deep Learning: An Overview. *Neural Networks*, 140:65–99, Aug. 2021. ISSN 0893-6080. doi: 10.1016/j.neunet.2021.03.004. URL https://www.sciencedirect.com/science/article/pii/S0893608021000848. 39

[32] B. Balle and Y.-X. Wang. Improving the Gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 394–403. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/balle18a.html. 25

[33] S. Banerjee and A. Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005. 112

[34] M. Barbaro, T. Zeller, and S. Hansell. A face is exposed for AOL searcher no. 4417749. *New York Times*, 9(2008):8For, August 2006. 3, 6, 37, 76, 100

[35] G. Baril, P. Cardinal, and A. L. Koerich. Named Entity Recognition for Audio De-Identification, Apr. 2022. URL http://arxiv.org/abs/2204.12622. 53

[36] G. Barlas and E. Stamatatos. Cross-Domain Authorship Attribution Using Pre-trained Language Models. In I. Maglogiannis, L. Iliadis, and E. Pimenidis, editors, *Artificial Intelligence Applications and Innovations*, IFIP Advances in Information and Communication Technology, pages 255–266, Cham, 2020. Springer International Publishing. ISBN 978-3-030-49161-1. doi: 10.1007/978-3-030-49161-1_22. 36

[37] R. Bassily, A. Smith, and A. Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 464–473, 2014. doi: 10.1109/FOCS.2014.56. 18, 119

[38] B. Bebensee. Local Differential Privacy: A tutorial. *arXiv:1907.11908 [cs]*, July 2019. URL http://arxiv.org/abs/1907.11908. 19, 171

[39] F. Béchet. Named Entity Recognition. In *Spoken Language Understanding*, chapter 10, pages 257–290. John Wiley & Sons, Ltd, 2011. ISBN 978-1-119-99269-1. doi: 10.1002/9781119992691.ch10. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119992691.ch10. 38

[40] G. Beigi, K. Shu, R. Guo, S. Wang, and H. Liu. I Am Not What I Write: Privacy Preserving Text Representation Learning, July 2019. URL http://arxiv.org/abs/1907.03189. 49

[41] G. Beigi, K. Shu, R. Guo, S. Wang, and H. Liu. Privacy Preserving Text Representation Learning. In *Proceedings of the 30th ACM Conference on Hypertext and Social Media*, HT '19, pages 275–276, New York, NY, USA, Sept. 2019. Association for Computing Machinery. ISBN 978-1-4503-6885-8. doi: 10.1145/3342220.3344925. URL https://doi.org/10.1145/3342220.3344925. 49

[42] K. G. Bennett, S. C. Bonawitz, and C. J. Vercler. Guidelines for the Ethical Publication of Facial Photographs and Review of the Literature. *The Cleft Palate Craniofacial Journal*, 56(1):7–14, Jan. 2019. ISSN 1055-6656. doi: 10.1177/1055665618774026. URL https://doi.org/10.1177/1055665618774026. 57

[43] J. Bevendorff, M. Potthast, M. Hagen, and B. Stein. Heuristic Authorship Obfuscation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*,

pages 1098–1108, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1104. URL https://aclanthology.org/P19-1104. 53

[44] M. Bhargava, P. Mehndiratta, and K. Asawa. Stylometric Analysis for Authorship Attribution on Twitter. In V. Bhatnagar and S. Srinivasa, editors, *Big Data Analytics*, Lecture Notes in Computer Science, pages 37–47, Cham, 2013. Springer International Publishing. ISBN 978-3-319-03689-2. doi: 10.1007/978-3-319-03689-2_3. 36, 37

[45] B. H. Bloom. Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, 13(7):422–426, 1970. 163

[46] H. Bo, S. H. H. Ding, B. C. M. Fung, and F. Iqbal. ER-AE: Differentially Private Text Generation for Authorship Anonymization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3997–4007, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.314. URL https://www.aclweb.org/anthology/2021.naacl-main.314. 12, 63, 69, 100, 104, 116, 171

[47] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai. Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/hash/a486cd07e4ac3d270571622f4f316ec5-Abstract.html. 45

[48] J.-F. Bonastre, D. Matrouf, and C. Fredouille. Artificial Impostor Voice Transformation Effects on False Acceptance Rates. In *Interspeech*, Antwerp, Belgium, 2007. URL https://hal.archives-ouvertes.fr/hal-02157147. 53

[49] A. Bourka and P. Drogkaris, editors. *Recommendations on Shaping Technology According to GDPR Provisions: An Overview on Data Pseudonymisation*. European Union Agency for Network and Information Security (ENISA), LU, Nov. 2018. ISBN 978-92-9204-281-3. URL https://data.europa.eu/doi/10.2824/74954. 5

[50] S. R. Bowman, L. Vilnis, O. Vinyals, A. Dai, R. Jozefowicz, and S. Bengio. Generating Sentences from a Continuous Space. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21, Berlin, Germany, Aug. 2016. Association for Computational Linguistics. doi: 10.18653/v1/K16-1002. URL https://aclanthology.org/K16-1002. 104, 108, 113

[51] M. Brennan, S. Afroz, and R. Greenstadt. Adversarial stylometry: Circumventing authorship recognition to preserve privacy and anonymity. *ACM Transactions on*

Information and System Security (TISSEC), 15(3):12, 2012. URL http://dl.acm.org/citation.cfm?id=2382450. 51, 95

[52] M. R. Brennan and R. Greenstadt. Practical Attacks Against Authorship Recognition Techniques. In *Twenty-First IAAI Conference*, Apr. 2009. URL https://www.aaai.org/ocs/index.php/IAAI/IAAI09/paper/view/257. 51

[53] K. Brkic, I. Sikiric, T. Hrkac, and Z. Kalafatic. I Know That Person: Generative Full Body and Face De-identification of People in Images. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1319–1328, Honolulu, HI, USA, July 2017. IEEE. ISBN 978-1-5386-0733-6. doi: 10.1109/CVPRW.2017.173. URL http://ieeexplore.ieee.org/document/8014907/. 61

[54] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language Models are Few-Shot Learners, July 2020. URL http://arxiv.org/abs/2005.14165. 172

[55] S. Burrows and S. Tahaghoghi. Source code authorship attribution using n-grams. In *Proceedings of the Twelth Australasian Document Computing Symposium, Melbourne, Australia, RMIT University*, pages 32–39. Citeseer, 2007. 37

[56] S. Busemann, S. Schmeier, and R. Arens. Message classification in the call center. In *Proceedings of the sixth conference on Applied natural language processing*, pages 158–165. Association for Computational Linguistics, 2000. 75

[57] F. S. Bäumer, N. Grote, J. Kersting, and M. Geierhos. Privacy Matters: Detecting Nocuous Patient Data Exposure in Online Physician Reviews. In R. Damaševičius and V. Mikašytė, editors, *Information and Software Technologies*, Communications in Computer and Information Science, pages 77–89, Cham, 2017. Springer International Publishing. ISBN 978-3-319-67642-5. doi: 10.1007/978-3-319-67642-5_7. 3, 100

[58] A. Caliskan and R. Greenstadt. Translate once, translate twice, translate thrice and attribute: Identifying authors and machine translation tools in translated text. In *Semantic Computing (ICSC), 2012 IEEE Sixth International Conference On*, pages 121–125. IEEE, 2012. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6337093. 51, 95

[59] A. Caliskan-Islam, F. Yamaguchi, E. Dauber, R. Harang, K. Rieck, R. Greenstadt, and A. Narayanan. When coding style survives compilation: De-anonymizing programmers from executable binaries. *arXiv preprint arXiv:1512.08546*, 2015. 37

[60] J. P. Campbell, W. Shen, W. M. Campbell, R. Schwartz, J.-F. Bonastre, and D. Matrouf. Forensic speaker recognition. *IEEE Signal Processing Magazine*, 26(2):95–103, Mar. 2009. ISSN 1558-0792. doi: 10.1109/MSP.2008.931100. 39

[61] A. Caubrière, S. Rosset, Y. Estève, A. Laurent, and E. Morin. Where are we in Named Entity Recognition from Speech? In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4514–4520, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.556. 38

[62] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, et al. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*, 2018. 112

[63] M. A. P. Chamikara, P. Bertok, I. Khalil, D. Liu, and S. Camtepe. Privacy Preserving Face Recognition Utilizing Differential Privacy. *Computers & Security*, 97:101951, Oct. 2020. ISSN 0167-4048. doi: 10.1016/j.cose.2020.101951. URL https://www.sciencedirect.com/science/article/pii/S0167404820302273. 65

[64] P. Champion, D. Jouvet, and A. Larcher. Speaker information modification in the VoicePrivacy 2020 toolchain. Research Report, INRIA Nancy, équipe Multispeech ; LIUM - Laboratoire d'Informatique de l'Université du Mans, Nov. 2020. URL https://hal.archives-ouvertes.fr/hal-02995855. 55

[65] K. Chatzikokolakis, M. E. Andrés, N. E. Bordenabe, and C. Palamidessi. Broadening the Scope of Differential Privacy Using Metrics. In E. De Cristofaro and M. Wright, editors, *Privacy Enhancing Technologies*, Lecture Notes in Computer Science, pages 82–102, Berlin, Heidelberg, 2013. Springer. ISBN 978-3-642-39077-7. doi: 10.1007/978-3-642-39077-7_5. 17, 20, 27, 48, 62, 64, 65, 66, 67, 96, 121, 122, 123, 130, 140, 144

[66] K. Chatzikokolakis, C. Palamidessi, and M. Stronati. Location Guard: location privacy for the rest of us. In *8th Workshop on Hot Topics in Privacy Enhancing Technologies (HotPETs 2015)*, 2015. URL https://github.com/chatziko/location-guard. 67, 144

[67] K. Chatzikokolakis, E. ElSalamouny, and C. Palamidessi. Efficient Utility Improvement for Location Privacy. *Proceedings on Privacy Enhancing Technologies*, 2017(4),

Jan. 2017. ISSN 2299-0984. doi: 10.1515/popets-2017-0051. URL http://content.sciendo.com/view/journals/popets/2017/4/article-p308.xml. 171

[68] K. Chatzikokolakis, E. ElSalamouny, C. Palamidessi, and A. Pazii. Methods for Location Privacy: A comparative overview. *Foundations and Trends® in Privacy and Security*, 1(4):199–257, 2017. ISSN 2474-1558, 2474-1566. doi: 10.1561/3300000017. URL http://www.nowpublishers.com/article/Details/SEC-017. 62

[69] E. Chatzikyriakidis, C. Papaioannidis, and I. Pitas. Adversarial Face De-Identification. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 684–688, Taipei, Taiwan, Sept. 2019. IEEE. ISBN 978-1-5386-6249-6. doi: 10.1109/ICIP.2019.8803803. URL https://ieeexplore.ieee.org/document/8803803/. 60

[70] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(29):1069–1109, 2011. URL http://jmlr.org/papers/v12/chaudhuri11a.html. 119

[71] B. Chen, G. Xu, X. Wang, P. Xie, M. Zhang, and F. Huang. AISHELL-NER: Named Entity Recognition from Chinese Speech. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8352–8356, May 2022. doi: 10.1109/ICASSP43922.2022.9746955. 38

[72] X. Chen, L. Jin, Y. Zhu, C. Luo, and T. Wang. Text Recognition in the Wild: A Survey. *ACM Computing Surveys*, 54(2):1–35, Mar. 2022. ISSN 0360-0300, 1557-7341. doi: 10.1145/3440756. URL https://dl.acm.org/doi/10.1145/3440756. 39

[73] D. Chicco and G. Jurman. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC genomics*, 21(1):1–13, 2020. 112

[74] E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1082–1090, 2011. 159

[75] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio. On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-4012. URL https://aclanthology.org/W14-4012. 63, 172

[76] K. Cho, B. van Merrienboer, Ç. Gülçehre, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078, 2014. URL http://arxiv.org/abs/1406.1078. 108

[77] J. Chord and S. Ellis. Taking charge of your data: Using Cloud DLP to de-identify and obfuscate sensitive information, Nov. 2018. URL https://cloud.google.com/blog/products/identity-security/taking-charge-of-your-data-using-cloud-dlp-to-de-identify-and-obfuscate-sensitive-information/. 42

[78] R. Clement and D. Sharp. Ngram and Bayesian Classification of Documents for Topic and Authorship. *Literary and Linguistic Computing*, 18(4):423–447, Nov. 2003. ISSN 0268-1145. doi: 10.1093/llc/18.4.423. URL https://academic.oup.com/dsh/article/18/4/423/957782. 36

[79] A. J. P. Clover, E. Fitzpatrick, and C. Healy. Analysis of methods of providing anonymity in facial photographs; a randomised controlled study. *Irish Medical Journal*, 103(8):243–245, Sept. 2010. ISSN 0332-3102. 57

[80] M. Coavoux, S. Narayan, and S. B. Cohen. Privacy-preserving Neural Representations of Text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1–10, Brussels, Belgium, Oct. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1001. URL https://aclanthology.org/D18-1001. 44

[81] A. Cohen-Hadria, M. Cartwright, B. McFee, and J. P. Bello. Voice Anonymization in Urban Sound Recordings. In *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, Pittsburgh, PA, USA, Oct. 2019. IEEE. ISBN 978-1-72810-824-7. doi: 10.1109/MLSP.2019.8918913. URL https://ieeexplore.ieee.org/document/8918913/. 53

[82] I. Cohn, I. Laish, G. Beryozkin, G. Li, I. Shafran, I. Szpektor, T. Hartman, A. Hassidim, and Y. Matias. Audio De-identification - a New Entity Recognition Task. In *Proceedings of the 2019 Conference of the North*, pages 197–204, Minneapolis - Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-2025. URL http://aclweb.org/anthology/N19-2025. 53

[83] A. Conway. Face blurring: When footage requires anonymity, July 2012. URL https://blog.youtube/news-and-events/face-blurring-when-footage-requires/. 58

[84] A. Conway. Blur moving objects in your video with the new Custom blurring tool on YouTube, Feb. 2016. URL https://blog.youtube/news-and-events/blur-moving-objects-in-your-video-with/. 58

[85] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685, June 2001. ISSN 1939-3539. doi: 10.1109/34.927467. 59

[86] R. M. Coyotl-Morales, L. Villaseñor-Pineda, M. Montes-y-Gómez, and P. Rosso. Authorship Attribution Using Word Sequences. In J. F. Martínez-Trinidad, J. A. Carrasco Ochoa, and J. Kittler, editors, *Progress in Pattern Recognition, Image Analysis and Applications*, Lecture Notes in Computer Science, pages 844–853, Berlin, Heidelberg, 2006. Springer. ISBN 978-3-540-46557-7. doi: 10.1007/11892755_87. 36

[87] W. L. Croft, J.-R. Sack, and W. Shi. Differentially Private Obfuscation of Facial Images. In A. Holzinger, P. Kieseberg, A. M. Tjoa, and E. Weippl, editors, *Machine Learning and Knowledge Extraction*, Lecture Notes in Computer Science, pages 229–249, Cham, 2019. Springer International Publishing. ISBN 978-3-030-29726-8. doi: 10.1007/978-3-030-29726-8_15. 66

[88] W. L. Croft, J.-R. Sack, and W. Shi. Obfuscation of Images via Differential Privacy: From Facial Images to General Images. *Peer-to-Peer Networking and Applications*, 14 (3):1705–1733, May 2021. ISSN 1936-6450. doi: 10.1007/s12083-021-01091-9. URL https://doi.org/10.1007/s12083-021-01091-9. 65

[89] W. L. Croft, J.-R. Sack, and W. Shi. Differentially private facial obfuscation via generative adversarial networks. *Future Generation Computer Systems*, 129:358–379, Apr. 2022. ISSN 0167-739X. doi: 10.1016/j.future.2021.11.032. URL https://www.sciencedirect.com/science/article/pii/S0167739X21004763. 66, 69, 171

