Scaling Up secure Processing, Anonymization and generation of Health Data for EU cross border collaborative research and Innovation

D2.1 — Interim report on data anonymization, de-anonymization and synthetic data generation techniques, tools and services

Funded by
the European Union

Grant Agreement Nr. 10109571

Project Information

Project Title	Scaling Up Secure Processing, Anonymization and Generation of Health Data for EU Cross Border Collaborative Research and Innovation		
Project Acronym	SECURED	**Project No.**	10109571
Start Date	01 January 2023	**Project Duration**	36 months
Project Website	https://secured-project.eu/		

Project Partners

Num.	Partner Name	Short Name	Country
1 (C)	Universiteit van Amsterdam	UvA	NL
2	Erasmus Universitair Medisch Centrum Rotterdam	EMC	NL
3	Budapesti Muszaki Es Gazdasagtudomanyi Egyetem	BME	HU
4	ATOS Spain SA	ATOS	ES
5	NXP Semiconductors Belgium NV	NXP	BE
6	THALES SIX GTS France SAS	THALES	FR
7	Barcelona Supercomputing Center Centro Nacional De Supercomputacion	BSC CNS	ES
8	Fundacion Para La Investigacion Biomedica Hospital Infantil Universitario Nino Jesus	HNJ	ES
9	Katholieke Universiteit Leuven	KUL	BE
10	Erevnitiko Panepistimiako Institouto Systimaton Epikoinonion Kai Ypolgiston-emp	ICCS	EL
11	Athina-Erevnitiko Kentro Kainotomias Stis Technologies Tis Pliroforias, Ton Epikoinonion Kai Tis Gnosis	ISI	EL
12	University College Cork - National University of Ireland, Cork	UCC	IE
13	Università Degli Studi di Sassari	UNISS	IT
14	Semmelweis Egyetem	SEM	HU
15	Fundacio Institut De Recerca Contra La Leucemia Josep Carreras	JCLRI	ES
16	Catalink Limited	CTL	CY
17	Circular Economy Foundation	CEF	BE

Project Coordinator: Francesco Regazzoni - University of Amsterdam - Amsterdam, The Netherlands

Copyright

Disclaimer

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the Health and Digital Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

Except as otherwise expressly provided, the information in this document is provided by SECURED members "as is" without warranty of any kind, expressed, implied or statutory, including but not limited to any implied warranties of merchantability, fitness for a particular purpose and no infringement of third party’s rights.

SECURED shall not be liable for any direct, indirect, incidental, special or consequential damages of any kind or nature whatsoever (including, without limitation, any damages arising from loss of use or lost business, revenue, profits, data or goodwill) arising in connection with any infringement claims by third parties or the specification, whether in an action in contract, tort, strict liability, negligence, or any other theory, even if advised of the possibility of such damages.

Deliverable Information

Workpackage	WP2
Workpackage Leader	Alberto Gutierrez-Torre (BSC)
Deliverable No.	D2.1
Deliverable Title	Interim report on data anonymization, de-anonymization and synthetic data generation techniques, tools and services
Lead Beneficiary	UCC
Type of Deliverable	Report
Dissemination Level	Public
Due Date	29/02/2024

Document Information

Delivery Date	27/03/2024
No. pages	57
Version | Status	1.6 | Final
Deliverable Leader	Paolo Palmieri (UCC)
Internal Reviewer #1	Joppe Bos (NXP)
Internal Reviewer #2	Gergely Acs (BME)

Quality Control

Approved by Internal Reviewer #1	19/02/2024
Approved by Internal Reviewer #2	19/02/2024
Approved by Workpackage Leader	19/03/2024
Approved by Quality Manager	27/03/2024
Approved by Project Coordinator	27/03/2024

List of Authors

Name(s)	Partner
Paolo Palmieri, Hamza Aguelal	UCC
Francesco Regazzoni, Marco Brohet, Kyrian Maat, Georgios Tasopoulos	UvA
Alberto Gutierrez-Torre	BSC
Gergely Acs	BME
Solène Blasco-Lopez, Vincent Thouvenot	THALES
Miryam Villegas Jimenez, Juan Carlos Perez Baun	ATOS
Ioannis N. Tzortzis, Charalampos Zafeiropoulos, Dimitrios Kalogeras	ICCS

The list of authors reflects the major contributors to the activity described in the document. The list of authors does not imply any claim of ownership on the Intellectual Properties described in this document. The authors and the publishers make no expressed or implied warranty of any kind and assume no responsibilities for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information contained in this document.

Revision History

Date	Ver.	Author(s)	Summary of main changes
20.11.2023	0.1	Francesco Regazzoni	Created the document
21.11.2023	0.2	Paolo Palmieri	Added initial version of the contents
7.12.2023	0.3	Paolo Palmieri	Introduction and structure of the document
16.12.2023	0.4	Alberto Gutierrez-Torre	Initial contribution on section 3 (data types)
11.01.2024	0.5	Juan Carlos Perez Baun	Initial contribution section 4
18.01.2024	0.6	Alberto Gutierrez-Torre, Gergely Acs, Vincent Theouvenot and Ioannis N. Tzortzis	Draft of section 6 (synthetic data)
20.01.2024	0.7	Alberto Gutierrez-Torre	Draft of section 3 (data types)
26.01.2024	0.8	Hamza Aguelal and Paolo Palmieri	Main contents of section 5 (de-anonymization) integrated into the document
05.02.2024	0.9	Francesco Regazzoni, Marco Brohet, Kyrian Maat, Georgios Tasopoulos (UvA)	Initial contribution section 7
07.02.2024	1.0	Alberto Gutierrez-Torre, Gergely Acs, Vincent Theouvenot and Ioannis N. Tzortzis	Main contents of section 6 (synthetic data)
09.02.2024	1.1	Paolo Palmieri and Alberto Gutierrez-Torre	Executive summary and conclusions
10.02.2024	1.2	Alberto Gutierrez-Torre	Main contents of section 3
20.02.2024	1.3	Alberto Gutierrez-Torre, Vincent Theouvenot and Hamza Aguelal	Corrections from internal reviews on Sections 2, 3, 5 and 6
24.02.2024	1.4	Gergely Acs, Juan Carlos Perez Baun and Ioannis N. Tzortzis	Corrections from internal reviews on Sections 2, 3, 5 and 6

Date	Ver.	Author(s)	Summary of main changes
27.02.2024	1.5	Alberto Gutierrez-Torre	Minor final corrections
20.03.2024	1.6	Paolo Palmieri	Final editing

Table of Contents

Acronyms and Abbreviations

AI Artificial Intelligence.

