UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Big Data Privacy by Design Computation Platform

Rui Nuno Lopes Claro

Supervisor: Doctor Miguel Filipe Leitão Pardal
Co-Supervisor: Doctor José Miguel Ladeira Portêlo

Thesis to obtain the Masters Degree in
Information Systems and Computer Engineering

2018

Resumo

Abstract

We live in the age of Big Data. Personal user data, in particular, is necessary for the operation and improvement of everyday Internet services like Google, Facebook, WhatsApp, Spotify, etc. Many times, the capture and use of personal data is not made explicit to the users, but it is central to the business model of companies. However, each person's right to privacy has to be respected. How can these two conflicting needs be reconciled, *i.e.*, how can we build useful Big Data systems that are respectful of user privacy? The goal of this work is to design and implement a proof-of-concept of a platform for performing privacy preserving computations, providing an easy-to-use method to implement privacy-preserving techniques. This system could be used, for example, to monitor the vital signs of patients (without exposing them to other people), to produce real time recommendations based on location (without disclosing location to others), sports/fitness applications, etc. This proof-of-concept will implement privacy-preserving versions of Machine Learning algorithms and compare them against a baseline reference.

so the trade-offs can be quantified and better understood.

Palavras-Chave
Keywords

Palavras-Chave

Preservação de privacidade em Computações
Aprendizagem automática
Extracção de Informação
Big Data
Processamento de Dados
Computação Multi-Entidade Segura

Keywords

Privacy-preserving Computations
Machine Learning
Data Mining
Big Data
Data Processing
Secure Multi-Party Computation

Acknowledgements

Contents

List of Figures

List of Tables

List of Acronyms

PPDM Privacy-Preserving Data Mining

PII Personal Identifiable Information

SMPC Secure Multi-Party Computation

STPC Secure Two-Party Computation

OT Oblivious Transfer

GC Garbled Circuits

DP Differential Privacy

HE Homomorphic Encryption

PHE Partial Homomorphic Encryption

FHE Fully Homomorphic Encryption

FE Functional Encryption

ENISA European Union Agency for Network and Information Security

EU European Union

GDPR General Data Protection Regulation

k-NN k-Nearest Neighbors

SVM Support Vector Machine

CMU Carnegie Mellon University

ML Machine Learning

BARD Big dAta pRivacy by Design platform

1 Introduction

1.1 Motivation

With the so called "Big Data revolution", vast amounts of data are now being analyzed and processed by companies. In this way, meaningful information can be obtained to improve existing systems or to discover new approaches in business models. It is common practice in most of these companies to deploy *Data Mining* algorithms to better understand their customers, and to devise better recommendation systems, in order to surpass their competitors in customer satisfaction. These practices are not only limited to customer satisfaction. In healthcare, for example, it can be beneficial to match patient records from different hospitals in order to identify inefficiencies and develop best practices [25].

Most times data contains private information about individuals, such as health records or daily routines. This kind of data cannot be freely processed because that goes against privacy laws and could lead to breaches of private information. Despite the value of Data Mining results, people are showing an increasing concern relatively to the privacy threats posed by Data Mining [5]. The privacy of an individual may be violated due to, for example, unauthorized access to personal data, or the use of personal data for purposes other than the one for which data was collected.

To deal with the privacy issues in Data Mining, a sub-field known as Privacy-Preserving Data Mining (PPDM) has been gaining influence over the last years [7]. The objective of PPDM is to guarantee the privacy of sensitive information, while at the same time preserve the utility of the data for Data Mining purposes [1]. This can be achieved by using one or more privacy-preserving techniques, such as Secure Multi-Party Computation (SMPC) [7] or Differential Privacy (DP) [8].

Machine Learning algorithms in the context of Big Data processing are also producing significant results, so that it is possible to do knowledge learning from datasets in order to predict future labels or clusters for new data. An example of an application of Machine Learning algorithms in Data Mining is *Classification* [23], in which a training set is processed in order to create a classifier for data, and then that classifier is used to predict class labels for new data. Some examples of Machine Learning algorithms include Decision Tree, k-Nearest Neighbors (k-NN), and Support Vector Machine (SVM) [23].

By combining Machine Learning algorithms and privacy-preserving techniques, it is possible to create Data Mining processes that, not only allow for knowledge learning on large datasets but also to maintain a level of privacy that is desirable by individuals and that complies with the laws in force.

1.2 Contributions

1.3 Structure of this Document

The remainder of this thesis is structured as follows.

2 Related Work

This chapter provides and overview on the privacy-preserving Machine Learning paradigm. We start by explaining Data Security and Data Privacy, in sections 2.1 and 2.2 respectively. We detail each concept so that it gives an understanding of the differences between them. Section 2.3 presents the concepts of data processing and Data Mining, gives an overview of the CRISP-DM model, and defines the attack models that can be assumed when developing a Privacy-Preserving Data Mining (PPDM) algorithm. For developing a PPDM algorithm, one can implement one or more of the privacy-preserving techniques briefly explained in section 2.4. We join the previous concepts with Machine Learning in section 2.5, presenting a brief description and an example of Machine Learning applied in Data Mining. Finally, we discuss known use cases that are of interest in the field in section 2.6.

2.1 Data Security

Data Security refers to protective digital measures that are applied to prevent unauthorized access to computers, databases, and websites, as well as to prevent destruction or alteration of data.

2.1.1 Data Protection Goals

We define here the core protection goals widely accepted in the literature, often known as the CIA triad (*Confidentiality*, *Integrity*, and *Availability*) [19].

- *Confidentiality* is defined as the property that data, and services that process such data, cannot be accessed by unauthorized entities.