[90] C. Cutting, D. Paindaveine, and T. Verdebout. *Tests of Concentration for Low-Dimensional and High-Dimensional Directional Data*, pages 209–227. Springer International Publishing, Cham, 2017. 123, 139, 140, 147, 164

[91] E. Dauber, R. Erbacher, G. Shearer, M. Weisman, F. Nelson, and R. Greenstadt. Supervised Authorship Segmentation of Open Source Code Projects. *Proceedings on Privacy Enhancing Technologies*, 2021(4):464–479, Oct. 2021. ISSN 2299-0984. doi: 10.2478/popets-2021-0080. URL https://petsymposium.org/popets/2021/popets-2021-0080.php. 37

[92] Y.-A. de Montjoye, C. A. Hidalgo, M. Verleysen, and V. D. Blondel. Unique in the Crowd: The privacy bounds of human mobility. *Scientific Reports*, 3(1):1376, Mar. 2013. ISSN 2045-2322. doi: 10.1038/srep01376. URL https://www.nature.com/articles/srep01376;. 3, 76, 100, 173

[93] Y.-A. de Montjoye, L. Radaelli, V. K. Singh, and A. S. Pentland. Unique in the shopping mall: On the reidentifiability of credit card metadata. *Science*, 347(6221): 536–539, Jan. 2015. doi: 10.1126/science.1256297. URL https://www.science.org/doi/10.1126/science.1256297. 3, 100

[94] O. de Vel, A. Anderson, M. Corney, and G. Mohay. Mining e-mail content for author identification forensics. *ACM SIGMOD Record*, 30(4):55–64, Dec. 2001. ISSN 0163-5808. doi: 10.1145/604264.604272. URL https://dl.acm.org/doi/10.1145/604264.604272. 37

[95] O. de Vel, A. Anderson, M. Corney, and G. Mohay. Multi-topic e-mail authorship attribution forensics. In *Proceedings ACM Conference on Computer Security-Workshop on Data Mining for Security Applications*, 2001. 37

[96] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet. Front-End Factor Analysis for Speaker Verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):788–798, May 2011. ISSN 1558-7924. doi: 10.1109/TASL.2010.2064307. 39

[97] J. Demanet, K. Dhont, L. Notebaert, S. Pattyn, and A. Vandierendonck. Pixelating Familiar People in the Media: Should Masking Be Taken at Face Value? *Psychologica Belgica*, 47(4):261–276, 2007. ISSN 0033-2879. URL http://dx.doi.org/10.5334/pb-47-4-261. 58, 61

[98] F. Dernoncourt, J. Y. Lee, O. Uzuner, and P. Szolovits. De-identification of patient notes with recurrent neural networks. *Journal of the American Medical Informatics Association*, 24(3):596–606, May 2017. ISSN 1067-5027. doi: 10.1093/jamia/ocw156. URL https://doi.org/10.1093/jamia/ocw156. 51

[99] D. Desfontaines. Local vs. central differential privacy, Sept. 2021. URL https://desfontain.es/privacy/local-global-differential-privacy.html. 171

[100] D. Desfontaines. A list of real-world uses of differential privacy, Jan. 2022. URL https://desfontain.es/privacy/real-world-differential-privacy.html. 18

[101] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423. 48, 111, 172

[102] L. Devroye. Nonuniform random variate generation. *Handbooks in operations research and management science*, 13:83–121, 2006. 138, 140

[103] I. S. Dhillon and S. Sra. Modeling data using directional distributions. Technical Report TR-03-06, Dept. of CS, Univ. of Texas at Austin, 2003. 8, 172

[104] M. Diaz, M. A. Ferrer, D. Impedovo, M. I. Malik, G. Pirlo, and R. Plamondon. A Perspective Analysis of Handwritten Signature Technology. *ACM Computing Surveys*, 51(6):1–39, Nov. 2019. ISSN 0360-0300, 1557-7341. doi: 10.1145/3274658. URL https://dl.acm.org/doi/10.1145/3274658. 41

[105] A. D. Diego. Api_key_detector, Oct. 2017. URL https://github.com/alessandrodd/api_key_detector. 35

[106] A. D. Diego. Apk_api_key_extractor, Oct. 2017. URL https://github.com/alessandrodd/apk_api_key_extractor. 35

[107] A. D. Diego. *Automatic Extraction of API Keys from Android Applications*. PhD thesis, UNIVERSITA DEGLI STUDI DI ROMA "TOR VERGATA", 2017. 35

[108] Differential Privacy Team, Apple. Learning with Privacy at Scale. Technical report, Apple Machine Learning Research, Dec. 2017. URL https://machinelearning.apple.com/research/learning-with-privacy-at-scale. 19

[109] B. Ding, J. Kulkarni, and S. Yekhanin. Collecting Telemetry Data Privately. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/hash/253614bbac999b38b5b60cae531c4969-Abstract.html. 19

[110] C. Doersch. Tutorial on variational autoencoders. *CoRR*, abs/1606.05908, 2016. URL http://arxiv.org/abs/1606.05908. 101

[111] J. Domingo-Ferrer, D. Sánchez, and A. Blanco-Justicia. The limits of differential privacy (and its misuse in data release and machine learning). *Communications*

of the ACM, 64(7):33–35, June 2021. ISSN 0001-0782. doi: 10.1145/3433638. URL https://doi.org/10.1145/3433638. 19

[112] L. Du, M. Yi, E. Blasch, and H. Ling. GARP-face: Balancing privacy protection and utility preservation in face de-identification. In *IEEE International Joint Conference on Biometrics*, pages 1–8, Sept. 2014. doi: 10.1109/BTAS.2014.6996249. 59

[113] S. Du, M. Ibrahim, M. Shehata, and W. Badawy. Automatic License Plate Recognition (ALPR): A State-of-the-Art Review. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(2):311–325, Feb. 2013. ISSN 1558-2205. doi: 10.1109/TCSVT.2012.2203741. 40

[114] J. C. Duchi, M. I. Jordan, and M. J. Wainwright. Local privacy and statistical minimax rates. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 429–438, Oct 2013. 19

[115] C. Dwork. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pages 1–19. Springer, 2008. 96

[116] C. Dwork and A. Roth. *The Algorithmic Foundations of Differential Privacy*, volume 9. Now Publishers Inc., Hanover, MA, USA, aug 2014. doi: 10.1561/0400000042. URL https://doi.org/10.1561/0400000042. 7, 15, 17, 18, 24, 25, 81, 82, 83, 105, 107

[117] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference*, pages 265–284. Springer, 2006. 3, 6, 7, 15, 17, 23, 79, 100, 122, 141, 155

[118] M. Eder. Does size matter? Authorship attribution, small samples, big problem. *Digital Scholarship in the Humanities*, page fqt066, 2013. URL http://dsh.oxfordjournals.org/content/early/2014/12/02/llc.fqt066.abstract. 36

[119] Y. Elazar and Y. Goldberg. Adversarial Removal of Demographic Attributes from Text Data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 11–21, Brussels, Belgium, Oct. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1002. URL https://aclanthology.org/D18-1002. 44, 69

[120] E. ElSalamouny and S. Gambs. Differential Privacy Models for Location-Based Services. *Transactions on Data Privacy*, 9(1):15, 2016. URL https://hal.inria.fr/hal-01418136. 67

[121] C. Emmery, E. Manjavacas, and G. Chrupała. Style Obfuscation by Invariance. *arXiv:1805.07143 [cs]*, May 2018. URL http://arxiv.org/abs/1805.07143. 52

[122] D. Ericsson, A. Östberg, E. L. Zec, J. Martinsson, and O. Mogren. Adversarial representation learning for private speech generation, June 2020. URL http://arxiv.org/abs/2006.09114. 54

[123] U. Erlingsson, V. Pihur, and A. Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, CCS '14, page 1054–1067, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450329576. doi: 10.1145/2660267.2660348. URL https://doi.org/10.1145/2660267.2660348. 19, 96, 163

[124] F. Z. Errounda and Y. Liu. An Analysis of Differential Privacy Research in Location and Trajectory Data. Preprint, In Review, Oct. 2020. URL https://www.researchsquare.com/article/rs-94765/v1. 67

[125] European Parliament and Council of the European Union. Regulation on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). *Official Journal of the European Union*, L 119:1–88, May 2016. URL https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX:32016R0679. 4, 5, 6

[126] A. Evfimievski, J. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *Proceedings of the twenty-second ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 211–222, 2003. 19

[127] M. Fabien, E. Villatoro-Tello, P. Motlicek, and S. Parida. BertAA : BERT fine-tuning for authorship attribution. In *Proceedings of the 17th International Conference on Natural Language Processing (ICON)*, pages 127–137, Indian Institute of Technology Patna, Patna, India, Dec. 2020. NLP Association of India (NLPAI). URL https://aclanthology.org/2020.icon-main.16. 3, 100

[128] M. Fabien, E. Villatoro-Tello, P. Motlicek, and S. Parida. BertAA : BERT fine-tuning for Authorship Attribution. In *Proceedings of the 17th International Conference on Natural Language Processing (ICON)*, pages 127–137, Indian Institute of Technology Patna, Patna, India, Dec. 2020. NLP Association of India (NLPAI). URL https://aclanthology.org/2020.icon-main.16. 36, 111

[129] A. Fan, M. Lewis, and Y. Dauphin. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1082. URL https://aclanthology.org/P18-1082. 107

[130] L. Fan. Image Pixelization with Differential Privacy. In F. Kerschbaum and S. Paraboschi, editors, *Data and Applications Security and Privacy XXXII*, volume 10980 of *Lecture Notes in Computer Science*, pages 148–162, Cham, 2018. Springer International Publishing. ISBN 978-3-319-95729-6. doi: 10.1007/978-3-319-95729-6_10. URL https://doi.org/10.1007/978-3-319-95729-6_10. 65

[131] L. Fan. Differential Privacy for Image Publication. In *The 2019 Theory and Practice of Differential Privacy Workshop (TPDP)*, 2019. 65

[132] L. Fan. Practical Image Obfuscation with Provable Privacy. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 784–789, July 2019. doi: 10.1109/ICME.2019.00140. 65

[133] F. Fang, X. Wang, J. Yamagishi, I. Echizen, M. Todisco, N. Evans, and J.-F. Bonastre. Speaker Anonymization Using X-vector and Neural Waveform Models, May 2019. URL http://arxiv.org/abs/1905.13561. 54, 55, 56

[134] G. Fanti, V. Pihur, and Ú. Erlingsson. Building a rappor with the unknown: Privacy-preserving learning of associations and data dictionaries. *Proceedings on Privacy Enhancing Technologies*, 2016(3):41–61, 2016. 19

[135] C. C. Farmer, S. C. Pang, D. Kevat, J. Dean, D. Panaccio, and P. D. Mahar. Medico-legal implications of audiovisual recordings of telehealth encounters. *Medical Journal of Australia*, 214(8):357, May 2021. ISSN 0025-729X, 1326-5377. doi: 10.5694/mja2.51008. URL https://onlinelibrary.wiley.com/doi/10.5694/mja2.51008. 38

[136] B. Favre, F. Béchet, and P. Nocéra. Robust Named Entity Extraction from Large Spoken Archives. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 491–498, Vancouver, British Columbia, Canada, Oct. 2005. Association for Computational Linguistics. URL https://aclanthology.org/H05-1062. 38

[137] N. Fernandes, M. Dras, and A. McIver. Author Obfuscation Using Generalised Differential Privacy. *arXiv:1805.08866 [cs]*, May 2018. URL http://arxiv.org/abs/1805.08866. 48, 62

[138] N. Fernandes, M. Dras, and A. McIver. Generalised Differential Privacy for Text Document Processing. In F. Nielson and D. Sands, editors, *Principles of Security and Trust*, volume 11426, pages 123–148. Springer International Publishing, Cham, 2019. ISBN 978-3-030-17137-7 978-3-030-17138-4. doi: 10.1007/978-3-030-17138-4_6. URL http://link.springer.com/10.1007/978-3-030-17138-4_6. 48, 62, 100, 104

[139] C. Feutry, P. Piantanida, Y. Bengio, and P. Duhamel. Learning Anonymized Representations with Adversarial Neural Networks, Feb. 2018. URL http://arxiv.org/abs/1802.09386. 45

[140] O. Feyisetan and S. Kasiviswanathan. Private Release of Text Embedding Vectors. In *Proceedings of the First Workshop on Trustworthy Natural Language Processing*, pages 15–27, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.trustnlp-1.3. URL https://aclanthology.org/2021.trustnlp-1.3. 48

[141] O. Feyisetan, T. Diethe, and T. Drake. Leveraging Hierarchical Representations for Preserving Privacy and Utility in Text. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 210–219, Nov. 2019. doi: 10.1109/ICDM.2019.00031. 48, 63, 100, 104

[142] O. Feyisetan, B. Balle, T. Drake, and T. Diethe. Privacy- and Utility-Preserving Textual Analysis via Calibrated Multivariate Perturbations. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 178–186, Houston TX USA, Jan. 2020. ACM. ISBN 978-1-4503-6822-3. doi: 10.1145/3336191.3371856. URL https://dl.acm.org/doi/10.1145/3336191.3371856. 48, 49, 62, 100, 104

[143] S. Fiel and R. Sablatnig. Writer Identification and Retrieval Using a Convolutional Neural Network. In G. Azzopardi and N. Petkov, editors, *Computer Analysis of Images and Patterns*, Lecture Notes in Computer Science, pages 26–37, Cham, 2015. Springer International Publishing. ISBN 978-3-319-23117-4. doi: 10.1007/978-3-319-23117-4_3. 41

[144] M. Figurnov, S. Mohamed, and A. Mnih. Implicit reparameterization gradients. In S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 439–450, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/92c8c96e4c37100777c7190b76d28233-Abstract.html. 104

[145] M. Fiore, P. Katsikouli, E. Zavou, M. Cunche, F. Fessant, D. L. Hello, U. M. Aivodji, B. Olivier, T. Quertier, and R. Stanica. Privacy in trajectory micro-data publishing: A

survey. *Transactions on Data Privacy*, 13:91, 2020. URL https://hal.inria.fr/hal-02968279. 62

[146] R. Fisher. Dispersion on a Sphere. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 217(1130):295–305, 1953. 127

[147] G. Frantzeskou, S. Gritzalis, and S. MacDonell. Source code authorship analysis for supporting the cybercrime investigation process. *Handbook of Research on Computational Forensics, Digital Crime, and Investigation: Methods and Solutions*, pages 470–495, 2004. 37

[148] G. Frantzeskou, E. Stamatatos, S. Gritzalis, C. Chaski, and B. Howald. Identifying authorship by byte-level n-grams: The source code author profile (scap) method. *International Journal of Digital Evidence*, 6(1):1–18, 2007. 37

[149] M. Fredrikson, E. Lantz, S. Jha, S. M. Lin, D. Page, and T. Ristenpart. Privacy in Pharmacogenetics: An End-to-End Case Study of Personalized Warfarin Dosing. In K. Fu and J. Jung, editors, *Proceedings of the 23rd USENIX Security Symposium, San Diego, CA, USA, August 20-22, 2014*, pages 17–32. USENIX Association, 2014. URL https://www.usenix.org/node/184490. 46

[150] M. Fredrikson, S. Jha, and T. Ristenpart. Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1322–1333, Denver Colorado USA, Oct. 2015. ACM. ISBN 978-1-4503-3832-5. doi: 10.1145/2810103.2813677. URL https://dl.acm.org/doi/10.1145/2810103.2813677. 46