API Application Programming Interface.

BDS RD Big Data and Security Research and Development.

CA Consortium Agreement.

CD/CI Continuous Development/Continuous Integration.

CNN Convolutional Neural Network.

CTG Cardiotocography.

D Deliverable.

DANS Data ANonymization Service.

DEAN De-Anonymization/Re-Identification.

DICOM Digital Imaging and Communication In Medicine.

DL Deliverable Leader.

DoA Description of Actions.

DP ε-Differential Privacy.

DP-SGD Differential Privacy Stochastic Gradient Descent.

DTW Dynamic Time Warping.

EC European Commission.

ECG Electrocardiogram.

EHR Electronic Health Record.

GA Grant Agreement.

GAN Generative Adversarial Networks.

GDPR General Data Protection Regulation.

GI Genome Inference.

GUI Graphical User Interface.

GWAS Genome-Wide Association Studies.

JSON JavaScript Object Notation.

LD Linkage Disequilibrium.

MRI Magnetic Resonance Imaging.

NLP Natural Language Processing.

PC Project Coordinator.

PCT Project Coordination Team.

PEB Project Executive Board.

PETs Privacy-Enhancing Technologies.

PGA Project General Assembly.

PM Project Manager.

PMT Project Management Team.

PO Project Office (European Commission).

QI Query Inference.

RDP Rényi Differential Privacy.

RNN Recurrent Neural Network.

RST reStructredText.

SDG Synthetic Data Generation.

SIMD Single Instruction, Multiple Data.

SNP Single Nucleotide Polymorphism.

T Task.

TL Task Leader.

WGAN Wasserstein Generative Adversarial Network.

WGS Whole Genome Sequencing.

WP Work Pakage.

WPL Work Pakage Leader.

1 Executive Summary

Data management has grown more difficult in an era where data is both more easily accessible and more valuable than ever. This is especially true when handling sensitive data subject to stringent data protection regulations like the General Data Protection Regulation (GDPR). This problem is nowhere more apparent than in health data, which is considered to be highly private. The SECURED project is trageted at scaling up the secure processing, anonymization and synthetic generation of health data. As part of the project, Work Package 2 (WP2) is specifically aimed at the design and implementation of techinques for the anonymization of data (T2.1); the validation of anonymization through re-identification attacks (T2.2); the augmentation of data through synthetic generation (T2.3); and the implementation and integration of the above components into a coherent and cohesive software library (with hardware accelleration suppport where warranted).

This deliverable, *D2.1 - Interim report on data anonymization, de-anonymization and synthetic data generation techniques, tools and services*, provides a detailed summary of the current status of WP2 and related tasks, and describes the interim results of the SECURED project in the design and development of anonymization, de-anonymization and synthetic data generation techniques for health data.

In details, this deliverable presents:

- A brief introduction to the WP and the underlying Data Flow of the SECURED design, which, together with the Processing Flow addressed in WP3, underpins the SECURED technological and innovation process.
- A detailed description of the data types and dataset that have been identified as useful in the design and development of the WP techniques, both originating from open data and from partner institutions in the SECURED consortium.
- The status of the current work on the development of anonymization solutions for health data being investigated as part of T2.1, lead by ATOS.
- A description of the de-anonymization attack techniques and related data types that have been identified as useful in the development of an assesment strategy for the re-identification risk of anonymized datasets (T2.2, UCC).
- The current research lines and status of the synthetic data generation techniques that are being researched as part of T2.3, lead by BSC and contributed by BME, Thales and ICCS.
- An early overview of the development of the software library implementing and integrating the WP2 techniques, and an assessment of where hardware accelleration may be required (T2.4, UvA).

This deliverable, hence, serves as a reference on the direction of WP2 as well as a summary of the current status and results of the SECURED project.

1.1 Related documents

- SECURED Deliverable D4.1 - State of the Art and initial technical requirements
- SECURED Deliverable D3.1 - Interim report on Scalable Secure Multiparty Computation, Federated Learning and Unbiased AI techniques and tools
- SECURED Deliverable D1.2 - GDPR and Ethics Project Guidelines
- SECURED Deliverable D1.6 - Data Management Plan

Also of interest:

- Directorate-General for Health and Food Safety - Proposal for a regulation - The European Health Data Space (COM(2022) 197/2), 3 May 2022

2 Introduction

SECURED is aimed at providing a platform and architecture for the secure, trusted, efficient, decentralized and cooperative processing of health data. For this purpose, a number of techniques are being investigated, optimized and tested, in the domains of secure computation, data anonymization (and anonymization assessment via pre-emptive de-anonymization), as well as the generation of new, high quality and private synthetic data. The techniques are being implemented as part of a library providing the tools for the secure interconnection of EU health data hubs, the health data analytics research community, health application innovators (such as e-health SMEs) as well as end users, ultimately enabling health datasets to be shared and made available across Europe. The SECURED approach follows two parallel, independent yet interacting flows to innovation, the data flow and the processing flow. This deliverable is focused on the former, while a separate parallel deliverable (D3.1 *Interim report on Scalable Secure Multiparty Computation, Federated Learning and Unbiased AI techniques and tools*) focuses on the latter.

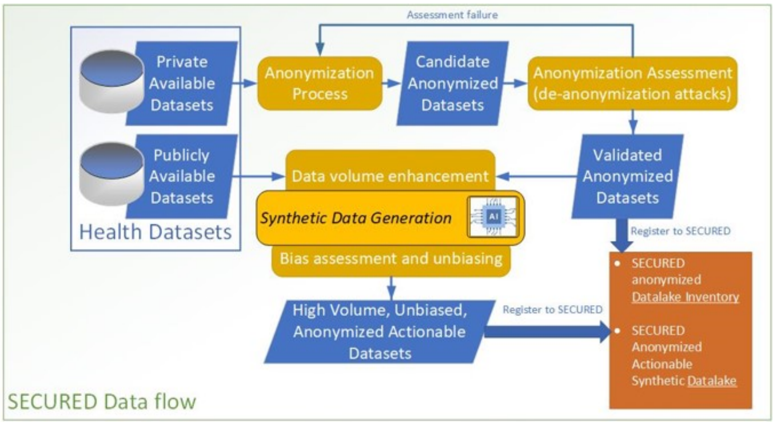
Assessment failure
Private Available Datasets
Publicly Available Datasets
Health Datasets
Anonymization Process
Candidate Anonymized Datasets
Anonymization Assessment (de-anonymization attacks)
Validated Anonymized Datasets
Data volume enhancement
Synthetic Data Generation
AI
Bias assessment and unbiasing
Register to SECURED
SECURED anonymized Datalake Inventory
High Volume, Unbiased, Anonymized Actionable Datasets
Register to SECURED
SECURED Anonymized Actionable Synthetic Datalake
SECURED Data flow

Figure 1 – The data flow is an integral part of the SECURED architecture and concept.