- *Integrity* is defined as the property that data, and services that process such data, cannot be modified in an unauthorized or undetected manner.

- *Availability* is defined as the property that access to data, and services that process such data, is always granted in a comprehensible, processable, and timely manner.

For applying Data Security measures, various technologies can be implemented, such as:

- *Data backups* ensure that data that has been lost can be recovered. This technique is standard procedure for most companies since the permanent loss of crucial data can seriously cripple a company.

- *Data erasure*, in contrast to backups, is a technique to permanently delete data from a hard drive or other digital media, to ensure that no sensitive data is leaked when a company wants to permanently remove an asset from usage.

- *Data encryption*, or disk encryption, refers to techniques that allow a user to encrypt data in a disk or part of it, such that it remains protected and cannot be decrypted easily by an unauthorized party.

- *Identity-based security*, as the name states, is a method to limit the access to data such that only a user that has been authenticated and has permission to access a piece of data can do so.

These techniques offer ways to protect data, but sometimes this is not enough. Either due to incorrect programming or due to the existence of bugs in the system, vulnerabilities occur in the software that allows unauthorized parties to bypass these mechanisms and get access to data that should be confidential.

2.1.2 Examples of Data Security Breaches

Data Security breaches refer to attacks, usually by unauthorized access, to systems that contain private data. These attacks are commonly made by organized hacker groups to gain leverage against companies or to make a profit by selling the data in black markets online. Next, we present some examples of recent Data Security breaches.

- Sony Pictures hack[1]. In 2014, a hacker group leaked confidential data from Sony Pictures in an attempt to gain leverage with the company to make it comply to their demands. The hacker group threatened to commit acts of terrorism in theaters if Sony released a movie related to the North Korean leader.

- Yahoo! data breach[2]. In 2016, Yahoo! reported two separate data breaches occurring in 2014 and 2013, of over 1.5 billion user accounts.

- Ashley Madison data breach[3]. In 2015, a group of hackers stole user data from the website Ashley Madison, and threatened to release usernames and personally identifying information if the website was not shut down.

2.2 Data Privacy

Privacy is an important field in information security because it gives a person his/her personal space and defines his/her personal private information, giving the person the right to decide which personal information is for sharing and which should be kept confidential. Privacy also limits the access that other entities, being them the government or private companies, have to personal data.

[1] https://www.washingtonpost.com/news/the-switch/wp/2014/12/18/the-sony-pictures-hack-explained/
[2] https://www.theguardian.com/technology/2016/dec/14/yahoo-hack-security-of-one-billion-accounts-breached
[3] http://fortune.com/2015/08/26/ashley-madison-hack/

2.2.1 Privacy Definitions

Privacy can be defined as the ability or right that an individual has of protecting his personal information and extends to the ability or right to prevent invasions on the personal space of said individual [2].

One of the prime examples of Privacy applied to information technology problems and mentioned in the literature is related to medical records [25]. These records must be handled with extra care because they contain a large number of sensitive information about the patients. The Patient Record Systems should be able to disclose information only to selected personnel, but not all the information about the patient, only what is necessary to proceed in helping the patient. This example illustrates the tension between having access to the data, that can be useful, but at the same time keeping it closed to other users.

2.2.2 Privacy Protection Goals

New concepts have arisen in recent years for privacy specific protection goals [8]. Their definitions are as follows:

- *Unlinkability* is defined as the property that ensures privacy-relevant data cannot be linked across domains that are constituted by a common purpose and context. In other words, multiple actions from the same user/entity must be unlinkable.

- *Transparency* is defined as the property that ensures all privacy-relevant data processing can be understood and reconstructed at any time. Transparency has to cover not only the actual processing but also the planned processing and after processing to fully know what has happened. Transparency is related to the principles concerning openness and it is a prerequisite to accountability. The user/entity must know and understand how his private data is being handled.

- *Intervenability* is defined as the property that ensures intervention is possible concerning privacy-relevant data processing, in particular by those persons whose data is being processed. Intervenability is related to the principles concerning the rights of an individual, in a way that the owner of privacy-relevant data must have the means to rectify or erase said data.

2.2.3 European Union Legislation

It is also important to mention in the context of this work the current legislation in the European Union (EU) regarding data protection.

The Data Protection Directive[4] is the current law regarding privacy in the EU and is in force since 1995. More recently a replacement has been proposed and accepted in the EU, the General Data Protection Regulation (GDPR)[5], that will take effect in May 2018. Both these laws are regulated by European entities, namely the European Union Agency for Network and Information Security (ENISA)[6] and the Article 29 Data Protection Working Party[7].

According to ENISA, EU data protection law applies to any processing of personal data [7]. This personal data is defined as any information related to an identified or identifiable natural person. In the context of Big Data analysis, the focus is more on indirect identification, which translates into three different approaches: *i)* The possibility of isolating some or all records which identify an individual in a dataset; *ii)* The linking of at least two records concerning the same individual in the same database or in different databases; and *iii)* The possibility to infer the value of an attribute in a dataset from the value of other attributes.

Another important cornerstone of EU data protection law are the principles relating to data quality:

- The *fairness principle* requires that personal data should never be processed without the individual being actually aware of it.

[4] http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex:31995L0046

[5] http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv:OJ.L_.2016.119.01.0001.01.ENG&toc=OJ:L:2016:119:TOC

[6] https://www.enisa.europa.eu/

[7] http://ec.europa.eu/newsroom/just/item-detail.cfm?item_id=50083

- The *purpose limitation principle* implies that data can only be collected for specified, explicit and legitimate purposes.