[151] M. Friedrich, A. Köhn, G. Wiedemann, and C. Biemann. Adversarial Learning of Privacy-Preserving Text Representations for De-Identification of Medical Records. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5829–5839, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1584. URL https://aclanthology.org/P19-1584. 44

[152] A. Frome, G. Cheung, A. Abdulkader, M. Zennaro, B. Wu, A. Bissacco, H. Adam, H. Neven, and L. Vincent. Large-scale Privacy Protection in Google Street View. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2373–2380, Kyoto, Sept. 2009. IEEE. ISBN 978-1-4244-4420-5. doi: 10.1109/ICCV.2009.5459413. URL http://ieeexplore.ieee.org/document/5459413/. 57

[153] S. Furui. An Overview of Speaker Recognition Technology. In C.-H. Lee, F. K. Soong, and K. K. Paliwal, editors, *Automatic Speech and Speaker Recognition: Advanced Topics*,

The Kluwer International Series in Engineering and Computer Science, pages 31–56. Springer US, Boston, MA, 1996. ISBN 978-1-4613-1367-0. doi: 10.1007/978-1-4613-1367-0_2. URL https://doi.org/10.1007/978-1-4613-1367-0_2. 39

[154] O. Gafni, L. Wolf, and Y. Taigman. Live Face De-Identification in Video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9378–9387, 2019. URL https://openaccess.thecvf.com/content_ICCV_2019/html/Gafni_Live_Face_De-Identification_in_Video_ICCV_2019_paper.html. 61

[155] Y. Ganin and V. Lempitsky. Unsupervised Domain Adaptation by Backpropagation. In *International Conference on Machine Learning*, pages 1180–1189, Lille, France, 2015. PMLR. 44, 52

[156] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-Adversarial Training of Neural Networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016. 44, 45

[157] S. Ghannay, A. Caubrière, Y. Estève, N. Camelin, E. Simonnet, A. Laurent, and E. Morin. End-To-End Named Entity And Semantic Concept Extraction From Speech. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 692–699, Dec. 2018. doi: 10.1109/SLT.2018.8639513. 38

[158] S. Ghannay, A. Caubrière, Y. Estève, A. Laurent, and E. Morin. End-to-end named entity extraction from speech, May 2018. URL http://arxiv.org/abs/1805.12045. 38

[159] M. Gil, F. Alajaji, and T. Linder. Rényi divergence measures for commonly used univariate continuous distributions. *Information Sciences*, 249:124–131, 2013. 26

[160] M. Gong, J. Liu, H. Li, Y. Xie, and Z. Tang. Disentangled Representation Learning for Multiple Attributes Preserving Face Deidentification. *IEEE Transactions on Neural Networks and Learning Systems*, 33(1):244–256, Jan. 2022. ISSN 2162-2388. doi: 10.1109/TNNLS.2020.3027617. 60, 116

[161] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://papers.nips.cc/paper/2014/hash/5ca3e9b122f61f8f06494c97b1afccf3-Abstract.html. 45, 52, 60, 61

[162] D. Goodin. PSA: Don't upload your important passwords to GitHub, Jan. 2013. URL https://arstechnica.com/information-technology/2013/01/psa-dont-upload-your-important-passwords-to-github/. 35

[163] Google. Cloud Data Loss Prevention, . URL https://cloud.google.com/dlp. 42

[164] Google. Google-Contributed Street View Imagery Policy, . URL https://www.google.com/streetview/policy/. 39, 53

[165] J. Gorodkin. Comparing two K-category assignments by a K-category correlation coefficient. *Computational Biology and Chemistry*, 28(5):367–374, 2004. ISSN 1476-9271. doi: https://doi.org/10.1016/j.compbiolchem.2004.09.006. URL https://www.sciencedirect.com/science/article/pii/S1476927104000799. 112

[166] I. S. Gradshteyn and I. M. Ryzhik. *Table of Integrals, Series, and Products*. Academic Press, 2014. 128, 135

[167] N. Graham, G. Hirst, and B. Marthi. Segmenting documents by stylistic character. *Natural Language Engineering*, 11(4):397–415, Dec. 2005. ISSN 1469-8110, 1351-3249. doi: 10.1017/S1351324905003694. URL https://www.cambridge.org/core/journals/natural-language-engineering/article/abs/segmenting-documents-by-stylistic-character/3498721D8CCE5EAA6F7DEF355126E048. 37

[168] R. Gross, E. Airoldi, B. Malin, and L. Sweeney. Integrating Utility into Face De-identification. In G. Danezis and D. Martin, editors, *Privacy Enhancing Technologies*, Lecture Notes in Computer Science, pages 227–242, Berlin, Heidelberg, 2006. Springer. ISBN 978-3-540-34746-0. doi: 10.1007/11767831_15. 59

[169] R. Gross, L. Sweeney, F. de la Torre, and S. Baker. Model-Based Face De-Identification. In *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*, pages 161–161, June 2006. doi: 10.1109/CVPRW.2006.125. 59

[170] M. Guevara, D. Desfontaines, J. Waldo, and T. Coatta. Differential privacy: The pursuit of protections by default: A discussion with miguel guevara, damien desfontaines, jim waldo, and terry coatta. *Queue*, 18(5):93–112, 2020. 2

[171] Y. Guo, J. Liu, W. Tang, and C. Huang. Exsense: Extract sensitive information from unstructured data. *Computers & Security*, 102:102156, Mar. 2021. ISSN 0167-4048. doi: 10.1016/j.cose.2020.102156. URL https://www.sciencedirect.com/science/article/pii/S0167404820304296. 34

[172] R. Gylberth, R. Adnan, S. Yazid, and T. Basaruddin. Differentially private optimization algorithms for deep neural networks. In *2017 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pages 387–394, 2017. doi: 10.1109/ICACSIS.2017.8355063. 119

[173] T. Ha, T. K. Dang, T. T. Dang, T. A. Truong, and M. T. Nguyen. Differential Privacy in Deep Learning: An Overview. In *2019 International Conference on Advanced Computing and Applications (ACOMP)*, pages 97–102, Nov. 2019. doi: 10.1109/ACOMP.2019.00022. 18

[174] I. Habernal. When differential privacy meets NLP: The devil is in the detail. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1522–1528, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.114. URL https://aclanthology.org/2021.emnlp-main.114. 64

[175] I. Habernal. How reparametrization trick broke differentially-private text representation learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 771–777, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.87. URL https://aclanthology.org/2022.acl-short.87. 49

[176] L. G. Hafemann, R. Sabourin, and L. S. Oliveira. Offline Handwritten Signature Verification - Literature Review. In *2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–8, Nov. 2017. doi: 10.1109/IPTA.2017.8310112. URL http://arxiv.org/abs/1507.07909. 41

[177] P. Haghighatkhah, A. Fokkens, P. Sommerauer, B. Speckmann, and K. Verbeek. Better Hit the Nail on the Head than Beat around the Bush: Removing Protected Attributes with a Single Projection. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8395–8416, Dec. 2022. URL https://aclanthology.org/2022.emnlp-main.575. 45

[178] A. Hamdi, K. Shaban, A. Erradi, A. Mohamed, S. K. Rumi, and F. D. Salim. Spatiotemporal data mining: A survey on challenges and open problems. *Artificial Intelligence Review*, 55(2):1441–1488, Feb. 2022. ISSN 1573-7462. doi: 10.1007/s10462-021-09994-y. URL https://doi.org/10.1007/s10462-021-09994-y. 173

[179] A. Hamilton and J. Madison. *The Federalist Papers*. Ryerson University, Feb. 2022. URL https://openlibrary-repo.ecampusontario.ca/jspui/handle/123456789/1298. 35

[180] Y. Han, S. Li, Y. Cao, Q. Ma, and M. Yoshikawa. Voice-Indistinguishability: Protecting Voiceprint In Privacy-Preserving Speech Data Release. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, July 2020. doi: 10.1109/ICME46284.2020.9102875. 64, 69, 171

[181] M. Hay, C. Li, G. Miklau, and D. Jensen. Accurate estimation of the degree distribution of private networks. In *Data Mining, 2009. ICDM'09. Ninth IEEE International Conference on*, pages 169–178. IEEE, 2009. 96

[182] M. Hay, A. Machanavajjhala, G. Miklau, Y. Chen, and D. Zhang. Principled evaluation of differentially private algorithms using dpbench. In *Proceedings of the 2016 International Conference on Management of Data*, pages 139–154, 2016. 163

[183] M. Hay, A. Machanavajjhala, G. Miklau, Y. Chen, D. Zhang, and G. Bissias. Exploring privacy-accuracy tradeoffs using dpcomp. In *Proceedings of the 2016 International Conference on Management of Data*, pages 2101–2104, 2016. 163

[184] B. He, Y. Guan, J. Cheng, K. Cen, and W. Hua. CRFs based de-identification of medical records. *Journal of biomedical informatics*, 58:S39–S46, 2015. 51

[185] S. He and L. Schomaker. GR-RNN: Global-context residual recurrent neural networks for writer identification. *Pattern Recognition*, 117:107975, 2021. 41

[186] X. He, A. Machanavajjhala, and B. Ding. Blowfish privacy: Tuning privacy-utility trade-offs using policies. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pages 1447–1458. ACM, 2014. 96

[187] A. Hern. Fitness tracking app strava gives away location of secret US army bases. 2018. URL https://www.theguardian.com/world/2018/jan/28/fitness-tracking-app-gives-away-location-of-secret-us-army-bases. 3, 122

[188] S. Hill, Z. Zhou, L. Saul, and H. Shacham. On the (In)effectiveness of Mosaicing and Blurring as Tools for Document Redaction. *Proceedings on Privacy Enhancing Technologies*, 2016(4):403–417, Oct. 2016. ISSN 2299-0984. doi: 10.1515/popets-2016-0047. URL https://petsymposium.org/popets/2016/popets-2016-0047.php. 58

[189] G. Hinton, N. Srivastava, and K. Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, 14(8):2, 2012. 113

[190] E. Hjelmås and B. K. Low. Face Detection: A Survey. *Computer Vision and Image Understanding*, 83(3):236–274, Sept. 2001. ISSN 1077-3142. doi: 10.1006/

cviu.2001.0921. URL https://www.sciencedirect.com/science/article/pii/S107731420190921X. 41

[191] S. Hochreiter and J. Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9 (8):1735–1780, Nov. 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. 52, 172

[192] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*, 2019. 107

[193] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd. spaCy: Industrial-strength Natural Language Processing in Python, 2020. URL https://github.com/explosion/spaCy. 51

[194] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed. HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units, June 2021. URL http://arxiv.org/abs/2106.07447. 55

[195] T. Huddleston, Jr. Can you get sued over a negative Yelp review?, Oct. 2019. URL https://www.cnbc.com/2019/10/10/can-you-get-sued-over-a-negative-yelp-review.html. 3, 100

[196] H. Hukkelås and F. Lindseth. DeepPrivacy2: Towards Realistic Full-Body Anonymization, Nov. 2022. URL http://arxiv.org/abs/2211.09454. 61

[197] H. Hukkelås, R. Mester, and F. Lindseth. DeepPrivacy: A Generative Adversarial Network for Face Anonymization. In G. Bebis, R. Boyle, B. Parvin, D. Koracin, D. Ushizima, S. Chai, S. Sueda, X. Lin, A. Lu, D. Thalmann, C. Wang, and P. Xu, editors, *Advances in Visual Computing*, volume 11844 of *Lecture Notes in Computer Science*, pages 565–578, Cham, 2019. Springer International Publishing. ISBN 978-3-030-33720-9. doi: 10.1007/978-3-030-33720-9_44. 61

[198] P. Humbert. Sur les fonctions hypercylindriques. *C. R. Acad. Sci., Paris*, 171:490–492, 1920. 129

[199] P. Humbert. IX.—The Confluent Hypergeometric Functions of Two Variables. *Proceedings of the Royal Society of Edinburgh*, 41:73–96, 1922. 129

[200] P. Ilia, I. Polakis, E. Athanasopoulos, F. Maggi, and S. Ioannidis. Face/Off: Preventing Privacy Leakage From Photos in Social Networks. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 781–792, Denver Colorado USA, Oct. 2015. ACM. ISBN 978-1-4503-3832-5. doi: 10.1145/2810103.2813603. URL https://dl.acm.org/doi/10.1145/2810103.2813603. 58

[201] D. Impedovo and G. Pirlo. Automatic signature verification: The state of the art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(5):609–635, 2008. 40

[202] F. Iqbal, R. Hadjidj, B. C. M. Fung, and M. Debbabi. A novel approach of mining write-prints for authorship attribution in e-mail forensics. *Digital Investigation*, 5: S42–S51, Sept. 2008. ISSN 1742-2876. doi: 10.1016/j.diin.2008.05.001. URL https://www.sciencedirect.com/science/article/pii/S1742287608000315. 37

[203] F. Iqbal, M. Debbabi, and B. C. Fung. *Machine Learning for Authorship Attribution and Cyber Forensics*. Springer, 2020. 37

[204] A. Irum and A. Salman. Speaker Verification Using Deep Neural Networks: A Review. *International Journal of Machine Learning and Computing*, 9(1):20–25, 2019. ISSN 20103700. doi: 10.18178/ijmlc.2019.9.1.760. URL http://www.ijmlc.org/show-83-883-1.html. 39

[205] R. Jafri and H. R. Arabnia. A Survey of Face Recognition Techniques. *Journal of Information Processing Systems*, 5(2):41–68, 2009. ISSN 1976-913X. doi: 10.3745/JIPS.2009.5.2.041. URL https://koreascience.kr/article/JAKO200920237949770.page. 42

[206] S. R. Jammalamadaka and T. Kozubowski. A new family of circular models: The wrapped laplace distributions. *Advances and applications in statistics*, 3(1):77–103, 2003. 141

[207] M. Jawurek, M. Johns, and K. Rieck. Smart metering de-pseudonymization. In *Proceedings of the 27th Annual Computer Security Applications Conference*, ACSAC '11, pages 227–236, New York, NY, USA, Dec. 2011. Association for Computing Machinery. ISBN 978-1-4503-0672-0. doi: 10.1145/2076732.2076764. URL https://doi.org/10.1145/2076732.2076764. 3, 76, 100

[208] H. Jiang, J. Li, P. Zhao, F. Zeng, Z. Xiao, and A. Iyengar. Location Privacy-preserving Mechanisms in Location-based Services: A Comprehensive Survey. *ACM Computing Surveys*, 54(1):4:1–4:36, Jan. 2021. ISSN 0360-0300. doi: 10.1145/3423165. URL https://doi.org/10.1145/3423165. 62

[209] Q. Jin, A. R. Toth, T. Schultz, and A. W. Black. Speaker de-identification via voice transformation. In *2009 IEEE Workshop on Automatic Speech Recognition & Understanding*, pages 529–533, Moreno, Italy, Dec. 2009. IEEE. ISBN 978-1-4244-5478-5. doi: 10.1109/ASRU.2009.5373356. URL http://ieeexplore.ieee.org/document/5373356/. 53

[210] Q. Jin, A. R. Toth, T. Schultz, and A. W. Black. Voice convergin: Speaker de-identification by voice transformation. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3909–3912, Apr. 2009. doi: 10.1109/ICASSP.2009.4960482. 53

[211] M. L. Jockers and D. M. Witten. A comparative study of machine learning methods for authorship attribution. *Literary and Linguistic Computing*, 25(2):215–223, June 2010. ISSN 0268-1145. doi: 10.1093/llc/fqq001. URL https://doi.org/10.1093/llc/fqq001. 36

[212] F. Johansson et al. *mpmath: a Python library for arbitrary-precision floating-point arithmetic (version 1.1.0)*, 12 2018. URL http://mpmath.org/. 146