The SECURED data flow, visualised in Figure 1 and implemented in the project Work Package 2 (WP2), is targeted at securing health data by applying appropriate privacy-preservation techniques, with a specific focus on anonymization and de-anonymization, as well as synthetic data generation. The utimate goal is twofold: first, to enable stakeholders that generate and hold health data (such as hospitals, healthcare facilities, as well as EU health data hubs) to properly anonymize their datasets, using novel anonymization techniques whose efficacy can be validated through new de-anonymization technologies, both developed by SECURED; and secondly to augment the datasets through novel mechanisms for privacy-preserving synthetic data generation, in order to generate sufficient volume for training artificial intelligence and machine learning models, as well as performing other data analyses.

With respect to the first objective, SECURED is developing a suite of novel anonymization tools (Section 4) as well as an anonymity assessment mechanism that can validate the anonymization by performing de-anonymization attacks (Section 5). This assessment is targeted at ultimately identifying a de-anonymization risk metric that can intuitively convey the level of protection offered by the anonymization. This could be used, for instance, in a setting where specific privacy requirements are to be met: if a given anonymized dataset fails to reach a certain threshold determined by the data owner, the anonymization process can be repeated with different parameters and/or techniques.

In relation to the second objective above, privacy-preserving synthetic data generation techniques are being developed in SECURED (Section 6), in order to leverage and augment health data. This is fundamental in two instances: where an (anonymized) dataset is insufficient in volume to be useful in the construction of a machine/deep learning model, to generate additional meaningful data; and where a dataset cannot be shared or transfered, but its fundamental feature can be extracted to produce related but private synthetic data to be used for further analysis.

The techniques produced by the SECURED data flow in the project Work Package 2 are being implemented in a library (Section 7) which, combined with the complementary library for secure computation and private data analysis, will provide the required tools to (de-)anonymize and synthetize, as well as securely share, process and analyze health data.

Ultimately, the end outcome of the SECURED data flow is enabling the generation and sharing of unbiased, anonymized actionable datasets. This will be achieved, in part, through the SECURED Innohub, a privacy-enhancing platform that will provide tools, services, and overall support to stakeholders in the healthcare domain, including researchers, innovators or health data users, as well as EU data hubs across Europe, thus enabling them to reap the benefits of accurate data analysis while preserving the privacy of the data, processed in a distributed and private manner. The SECURED hub will promote collaboration among parties by acting as a one-stop collaboration platform for stakeholders.

2.1 Structure of the document

This document provides a snapshot of the progress of the SECURED project on the stated objectives described above. The text is organised as follows: in Section 3 the data types and sets that will enable the development and validation of the technologies produced in the data flow are presented. Sections 4, 5 and 6 present the anonymization, de-anonymization and synthetic data generation technologies being developed, respectively. In Section 7 the current state of development of the library where the technologies will converge is discussed. Finally, Section 8 provides conclusions and an overview of the way forward towards successful completion of the work in WP 2.

3 Health data types and datasets

Data is a key component in modern Healthcare organizations, as it is the tool for the healthcare professionals to analyze and diagnose patients. Moreover, the rise of usage of *Machine Learning*, specially *Deep Learning*, has shown the possibilities of using medical data to assist the healthcare professionals in their tasks [1]. Typically, each hospital can be considered a silo that contains data from their patients. However, the *European Health Data Space* promoted from the EU commission promises to break this problem by harmonizing and interconnecting all the data. With this kind of approach, medical research can advance faster as the data can be accessed in a more easy way. This has risks, as privacy has to be assessed and preserved. For this reason, under the SECURED project, we aim to provide tools to make use of the data without leaking any sensitive information. Knowing which data is going to be used and the problem that is attacked with it is important as it conditions the approaches and measures to secure the processes.

In particular, the work performed under Work Package 2 (WP2) is tailored to the data types that are available, as the techniques presented in this document are not agnostic of the data they are treating. Therefore it is important to define which are the data types that are going to be anonymized, re-identified or synthetically generated. In the following subsections we define the data types that we aim to use in SECURED and the current datasets available, both open data and private from the use cases within the consortium.

3.1 Data type definition

In this section we present the different data types considered within the scope of the SECURED project, complementing the information provided in Section 5.1 of deliverable *D4.1 State of the Art and initial technical requirements* and the datasets found in *D1.6 Management Plan*.

3.1.1 Genomic data

Genomic data, a cornerstone of contemporary health research, encompasses crucial information encoded within genetic markers or Single Nucleotide Polymorphisms (SNPs). These genetic markers play a pivotal role in identifying specific disease-related characteristics, serving as molecular signposts that guide researchers in understanding the genetic underpinnings of various health conditions. The significance of genetic markers lies in their ability to pinpoint unique variations in an individual's genetic code, offering invaluable insights into disease predispositions and susceptibilities.

The extraction of phenotypes from genomic data further enhances the depth of analysis. Phenotypes, observable traits resulting from the interplay of genetic and environmental factors, provide a tangible link between genetic markers and the expression of specific characteristics. Unravelling phenotypic information from genomic data unlocks a wealth of potential insights, shedding light on the intricate connections between genetics and health outcomes. This comprehensive understanding sets the stage for exploring vulnerabilities in de-anonymization attacks, particularly identifying individuals based on their unique genetic profiles.

3.1.2 Medical image data

Medical imagery is usually a relevant piece of data for both healthcare professionals and medical data analists. These images, usually formatted according to the DICOM standard, encapsulate detailed information from diagnostic procedures. Incorporating DICOM data extends the scope of analysis to encompass medical scans, enabling researchers to correlate genetic and neurological findings with visual representations of anatomical structures. We can find different modalities such as Histopathological Images, Mammograms, Magnetic Resonance Imaging (MRI) or ultrasound imaging from different parts of the body.