- The *data minimization principle* states that data processed should be the one which is necessary for the specific purpose previously determined by the data controller.

These three principles altogether indicate that data processing must be done with the consent of the subject, for predetermined purposes communicated to the subject, and data must only be used for those predetermined purposes.

Finally, and also important to mention in the context of this work, are the rights of the data subject according to the EU law. There are two important rights that a subject has: the *right of access* and the *right to object*. The *right of access* ensures that any data subject is entitled to obtain from the data controllers communication of the data that is subjected to processing and to know the logic involved in any processing of data concerning him. This is particularly relevant in the context of Big Data analysis because it limits technological lock-ins and other competition impediments, and it enhances transparency and trust between users and service providers. The *right to object* ensures that data subjects have a right to revoke any prior consent, and to object to the processing of data relating to them, giving to the subject the power to remove himself completely or partially to any data processing mechanisms using his personal data.

2.2.4 Examples of Data Privacy Breaches

In recent years we can find a number of attacks made to systems that handle personal information. Next are a number of examples found relevant regarding Data Privacy breaches.

- Target Pregnancy Leak[8]. In 2012, Target, an American retail company, started merging data from user searches and demographics data in order to learn when their customers

[8] https://www.forbes.com/sites/kashmirhill/2012/02/16/how-target-figured-out-a-teen-girl-was-pregnant-befor #3001668b6668

were pregnant, in order to approach them with a specific advertisement. This is a clear violation of private sensitive information about their customers and their private life.

- Netflix Prize[9]. In 2007, Netflix created a contest to improve their recommendation system. For that, they released a training dataset, with all the personal information regarding customers removed and customer *ids* replaced by randomized *ids*. Later it was shown that it was not enough when a group of researchers linked public information in another movie-rating website (IMDB) with the released dataset and were able to partially de-anonymize the training dataset, compromising the identity of some users.

- AOL search data leak[10]. In 2006, AOL released to the general public a text file containing search keywords for numerous of their users, intended for research purposes. The users were not identified, but personally identifiable information was present in many of the queries. These queries contained a user *id* attributed by AOL, and an individual could be identified and matched to their account and search history by such information.

- Massachusetts GIC medical encounter database[11]. Researcher from Carnegie Mellon University (CMU) linked the anonymized database (which contained birth date, sex and ZIP code) with voter registration and was able to link medical records with individuals.

2.3 Privacy Implications of Personal Data Processing

Data processing is the conversion of raw data to meaningful information through a process. Data is manipulated to produce results that lead to a resolution of a problem or improvement of an existing situation.

Data mining is the process of discovering interesting patterns and knowledge from large amounts of data [18]. We will describe the Data Mining process according to the widely used

[9] https://www.wired.com/2009/12/netflix-privacy-lawsuit
[10] https://techcrunch.com/2006/08/06/aol-proudly-releases-massive-amounts-of-user-search-data
[11] https://techpinions.com/can-you-be-identified-from-anonymous-data-its-not-so-simple/7627

CRISP-DM model [39], in which the process is separated into six major phases, as described next and in Figure 2.1.

- **Business Understanding:** In this initial phase, the project goals and requirements must be understood from a business perspective, and then converted into a Data Mining problem.

- **Data Understanding:** During this phase, an initial data collection is done, followed by a number of activities in order to get familiar with the data, to understand how the data is organized, to identify if the data has quality problems, or even to detect interesting subsets in the data collected.

- **Data Preparation:** This phase covers all the data preparation tasks to construct the final dataset from the initial raw data collected. These tasks include attribute selection, data cleaning, and transformation of data to fit the modeling tools.

- **Modeling:** In this phase, various modeling techniques are selected and applied, and the underlying parameters are calibrated to optimal values. Usually, there are several techniques for the same Data Mining problem type, and some of these techniques require specific data formats.

- **Evaluation:** At this point in the Data Mining process, one or more models have been developed that appear to have high quality, from a data analysis perspective. These models must be evaluated thoroughly so that we can be certain that they properly achieve the business goals.

- **Deployment:** In this last phase, the knowledge gained by the Data Mining process will need to be organized and presented in a way that the customer can use it. This, of course, depends on the requirements presented at the beginning of the process. An example of a common deployment is a simple report on the knowledge obtained.