[213] V. John, L. Mou, H. Bahuleyan, and O. Vechtomova. Disentangled representation learning for non-parallel text style transfer. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 424–434, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1041. URL https://aclanthology.org/P19-1041. 11, 101, 108, 109

[214] A. E. W. Johnson, L. Bulgarelli, and T. J. Pollard. Deidentification of Free-Text Medical Records Using Pre-Trained Bidirectional Transformers. In *Proceedings of the ACM Conference on Health, Inference, and Learning*, CHIL '20, pages 214–221, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 978-1-4503-7046-2. doi: 10.1145/3368555.3384455. URL https://doi.org/10.1145/3368555.3384455. 51

[215] E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python, 2001–. URL http://www.scipy.org/. 87

[216] A. Jourabloo, X. Yin, and X. Liu. Attribute Preserved Face De-identification. In *2015 International Conference on Biometrics (ICB)*, pages 278–285, Phuket, Thailand, May 2015. IEEE. ISBN 978-1-4799-7824-3. doi: 10.1109/ICB.2015.7139096. URL http://ieeexplore.ieee.org/document/7139096/. 59

[217] P. Juola, J. Sofko, and P. Brennan. A Prototype for Authorship Attribution Studies. *Literary and Linguistic Computing*, 21(2):169–178, June 2006. ISSN 0268-1145. doi: 10.1093/llc/fql019. URL https://doi.org/10.1093/llc/fql019. 36

[218] M. M. Kabir, M. F. Mridha, J. Shin, I. Jahan, and A. Q. Ohi. A Survey of Speaker Recognition: Fundamental Theories, Recognition Methods and Opportunities. *IEEE Access*, 9:79236–79263, May 2021. ISSN 2169-3536. doi: 10.1109/ACCESS.2021.3084299. 39

[219] G. Kacmarcik and M. Gamon. Obfuscating document stylometry to preserve author anonymity. In *Proceedings of the COLING/ACL on Main Conference Poster Sessions*, pages 444–451. Association for Computational Linguistics, 2006. URL http://dl.acm.org/citation.cfm?id=1273131. 51, 52, 95

[220] D. Kagan, G. F. Alpert, and M. Fire. Zooming Into Video Conferencing Privacy and Security Threats, July 2020. URL http://arxiv.org/abs/2007.01059. 39

[221] H. Kai, S. Takamichi, S. Shiota, and H. Kiya. Lightweight Voice Anonymization Based on Data-Driven Optimization of Cascaded Voice Modification Modules. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 560–566, Jan. 2021. doi: 10.1109/SLT48900.2021.9383535. 54

[222] M. K. Kalera, S. Srihari, and A. Xu. Offline signature verification and identification using distance statistics. *International Journal of Pattern Recognition and Artificial Intelligence*, 18(07):1339–1360, Nov. 2004. ISSN 0218-0014. doi: 10.1142/S0218001404003630. URL https://www.worldscientific.com/doi/abs/10.1142/s0218001404003630. 41

[223] M. Kaplan. May I Ask Who's Calling? Named Entity Recognition on Call Center Transcripts for Privacy Law Compliance, Oct. 2020. URL http://arxiv.org/abs/2010.15598. 38

[224] G. Karadzhov, T. Mihaylova, Y. Kiprov, G. Georgiev, I. Koychev, and P. Nakov. The Case for Being Average: A Mediocrity Approach to Style Masking and Author Obfuscation. In G. J. Jones, S. Lawless, J. Gonzalo, L. Kelly, L. Goeuriot, T. Mandl, L. Cappellato, and N. Ferro, editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, Lecture Notes in Computer Science, pages 173–185, Cham, 2017. Springer International Publishing. ISBN 978-3-319-65813-1. doi: 10.1007/978-3-319-65813-1_18. 52

[225] M. Kasar, D. Bhattacharyya, and T.-H. Kim. Face Recognition Using Neural Network: A Review. *International Journal of Security and Its Applications*, 10:81–100, Mar. 2016. doi: 10.14257/ijsia.2016.10.3.08. 42

[226] S. Kasiviswanathan, K. Nissim, S. Raskhodnikova, and A. Smith. Analyzing graphs with node differential privacy. In *Theory of Cryptography*, pages 457–476. Springer, 2013. 96

[227] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith. What can we learn privately? *SIAM Journal on Computing*, 40(3):793–826, 2011. 19

[228] L. G. Kersta. Voiceprint identification. *The Journal of the Acoustical Society of America*, 34(5):725–725, 1962. 39

[229] Y. Keswani, H. Trivedi, P. Mehta, and P. Majumder. Author masking through translation. In *CLEF*, 2016. 52

[230] K. Khin, P. Burckhardt, and R. Padman. A Deep Learning Architecture for De-identification of Patient Notes: Implementation and Evaluation, Oct. 2018. URL http://arxiv.org/abs/1810.01570. 51

[231] J. W. Kim, D.-H. Kim, and B. Jang. Application of local differential privacy to collection of indoor positioning data. *Ieee Access*, 6:4276–4286, 2018. 163

[232] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2017. 113

[233] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In Y. Bengio and Y. LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6114. 100, 101, 104

[234] D. P. Kingma and M. Welling. An introduction to variational autoencoders. *CoRR*, abs/1906.02691, 2019. URL http://arxiv.org/abs/1906.02691. 101

[235] M. Koepf, F. Kleber, and R. Sablatnig. Writer Identification and Writer Retrieval Using Vision Transformer for Forensic Documents. In S. Uchida, E. Barney, and V. Eglin, editors, *Document Analysis Systems*, Lecture Notes in Computer Science, pages 352–366, Cham, 2022. Springer International Publishing. ISBN 978-3-031-06555-2. doi: 10.1007/978-3-031-06555-2_24. 41

[236] M. Koppel and J. Schler. Authorship Verification As a One-class Classification Problem. In *Proceedings of the Twenty-first International Conference on Machine Learning*, ICML '04, pages 62–, New York, NY, USA, 2004. ACM. ISBN 978-1-58113-838-2. doi: 10.1145/1015330.1015448. URL http://doi.acm.org/10.1145/1015330.1015448. 51

[237] M. Koppel, J. Schler, and E. Bonchek-Dokow. Measuring Differentiability: Unmasking Pseudonymous Authors. *Journal of Machine Learning Research*, 8(6):1261–1276, 2007. URL http://u.cs.biu.ac.il/~koppel/papers/authorship-jmlr-final.pdf. 51

[238] M. Kosinski. Facial recognition technology can expose political orientation from naturalistic facial images. *Scientific Reports*, 11(1):100, Jan. 2021. ISSN 2045-2322. doi:

10.1038/s41598-020-79310-1. URL https://www.nature.com/articles/s41598-020-79310-1. 39

[239] M. Kotadia. AWS urges developers to scrub GitHub of secret keys, Mar. 2014. URL https://www.itnews.com.au/news/aws-urges-developers-to-scrub-github-of-secret-keys-375785. 35

[240] F. Koufogiannis, S. Han, and G. J. Pappas. Optimality of the laplace mechanism in differential privacy. *arXiv preprint arXiv:1504.00065*, 2015. 27

[241] S. Krishna, R. Gupta, and C. Dupuy. ADePT: Auto-encoder based Differentially Private Text Transformation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2435–2439, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.207. URL https://aclanthology.org/2021.eacl-main.207. 63, 116

[242] J. L. Kröger, O. H.-M. Lutz, and P. Raschke. Privacy Implications of Voice and Speech Analysis – Information Disclosure by Inference. In M. Friedewald, M. Önen, E. Lievens, S. Krenn, and S. Fricker, editors, *Privacy and Identity Management. Data for Better Living: AI and Privacy: 14th IFIP WG 9.2, 9.6/11.7, 11.6/SIG 9.2.2 International Summer School, Windisch, Switzerland, August 19–23, 2019, Revised Selected Papers*, IFIP Advances in Information and Communication Technology, pages 242–258. Springer International Publishing, Cham, 2020. ISBN 978-3-030-42504-3. doi: 10.1007/978-3-030-42504-3_16. URL https://doi.org/10.1007/978-3-030-42504-3_16. 38

[243] J. Krumm. A survey of computational location privacy. *Personal and Ubiquitous Computing*, 13(6):391–399, Aug. 2009. ISSN 1617-4909, 1617-4917. doi: 10.1007/s00779-008-0212-5. URL http://link.springer.com/10.1007/s00779-008-0212-5. 3, 122

[244] A. Kumar, A. Kaur, and M. Kumar. Face detection techniques: A review. *Artificial Intelligence Review*, 52(2):927–948, Aug. 2019. ISSN 1573-7462. doi: 10.1007/s10462-018-9650-2. URL https://doi.org/10.1007/s10462-018-9650-2. 41

[245] M. Kumar. Hundreds of SSH Private Keys exposed via GitHub Search, Jan. 2013. URL https://thehackernews.com/2013/01/hundreds-of-ssh-private-keys-exposed.html. 35

[246] E. Kummer. Über die hypergeometrische Reihe ... . *Journal für die reine und angewandte Mathematik*, 15:39–83, 1836. 129

[247] E. Kummer. Über die hypergeometrische Reihe ... . (Fortsetzung). *Journal für die reine und angewandte Mathematik*, 15:127–172, 1836. 129

[248] H.-K. J. Kuo, Z. Tüske, S. Thomas, Y. Huang, K. Audhkhasi, B. Kingsbury, G. Kurata, Z. Kons, R. Hoory, and L. Lastras. End-to-End Spoken Language Understanding Without Full Transcripts. In *Interspeech 2020*, pages 906–910. ISCA, Oct. 2020. doi: 10.21437/Interspeech.2020-2924. URL https://www.isca-speech.org/archive/interspeech_2020/kuo20_interspeech.html. 38

[249] G. Kurz and U. D. Hanebeck. Stochastic sampling of the hyperspherical von mises–fisher distribution without rejection methods. In *2015 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pages 1–6, Oct 2015. 11, 127, 133, 138, 140, 164

[250] V. Kužina, E. Vušak, and A. Jović. Methods for Automatic Sensitive Data Detection in Large Datasets: A Review. In *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, pages 187–192, Sept. 2021. doi: 10.23919/MIPRO52101.2021.9596735. 43

[251] S. K. Lam, A. Pitrou, and S. Seibert. Numba: A llvm-based python jit compiler. In *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*, LLVM '15, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450340052. doi: 10.1145/2833157.2833162. URL https://doi.org/10.1145/2833157.2833162. 146

[252] K. Lander, V. Bruce, and H. Hill. Evaluating the Effectiveness of Pixelation and Blurring on Masking the Identity of Familiar Faces. *Applied Cognitive Psychology*, 15(1):101–116, 2001. ISSN 1099-0720. doi: 10.1002/1099-0720(200101/02)15:1<101::AID-ACP697>3.0.CO;2-7. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/1099-0720%28200101/02%2915%3A1%3C101%3A%3AAID-ACP697%3E3.0.CO%3B2-7. 58

[253] F. Leclerc and R. Plamondon. Automatic signature verification: The state of the art—1989–1993. *Progress in Automatic Signature Verification*, pages 3–20, 1994. 41

[254] G. Letournel, A. Bugeau, V.-T. Ta, and J.-P. Domenger. Face de-identification with expressions preservation. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 4366–4370, Sept. 2015. doi: 10.1109/ICIP.2015.7351631. 61

[255] A. Li, J. Guo, H. Yang, and Y. Chen. *DeepObfuscator: Adversarial Training Framework for Privacy-Preserving Image Classification*. Sept. 2019. 46

[256] J. Li, R. Zheng, and H. Chen. From Fingerprint to Writeprint. *Communications of the ACM*, 49(4):76–82, 2006. URL http://dl.acm.org/citation.cfm?id=1121951. 36

[257] J. Li, A. Sun, J. Han, and C. Li. A Survey on Deep Learning for Named Entity Recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):50–70, Jan. 2022. ISSN 1558-2191. doi: 10.1109/TKDE.2020.2981314. 34

[258] N. Li, T. Li, and S. Venkatasubramanian. T-closeness: Privacy beyond k-anonymity and l-diversity. In *2007 IEEE 23rd International Conference on Data Engineering*, pages 106–115. IEEE, 2007. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4221659. 61

[259] N. Li, M. Lyu, D. Su, and W. Yang. Differential privacy: From theory to practice. *Synthesis Lectures on Information Security, Privacy, & Trust*, 2016. 15, 17

[260] T. Li and C. Clifton. Differentially Private Imaging via Latent Space Manipulation, Apr. 2021. URL http://arxiv.org/abs/2103.05472. 66

[261] T. Li and L. Lin. AnonymousNet: Natural Face De-Identification With Measurable Privacy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. URL https://openaccess.thecvf.com/content_CVPRW_2019/html/CV-COPS/Li_AnonymousNet_Natural_Face_De-Identification_With_Measurable_Privacy_CVPRW_2019_paper.html. 61

[262] Y. Li, N. Vishwamitra, B. P. Knijnenburg, H. Hu, and K. Caine. Effectiveness and Users' Experience of Obfuscation as a Privacy-Enhancing Technology for Sharing Photos. *Proceedings of the ACM on Human-Computer Interaction*, 1(CSCW):67:1–67:24, Dec. 2017. doi: 10.1145/3134702. URL https://doi.org/10.1145/3134702. 58

[263] Y. Li, T. Baldwin, and T. Cohn. Towards Robust and Privacy-preserving Text Representations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 25–30, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2005. URL https://aclanthology.org/P18-2005. 44

[264] B. Liu. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167, 2012. 76

[265] Z. Liu, Y. Chen, B. Tang, X. Wang, Q. Chen, H. Li, J. Wang, Q. Deng, and S. Zhu. Automatic de-identification of electronic medical records using token-level and character-level conditional random fields. *Journal of biomedical informatics*, 58:S47–S52, 2015. 51

[266] Z. Liu, B. Tang, X. Wang, and Q. Chen. De-identification of clinical notes via recurrent neural network and conditional random field. *Journal of Biomedical Informatics*, 75:S34–S42, 2017. ISSN 1532-0464. doi: 10.1016/j.jbi.2017.05.023. URL https://www.sciencedirect.com/science/article/pii/S1532046417301223. 51

[267] S. Lounici, M. Rosa, C. Negri, S. Trabelsi, and M. Önen. Optimizing Leak Detection in Open-source Platforms with Machine Learning Techniques:. In *Proceedings of the 7th International Conference on Information Systems Security and Privacy*, pages 145–159, Online Streaming, — Select a Country —, 2021. SCITEPRESS - Science and Technology Publications. ISBN 978-989-758-491-6. doi: 10.5220/0010238101450159. URL https://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0010238101450159. 35, 42

[268] L. Lyu, X. He, and Y. Li. Differentially Private Representation for NLP: Formal Guarantee and An Empirical Study on Privacy and Fairness. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2355–2365, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.213. URL https://aclanthology.org/2020.findings-emnlp.213. 49

[269] L. Lyu, Y. Li, X. He, and T. Xiao. Towards Differentially Private Text Representations. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1813–1816, New York, NY, USA, July 2020. Association for Computing Machinery. ISBN 978-1-4503-8016-4. URL https://doi.org/10.1145/3397271.3401260. 48

[270] P. Ma, B. Jiang, Z. Lu, N. Li, and Z. Jiang. Cybersecurity named entity recognition using bidirectional long short-term memory with conditional random fields. *Tsinghua Science and Technology*, 26(3):259–265, June 2021. ISSN 1007-0214. doi: 10.26599/TST.2019.9010033. URL https://ieeexplore.ieee.org/document/9220752/. 34

[271] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL https://aclanthology.org/P11-1015. 111