Also, neuroimaging data emerges as a valuable source for exploration. This kind of imaging comes in both NIfTI and DICOM formats, being NIfTI preferred by the neuroscience community [2]. The extraction of features from neuroimaging data becomes a focal point, allowing researchers to discern patterns intricately linked to genetic markers and associated pathologies. Neuroimaging captures the brain's structural and functional aspects, providing a unique perspective on the interplay between genetics and neurological traits. In this context, beacons act as navigational indicators, spotlighting specific genetic features within the vast landscape of neuroimaging data. Beacons guide attention to regions of interest and facilitate a nuanced analysis that intertwines genetic signatures with neuroimaging patterns.

3.1.3 Time series

Time series is a very common type of data in healthcare, as many sensors provide values of desired variables to measure continuously.

For example, Electro CardioGrams (ECG) which measure the heart's electrical activity over a specific duration and Cardiotocographies (CTG), which measures the baby's heartbeat along with the mother's uterine contractions. Each data point in the time series corresponds to a specific time and contains the electrical voltage generated by the heart during a particular cardiac cycle.

Following a similar structure to ECG time series, there is also telemetry from devices that measure breath. These devices can measure variables such as O_2 saturation or CO_2 concentration that are useful to evaluate the condition of a patient.

3.1.4 Electronic Health Records

Electronic Health Records (EHR) can be seen as the electronic version of the patients medical history. These records are usually saved in the form of tables that contain all the information from the patients, from the demographics to the interventions and sickness that the patient has gone through. This kind of data encompasses a potential benefit to healthcare systems as the data can be shared between hospitals to treat better a given patient and also can help in research as those EHR can be used for new potential application in many different areas such as personalized medicine.

3.2 Datasets

In this subsection we are covering, from the previous different data types, the datasets that we have detected that are relevant for the project. This list includes an update from the datasets provided by the use cases in the Data Management Plan (D1.6).

In order to keep track of the data relevant to the project, two different data registries have been created: open data and project data registries. Both of them are intended to keep track on what data is available, how to obtain it and who is using it inside the project. This last part is relevant as it provides a way of link the partners that are using the same dataset and fosters collaboration between them. A summary of these two registries is presented in Table 4, which show the potential datasets detected at the moment. Notice that the datasets from the project have two identifiers: first is of the actual dataset and second is the identifier of the general data collection defined in the Data Management Plan. Some of these datasets have already been explored and being in use, such as the *CSAW* and *InBreast* open datasets. Regarding project data, the consortium is currently working on how to establish the legal agreements between use case data and the interested partners. As a first success case, the *HNJ1* dataset is currently being in use for synthetic data generation. In other cases, the work in the WP2 has started with open data as this data is from the start more accessible. However, this open data is of the same modalities that the use case data have as the idea is to start with open data and then fine tune the methodology when the use case data when it is accessible for the partners interested on it. These agreements have to take care of two different things:

- Ethics: usually taken care by the ethics board from the data origin and the partners from SECURED to provide a description of the work to be done.

Data type	Dataset name	Availability	Process summary
Mammograms	InBreast	Open	Request with signed contract [3]
Mammograms	CSAW	Open	Website request [4]
Mammograms	Optimam	Open	Website request [5]
Breast MRI	Duke Breast Cancer MRI	Open	Direct download [6]
Lymph node Histopathologic scan	PatchCamelyon	Open	Direct download [7]
Chest X-ray	ChestX-ray14	Open	Direct download [8]
Chest X-ray	NODE21	Open	Direct download [9]
Chest X-ray	CheXpert	Open	Direct download [10]
Genomic data	St. Jude genomic datasets	Open	Request with signed contract [11]
Ultrasound vascular imaging data	EMC-US/DS1	Project	Internal agreement
MRI vascular imaging data	EMC-MRI/DS2	Project	Internal agreement
Sensor data from patients at home	HNJ1/DS6	Project	Internal agreement (Outside of GDPR)
Sensor data from patients at UCIP	HNJ2/DS6	Project	Internal agreement
Scanned whole slide colorectal images	SEM-CRC/DS7	Project	Internal agreement
Annotations for SEM-CRC	SEM-CRC-TAG/DS7	Project	Internal agreement
Mammograms	SEM-MAMMO/DS7	Project	Internal agreement
Electrocardiograms	SEM-ECG/DS7	Project	Internal agreement
Cardiotogography (Fetal heartbeat)	SEM-CTG/DS7	Project	Internal agreement
EHR for reimbursement	SEM-TAB/DS7	Project	Internal agreement
Genotyping arrays and clinical data	JCLRI1/DS8	Project	Internal agreement

Table 4 – Open and Project data

- GDPR and national laws: to be addressed by the legal teams of the data origin and the destination. Notice that even if GDPR does not apply to a given dataset, national laws from the origin and destination countries may still apply. For further reference, check *D1.2 GDPR and Ethics Project Guidelines*.

Regarding datasets that are provided by partners in the consortium, their availability is subject to data sharing agreements being concluded between the dataset owner/originator and the partner institution(s). A number of data sharing agreements are in place or are currently being negotiated.

4 Data anonymisation techniques

The health sector generates several heterogenous data sources such as patient electronic health records (EHR), laboratory tests and medical imaging for diagnosis, genomic data, device-generated medical and wellbeing data, treatment and drug prescriptions or administrative data. The analysis of these data provides valuable information for doctors, researchers and healthcare bodies for improving diagnosis, treatment and at the end the healthcare services. Ensuring the security access to data and protect the patient privacy and data confidentiality implies the treatment of these heterogeneous and huge amount of data applying different tools and techniques. The use of cryptographic and anonymisation techniques helps preserve the privacy of patients' health data. While health data treatment cryptographic techniques are covered by WP3, the anonymisation techniques are covered by WP2 and namely in T2.1.

In the next subsections are provided details of the anonymisation tools and libraries that make up the building blocks of the prototype of the SECURED anonymisation tool and the rational for selecting them. Then, an initial architecture of this tool is depicted beside the envisaged deployment as microservices. Also, linked with Section 3 Health data types and datasets, a short description of the data types to be used is provided. Finally, a summary of the progress made for developing this anonymisation tool, including initial results and the next steps are depicted.