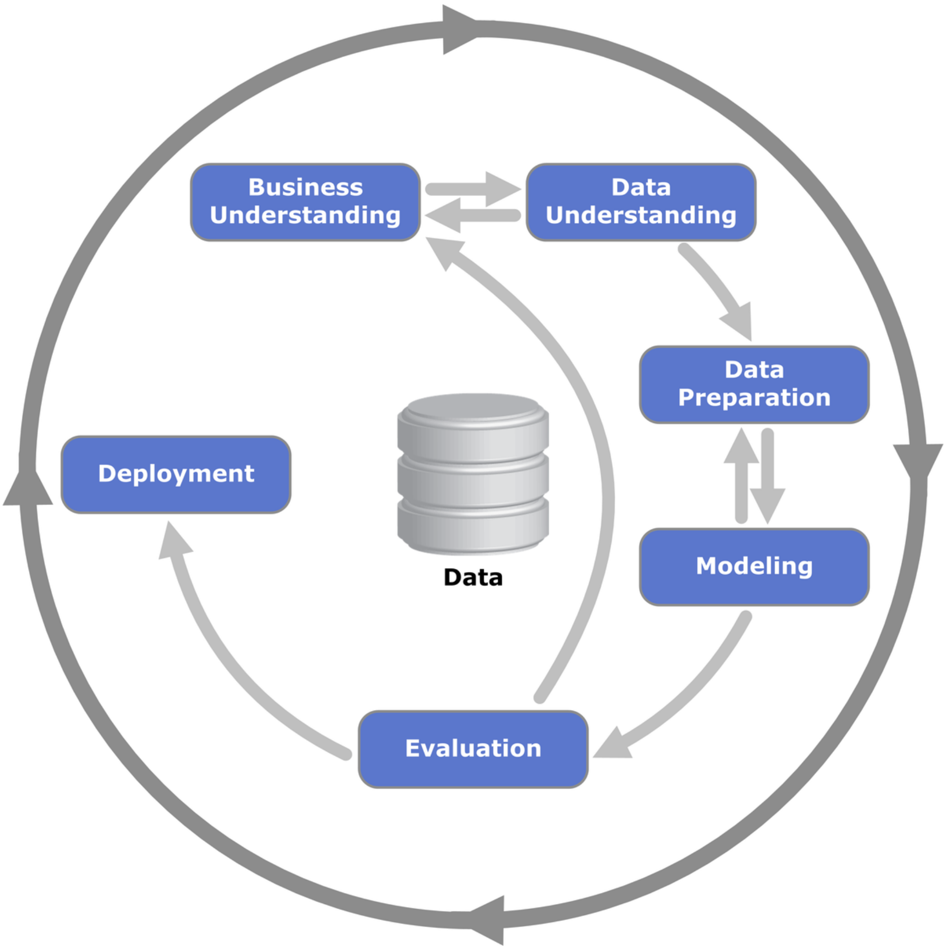


Figure 2.1: Process diagram showing the relationship between the different phases of CRISP-DM [39]

.

In the Data Mining process, we must be aware that, sometimes, private information about individuals can be used and it may lead to breaches of privacy. We define this private information as Personal Identifiable Information (PII) or Sensitive Personal Information. This concept can be defined as information that can be used alone or in conjunction with other information to identify, contact, or locate a single person, or to identify an individual in a context.

2.3.1 Attack Models

When considering the security of a system, one must have into account different concepts to understand how to better implement an efficient and trustworthy security layer.

A threat model specifies the potential threats that can be used against a system, by identifying, enumerating and prioritizing them. A well known and defined threat model is the STRIDE model developed by Microsoft [17].

An attack model is, in terms of cryptanalysis, a classification of cryptographic attacks specifying the type of access the attacker has to a system when attempting to break an encrypted message. We can summarize the attack models in the following four categories:

- **Ciphertext-only attack:** In this type of attack, the cryptanalyst has access only to the ciphertext and has no access to the plaintext. This is the most common type of attack, and it is a requirement for modern ciphers to be resistant to it. An example of a ciphertext-only attack is the brute force attack, where the attacker makes a trial-and-error approach to decrypt the ciphertext.

- **Known-plaintext attack:** In this type of attack, the cryptanalyst has access to a number of pairs of plaintext and the corresponding ciphertext.

- **Chosen-plaintext attack:** In this type of attack, the cryptanalyst is able to encrypt arbitrary plaintext and have access to the resulting ciphertext, allowing him to make a statistical analysis on the plaintext state space.

- **Chosen-ciphertext attack:** In this type of attack, the cryptanalyst is able to choose arbitrary ciphertext and obtain the corresponding plaintext.

We can also classify the types of adversaries according to their strategies in two major groups: *honest-but-curious* and *malicious* adversaries. The honest-but-curious adversary follows protocol, but he will try to extract information from his viewpoint of the protocol to gain some form of advantage or gain access to confidential information. The malicious adversary is one that can deviate from the protocol specification as he desires, and will try to disrupt and/or collect as many information as he can.

Two different trust models can be considered when developing a system: *semi-honest(honest-but-curious) security*, where we assume corrupted parties just gather information they can reach, they do not deviate from the protocol; and *malicious security*, in which we assume corrupted parties can disrupt the protocol execution.

2.4 Privacy-Preserving Techniques

In this section, we will describe some of the techniques used in privacy-preserving Machine Learning related to Data Privacy, namely anonymization, Differential Privacy (DP), Secure Multi-Party Computation (SMPC), Garbled Circuits (GC), Oblivious Transfer (OT), Homomorphic Encryption (HE) and Functional Encryption (FE).

2.4.1 Anonymization

Data anonymization is a type of information sanitization that has the final intent of privacy protection. This can be achieved by either removing or encrypting PII from datasets so that the individuals whom the data describe remain anonymous [31].

Anonymization techniques use a variety of approaches, namely suppression, where a piece of information (*e.g.*, name, age) is removed from the dataset; generalization, where data is coarsened into less refined sets; perturbation, where data is modified by adding noise; and permutation, where sensitive associations between entities in the dataset are swapped.

The goals behind anonymization of data are tightly intertwined with the privacy goals we want to achieve for the data being processed. Usually, one or more techniques mentioned above are applied to the data until certain properties are met, for example, *k-anonymity*, *ℓ-diversity* or *t-closeness*.

- *k-anonymity* property states that, for each individual whose data is released in some dataset, he must not be distinguishable from at least k individuals that are also present in the release [36].

- *ℓ-diversity* property is an extension of the *k-anonymity* property, and furthers the anonymization of data by reducing the granularity of the data representation such that for any given record, there exists at least ℓ different sensitive attribute values, in addition to the guarantees made by *k-anonymity* [26].

- *t-closeness* property is a further refinement of *ℓ-diversity*. This property requires that the distribution of a sensitive attribute in any equivalence class is close to the distribution of the attribute in the overall table, effectively limiting the amount of individual-specific information an observer can learn [24].