[272] A. Machanavajjhala, A. Korolova, and A. Sarma. Personalized social recommendations: accurate or private. *Proceedings of the VLDB Endowment*, 4(7):440–450, 2011. 96

[273] D. Machanavajjhala, A.and Kifer, J. Abowd, J. Gehrke, and L. Vilhuber. Privacy: Theory meets practice on the map. In *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, pages 277–286. IEEE Computer Society, 2008. 96

[274] A. Mahendran and A. Vedaldi. Understanding Deep Image Representations by Inverting Them. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5188–5196, Boston, MA, USA, June 2015. IEEE. ISBN 978-1-4673-6964-0. doi: 10.1109/CVPR.2015.7299155. URL http://ieeexplore.ieee.org/document/7299155/. 46

[275] G. Maheshwari, P. Denis, M. Keller, and A. Bellet. Fair NLP Models with Differentially Private Text Encoders. (arXiv:2205.06135), May 2022. URL http://arxiv.org/abs/2205.06135. 49, 69, 171

[276] A. Mahmood, F. Ahmad, Z. Shafiq, P. Srinivasan, and F. Zaffar. A girl has no name: Automated authorship obfuscation using mutant-x. *Proceedings on Privacy Enhancing Technologies*, 2019(4):54–71, 2019. doi: doi:10.2478/popets-2019-0058. URL https://doi.org/10.2478/popets-2019-0058. 53

[277] G. Maldoff. Top 10 operational impacts of the GDPR: Part 8 - Pseudonymization, Feb. 2016. URL https://iapp.org/news/a/top-10-operational-impacts-of-the-gdpr-part-8-pseudonymization/. 5

[278] K. V. Mardia and P. E. Jupp. *Directional statistics*, volume 494. John Wiley & Sons, 2000. 124

[279] I. Martínez-Ponte, X. Desurmont, J. Meessen, and J.-F. Delaigle. ROBUST HUMAN FACE HIDING ENSURING PRIVACY. In *Proceedings of the International Workshop on Image Analysis for Multimedia Interactive Services*, volume 4, 2005. 57

[280] J. Martinsson, E. L. Zec, D. Gillblad, and O. Mogren. Adversarial representation learning for synthetic replacement of private attributes. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 1291–1299, Dec. 2021. doi: 10.1109/BigData52589.2021.9671802. 46, 54

[281] I. Masi, Y. Wu, T. Hassner, and P. Natarajan. Deep Face Recognition: A Survey. In *2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 471–478, Oct. 2018. doi: 10.1109/SIBGRAPI.2018.00067. 42

[282] D. Matrouf, J.-F. Bonastre, and C. Fredouille. Effect of Speech Transformation on Impostor Acceptance. In *2006 IEEE International Conference on Acoustics Speech and*

Signal Processing Proceedings, volume 1, pages I–I, May 2006. doi: 10.1109/ICASSP.2006.1660175. 53

[283] Matt D'Zmura. Behind the scenes: popular times and live busyness information, Oct. 2020. URL https://blog.google/products/maps/maps101-popular-times-and-live-busyness-information/. 2

[284] J. Mattern, Z. Jin, B. Weggenmann, B. Schoelkopf, and M. Sachan. Differentially Private Language Models for Secure Data Sharing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4860–4873, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.323. 12

[285] J. Mattern, B. Weggenmann, and F. Kerschbaum. The Limits of Word Level Differential Privacy. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 867–881, Seattle, United States, July 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.findings-naacl.65. 12, 63

[286] J. Mattern, B. Weggenmann, and F. Kerschbaum. The Limits of Word Level Differential Privacy, May 2022. URL http://arxiv.org/abs/2205.02130. 12

[287] B. W. Matthews. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA) - Protein Structure*, 405(2):442–451, 1975. ISSN 0005-2795. doi: https://doi.org/10.1016/0005-2795(75)90109-9. URL https://www.sciencedirect.com/science/article/pii/0005279575901099. 112

[288] C. O. Mawalim, K. Galajit, J. Karnjana, S. Kidani, and M. Unoki. Speaker anonymization by modifying fundamental frequency and x-vector singular value. *Computer Speech & Language*, 73:101326, May 2022. ISSN 0885-2308. doi: 10.1016/j.csl.2021.101326. URL https://www.sciencedirect.com/science/article/pii/S0885230821001194. 55

[289] S. Mcadams. *Spectral Fusion, Spectral Parsing and the Formation of Auditory Images*. PhD thesis, Stanford University, May 1984. 54

[290] A. W. McDonald, S. Afroz, A. Caliskan, A. Stolerman, and R. Greenstadt. Use fewer instances of the letter "i": Toward writing style anonymization. In *International Symposium on Privacy Enhancing Technologies Symposium*, pages 299–318. Springer, 2012. URL http://link.springer.com/chapter/10.1007/978-3-642-31680-7_16. 36, 52, 78, 90, 95

[291] R. McPherson, R. Shokri, and V. Shmatikov. Defeating Image Obfuscation with Deep Learning, Sept. 2016. URL http://arxiv.org/abs/1609.00408. 58

[292] F. McSherry. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, pages 19–30. ACM, 2009. 64

[293] F. McSherry. How many secrets do you have? *Blog post*, 2017. URL https://github.com/frankmcsherry/blog/blob/master/posts/2017-02-08.md. 18

[294] F. McSherry and K. Talwar. Mechanism design via differential privacy. In *Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on*, pages 94–103. IEEE, 2007. 9, 28, 29, 48, 84, 163

[295] B. Meden, Z. Emersic, V. Struc, and P. Peer. κ-Same-Net: Neural-Network-Based Face Deidentification. In *2017 International Conference and Workshop on Bioinspired Intelligence (IWOBI)*, pages 1–7, July 2017. doi: 10.1109/IWOBI.2017.7985521. 59

[296] B. Meden, R. C. Mallı, S. Fabijan, H. K. Ekenel, V. Štruc, and P. Peer. Face deidentification with generative deep neural networks. *IET Signal Processing*, 11(9):1046–1054, 2017. ISSN 1751-9683. doi: 10.1049/iet-spr.2017.0049. URL https://onlinelibrary.wiley.com/doi/abs/10.1049/iet-spr.2017.0049. 59

[297] B. Meden, Ž. Emeršič, V. Štruc, and P. Peer. K-Same-Net: K-Anonymity with Generative Deep Neural Networks for Face Deidentification. *Entropy*, 20(1):60, Jan. 2018. ISSN 1099-4300. doi: 10.3390/e20010060. URL https://www.mdpi.com/1099-4300/20/1/60. 59

[298] B. Meden, P. Rot, P. Terhörst, N. Damer, A. Kuijper, W. J. Scheirer, A. Ross, P. Peer, and V. Štruc. Privacy–Enhancing Face Biometrics: A Comprehensive Survey. *IEEE Transactions on Information Forensics and Security*, 16:4147–4183, 2021. ISSN 1556-6021. doi: 10.1109/TIFS.2021.3096024. 62

[299] N. Mehdy, C. Kennington, and H. Mehrpouyan. Privacy Disclosures Detection in Natural-Language Text Through Linguistically-Motivated Artificial Neural Networks. In J. Li, Z. Liu, and H. Peng, editors, *Security and Privacy in New Computing Environments*, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, pages 152–177, Cham, 2019. Springer International Publishing. ISBN 978-3-030-21373-2. doi: 10.1007/978-3-030-21373-2_14. 34

[300] M. Meli, M. R. McNiece, and B. Reaves. How Bad Can It Git? Characterizing Secret Leakage in Public GitHub Repositories. In *Proceedings 2019 Network and Distributed System Security Symposium*, San Diego, CA, 2019. Internet Society. ISBN 978-1-891562-55-6. doi: 10.14722/ndss.2019.23418. URL https://www.ndss-symposium.org/wp-content/uploads/2019/02/ndss2019_04B-3_Meli_paper.pdf. 35

[301] J. Memon, M. Sami, R. A. Khan, and M. Uddin. Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR). *IEEE Access*, 8:142642–142668, 2020. ISSN 2169-3536. doi: 10.1109/ACCESS.2020.3012542. 39

[302] X. Miao, X. Wang, E. Cooper, J. Yamagishi, and N. Tomashenko. Analyzing Language-Independent Speaker Anonymization Framework under Unseen Conditions, Mar. 2022. URL http://arxiv.org/abs/2203.14834. 55

[303] X. Miao, X. Wang, E. Cooper, J. Yamagishi, and N. Tomashenko. Language-Independent Speaker Anonymization Approach using Self-Supervised Pre-Trained Models, Apr. 2022. URL http://arxiv.org/abs/2202.13097. 55

[304] Microsoft. Presidio - Data Protection and Anonymization API. Microsoft, May 2022. URL https://github.com/microsoft/presidio. 43, 51

[305] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. 87

[306] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013. 87

[307] S. Minaee, A. Abdolrashidi, H. Su, M. Bennamoun, and D. Zhang. Biometrics Recognition Using Deep Learning: A Survey, Feb. 2021. URL http://arxiv.org/abs/1912.00271. 32

[308] À. Miranda-Pascual, P. Guerra-Balboa, J. Parra-Arnau, T. Strufe, and J. Forné. SoK: Differentially Private Publication of Trajectory Data. In *Proceedings on Privacy Enhancing Technologies (PoPETs)*, 2023. URL https://publikationen.bibliothek.kit.edu/1000154050. 67

[309] I. Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275, 2017. doi: 10.1109/CSF.2017.11. 18, 20, 21, 22, 25, 27, 101, 107

[310] M. Mirza and S. Osindero. Conditional Generative Adversarial Nets, Nov. 2014. URL http://arxiv.org/abs/1411.1784. 61

[311] R. Mohd Hanifa, K. Isa, and S. Mohamad. A review on speaker recognition: Technology and challenges. *Computers & Electrical Engineering*, 90:107005, Mar. 2021. ISSN 0045-7906. doi: 10.1016/j.compeleceng.2021.107005. URL https://www.sciencedirect.com/science/article/pii/S0045790621000318. 39

[312] B. Mohit. Named Entity Recognition. In I. Zitouni, editor, *Natural Language Processing of Semitic Languages*, Theory and Applications of Natural Language Processing, pages 221–245. Springer, Berlin, Heidelberg, 2014. ISBN 978-3-642-45358-8. doi: 10.1007/978-3-642-45358-8_7. URL https://doi.org/10.1007/978-3-642-45358-8_7. 34

[313] A. Morales, J. Fierrez, R. Vera-Rodriguez, and R. Tolosana. SensitiveNets: Learning Agnostic Representations with Application to Face Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(6):2158–2164, June 2021. ISSN 0162-8828, 2160-9292, 1939-3539. doi: 10.1109/TPAMI.2020.3015420. URL https://ieeexplore.ieee.org/document/9163294/. 47

[314] A. Mosallanezhad, G. Beigi, and H. Liu. Deep Reinforcement Learning-based Text Anonymization against Private-Attribute Inference. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2360–2369, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1240. URL https://aclanthology.org/D19-1240. 44

[315] F. Mosteller and D. Wallace. Inference in an authorship problem: A comparative study of discrimination methods applied to the authorship of the disputed federalist papers. *Journal of the American Statistical Association*, 58(302):275–309, 1963. 35, 76

[316] K. Murugadoss, A. Rajasekharan, B. Malin, V. Agarwal, S. Bade, J. R. Anderson, J. L. Ross, W. A. Faubion, J. D. Halamka, V. Soundararajan, and S. Ardhanari. Building a best-in-class automated de-identification tool for electronic health records through ensemble learning. *Patterns*, 2(6):100255, June 2021. ISSN 2666-3899. doi: 10.1016/j.patter.2021.100255. URL https://www.sciencedirect.com/science/article/pii/S2666389921000817. 51

[317] A. Narayanan and V. Shmatikov. How To Break Anonymity of the Netflix Prize Dataset, Nov. 2007. URL http://arxiv.org/abs/cs/0610105. 6, 37

[318] A. Narayanan and V. Shmatikov. Robust De-anonymization of Large Sparse Datasets. In *2008 IEEE Symposium on Security and Privacy (SP 2008)*, pages 111–125, May 2008. doi: 10.1109/SP.2008.33. 3, 37, 76, 100

[319] National Sleep Foundation. 2011 Sleep in America Poll – Technology Use and Sleep. 1(2):e10, 2015. ISSN 2352-7218. doi: 10.1016/j.sleh.2015.04.010. URL http://www.sciencedirect.com/science/article/pii/S2352721815000716. 155

[320] T. Neal, K. Sundararajan, A. Fatima, Y. Yan, Y. Xiang, and D. Woodard. Surveying Stylometry Techniques and Applications. *ACM Computing Surveys*, 50(6):1–36, Nov. 2018. ISSN 0360-0300, 1557-7341. doi: 10.1145/3132039. URL https://dl.acm.org/doi/10.1145/3132039. 36

[321] I. Neamatullah, M. M. Douglass, L.-w. H. Lehman, A. Reisner, M. Villarroel, W. J. Long, P. Szolovits, G. B. Moody, R. G. Mark, and G. D. Clifford. Automated de-identification of free-text medical records. *BMC Medical Informatics and Decision Making*, 8(1):32, July 2008. ISSN 1472-6947. doi: 10.1186/1472-6947-8-32. URL https://doi.org/10.1186/1472-6947-8-32. 50

[322] J. Neerbek, I. Assent, and P. Dolog. Detecting Complex Sensitive Information via Phrase Structure in Recursive Neural Networks. In D. Phung, V. S. Tseng, G. I. Webb, B. Ho, M. Ganji, and L. Rashidi, editors, *Advances in Knowledge Discovery and Data Mining*, Lecture Notes in Computer Science, pages 373–385, Cham, 2018. Springer International Publishing. ISBN 978-3-319-93040-4. doi: 10.1007/978-3-319-93040-4_30. 34

[323] J. Newsham. A Bad Glassdoor Review Led to a $1 Million Lawsuit, Aug. 2021. URL https://www.businessinsider.com/bad-glassdoor-reddit-review-led-to-a-1-million-lawsuit-2021-8. 3, 100

[324] E. Newton, L. Sweeney, and B. Malin. Preserving privacy by de-identifying face images. *IEEE Transactions on Knowledge and Data Engineering*, 17(2):232–243, Feb. 2005. ISSN 1558-2191. doi: 10.1109/TKDE.2005.32. 57, 58

[325] P.-G. Noé, M. Mohammadamini, D. Matrouf, T. Parcollet, A. Nautsch, and J.-F. Bonastre. Adversarial Disentanglement of Speaker Representation for Attribute-Driven Privacy Preservation, June 2021. URL http://arxiv.org/abs/2012.04454. 45

[326] P. Nousi, S. Papadopoulos, A. Tefas, and I. Pitas. Deep autoencoders for attribute preserving face de-identification. *Signal Processing: Image Communication*, 81:115699,

Feb. 2020. ISSN 0923-5965. doi: 10.1016/j.image.2019.115699. URL https://www.sciencedirect.com/science/article/pii/S0923596519304667. 59, 116

[327] Office for Civil Rights (OCR). Guidance Regarding Methods for De-identification of Protected Health Information in Accordance with the Health Insurance Portability and Accountability Act (HIPAA) Privacy Rule, Sept. 2012. URL https://www.hhs.gov/hipaa/for-professionals/privacy/special-topics/de-identification/index.html. 3, 4, 32, 33, 62, 96

[328] J. Oglesby and J. Mason. Optimisation of neural models for speaker identification. In *International Conference on Acoustics, Speech, and Signal Processing*, pages 261–264 vol.1, Apr. 1990. doi: 10.1109/ICASSP.1990.115617. 39