4.1 Explored techniques

The objective of anonymisation process is to reduce the risk of re-identification while preserving data utility, when personal and sensitive data are shared. Different privacy-preserving techniques, such as generalization, suppression, and noise application, can maintain high privacy levels while impacting predictive performance. T2.1 devoted to the data anonymisation started in June 2023, the activities performed until now are based on the work done in D4.1 “State of the Art and initial technical requirements”. SECURED D4.1 [12] provides an overview of the privacy models and anonymisation techniques which can be applied to different types of datasets, both structured and unstructured, with a focus on respecting data usefulness and truthfulness while safeguarding user privacy. The study of the state of the art made in D4.1, and the analysis of the strengths and weaknesses of various anonymisation techniques, emphasizes the importance of combining multiple methods to achieve effective privacy protection while maintaining utility. Beside this analysis, an overview of the current open-source tools covering the privacy models and anonymisation techniques has been performed as well. According to the preliminary evaluation of the different tools provided in D4.1, two of them, the DANS tool (from now it will be named legacy DANS) and open-source Amnesia library, have been initially selected for providing an initial prototype for SECURED project. The rationale for selecting them are based on the following aspects: (i) providing at least k-anonymity and DP as privacy models, (ii) capable to manage different type of data, covering those provided by the different use cases/pilots, (iii) possibility to integrate in an easy way and (iv) the documentation is available and provide regular update of the tool/library.

Table 5 provides the properties the selected tools/libraries fulfil. Additional details are included in sections 4.1.1 and 4.1.2.

Table 5 – Properties of anonymisation tools: legacy DANS and Amnesia

Solution	Privacy models/Techniques	Type of data covered	Integration	Info	Last update
Legacy DANS	k-anonymity, l-diversity, t-closeness, Differential Privacy	Structured	Yes	Yes	2022
Amnesia	k-anonymity, km-anonymity, Differential Privacy, Masking.	Structured, Unstructured: DICOM, time-series, genomic	Yes	Yes	2022

Others open-source tools and libraries are being considered such as µ-ARGUS, sdcMicro among others. These tools and others which can fit with the SECURED project will be explored in a second stage of the project.

4.1.1 Legacy DANS

The Data anonymisation Service (DANS) is an enhanced anonymisation tool developed by the BDS RD Spain[1] (Eviden[2]- an Atos group business), in the context of the CyberSecurityforEurope project[3]. This tool is based on the open-source ARX library[4], being designed as a modular solution offering enough flexibility to users for customising the anonymisation process based on the user needs, handling large datasets. Provides k-anonymity [13], ℓ-diversity [14] and t-closeness [15] privacy models. Additionally, it supports several anonymisation techniques such as generalization, suppression, and micro aggregation. During the anonymisation process removing only the identifier attributes is not sufficient for reducing the privacy risk and maintain the data utility (the more privacy protection the less utility reached, and the other way around). Thus, it is necessary to combine different privacy models and techniques, with the aim to find an adequate trade-off of privacy vs utility, achieving different level of privacy protection depending on the sensitivity of the data. In this way, legacy DANS is focused on privacy quality but keeping the balance between the user-privacy preservation and the data utility for analytics. The graphical user interface (GUI) provided by this tool eases the execution of the anonymisation process by users with low anonymisation knowledge. On top of that, it helps organisations to accomplish with data protection regulations such as GDPR or Data Act. Moreover, this tool can be applicable to other domains (finance, insurance, eucation) than the health one. The main reasons for selecting legacy DANS are the following:

- The anonymisation library embedded in DANS tool provides several privacy models facilitating the automatic anonymisation of large datasets in the health domain. Currently legacy DANS is focused on k-anonymity privacy model which assure that each element in the anonymised dataset will be indistinguishable from other k-1 elements in that dataset considering the quasi-identifier attributes. Moreover, the use of l-diversity together with k-anonymity increases the privacy protection of the sensitive attributes, preventing attribute disclosure. Beside these privacy models the addition of t-closeness and DP is expected.
- Able to manage large datasets in the health domain but can be used in other domains such as financial, education or mobility.
- Is designed in a modular manner, facilitating the addition of new open-source libraries, to enhance the type of data to be managed, including structured and unstructured data.
- Is offered in two-fold, as a java library to be embedded in legacy systems and as a microservice for being deployed on the data provider premises. This aspect eases the integration with other services, helping the adoption of this kind of tools by data providers for protecting data privacy.
- The original library is updated on a regular basis and provide available documentation [16].

4.1.2 Amnesia

The Amnesia open-source library has also been selected to be included into the SECURED anonymisation toolset. It is focused on health data as well, providing k-anonymity and km -anonymity privacy models, using generalisation or suppression mechanisms. Km-anonymity is a relaxed form of k-anonymity, requiring that each combination of m quasi-identifier attributes must appear at least k times in the anonymised dataset [17], protecting only from attackers knowing up to m values of the quasi-identifier attributes, providing a better information quality, preventing identity disclosure.

[1] https://booklet.evidenresearch.eu/about-us
[2] https://eviden.com/
[3] https://cybersec4europe.eu/
[4] https://github.com/arx-deidentifier/arx

The main reasons for selecting Amnesia are the following:

- Ease to integrate in an anonymisation framework , for using their functionalities through a ReST API.
- Provides k-anonymity, km-anonymity and allows to manage small and medium datasets and the possibility to anonymise metadata of DICOM images.
- Provides privacy risk and utility information of the anonymisation process.
- The original library is updated time to time and provide available documentation.

4.2 Scaling up approach

The anonymisation toolset (from now named DANS 2.0) to be developed in SECURED project, is intended to protect and preserve the privacy of personal and sensitive information by removing or modify identifiable information avoiding the re-identification of the data subject, ensuring that the anonymisation process will maintain the utility of the protected data for further analysis. DANS 2.0 is an enhanced version of legacy DANS, including new open-source libraries (e.g., Amnesia library) providing additional properties.

For facilitating the implementation and deployment process of the DANS 2.0 and allow the adoption of new open source anonymisation libraries, the design and development of this asset follows a modular design based on a microservice approach. This modular design will support the scaling-up needs when big data analysis is required. The different modules will be deployed as microservices, which will allow the addition of new anonymisation techniques and privacy models in an easy way facilitating the deployment and scale-up of the tool. Figure 2 depicts a high-level overview of the modular architecture.