2.4.2 Differential Privacy

The concept of Differential Privacy (DP) arose due to recent Data Privacy breaches mentioned in section 2.2.4. In [11] it is shown that the security standard for statistical databases, that states that access to a statistical database should not enable one to learn anything about an individual that could not be learned without access, is not achievable because of the existence of auxiliary information, *i.e.*, information available from sources other than the statistical database.

DP is a process of maximizing the accuracy of queries in statistical databases while minimizing the chances of identifying its records. The core of the procedure is based on *randomized response* [38], giving the possibility to infer statistical information from the dataset, while still ensuring high levels of privacy.

Detailed information about DP and algorithms designed to achieve it are described in [12].

2.4.3 Secure Multi-Party Computation

Secure Multi-Party Computation (SMPC) (also known as secure computation, multi-party computation, or privacy-preserving computation) is a protocol with the goal of creating methods for parties to jointly compute a function over their inputs while keeping those inputs private. The problem of computing functions while also preserving the privacy of the inputs is referred in the literature as a SMPC problem [40]. Generally speaking, a SMPC problem deals with computing any probabilistic function on any input, while also ensuring the correctness of the computation and guaranteeing that no more information is revealed to a participant in the computation that can be inferred from that participant's input and output [16].

A strategy to solve these problems is to trust an external entity (a trusted third party), that can mediate the computation. This approach can be risky because it requires a third party that all participants agree to trust, which can sometimes be difficult to find. Sometimes, the data has such high degree of importance to the participants that even disclosing them to a trusted third party is not viable.

The first implementation of secure computation was introduced as Secure Two-Party Computation (STPC) [40]. It is a simplification of the problem of SMPC, and a known protocol for STPC is Yao's GC protocol, which is detailed in subsection 2.4.5.

When building a SMPC protocol, the most important properties that must be ensured are *input privacy* and *correctness* [15]:

- *Input privacy* property states that no information about the private data held by the parties can be inferred during the execution of the protocol. The only inferences about private data are those that could be inferred from seeing the output of the computation made by the protocol.

- *Correctness* property relates to the existence of malicious parties that could try to deviate from the normal functioning of the protocol. In these cases, the protocol should prevent honest parties to output incorrect results. The approach to implementing the correctness property comes in two alternatives: either the protocol guarantees that the honest parties compute the correct output, a robust protocol; or the honest parties abort the computation if they find an error during the execution of the protocol.

Recent implementations of SMPC protocols are based on *secret sharing*. In 2PC there are only two participants in the computation, and usually one is responsible for starting and encoding the computation mechanism, while the other is the one responsible for evaluating the computation. In multi-party computation, the parties have no special roles and, instead, the encoding is shared amongst the parties, by secret sharing, and the evaluation is made by a protocol.

Secret sharing allows one to distribute a secret over a number of parties by distributing shares to each party. Three of the types of secret sharing techniques more commonly used are: Shamir Secret Sharing [34], Additive Secret Sharing and Replicated Secret Sharing.

2.4.4 Oblivious Transfer

Oblivious Transfer (OT) [30] is a protocol in which a sender transfers one of the potentially many pieces of information he has to a receiver, but remains oblivious as to what piece has been transferred. Let s^0 and s^1 be two strings held by a sender that wants to transfer to a receiver, holding a selection bit b; the protocol allows for only one of the inputs s^b to be transfered; the receiver learns nothing about s^{1-b}, and the sender does not learn b.

An interesting implementation of OT is the one done by Pinkas and Naor [27]. In it, the authors describe an extension to the basic 1-out-of-2 OT protocol, to a 1-out-of-N protocol, and a k-out-of-N protocol.

2.4.5 Garbled Circuits

Yao's Garbled Circuits (GC) [41] is a cryptographic protocol that allows two mistrusting parties to evaluate a function without resorting to a trusted third party. In other words, GC allow parties holding input x and y to evaluate an arbitrary function $f(x, y)$ without leaking any information about their inputs beyond what is inferred from the function output. The idea behind GC is that one party prepares an encrypted version of a circuit that computes f and the second party then computes the output of the circuit without learning any intermediate values.

Some optimizations have been proposed for Yao's GC. Kolesnikov and Schneider [22] present a technique that eliminates the need to garble XOR gates. Pinkas *et al.* present a technique that reduces the size of a garbled table from four to three ciphertexts [29].

2.4.6 Homomorphic Encryption

Homomorphic Encryption (HE) [32] is a cryptographic technique that allows computations to be carried in the ciphertext, so that, when decrypted, the resulting plaintext reflects the computation made. In other words, HE allows to make some computation over the ciphertext, for example, addition, without decrypting it, and the result is the same as making that computation on the plaintext. This is of great importance because it allows chaining multiple services that make computations on a ciphertext, without the need to expose the data to those services. A prime example of HE in real-world problems is electronic voting [21].

Homomorphic cryptosystems can be classified into two distinct groups: partially homomorphic cryptosystems and fully homomorphic cryptosystems.

- In Partial Homomorphic Encryption (PHE), there is a operation or operations, like addition or multiplication, that offer the homomorphic property, but not for all possible operations. Some examples of existing partially homomorphic cryptosystems are: ElGamal cryptosystem [13]; Unpadded RSA [33]; and Pailier cryptosystem [28].

- In Fully Homomorphic Encryption (FHE), it is possible to make any arbitrary computation on the ciphertext. This concept was first introduced in [32], and for many years it remained just as a concept, until recent years when fully homomorphic implementations were developed, for instance, Gentry's cryptosystem [14].