[329] S. J. Oh, R. Benenson, M. Fritz, and B. Schiele. Faceless Person Recognition: Privacy Implications in Social Media. In B. Leibe, J. Matas, N. Sebe, and M. Welling, editors, *Computer Vision – ECCV 2016*, Lecture Notes in Computer Science, pages 19–35, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46487-9. doi: 10.1007/978-3-319-46487-9_2. 58, 61

[330] S. J. Oh, M. Fritz, and B. Schiele. Adversarial Image Perturbation for Privacy Protection A Game Theory Perspective. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1491–1500, Venice, Oct. 2017. IEEE. ISBN 978-1-5386-1032-9. doi: 10.1109/ICCV.2017.165. URL http://ieeexplore.ieee.org/document/8237427/. 60

[331] T. E. Oliphant. *A guide to NumPy*, volume 1. Trelgol Publishing USA, 2006. 146

[332] T. Orekondy, M. Fritz, and B. Schiele. Connecting Pixels to Privacy and Utility: Automatic Redaction of Private Information in Images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8466–8475, 2018. URL https://openaccess.thecvf.com/content_cvpr_2018/html/Orekondy_Connecting_Pixels_to_CVPR_2018_paper.html. 57

[333] M. Osakwe. Announcing The Nightfall Developer Platform, Nov. 2021. URL https://nightfall.ai/announcing-the-nightfall-developer-platform. 42

[334] S. A. Osia, A. Taheri, A. S. Shamsabadi, K. Katevas, H. Haddadi, and H. R. Rabiee. Deep Private-Feature Extraction. *IEEE Transactions on Knowledge and Data Engineering*, 32(1):54–66, Jan. 2020. ISSN 1558-2191. doi: 10.1109/TKDE.2018.2878698. 46

[335] A. E. Ouadrhiri and A. Abdelhadi. Differential Privacy for Deep and Federated Learning: A Survey. *IEEE Access*, 10:22359–22380, 2022. ISSN 2169-3536. doi: 10.1109/ACCESS.2022.3151670. 18

[336] J. R. Padilla-López, A. A. Chaaraoui, and F. Flórez-Revuelta. Visual privacy protection methods: A survey. *Expert Systems with Applications*, 42(9):4177–4195, June 2015. ISSN 0957-4174. doi: 10.1016/j.eswa.2015.01.041. URL https://www.sciencedirect.com/science/article/pii/S0957417415000561. 62

[337] J.-s. Park, G.-w. Kim, and D.-h. Lee. Sensitive Data Identification in Structured Data through GenNER Model based on Text Generation and NER. In *Proceedings of the 2020 International Conference on Computing, Networks and Internet of Things*, CNIOT2020, pages 36–40, New York, NY, USA, Apr. 2020. Association for Computing Machinery. ISBN 978-1-4503-7771-3. doi: 10.1145/3398329.3398335. URL https://doi.org/10.1145/3398329.3398335. 34

[338] J. Patino, M. Todisco, A. Nautsch, and N. Evans. Speaker anonymisation using the McAdams coefficient. Technical Report RR-20-343, EURECOM, Feb. 2020. URL https://www.eurecom.fr/publication/6190. 54

[339] J. Patino, N. Tomashenko, M. Todisco, A. Nautsch, and N. Evans. Speaker Anonymisation Using the McAdams Coefficient. In *Interspeech 2021*, pages 1099–1103. ISCA, Aug. 2021. doi: 10.21437/Interspeech.2021-1070. URL https://www.isca-speech.org/archive/interspeech_2021/patino21_interspeech.html. 54

[340] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. 90

[341] B. Pellom and J. Hansen. An experimental study of speaker verification sensitivity to computer voice-altered imposters. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258)*, volume 2, pages 837–840 vol.2, Mar. 1999. doi: 10.1109/ICASSP.1999.759801. 53

[342] J. Pennington, R. Socher, and C. Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162. URL https://aclanthology.org/D14-1162. 62, 87, 88

[343] J. M. Perero-Codosero, F. M. Espinoza-Cuadros, and L. A. Hernández-Gómez. X-vector anonymization using autoencoders and adversarial training for preserving speech privacy. *Computer Speech & Language*, 74:101351, July 2022. ISSN 0885-2308. doi: 10.1016/j.csl.2022.101351. URL https://www.sciencedirect.com/science/article/pii/S088523082200002X. 55

[344] D. Petro. Never Use Text Pixelation To Redact Sensitive Information, Feb. 2022. URL https://bishopfox.com/blog/unredacter-tool-never-pixelation. 58

[345] D. Petro. Unredacter. Bishop Fox, Aug. 2022. URL https://github.com/BishopFox/unredacter. 58

[346] F. Pittaluga, S. Koppal, and A. Chakrabarti. Learning Privacy Preserving Encodings Through Adversarial Training. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 791–799, Jan. 2019. doi: 10.1109/WACV.2019.00089. 46

[347] R. Plamondon and G. Lorette. Automatic signature verification and writer identification—the state of the art. *Pattern recognition*, 22(2):107–131, 1989. 41

[348] R. Plant, D. Gkatzia, and V. Giuffrida. CAPE: Context-Aware Private Embeddings for Private Language Learning, Aug. 2021. URL http://arxiv.org/abs/2108.12318. 49

[349] A. Polyak, Y. Adi, J. Copet, E. Kharitonov, K. Lakhotia, W.-N. Hsu, A. Mohamed, and E. Dupoux. Speech Resynthesis from Discrete Disentangled Self-Supervised Representations. In *Interspeech 2021*, pages 3615–3619. ISCA, Aug. 2021. doi: 10.21437/Interspeech.2021-475. URL https://www.isca-speech.org/archive/interspeech_2021/polyak21_interspeech.html. 55

[350] M. Potthast, F. Schremmer, M. Hagen, and B. Stein. Overview of the author obfuscation task at PAN 2018: A new approach to measuring safety. In L. Cappellato, N. Ferro, J.-Y. Nie, and L. Soulier, editors, *Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, Avignon, France, September 10-14, 2018*, volume 2125 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2018. URL http://ceur-ws.org/Vol-2125/invited_paper_16.pdf. 52

[351] G. P. Prajapati, D. K. Singh, P. P. Amin, and H. A. Patil. Voice privacy using CycleGAN and time-scale modification. *Computer Speech & Language*, 74:101353, July 2022. ISSN 0885-2308. doi: 10.1016/j.csl.2022.101353. URL https://www.sciencedirect.com/science/article/pii/S0885230822000031. 53

[352] F. G. Preston, Y. Meng, Y. Zheng, J. Hsuan, K. J. Hamill, and A. G. McCormick. Informed Consent In Facial Photograph Publishing: A Cross-sectional Pilot Study To Determine The Effectiveness Of Deidentification Methods. *Journal of Empirical Research on Human Research Ethics*, 17(3):373–381, July 2022. ISSN 1556-2646. doi: 10.1177/15562646221075459. URL https://doi.org/10.1177/15562646221075459. 57

[353] V. Primault, S. B. Mokhtar, C. Lauradoux, and L. Brunie. Differentially Private Location Privacy in Practice, Oct. 2014. URL http://arxiv.org/abs/1410.7744. 3, 68, 122

[354] V. Primault, A. Boutet, S. B. Mokhtar, and L. Brunie. The Long Road to Computational Location Privacy: A Survey. *IEEE Communications Surveys & Tutorials*, 21(3):2772–2793, 2019. ISSN 1553-877X. doi: 10.1109/COMST.2018.2873950. 62

[355] S. Purkayastha. A rotationally symmetric directional distribution: obtained through maximum likelihood characterization. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 70–83, 1991. 127, 134

[356] A. Pyrgelis, C. Troncoso, and E. De Cristofaro. Knock Knock, Who's There? Membership Inference on Aggregate Location Data. *arXiv preprint arXiv:1708.06145*, 2017. 3, 122

[357] J. Qian, H. Du, J. Hou, L. Chen, T. Jung, X.-Y. Li, Y. Wang, and Y. Deng. VoiceMask: Anonymize and Sanitize Voice Input on Mobile Devices, Nov. 2017. URL http://arxiv.org/abs/1711.11460. 54, 64

[358] J. Qian, H. Du, J. Hou, L. Chen, T. Jung, and X.-Y. Li. Hidebehind: Enjoy Voice Input with Voiceprint Unclonability and Anonymity. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*, SenSys '18, pages 82–94, New York, NY, USA, Nov. 2018. Association for Computing Machinery. ISBN 978-1-4503-5952-8. doi: 10.1145/3274783.3274855. URL https://doi.org/10.1145/3274783.3274855. 64

[359] J. Qian, F. Han, J. Hou, C. Zhang, Y. Wang, and X.-Y. Li. Towards Privacy-Preserving Speech Data Publishing. In *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, pages 1079–1087, Apr. 2018. doi: 10.1109/INFOCOM.2018.8486250. 54

[360] C. Qu, W. Kong, L. Yang, M. Zhang, M. Bendersky, and M. Najork. Natural Language Understanding with Privacy-Preserving BERT. In *Proceedings of the 30th ACM*

International Conference on Information & Knowledge Management, pages 1488–1497, New York, NY, USA, Oct. 2021. Association for Computing Machinery. ISBN 978-1-4503-8446-9. URL https://doi.org/10.1145/3459637.3482281. 48

[361] J. C. Quiroz, L. Laranjo, A. B. Kocaballi, S. Berkovsky, D. Rezazadegan, and E. Coiera. Challenges of developing a digital scribe to reduce clinical documentation burden. *npj Digital Medicine*, 2(1):1–6, Nov. 2019. ISSN 2398-6352. doi: 10.1038/s41746-019-0190-1. URL https://www.nature.com/articles/s41746-019-0190-1. 38

[362] M. Raafat, B. Abdullah, M. Taher, and M. N. Moustafa. Towards Privacy-Preserving Driver's Drowsiness and Distraction Detection: A Differential Privacy Approach. *International Journal of Computing and Digital Systems*, 05(05), Sept. 2016. ISSN 2210-142X. doi: 10.12785/IJCDS/050501. URL https://journal.uob.edu.bh:443/handle/123456789/317. 65

[363] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language Models are Unsupervised Multitask Learners. 2019. 113, 172

[364] J. R. Rao and P. Rohatgi. Can Pseudonymity Really Guarantee Privacy? In *Proceedings of the 9th USENIX Security Symposium*, 2000. URL https://www.usenix.org/conference/9th-usenix-security-symposium/can-pseudonymity-really-guarantee-privacy. 3, 35, 51, 76, 100

[365] F. Y. Rashid. GitHub Search Makes Easy Discovery of Encryption Keys, Passwords In Source Code | SecurityWeek.Com, Jan. 2013. URL https://www.securityweek.com/github-search-makes-easy-discovery-encryption-keys-passwords-source-code. 35

[366] N. Raval, A. Machanavajjhala, and L. P. Cox. Protecting Visual Secrets Using Adversarial Nets. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1329–1332, Honolulu, HI, USA, July 2017. IEEE. ISBN 978-1-5386-0733-6. doi: 10.1109/CVPRW.2017.174. URL http://ieeexplore.ieee.org/document/8014908/. 60, 69

[367] S. Ravfogel, Y. Elazar, H. Gonen, M. Twiton, and Y. Goldberg. Null It Out: Guarding Protected Attributes by Iterative Nullspace Projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7237–7256, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.647. URL https://aclanthology.org/2020.acl-main.647. 45

[368] D. Reilly and L. Fan. A Comparative Evaluation of Differentially Private Image Obfuscation. In *2021 Third IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)*, pages 80–89, Dec. 2021. doi: 10.1109/TPSISA52974.2021.00009. 65

[369] N. Reimers and I. Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL https://arxiv.org/abs/1908.10084. 112, 172

[370] Z. Ren, Y. J. Lee, and M. S. Ryoo. Learning to Anonymize Faces for Privacy Preserving Action Detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 620–636, 2018. URL https://openaccess.thecvf.com/content_ECCV_2018/html/Zhongzheng_Ren_Learning_to_Anonymize_ECCV_2018_paper.html. 61

[371] P. Resnick and R. Zeckhauser. Trust among strangers in internet transactions: Empirical analysis of eBay' s reputation system. In *Advances in Applied Microeconomics*, volume 11, pages 127–157. Emerald (MCB UP), Bingley, 2002. ISBN 978-0-7623-0971-9. doi: 10.1016/S0278-0984(02)11030-3. URL https://www.emerald.com/insight/content/doi/10.1016/S0278-0984(02)11030-3/full/html. 3, 100

[372] D. Reynolds and R. Rose. Robust text-independent speaker identification using Gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing*, 3(1):72–83, Jan. 1995. ISSN 1558-2353. doi: 10.1109/89.365379. 39

[373] D. A. Reynolds. Speaker identification and verification using Gaussian mixture speaker models. *Speech Communication*, 17(1):91–108, Aug. 1995. ISSN 0167-6393. doi: 10.1016/0167-6393(95)00009-D. URL https://www.sciencedirect.com/science/article/pii/016763939500009D. 39

[374] D. A. Reynolds. An overview of automatic speaker recognition technology. In *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages IV–4072–IV–4075, May 2002. doi: 10.1109/ICASSP.2002.5745552. 39

[375] D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, volume 32 of *JMLR Workshop and Conference Proceedings*, pages 1278–1286. JMLR.org, 2014. URL http://proceedings.mlr.press/v32/rezende14.html. 100, 104

[376] D. Rhodes. Author attribution with cnns. Technical report, 2015. URL https://cs224d.stanford.edu/reports/RhodesDylan.pdf. Accessed on 2021-10-15. 3, 36, 100

[377] S. Ribaric and N. Pavesic. An overview of face de-identification in still images and videos. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 04, pages 1–6, May 2015. doi: 10.1109/FG.2015.7285017. 62

[378] S. Ribaric, A. Ariyaeeinia, and N. Pavesic. De-identification for privacy protection in multimedia content: A survey. *Signal Processing: Image Communication*, 47:131–151, Sept. 2016. ISSN 0923-5965. doi: 10.1016/j.image.2016.05.020. URL https://www.sciencedirect.com/science/article/pii/S0923596516300856. 62

[379] A. Rocha, W. J. Scheirer, C. W. Forstall, T. Cavalcante, A. Theophilo, B. Shen, A. R. B. Carvalho, and E. Stamatatos. Authorship Attribution for Social Media Forensics. *IEEE Transactions on Information Forensics and Security*, 12(1):5–33, Jan. 2017. ISSN 1556-6021. doi: 10.1109/TIFS.2016.2603960. 37

[380] M. Roguljić, I. Buljan, N. Veček, R. Dragun, M. Marušić, E. Wager, and A. Marušić. Deidentification of facial photographs: A survey of editorial policies and practices. *Journal of Medical Ethics*, 48(1):56–60, Jan. 2022. ISSN 0306-6800, 1473-4257. doi: 10.1136/medethics-2019-105823. URL https://jme.bmj.com/content/48/1/56. 57

[381] A. Romanov, A. Fedotova, A. Kurtukova, and R. Meshcheryakov. Natural Text Anonymization Using Universal Transformer with a Self-attention. In *Proceedings of the III International Conference on Language Engineering and Applied Linguistics (PRLEAL-2019)*, page 15, Saint Petersburg, Russia, 2019. 52

[382] A. Romanov, A. Rumshisky, A. Rogers, and D. Donahue. Adversarial decomposition of text representation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 815–825, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1088. URL https://aclanthology.org/N19-1088. 109

[383] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI*

2015, Lecture Notes in Computer Science, pages 234–241, Cham, 2015. Springer International Publishing. ISBN 978-3-319-24574-4. doi: 10.1007/978-3-319-24574-4_28. 61

[384] N. Rosenblum, X. Zhu, and B. Miller. Who wrote this code? identifying the authors of program binaries. In *European Symposium on Research in Computer Security*, pages 172–189. Springer, 2011. 37