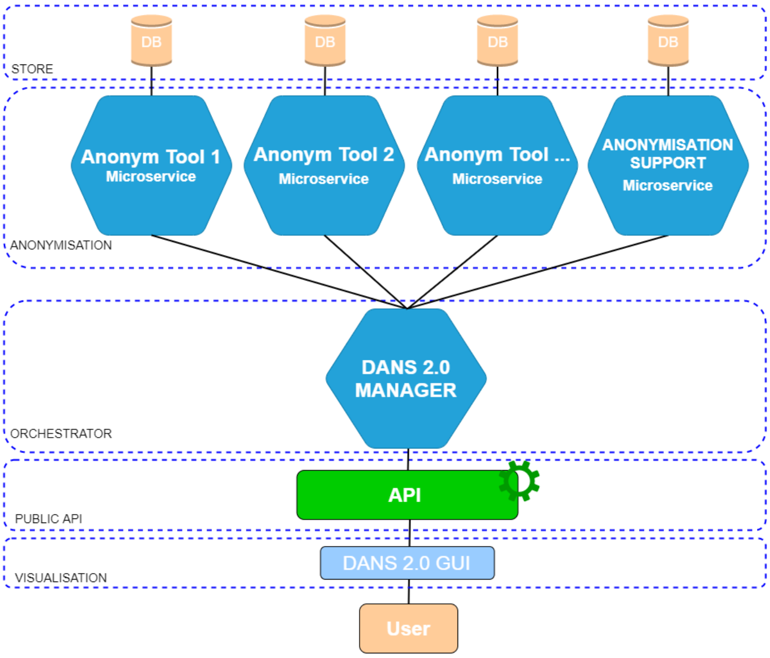

Figure 2 – High-level view of DANS 2.0 microservice architecture.

The adoption of a microservices architecture approach provides several advantages for scaling up the anonymisation tool:

- Decoupling: As each module is a self-contained element, the development (can be used different technologies for each module), deployment (CD/CI techniques can be applied) and maintenance of each service is independent of the rest of modules, improving the overall performance.
- Each module can be scaled independently according to their needs.
- Resilience: The isolation of the services facilitates bug-fixing minimising impact on the overall process.

4.3 Relation to datatypes and use cases

The use of the anonymisation tool on the different SECURED pilots has been discussed in several meetings with pilot owners for analysing the utility of the anonymisation tool for protecting their data. As a result of these meetings, the anonymisation tool DANS 2.0 will be applied for anonymise data in three SECURED use cases, namely in pilot 2 “Telemonitoring for children”, pilot 3 “Synthetic data generation for education” and pilot 4 "Genomic data". In the case of the Telemonitoring for children pilot, structured and time series data are managed. Regarding the synthetic data for education pilot, the anonymisation tool will be restricted to some scenarios, exploring the anonymisation of time-series data and metadata associated to electrocardiograms, the anonymisation of electronic health records for reimbursement (structured tabular data) and the mammography images scenario where the metadata associated to the images will be treated. In case of genomic data the metadata associated to these data will be managed. Depending on the structure and type of data (structured, semi-structured and unstructured data) different techniques and privacy models can be used for data anonymisation, as described in D4.1 [12]. Following this guide, for time series datasets the k-anonymity and differential privacy models can be used [18]. In the case of structured tabular data and associated metadata to images or genomic data the same approach will be adopted. Based on these considerations, the selected anonymisation tool (legacy DANS) and library (Amnesia) described in section 4.1.2, provide the required privacy models to be used for anonymising these data. Additional details of the data to be used in SECURED are included in Section 3.

4.4 Progress

The progress made from the start of the T2.1 are basically the next:

- Design of the anonymisation component providing the architecture of the tool. This architecture is based on the privacy-preserving requirements and the constraints of the SECURED pilots and health domain. A high-level architecture of DANS 2.0 (the SECURED anonymisation toolset) is provided in Figure 3. This architecture implies several layers including:

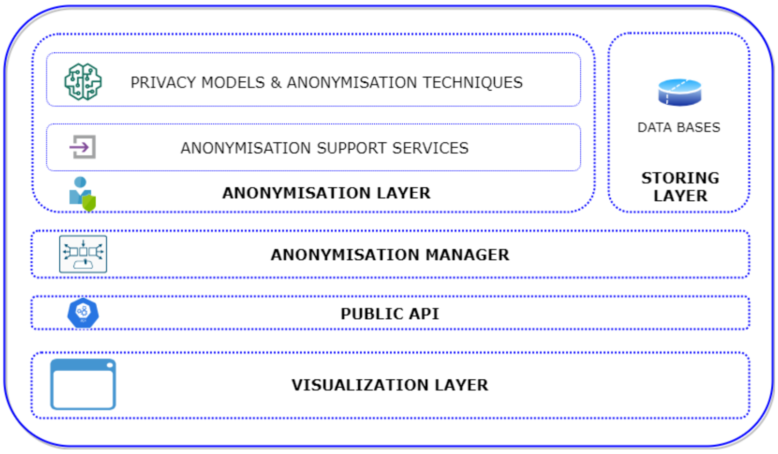

Figure 3 – High-level view of DANS 2.0 architecture.

  - **The anonymisation layer** which comprises:
    - The anonymisation libraries providing the different privacy models, such as k-anonymity, l-diversity or Differential Privacy and the privacy-preserving techniques (micro-aggregation, generalisation, sampling, masking, suppression, etc).
    - The anonymisation support services comprises the I/O data, the data specification of how to anonymise (classification of the attributes, parametrisation of the anonymisation methods to be applied, etc.), the hierarchy builder, or the risk assessment services. Reports on the anonymisation process can be generated also.

  - **Storing layer**: the anonymisation process rely on light data bases for allocating files on the data provider infrastructure and for storing configuration settings, hierarchies, etc.
  - An **anonymisation manager** for orchestrating the anonymisation process depending on the type of data-source (structured, unstructured, or semi-structured data formats) or size (large or small datasets) or the anonymisation tool to use.
  - A **public OpenAPI** for accessing the ReST services. The GUI or trusted third parties can use this API.
  - A **visualisation layer** for users to access the different services. The offered Graphical User Interface (GUI) facilitates to Low skilled users the performance of the anonymisation process.

- Design of the deployment of the tool DANS 2.0 (see Figure 2), considering the scalability of the asset and the later integration with other tools of the SECURED project.

- Analysis of the initially selected anonymisation tools: Legacy DANS and Amnesia.