2.4.7 Functional Encryption

In Functional Encryption (FE) systems, a decryption key allows a user to learn a specific function of the encryption data, while also stopping that same user from learning anything more about the encrypted data. In other words, having a secret key only allows for a specific computation of a function over the ciphertext [4]. When comparing FE with HE, the main difference is that, in HE, the result of a computation in the ciphertext remains encrypted, while in FE, the result of the computation is available in the clear.

2.5 Privacy-Preserving Machine Learning

In the context of Data Mining, Machine Learning comes as an important addition to the data processing step. An example of that is the Classification method. Classification is a subset of the Machine Learning applications on Data Mining. It is described by a two-step process, in which a classification algorithm is employed to build a classifier for the data by analyzing a training set made of tuples of data and their associated labels, and then the classifier is used to predict class labels for new data. As such, because the large size of the datasets produced in Big Data operations, classification algorithms have a large quantity of data to learn from, making them less prone to erroneous classification of new data.

The conjunction between Machine Learning and the privacy-preserving concept comes from the need to do knowledge learning over large datasets, while also maintaining an increased concern over the privacy of the data, without degrading the quality of data by using anonymization techniques.

In the field of Machine Learning, we can identify a number of algorithms that can be used in Data Mining. In Table 2.1, we list some of them, giving a brief description, and identifying which privacy-preserving technique could be used with it.

Table 2.1: PPDM algorithms.

Machine Learning Algorithms	Privacy-Preserving Techniques	Short summary	References
Decision Tree	SMPC	Protocol for distributed learning of decision-tree classifiers.	[6]
Naive Bayes	DP	Differentially private naive Bayes classifier. Centralized access to the dataset.	[37]
SVM	SMPC	Algorithm for support vector machine classification over vertically partitioned data.	[42]
k-NN	SMPC	Nearest neighbors of records in horizontally distributed data.	[35]
k-means	SMPC	k-means clustering based on additive secret sharing.	[10]

2.6 Use Cases

In terms of privacy-preserving Machine Learning and its applications, it is important to distinguish the context of what is being processed. The information contained in the data has to be treated differently depending on what that data contains. Different data can be subject to different constraints regarding laws and privacy. Some sensitive data may be only processable in a local environment, while other data can only be processed in a less individualized way.

We now detail three different subjects that are of importance in the area and are subject to different privacy constraints.

- **Health records:** The healthcare system is one of the examples where vast amounts of data are collected every day, and it is of relevance to do Data Mining on patient records, for a better understanding of patients and to improve the healthcare system. But patient records contain very sensitive information about individuals and cannot be processed without the Data Mining system being in compliance with the legislation on Data Privacy, therefore it is of interest to build privacy-preserving systems for the healthcare system, so that hospitals and other health-related organizations can share and infer knowledge without violating the privacy of their patients.

- **Students and taxes:** Bogdanov *et al.* [3] made in 2015 a statistical study using SMPC to look for correlations between working during university studies and failing to graduate in time. For this study, it was necessary to link the database of individual tax payments and the database of higher education universities. These types of government data are subject to strict legislation and cannot simply be handled without strong privacy guarantees.

  To solve this problem, a SMPC system was developed and deployed that could assure a level of privacy that would be in compliance with the laws on Data Privacy. The data processing steps were all made using SMPC between three parties, using OT so that each party would not know each other inputs.

  In the end, the study using SMPC was compared with an anonymized study using 3-anonymity. The loss of samples in the latter was 10%-30%, depending on the demographic group, thus suggesting that producing studies on existing databases using SMPC to enforce privacy can give more accurate results than the same study run using k-anonymity measures.

- **Human mobility:** Another subject that provides great challenges in the field of Data Privacy are the mobility traces generated by people when driving, walking, etc. Mobility traces are highly unique so it is possible, even after anonymizing the dataset, to link an individual to his mobility patterns, as shown by Montjoye *et al.* [9]. Since mobility data contains the approximate whereabouts of individuals, it can be used to reconstruct their movements across space and time. Applying privacy-preserving techniques to process this highly sensitive data can result in better geographic-based recommendation systems.

2.7 Summary

The previous sections provided an overview of the state of the art surrounding privacy-preserving Machine Learning.

expand

3 BARD

In this chapter we discuss the mechanisms and specifications used in building the BARD proof-of-concept platform.

Expand this introduction

3.1 Architecture

This section presents the architectural specifications of Big dAta pRivacy by Design platform (BARD).

Expand

3.1.1 Internal Structure

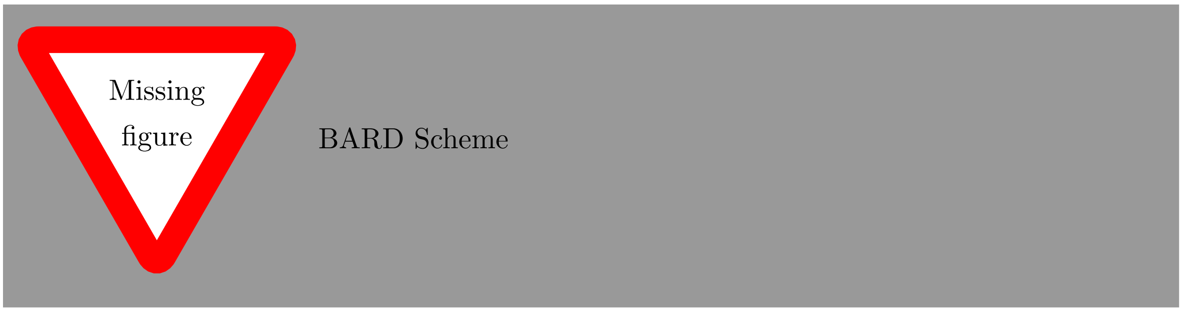
add a smallish intro to this section

BARD is composed of:

- A dataset to train the Machine Learning (ML) algorithm, or the values representing the already trained algorithm.
- A sample or a set of samples that represent the input of the "user", to be predicted.
- A prediction algorithm that depends on the ML algorithm and the privacy-preserving technique chosen.
- A set of toolkits for each of the techniques used.

more?