[385] P. C. Roy and V. N. Boddeti. Mitigating Information Leakage in Image Representations: A Maximum Entropy Approach. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2581–2589, Long Beach, CA, USA, June 2019. IEEE. ISBN 978-1-72813-293-8. doi: 10.1109/CVPR.2019.00269. URL https://ieeexplore.ieee.org/document/8953999/. 46

[386] S. Ruder, P. Ghaffari, and J. G. Breslin. Character-level and Multi-channel Convolutional Neural Networks for Large-scale Authorship Attribution. *arXiv:1609.06686 [cs]*, Sept. 2016. URL http://arxiv.org/abs/1609.06686. 36

[387] A. Saha, T. Denning, V. Srikumar, and S. K. Kasera. Secrets in Source Code: Reducing False Positives using Machine Learning. In *2020 International Conference on COMmunication Systems & NETworkS (COMSNETS)*, pages 168–175, Jan. 2020. doi: 10.1109/COMSNETS48256.2020.9027350. 35

[388] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz. A bayesian approach to filtering junk e-mail. In *Learning for Text Categorization: Papers from the 1998 workshop*, volume 62, pages 98–105, 1998. 75

[389] A. Sala, X. Zhao, C. Wilson, H. Zheng, and B. Zhao. Sharing graphs using differentially private graph models. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pages 81–98. ACM, 2011. 96

[390] M. U. Saleem, D. Reilly, and L. Fan. DP-Shield: Face Obfuscation with Differential Privacy. In J. Stoyanovich, J. Teubner, P. Guagliardo, M. Nikolic, A. Pieris, J. Mühlig, F. Özcan, S. Schelter, H. V. Jagadish, and M. Zhang, editors, *Proceedings of the 25th International Conference on Extending Database Technology, EDBT 2022, Edinburgh, UK, March 29 - April 1, 2022*, pages 2:578–2:581. OpenProceedings.org, 2022. doi: 10.48786/edbt.2022.55. URL https://doi.org/10.48786/edbt.2022.55. 65

[391] G. Salton, A. Wong, and C. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975. 75, 76

[392] P. Samarati and L. Sweeney. Protecting Privacy when Disclosing Information: K-Anonymity and Its Enforcement through Generalization and Suppression. Technical report, 1998. URL http://dataprivacylab.org/dataprivacy/projects/kanonymity/paper3.pdf. 59

[393] D. Sánchez, M. Batet, and A. Viejo. Detecting Sensitive Information from Textual Documents: An Information-Theoretic Approach. In D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, V. Torra, Y. Narukawa, B. López, and M. Villaret, editors, *Modeling Decisions for Artificial Intelligence*, volume 7647, pages 173–184. Springer, Berlin, Heidelberg, 2012. ISBN 978-3-642-34619-4 978-3-642-34620-0. doi: 10.1007/978-3-642-34620-0_17. URL http://link.springer.com/10.1007/978-3-642-34620-0_17. 34

[394] C. Sanderson and S. Guenter. Short Text Authorship Attribution via Sequence Kernels, Markov Chains and Author Unmasking: An Investigation. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 482–491, Sydney, Australia, July 2006. Association for Computational Linguistics. URL https://aclanthology.org/W06-1657. 36

[395] SAP Security Research. Credential Digger. SAP, 2020. URL https://github.com/SAP/credential-digger. 42

[396] J. G. Saw. A family of distributions on the m-sphere and some hypothesis tests. *Biometrika*, 65(1):69–73, 1978. 132

[397] J. Schatz. Depix-HMM, Feb. 2023. URL https://github.com/JonasSchatz/DepixHMM. 58

[398] A. Schaub, R. Bazin, O. Hasan, and L. Brunie. A Trustless Privacy-Preserving Reputation System. In J.-H. Hoepman and S. Katzenbeisser, editors, *ICT Systems Security and Privacy Protection*, IFIP Advances in Information and Communication Technology, pages 398–411, Cham, 2016. Springer International Publishing. ISBN 978-3-319-33630-5. doi: 10.1007/978-3-319-33630-5_27. 3, 100

[399] B. Schipper. Depix, Aug. 2022. URL https://github.com/beurtschipper/Depix. 58

[400] A. Schlapbach and H. Bunke. Using HMM Based Recognizers for Writer Identification and Verification. In *Ninth International Workshop on Frontiers in Handwriting Recognition*,

pages 167–172, Tokyo, Japan, 2004. IEEE. ISBN 978-0-7695-2187-9. doi: 10.1109/IWFHR.2004.107. URL http://ieeexplore.ieee.org/document/1363905/. 41

[401] A. Schlapbach and H. Bunke. Off-line Writer Identification Using Gaussian Mixture Models. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 3, pages 992–995, Aug. 2006. doi: 10.1109/ICPR.2006.894. 41

[402] H. Schütze, C. Manning, and P. Raghavan. *Introduction to Information Retrieval*, volume 39. Cambridge University Press, 2008. 77

[403] R. Schwartz, O. Tsur, A. Rappoport, and M. Koppel. Authorship Attribution of Micro-Messages. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1880–1891, Seattle, Washington, USA, Oct. 2013. Association for Computational Linguistics. URL https://aclanthology.org/D13-1193. 36

[404] A. Shabtai, Y. Elovici, and L. Rokach. *A Survey of Data Leakage Detection and Prevention Solutions*. SpringerBriefs in Computer Science. Springer US, Boston, MA, 2012. ISBN 978-1-4614-2052-1 978-1-4614-2053-8. doi: 10.1007/978-1-4614-2053-8. URL http://link.springer.com/10.1007/978-1-4614-2053-8. 43

[405] A. S. Shamsabadi, B. M. L. Srivastava, A. Bellet, N. Vauquier, E. Vincent, M. Maouche, M. Tommasi, and N. Papernot. Differentially Private Speaker Anonymization. *Proceedings on Privacy Enhancing Technologies*, 2023(1):98–114, Jan. 2023. ISSN 2299-0984. doi: 10.56553/popets-2023-0007. URL https://petsymposium.org/popets/2023/popets-2023-0007.php. 64, 69, 171

[406] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter. Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 1528–1540, Vienna Austria, Oct. 2016. ACM. ISBN 978-1-4503-4139-4. doi: 10.1145/2976749.2978392. URL https://dl.acm.org/doi/10.1145/2976749.2978392. 60

[407] J. Shashirangana, H. Padmasiri, D. Meedeniya, and C. Perera. Automated License Plate Recognition: A Survey on Methods and Techniques. *IEEE Access*, 9:11203–11225, 2021. ISSN 2169-3536. doi: 10.1109/ACCESS.2020.3047929. 40

[408] R. Shetty, B. Schiele, and M. Fritz. A4NT: Author attribute anonymity by adversarial training of neural machine translation. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 1633–1650, Baltimore, MD, Aug. 2018. USENIX Association. ISBN 978-1-939133-04-5. URL https://www.usenix.org/conference/usenixsecurity18/presentation/shetty. 52

[409] P. Shrestha, S. Sierra, F. A. González, M. Montes, P. Rosso, and T. Solorio. Convolutional neural networks for authorship attribution of short texts. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 669–674, 2017. 3, 36, 100

[410] T. Sibanda, O. Uzuner, and O. Uzuner. Role of Local Context in Automatic Deidentification of Ungrammatical, Fragmented Text. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 65–73, New York City, USA, June 2006. Association for Computational Linguistics. URL https://aclanthology.org/N06-1009. 51

[411] M. R. Silas, P. Grassia, and A. Langerman. Video recording of the operating room—is anonymity possible? *Journal of Surgical Research*, 197(2):272–276, Aug. 2015. ISSN 0022-4804. doi: 10.1016/j.jss.2015.03.097. URL https://www.sciencedirect.com/science/article/pii/S0022480415003625. 39, 61

[412] V. S. Sinha, D. Saha, P. Dhoolia, R. Padhye, and S. Mani. Detecting and Mitigating Secret-Key Leaks in Source Code Repositories. In *2015 IEEE/ACM 12th Working Conference on Mining Software Repositories*, pages 396–400, May 2015. doi: 10.1109/MSR.2015.48. 42

[413] B. Slaughter. *Handwritten Authorship Attribution Using Both Image Recognition and Natural Language Processing Techniques*. PhD thesis, Rose-Hulman Institute of Technology, Mar. 2021. 41

[414] A. F. Smith and V. J. Fortunato. Factors influencing employee intentions to provide honest upward feedback ratings. *Journal of Business and Psychology*, 22(3):191–207, 2008. 3, 100

[415] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur. Deep Neural Network Embeddings for Text-Independent Speaker Verification. In *Interspeech 2017*, pages 999–1003. ISCA, Aug. 2017. doi: 10.21437/Interspeech.2017-620. URL https://www.isca-speech.org/archive/interspeech_2017/snyder17_interspeech.html. 39

[416] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur. X-Vectors: Robust DNN Embeddings for Speaker Recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5329–5333, Calgary, AB, Apr. 2018. ISBN 978-1-5386-4658-8. doi: 10.1109/ICASSP.2018.8461375. URL https://ieeexplore.ieee.org/document/8461375/. 39, 54, 55, 64, 69, 116, 171

[417] S. Song, K. Chaudhuri, and A. D. Sarwate. Stochastic gradient descent with differentially private updates. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 245–248, 2013. doi: 10.1109/GlobalSIP.2013.6736861. 18, 119

[418] B. M. L. Srivastava, A. Bellet, M. Tommasi, and E. Vincent. Privacy-Preserving Adversarial Representation Learning in ASR: Reality or Illusion? In *Interspeech 2019*, pages 3700–3704, Sept. 2019. doi: 10.21437/Interspeech.2019-2415. URL http://arxiv.org/abs/1911.04913. 45

[419] B. M. L. Srivastava, N. Tomashenko, X. Wang, E. Vincent, J. Yamagishi, M. Maouche, A. Bellet, and M. Tommasi. Design Choices for X-vector Based Speaker Anonymization, May 2020. URL http://arxiv.org/abs/2005.08601. 56, 69

[420] B. M. L. Srivastava, N. Vauquier, M. Sahidullah, A. Bellet, M. Tommasi, and E. Vincent. Evaluating Voice Conversion-Based Privacy Protection against Informed Attackers. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2802–2806, May 2020. doi: 10.1109/ICASSP40776.2020.9053868. 55, 69

[421] B. M. L. Srivastava, M. Maouche, M. Sahidullah, E. Vincent, A. Bellet, M. Tommasi, N. Tomashenko, X. Wang, and J. Yamagishi. Privacy and Utility of X-Vector Based Speaker Anonymization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:2383–2395, 2022. ISSN 2329-9304. doi: 10.1109/TASLP.2022.3190741. 56, 69

[422] E. Stamatatos. A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology*, 60(3):538–556, 2009. URL http://onlinelibrary.wiley.com/doi/10.1002/asi.21001/full. 3, 36, 100

[423] B. Stein, M. Koppel, and E. Stamatatos, editors. *1st Workshop on Plagiarism Analysis, Authorship Identification, and Near-Duplicate Detection (PAN 2007) at SIGIR*, volume 276 of *CEUR Workshop Proceedings*. July 2007. URL http://ceur-ws.org/Vol-276. 36

[424] B. Stein, M. Koppel, and E. Stamatatos. Plagiarism analysis, authorship identification, and near-duplicate detection PAN'07. *ACM SIGIR Forum*, 41(2):68–71, Dec. 2007. ISSN 0163-5840. doi: 10.1145/1328964.1328976. URL https://dl.acm.org/doi/10.1145/1328964.1328976. 36

[425] N. Subramani, A. Matton, M. Greaves, and A. Lam. A Survey of Deep Learning

Approaches for OCR and Document Understanding, Feb. 2021. URL http://arxiv.org/abs/2011.13534. 39

[426] O. Sudana, I. W. Gunaya, and I. K. G. D. Putra. Handwriting identification using deep convolutional neural network method. *Telkomnika (Telecommunication Computing Electronics and Control)*, 18(4):1934–1941, 2020. 41

[427] Q. Sun, L. Ma, S. J. Oh, L. Van Gool, B. Schiele, and M. Fritz. Natural and Effective Obfuscation by Head Inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 61

[428] S. Swain, G. Mishra, and C. Sindhu. Recent approaches on authorship attribution techniques — An overview. In *2017 International Conference of Electronics, Communication and Aerospace Technology (ICECA)*, volume 1, pages 557–566, Apr. 2017. doi: 10.1109/ICECA.2017.8203599. 36

[429] L. Sweeney. Replacing personally-identifying information in medical records, the Scrub system. *Proceedings of the AMIA Annual Fall Symposium*, pages 333–337, 1996. ISSN 1091-8280. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2233179/. 50

[430] L. Sweeney. Simple demographics often identify people uniquely. *Health (San Francisco)*, 671:1–34, 2000. 6, 76

[431] L. Sweeney. K-ANONYMITY: A MODEL FOR PROTECTING PRIVACY. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05): 557–570, Oct. 2002. ISSN 0218-4885. doi: 10.1142/S0218488502001648. URL https://www.worldscientific.com/doi/abs/10.1142/S0218488502001648. 59

[432] G. Szarvas, R. Farkas, and R. Busa-Fekete. State-of-the-art Anonymization of Medical Records Using an Iterative Machine Learning Framework. *Journal of the American Medical Informatics Association*, 14(5):574–580, 2007. ISSN 1067-5027. doi: 10.1197/jamia.M2441. URL https://www.sciencedirect.com/science/article/pii/S1067502707001776. 51

[433] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks, Feb. 2014. URL http://arxiv.org/abs/1312.6199. 60, 61

[434] P. Thaine. Private AI Whitepaper. Technical report. URL https://www.private-ai.com/wp-content/uploads/2021/10/Private_AI_Technical_Whitepaper.pdf. 43

[435] The 104th United States Congress. Health Insurance Portability and Accountability Act, 1996. URL https://www.govinfo.gov/link/plaw/104/public/191?link-type=pdf&.pdf. 3, 4, 50

[436] A. Theophilo, R. Giot, and A. Rocha. Authorship Attribution of Social Media Messages. *IEEE Transactions on Computational Social Systems*, pages 1–14, 2021. ISSN 2329-924X. doi: 10.1109/TCSS.2021.3123895. 36

[437] S. A. Thompson and C. Warzel. Opinion | twelve million phones, one dataset, zero privacy. 2019. URL https://www.nytimes.com/interactive/2019/12/19/opinion/location-tracking-cell-phone.html. 3, 122

[438] M. Tölle, U. Köthe, F. André, B. Meder, and S. Engelhardt. Content-Aware Differential Privacy with Conditional Invertible Neural Networks. In S. Albarqouni, S. Bakas, S. Bano, M. J. Cardoso, B. Khanal, B. Landman, X. Li, C. Qin, I. Rekik, N. Rieke, H. Roth, D. Sheet, and D. Xu, editors, *Distributed, Collaborative, and Federated Learning, and Affordable AI and Healthcare for Resource Diverse Global Health*, Lecture Notes in Computer Science, pages 89–99, Cham, 2022. Springer Nature Switzerland. ISBN 978-3-031-18523-6. doi: 10.1007/978-3-031-18523-6_9. 66

[439] N. Tomashenko, B. M. L. Srivastava, X. Wang, E. Vincent, A. Nautsch, J. Yamagishi, N. Evans, J. Patino, J.-F. Bonastre, P.-G. Noé, and M. Todisco. Introducing the VoicePrivacy Initiative. In *Interspeech 2020*, pages 1693–1697, Oct. 2020. doi: 10.21437/Interspeech.2020-1333. URL http://arxiv.org/abs/2005.01387. 54, 55

[440] Truffle Security. TruffleHog. URL https://trufflesecurity.com/trufflehog. 42