- Regarding the specific progress on Legacy DANS tool:

  - Included persistency of database: database persistence is recommended when large datasets are managed. Different databases (e.g., Postgres, MySQL, ...) can be used for this purpose.
  - Checked and analysed updated documentation and functionalities of the basic ARX java library[5].
  - Updated tool with the latest version of opensource library (v3.9.1).
  - Tested and fixing detected bugs after updating embedded library.
  - Improved the exception management, fixed knowndetected bugs, cleaned the OpenAPI removing old endpoints not needed anymore.
  - Updated the l-diversity privacy model.
  - Exploring new privacy models, e.g., t-closeness and differential privacy. Initial tests with existing datasets and use case datasets (when available) are ongoing.
  - Enhancing the docker deployment with several environment variables to ease the deployment on the different pilots.
  - Tested simple examples of available use cases' datasets.

- Regarding the progress on Amnesia library:

  - Analysed documentation and functionalities of the basic Amnesia java library[6].
  - Tested the last version (v1.3.3) of the open-source library with dataset by using curl calls[7]. Even Amnesia library contains code for managing DICOM files and apply differential privacy, some problems arose during the initial testing phase. Additional work on this matter will be needed for leveraging these functionalities.
  - Defined a draft flow for anonymising datasets and retrieve an anonymisation report. Figure 4 depicts the steps of the process: (1) Open a session needed for the whole anonymisation process, (2) Load the dataset (accepting csv, xlsx, txt formats), (3) Identify the attributes (identifier, quasi-identifier, sensitive) included in the dataset, (4) Create a hierarchy file, JSON format, for integer quasi-identifier attributes, (5) Upload the hierarchy files associated with quasi-identifier attributes (age, dates, ...), if they don't exist a customised one can be generated, (6) Set hierarchies to quasi-identifier attributes and k parameter, (7) After k-anonymising, initial results are obtained in a JSON format, (8) The anonymised dataset can be retrieved depending on the different parametrisation, (9) Statistics and data loss can be obtained for a given solution.
  - Some constraints and limitations have been detected during the tests:

[5] https://github.com/arx-deidentifier/arx/releases

[6] https://amnesia.openaire.eu/features.html

[7] https://github.com/dTsitsigkos/Amnesia

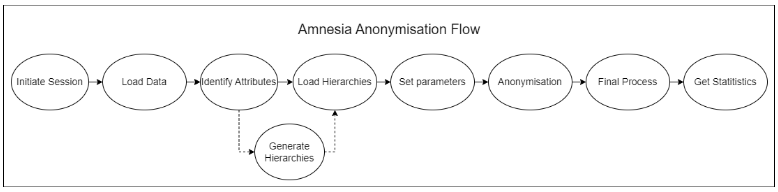

Figure 4 – anonymisation process for Amnesia.

    * The hierarchy files used and created have different format and structure than the used in DANS tool. It is necessary to analyse how to align this aspect.
    * The hierarchy files initially cover integer and date attributes. String format is not considered.
    * The anonymised attributes where the hierarchies are applied contains the associated code of the range instead of range itself. This need to be analysed.
  - Tested simple examples of available use cases' datasets.

In the short term, the next points will be developed:

- The anonymisation results and statistics report to be stored on databases.
- Integrating new privacy models, e.g., t-closeness and differential privacy into legacy DANS tool.
- Create the first draft of the OpenAPI for Amnesia to be deployed as a microservice, based on the flow described in Figure 4. Basically it contains the following endpoints: Init(): SessionId; LoadData (File dataset, SessionId); CreateHierarchy (Hierarchy Parameters, SessionId); LoadHierarchy(File hierarchy, SessionId); SetParameters (anonymisation Parameters, SessionId); anonymisation (SessionId); GetResult (SessionId): anonymisation file, statistic results.
- Create the first version of the amnesia microservice with basic functionalities (init, uploadFile, setParameters, anonymise).
- Designing the first version of the public OpenAPI for DANS 2.0:
  - **LoadData**: we can distinguish three main kinds of data: data to be anonymised, data anonymised and hierarchical data to be used in an anonymisation process. Data to be anonymised are temporarily stored in database if big size, or just kept on memory if small size. Data conforming a generic hierarchy can be stored in database for later use.
  - **CreateHierarchy**: Initially predefined hierarchies will be used for anonymisation.
  - **Anonymise**: The datasets will be anonymised according to a very detailed specification of how the user wants to anonymise their data (classification of attributes, privacy and generalization models, parametrization of each privacy model to be applied, and any other configuration).
  - **GetResults**: apart from the anonymised dataset, a statistical report can be retrieved for a given anonymisation. Such report plus a thorough analysis of the anonymised data, help the user to tune the anonymisation process until she gets a data set useful for her needs.

In the medium term the following points will be covered:

- The selected tools and libraries will test all the proper datasets used in the related use cases e.g., time-series datasets, metadata for DICOM files and genomic datasets.
- Work on km-anonymity model through Amnesia.

The updated design of the GUI is postponed to the next stage once the prototype is delivered and a stable anonymisation process is defined.

5 Data de-anonymization/re-identification techniques

The need to protect the confidentiality and integrity of health information has given rise to research on de-identification or anonymization techniques, procedures that make data anonymous and protect individual identities while maintaining the usefulness. Conversely, de-anonymization, or re-identification, enables the uncovering the original identities through anonymised data. The latter techniques are used by attackers to try and extract sensitive information from anonymized datasets, but can also be used constructively by dataset owner to assess the efficacy of anonymization techniques being applied to their data. This dynamic interplay between anonymization and de-anonymization necessitates careful exploration and investigation.

Anonymization plays a pivotal role in safeguarding the privacy and public confidence in health research, as ongoing efforts to anonymize health datasets are witnessing a consistent increase [19], yet anonymization in itself is not always sufficient to ensure long-term privacy, and hence maintain public confidence in health research. Therefore, comprehending the potential weaknesses and deficiencies present in de-identified datasets is crucial for enhancing data privacy. De-anonymization or re-identification, in this context, serves as a valuable mechanism for locating and fixing vulnerabilities that might still exist despite well-meaning anonymization attempts.

5.1 De-anonymization techniques

The de-anonymization research efforts that constitute the core of Task 2.2. The task will focus on the datasets described in Section 3, in line and in collaboration with Tasks 2.1 and 2.3.