- ...



fazer um esquema para o BARD. não só de arquitectura, mas como as pessoas interagem com o BARD.

passos no flow geral de processamento de dados : recolha de dados, pre-processamento, etc

3.2 Implementation

When building a platform in the scope of privacy-preserving ML, we must consider not only the traditional steps in data processing, but also have an increased care when preprocessing data to incorporate the cryptographic techniques that we used. This section describes the implementation of BARD. We start off with a description of the datasets chosen to evaluate our platform3.2.1. Then, we explain the preprocessing that was done to those datasets3.2.2. Subsection 3.2.3 presents the baseline implementation of the chosen ML algorithms, resorting to a widely used ML toolkit for Python. In subsection 3.2.4 it is detailed the implementation of the prediction phase of the algorithms. Finally, in subsection 3.2.5, we detail which cryptographic protocols we used, why, and how we implemented them, mentioning which toolkits were used.

3.2.1 Datasets Used

For running the experiments, we found some datasets that are highly used in the literature, and that fitted the subjects mentioned in 2.6. A brief description of each dataset can be found in Table 3.1.

Table 3.1: Datasets used in BARD.

Dataset	Subject	Instances	Features
Breast Cancer Wisconsin[1]	HealthCare	569	30
Pima Indians Diabetes[2]	HealthCare	768	8
Credit Approval[3]	Finance	690	15
Adult Income[4]	Governance	48842	14

3.2.2 Data Preprocessing

Although our data is obtained from publicly available data sources, it is still required to do some preprocessing on the data. The following techniques were used in the datasets described in 3.1.

- **One-hot Encoding**[20] was used to expand discrete variables in a machine readable and comparable way.

- **Scaling(or Feature Scaling?)** was needed to "fix the size of some of the features... ".

change description

-mencionar 1-hot-Encoding -mencionar scaling

quais os datasets que foram visados, e porque.

3.2.3 Baseline

The baseline approach consists on setting up ground values so that meaningful comparisons can be achieved. For understanding the overhead created by privacy-preserving technologies, we implemented a baseline for BARD using the publicly available ML toolkit, scikit-learn for Python[5].

[5] http://scikit-learn.org

listar as baselines criadas, e quais os datasets que foram considerados

3.2.4 Expanded Algorithms

Taking into account that the baseline implementation was done using a toolkit, we could not explicitly compare execution times, due to the fact that the operations in the toolkit were made in a "black box" mode. To solve this problem, we implemented the prediction part of the ML algorithms without using the toolkit.

We started off by extracting the coefficients of the ML algorithms from the toolkit (example: logReg: β_0 and β_1). Then we implemented our own version of the prediction based on the definitions found in Section .

add reference

Detalhar os algoritmos no related work? ou aqui?

detalhar quais os que foram feitos, como e em que linguagem, quais os datasets.

3.2.5 Cryptographic Domain

listar os toolkits que foram testados? ou só os que surtiram resultados? falar de: Garbled Circuits, FHE, PHE. como foi feito, quais os toolkits, etc.

3.3 Add meaningful title

change title of this section

title: use cases again?

Use case scenarios: focus on the healthcare?

orientar a possivel use cases. padroes de utilização (tamanho dataset, tipo pedidos, timers, etc) comparar com sistemas existentes do genero?

agregador de pedidos

3.4 Summary

Summary chapter 3.

4 Evaluation

4.1 Summary

5 Conclusions and Future Work

5.1 Conclusions

5.2 Future Work

Todo list

Bibliography

[1] Rakesh Agrawal and Ramakrishnan Srikant. Privacy-preserving data mining. In *ACM Sigmod Record*, volume 29, pages 439–450. ACM, 2000.

[2] Ross Anderson. *Security engineering*. John Wiley & Sons, 2008.

[3] Dan Bogdanov, Liina Kamm, Baldur Kubo, Reimo Rebane, Ville Sokk, and Riivo Talviste. Students and taxes: a privacy-preserving study using secure computation. *Proceedings on Privacy Enhancing Technologies*, 2016(3):117–135, 2016.

[4] Dan Boneh, Amit Sahai, and Brent Waters. Functional encryption: Definitions and challenges. In *Theory of Cryptography Conference*, pages 253–273. Springer, 2011.

[5] Ljiljana Brankovic and Vladimir Estivill-Castro. Privacy issues in knowledge discovery and data mining. In *Australian institute of computer ethics conference*, pages 89–99, 1999.

[6] Justin Brickell and Vitaly Shmatikov. Privacy-preserving classifier learning. In *International Conference on Financial Cryptography and Data Security*, pages 128–147. Springer, 2009.

[7] Giuseppe D'Acquisto, Josep Domingo-Ferrer, Panayiotis Kikiras, Vicenç Torra, Yves-Alexandre de Montjoye, and Athena Bourka. Privacy by design in big data: An overview of privacy enhancing technologies in the era of big data analytics. *European Union Agency for Network and Information Security*, 2015.

[8] George Danezis, Josep Domingo-Ferrer, Marit Hansen, Jaap-Henk Hoepman, Daniel Le Metayer, Rodica Tirtea, and Stefan Schiffner. Privacy and data protection by design-from policy to engineering. *European Union Agency for Network and Information Security*, 2015.