[441] Truffle Security. TruffleHog. Truffle Security, 2016. URL https://github.com/trufflesecurity/trufflehog. 42

[442] H. Turner, G. Lovisotto, and I. Martinovic. Speaker Anonymization with Distribution-Preserving X-Vector Generation for the VoicePrivacy Challenge 2020, Jan. 2021. URL http://arxiv.org/abs/2010.13457. 55

[443] Twitter Inc. How to add your location to a Tweet. URL https://help.twitter.com/en/using-twitter/tweet-location. 2

[444] J. Tyo, B. Dhingra, and Z. C. Lipton. On the State of the Art in Authorship Attribution and Authorship Verification, Oct. 2022. URL http://arxiv.org/abs/2209.06869. 36

[445] F. Ullah, M. Edwards, R. Ramdhany, R. Chitchyan, M. A. Babar, and A. Rashid. Data exfiltration: A review of external attack vectors and countermeasures. *Journal of Network and Computer Applications*, 101:18–54, Jan. 2018. ISSN 1084-8045. doi: 10.1016/j.jnca.2017.10.016. URL https://www.sciencedirect.com/science/article/pii/S1084804517303569. 42

[446] G. Ulrich. Computer generation of distributions on the m-sphere. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 33(2):158–163, 1984. 125, 127, 132, 138, 146, 164

[447] U.S. Dept. of Labor, Employee Benefits Security Administration. The Health Insurance Portability and Accountability Act of 1996 (HIPAA), 1996. URL http://www.hhs.gov/hipaa/. 4, 50, 62

[448] Ö. Uzuner, Y. Luo, and P. Szolovits. Evaluating the State-of-the-Art in Automatic De-identification. *Journal of the American Medical Informatics Association*, 14(5):550–563, Sept. 2007. ISSN 1067-5027. doi: 10.1197/jamia.M2444. URL https://doi.org/10.1197/jamia.M2444. 96

[449] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez. Deep neural networks for small footprint text-dependent speaker verification. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4052–4056, Florence, Italy, May 2014. IEEE. ISBN 978-1-4799-2893-4. doi: 10.1109/ICASSP.2014.6854363. URL http://ieeexplore.ieee.org/document/6854363/. 38, 55

[450] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html. 41, 172

[451] G. Venayagamoorthy, V. Moonasar, and K. Sandrasegaran. Voice recognition using neural networks. In *Proceedings of the 1998 South African Symposium on Communications and Signal Processing-COMSIG '98 (Cat. No. 98EX214)*, pages 29–32, Sept. 1998. doi: 10.1109/COMSIG.1998.736916. 39

[452] W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance*, 16(1):8–37, 1961. ISSN 00221082, 15406261. URL http://www.jstor.org/stable/2977633. 64

[453] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature Methods*, pages 1–12, 2020. 146

[454] R. von Mises. Über die "Ganzzahligkeit" der Atomgewicht und verwandte Fragen. *Physikalische Zeitschrift*, 19:490–500, 1918. 127

[455] A. Wali, Z. Alamgir, S. Karim, A. Fawaz, M. B. Ali, M. Adan, and M. Mujtaba. Generative adversarial networks for speech processing: A review. *Computer Speech & Language*, 72:101308, Mar. 2022. ISSN 0885-2308. doi: 10.1016/j.csl.2021.101308. URL https://www.sciencedirect.com/science/article/pii/S0885230821001066. 45

[456] S. Walt, S. Colbert, and G. Varoquaux. The numpy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30, 2011. 87, 146

[457] H. Wang, S. Xie, and Y. Hong. VideoDP: A Flexible Platform for Video Analytics with Differential Privacy. *Proceedings on Privacy Enhancing Technologies*, 2020(4): 277–296, Oct. 2020. ISSN 2299-0984. doi: 10.2478/popets-2020-0073. URL https://petsymposium.org/popets/2020/popets-2020-0073.php. 61

[458] J. Wang, B. Amos, A. Das, P. Pillai, N. Sadeh, and M. Satyanarayanan. Enabling Live Video Analytics with a Scalable and Privacy-Aware Framework. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 14(3s):64:1–64:24, June 2018. ISSN 1551-6857. doi: 10.1145/3209659. URL https://doi.org/10.1145/3209659. 61

[459] M. Wang and W. Deng. Deep face recognition: A survey. *Neurocomputing*, 429: 215–244, Mar. 2021. ISSN 0925-2312. doi: 10.1016/j.neucom.2020.10.081. URL https://www.sciencedirect.com/science/article/pii/S0925231220316945. 42

[460] S. Wang and S. Jia. Signature handwriting identification based on generative adversarial networks. In *Journal of Physics: Conference Series*, volume 1187, page 042047. IOP Publishing, 2019. 41

[461] T. Wang, X. Zhang, J. Feng, and X. Yang. A Comprehensive Survey on Local Differential Privacy toward Data Statistics and Analysis. *Sensors*, 20(24):7030, Jan. 2020. ISSN 1424-8220. doi: 10.3390/s20247030. URL https://www.mdpi.com/1424-8220/20/24/7030. 171

[462] Y. Wang, X. Wu, and L. Wu. Differential privacy preserving spectral graph analysis. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 329–340. Springer, 2013. 163

[463] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, Apr. 2004. ISSN 1941-0042. doi: 10.1109/TIP.2003.819861. 65

[464] S. L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309):63–69, 1965. 19, 162

[465] B. Weggenmann and F. Kerschbaum. SynTF: Synthetic and Differentially Private Term Frequency Vectors for Privacy-Preserving Text Mining. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, pages 305–314, New York, NY, USA, June 2018. Association for Computing Machinery. ISBN 978-1-4503-5657-2. doi: 10.1145/3209978.3210008. URL https://doi.org/10.1145/3209978.3210008. 11, 47, 62, 75, 100, 104

[466] B. Weggenmann and F. Kerschbaum. SynTF: Synthetic and Differentially Private Term Frequency Vectors for Privacy-Preserving Text Mining, May 2018. URL http://arxiv.org/abs/1805.00904. 11, 47, 62, 75, 100

[467] B. Weggenmann and F. Kerschbaum. Differential Privacy for Directional Data. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, CCS '21, pages 1205–1222, New York, NY, USA, Nov. 2021. Association for Computing Machinery. ISBN 978-1-4503-8454-4. doi: 10.1145/3460120.3484734. URL https://doi.org/10.1145/3460120.3484734. 11, 67, 121

[468] B. Weggenmann, V. Rublack, M. Andrejczuk, J. Mattern, and F. Kerschbaum. DP-VAE: Human-Readable Text Anonymization for Online Reviews with Differentially Private Variational Autoencoders. In *Proceedings of the ACM Web Conference 2022*, WWW '22, pages 721–731, New York, NY, USA, Apr. 2022. Association for Computing Machinery. ISBN 978-1-4503-9096-5. doi: 10.1145/3485447.3512232. URL https://doi.org/10.1145/3485447.3512232. 11, 99

[469] Y. Wen, B. Liu, M. Ding, R. Xie, and L. Song. IdentityDP: Differential Private Identification Protection for Face Images. *Neurocomputing*, 501:197–211, Aug. 2022. ISSN 0925-2312. doi: 10.1016/j.neucom.2022.06.039. URL https://www.sciencedirect.com/science/article/pii/S0925231222007597. 66, 69, 116, 171

[470] E. Wenger, S. Shan, H. Zheng, and B. Y. Zhao. SoK: Anti-Facial Recognition Technology, Dec. 2021. URL http://arxiv.org/abs/2112.04558. 61

[471] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush. Transformers: State-of-the-Art Natural Language Processing. pages 38–45. Association for Computational Linguistics, Oct. 2020. URL https://www.aclweb.org/anthology/2020.emnlp-demos.6. 51

[472] A. Wood, M. Altman, K. Nissim, and S. Vadhan. Designing Access with Differential Privacy. In S. Cole, I. Dhaliwal, A. Sautmann, and L. Vilhuber, editors, *Handbook on Using Administrative Data for Research and Evidence-based Policy*, chapter 6. 2021. URL https://admindatahandbook.mit.edu/book/v1.0/diffpriv.html. 19

[473] A. T. Wood. Simulation of the von mises fisher distribution. *Communications in Statistics - Simulation and Computation*, 23(1):157–164, 1994. doi: 10.1080/03610919408813161. URL https://doi.org/10.1080/03610919408813161. 127, 132, 138, 146, 164

[474] Y. Wu, F. Yang, Y. Xu, and H. Ling. Privacy-Protective-GAN for Privacy Preserving Face De-Identification. *Journal of Computer Science and Technology*, 34(1):47–60, Jan. 2019. ISSN 1860-4749. doi: 10.1007/s11390-019-1898-8. URL https://doi.org/10.1007/s11390-019-1898-8. 61

[475] T. Xiao, Y.-H. Tsai, K. Sohn, M. Chandraker, and M.-H. Yang. Adversarial Learning of Privacy-Preserving and Task-Oriented Representations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(07):12434–12441, Apr. 2020. ISSN 2374-3468, 2159-5399. doi: 10.1609/aaai.v34i07.6930. URL https://aaai.org/ojs/index.php/AAAI/article/view/6930. 46

[476] L. Xing and Y. Qiao. DeepWriter: A Multi-stream Deep CNN for Text-Independent Writer Identification. In *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 584–589, Oct. 2016. doi: 10.1109/ICFHR.2016.0112. 41

[477] G. Xu, C. Qi, H. Yu, S. Xu, C. Zhao, and J. Yuan. Detecting Sensitive Information of Unstructured Text Using Convolutional Neural Network. In *2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*, pages 474–479, Oct. 2019. doi: 10.1109/CyberC.2019.00087. 34

[478] J. Xu, Z. Zhang, X. Xiao, Y. Yang, G. Yu, and M. Winslett. Differentially private histogram publication. *The VLDB Journal*, 22(6):797–822, 2013. 79

[479] K. Xu, T. Cao, S. Shah, C. Maung, and H. Schweitzer. Cleaning the Null Space: A Privacy Mechanism for Predictors. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1), Feb. 2017. ISSN 2374-3468, 2159-5399. doi: 10.1609/aaai.v31i1.10935. URL https://ojs.aaai.org/index.php/AAAI/article/view/10935. 47

[480] Z. Xu, A. Aggarwal, O. Feyisetan, and N. Teissier. A Differentially Private Text Perturbation Method Using Regularized Mahalanobis Metric. In *Proceedings of the Second Workshop on Privacy in NLP*, pages 7–17, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.privatenlp-1.2. URL https://aclanthology.org/2020.privatenlp-1.2. 48, 171

[481] Z. Xu, A. Aggarwal, O. Feyisetan, and N. Teissier. On a Utilitarian Approach to Privacy Preserving Text Generation. In *Proceedings of the Third Workshop on Privacy in Natural Language Processing*, pages 11–20, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.privatenlp-1.2. URL https://aclanthology.org/2021.privatenlp-1.2. 64

[482] H. Yadav, S. Ghosh, Y. Yu, and R. R. Shah. End-to-End Named Entity Recognition from English Speech. In *Interspeech 2020*, pages 4268–4272. ISCA, Oct. 2020. doi: 10.21437/Interspeech.2020-2482. URL https://www.isca-speech.org/archive/interspeech_2020/yadav20b_interspeech.html. 38

[483] J. Yang, W. Zhang, J. Liu, J. Wu, and J. Yang. Generating De-identification facial images based on the attention models and adversarial examples. *Alexandria Engineering Journal*, 61(11):8417–8429, Nov. 2022. ISSN 1110-0168. doi: 10.1016/j.aej.2022.02.007. URL https://www.sciencedirect.com/science/article/pii/S1110016822000953. 60

[484] K. Yang, J. H. Yau, L. Fei-Fei, J. Deng, and O. Russakovsky. A Study of Face Obfuscation in ImageNet. In *Proceedings of the 39th International Conference on Machine Learning*, pages 25313–25330. PMLR, June 2022. URL https://proceedings.mlr.press/v162/yang22q.html. 58

[485] M. Yang, L. Lyu, J. Zhao, T. Zhu, and K.-Y. Lam. Local Differential Privacy and Its Applications: A Comprehensive Survey, Aug. 2020. URL http://arxiv.org/abs/2008.03686. 19, 171, 172

[486] W. Yang, L. Jin, and M. Liu. DeepWriterID: An End-to-End Online Text-Independent Writer Identification System. *IEEE Intelligent Systems*, 31(2):45–53, Mar. 2016. ISSN 1941-1294. doi: 10.1109/MIS.2016.22. 41

[487] X. Yue, M. Du, T. Wang, Y. Li, H. Sun, and S. S. M. Chow. Differential Privacy for Text Analytics via Natural Text Sanitization. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3853–3866, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.337. URL https://aclanthology.org/2021.findings-acl.337. 63, 69, 171

[488] S. Zafeiriou, C. Zhang, and Z. Zhang. A survey on face detection in the wild: Past, present and future. *Computer Vision and Image Understanding*, 138:1–24, Sept. 2015. ISSN 1077-3142. doi: 10.1016/j.cviu.2015.03.015. URL https://www.sciencedirect.com/science/article/pii/S1077314215000727. 41

[489] B. Zhang and S. N. Srihari. Binary vector dissimilarity measures for handwriting identification. In *Document Recognition and Retrieval X*, volume 5010, pages 28–38. SPIE, 2003. 41

[490] C. Zhang and Z. Zhang. A Survey of Recent Advances in Face Detection. June 2010. URL https://www.microsoft.com/en-us/research/publication/a-survey-of-recent-advances-in-face-detection/. 41

[491] P. Zhang. RSTC: A New Residual Swin Transformer for Offline Word-Level Writer Identification. *IEEE Access*, 10:57452–57460, 2022. ISSN 2169-3536. doi: 10.1109/ACCESS.2022.3178597. 41

[492] X.-Y. Zhang, G.-S. Xie, C.-L. Liu, and Y. Bengio. End-to-End Online Writer Identification With Recurrent Neural Network. *IEEE Transactions on Human-Machine Systems*, 47(2):285–292, Apr. 2017. ISSN 2168-2305. doi: 10.1109/THMS.2016.2634921. 41

[493] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 35(4):399–458, Dec. 2003. ISSN 0360-0300. doi: 10.1145/954339.954342. URL https://doi.org/10.1145/954339.954342. 42

[494] Y. Zhao and J. Chen. A Survey on Differential Privacy for Unstructured Data Content. *ACM Computing Surveys*, 54(10s):207:1–207:28, Sept. 2022. ISSN 0360-0300. doi: 10.1145/3490237. URL https://doi.org/10.1145/3490237. 9, 167

[495] Y.-C. Zheng and B.-Z. Yuan. Text-dependent speaker identification using circular hidden Markov models. In *ICASSP-88., International Conference on Acoustics, Speech, and Signal Processing*, pages 580,581,582–580,581,582. IEEE Computer Society, Jan. 1988. doi: 10.1109/ICASSP.1988.196651. URL https://www.computer.org/csdl/proceedings-article/icassp/1988/00196651/12OmNB8CiWg. 39

[496] B. Zhu, H. Fang, Y. Sui, and L. Li. Deepfakes for Medical Video De-Identification: Privacy Protection and Diagnostic Information Preservation. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 414–420. Association for Computing Machinery, New York, NY, USA, Feb. 2020. ISBN 978-1-4503-7110-0. URL https://doi.org/10.1145/3375627.3375849. 61

[497] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired Image-To-Image Translation Using Cycle-Consistent Adversarial Networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2223–2232, 2017. URL https://openaccess.thecvf.com/content_iccv_2017/html/Zhu_Unpaired_Image-To-Image_Translation_ICCV_2017_paper.html. 54