[9] Yves-Alexandre De Montjoye, César A Hidalgo, Michel Verleysen, and Vincent D Blondel. Unique in the crowd: The privacy bounds of human mobility. *Scientific reports*, 3: 1376, 2013.

[10] Mahir Can Doganay, Thomas B Pedersen, Yücel Saygin, Erkay Savaş, and Albert Levi. Distributed privacy preserving k-means clustering with additive secret sharing. In *Proceedings of the 2008 international workshop on Privacy and anonymity in information society*, pages 3–11. ACM, 2008.

[11] Cynthia Dwork. Differential privacy. *Proceedings of the 33rd International Colloquium on Automata, Languages and Programming*, pages 1–12, 2006. ISSN 03029743. doi: 10.1007/11787006_1.

[12] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[13] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE transactions on information theory*, 31(4):469–472, 1985.

[14] Craig Gentry et al. Fully homomorphic encryption using ideal lattices. In *Symposium on the Theory of Computing*, volume 9, pages 169–178, 2009.

[15] Oded Goldreich. Secure multi-party computation. *Manuscript. Preliminary version*, pages 86–97, 1998.

[16] Shafi Goldwasser. Multi party computations: past and present. In *Proceedings of the sixteenth annual ACM symposium on Principles of distributed computing*, pages 1–6. ACM, 1997.

[17] Munawar Hafiz, Paul Adamczyk, and Ralph E Johnson. Organizing security patterns. *IEEE software*, 24(4), 2007.

[18] Jiawei Han, Jian Pei, and Micheline Kamber. *Data mining: concepts and techniques*. Elsevier, 2011.

[19] Marit Hansen, Meiko Jensen, and Martin Rost. Protection goals for privacy engineering. *Proceedings - IEEE Security and Privacy Workshops, SPW*, pages 159–166, 2015. doi: 10.1109/SPW.2015.13.

[20] David Harris and Sarah Harris. *Digital design and computer architecture*. Morgan Kaufmann, 2010.

[21] Martin Hirt and Kazue Sako. Efficient receipt-free voting based on homomorphic encryption. In *Advances in Cryptology—EUROCRYPT 2000*, pages 539–556. Springer, 2000.

[22] Vladimir Kolesnikov and Thomas Schneider. Improved garbled circuit: Free xor gates and applications. *Automata, Languages and Programming*, pages 486–498, 2008.

[23] Lei Xu, Chunxiao Jiang, Jian Wang, Jian Yuan, Yong Ren, Lei Xu, Chunxiao Jiang, Jian Wang, Jian Yuan, and Yong Ren. Information Security in Big Data: Privacy and Data Mining. *IEEE Access*, 2:1149–1176, 2014. ISSN 2169-3536. doi: 10.1109/ACCESS.2014.2362522. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6919256.

[24] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *Data Engineering. ICDE. IEEE 23rd International Conference on*, pages 106–115. IEEE, 2007.

[25] Rongxing Lu, Hui Zhu, Ximeng Liu, Joseph Liu, and Jun Shao. Toward efficient and privacy-preserving computing in big data era. *IEEE Network*, 28(4):46–50, 2014. ISSN 08908044. doi: 10.1109/MNET.2014.6863131.

[26] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3, 2007.

[27] Moni Naor and Benny Pinkas. Computationally secure oblivious transfer. *Journal of Cryptology*, 18(1):1–35, 2005.

[28] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 223–238. Springer, 1999.

[29] Benny Pinkas, Thomas Schneider, Nigel P Smart, and Stephen C Williams. Secure two-party computation is practical. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 250–267. Springer, 2009.

[30] Michael O Rabin. How to exchange secrets with oblivious transfer. *IACR Cryptology ePrint Archive*, 2005:187, 2005.

[31] Balaji Raghunathan. *The Complete Book of Data Anonymization: From Planning to Implementation*. CRC Press, 2013.

[32] Ronald L Rivest, Len Adleman, and Michael L Dertouzos. On data banks and privacy homomorphisms. *Foundations of secure computation*, 4(11):169–180, 1978.

[33] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[34] Adi Shamir. How to share a secret. *Communications of the ACM*, 22(11):612–613, 1979.

[35] Mark Shaneck, Yongdae Kim, and Vipin Kumar. Privacy preserving nearest neighbor search. In *Data Mining Workshops. ICDM Workshops. Sixth IEEE International Conference on*, pages 541–545. IEEE, 2006.

[36] Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[37] Jaideep Vaidya, Basit Shafiq, Anirban Basu, and Yuan Hong. Differentially private naive bayes classification. In *Proceedings of the 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-Volume 01*, pages 571–576. IEEE Computer Society, 2013.

[38] Stanley L Warner. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309):63–69, 1965.

[39] Rüdiger Wirth and Jochen Hipp. Crisp-dm: Towards a standard process model for data mining. In *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*, pages 29–39. Citeseer, 2000.

[40] Andrew C Yao. Protocols for secure computations. In *Foundations of Computer Science, 1982. SFCS'08. 23rd Annual Symposium on*, pages 160–164. IEEE, 1982.

[41] Andrew Chi-Chih Yao. How to generate and exchange secrets. In *Foundations of Computer Science, 1986., 27th Annual Symposium on*, pages 162–167. IEEE, 1986.

[42] Hwanjo Yu, Jaideep Vaidya, and Xiaoqian Jiang. Privacy-preserving svm classification on vertically partitioned data. *Advances in Knowledge Discovery and Data Mining*, pages 647–656, 2006.