Electrocardiogram (ECGs) data, which record the heart's dynamic electrical activity, are essential for improving cardiac diagnosis and comprehending cardiovascular health. ECG datasets, which consist of time series data indicating voltage variations during particular cardiac cycles, present difficulties in maintaining patient privacy. To reduce the risk of re-identification, a specific focus on anonymization is required because every data point in the temporal series has the potential to be an identifier. The quantification of re-identification risks through the design of attacks is a contribution of Task 2.2, by performing de-anonymization in the process of improving the privacy afforded by anonymization.

Correspondingly, Electronic Health Records (EHRs) constitute a fundamental component of contemporary healthcare, comprising a thorough documentation of a patient's medical background and prescribed medical treatments. Though the sensitive nature of this data raises serious concerns regarding patient privacy and the possibility of de-anonymization threats, integrating EHRs into health research offers profound insights.

Another vital component of modern health research is genomic data, which provides deep insights into the complexities of human biology, disease susceptibility, and possible strategies for treatment. The human genome has vast information that can be used to improve personalised medicine, target specific treatments, and better understand how genetics affect health outcomes. However, maintaining privacy while utilising genetic data presents significant challenges due to its fundamental sensitivity.

The increasing availability of sensitive health information heightens privacy risks, requiring stronger anonymization methods and, in parallel, the evaluation of those methods through re-identification attacks. The decision to concentrate on ECGs, EHRs and genomic data stems from their availability, sensitivity, and rich information content, offering a compelling opportunity for potential de-anonymization attacks. This investigation aims to unveil the challenges in genomic data anonymization, emphasizing the critical need for privacy safeguards in health research.

In the following we describe the avenues for de-anonymization attacks that have been identified as the most promising for each of the data types above. Due to the ad-hoc nature of de-anonymization attacks, each technique may or may not be successful for a given dataset, and will need to be evaluated and fine-tuned individually to attack targets (the test datasets to be used in the research). As such, the final results of the research and eventual outcomes of T2.2 are subject to a high degree of potential future change (in terms of techniques used and investigation directions) during the execution of the research, even when compared to other technical research tasks. In other words, in order to successfully attack anonymised datasets, one must remain flexible as to what techniques are used, and as such the techniques proposed below are more of a strarting point than a well-defined roadmap.

1. **Advanced De-anonymization Techniques for ECG Data:**

   With its distinct time series characteristics, ECG data presents a prime target for de-anonymization attacks. Specialized algorithms like Dynamic Time Warping (DTW) [20] [21] and machine learning methods such as Support Vector Machines (SVMs) are instrumental in identifying unique physiological patterns in ECG signals. The challenge intensifies when comparing fully identified ECG datasets with their anonymized counterparts. To conduct effective de-anonymization, techniques such as Recurrent Neural Networks (RNNs) LSTM [22] [23] [24] for example, and Convolutional Neural Networks (CNNs) are employed for their proficiency in signal processing and feature extraction. These methods enhance the ability to detect and match identifiable markers within ECG data, thus facilitating the re-identification of individuals from datasets [25]. The integration of these approaches underlines the sophisticated nature of ECG data de-anonymization, bridging the gap between complex signal patterns and identifiable personal information.

2. **Electronic Health Records (EHR) - Vulnerabilities and De-anonymization Approaches:**

   Electronic Health Records (EHR) are a goldmine of personal health information, making them susceptible to de-anonymization attacks. These attacks exploit the rich, multidimensional nature of EHR data [26], which includes patient demographics, clinical history, laboratory results, and more. Techniques for de-anonymization in EHR data encompass a range of machine learning algorithms, including clustering for pattern recognition and anomaly detection, decision trees for dissecting hierarchical data structures, and neural networks for extracting complex interrelations among diverse data points. The utilization of NLP techniques also in interpreting unstructured data, such as clinical notes. By correlating de-identified EHR data with external datasets, attackers can re-identify individuals through unique health patterns or anomalies. Protecting EHR data requires an understanding of these methods and the implementation of robust anonymization techniques.

3. **Significance of Genetic Markers and Extraction of Phenotypes:**

   De-anonymization in genomic research involves using genetic markers, like Single Nucleotide Polymorphisms (SNPs), to link genetic traits to diseases. Techniques such as Hidden Markov Models (HMMs) and machine learning, including Support Vector Machines (SVMs), are crucial for interpreting these patterns for identifying individuals. Similarly, extracting phenotypes from genomic data is key, using methods like clustering algorithms, t-SNE, and deep learning models like RNNs and LSTMs, to link observable traits with genetic variations.

   Genome-Wide Association Studies (GWAS) and Whole Genome Sequencing (WGS) are instrumental in associating genetic variants with specific traits, using statistical methods and machine learning for precision. In de-anonymization, variant calling identifies unique genetic markers, and pathway analysis elucidates how genes contribute to diseases, crucial for re-identifying individuals from anonymized genomic data by understanding the intricate genotype-phenotype relationships.

4. **Neuroimaging Data - Genetic Associations and Pattern Analysis:**

   Our discussion primarily focuses on raw images; however, this doesn't preclude us from addressing random forests (RFs), given that RFs are typically employed for structured data including metadata. Therefore, in this context, applying RFs to raw images necessitates feature extraction to identify pertinent features. This implies that images must undergo preprocessing to extract these relevant features.

   Neuroimaging data reveal complex connections between genetic markers and brain characteristics, crucial for de-anonymization attacks. Techniques like Hidden Markov Models (HMMs) capture brain feature dependencies [27], while machine learning algorithms, including Random Forests and Support Vector Machines (SVMs) [28], classify neuroimaging patterns linked to genetics. Deep neural networks, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), efficiently extract complex patterns from neuroimaging data, elucidating the relationship between genetics and brain traits, thereby aiding in the re-identification process.

5. **Health Images and DICOM Data:**

   De-anonymization techniques apply to health imaging, particularly Digital Imaging and Communications in Medicine (DICOM) data, in the larger context of health research. The potential for health photos to disclose private medical information makes them significant. Convolutional neural networks (CNNs), in particular, are frequently used in machine learning for feature extraction in health imaging. Architectures like VGG (Visual Geometry Group) or ResNet (Residual Networks) prove effective in discerning patterns and abnormalities within medical images [29]. Image segmentation algorithms, such as U-Net, contribute to isolating regions of interest within health images. These methods allow for spotting trends and irregularities, improving the attacker's capacity to associate particular photographs with particular people. Deep learning methods highlight the susceptibility of de-identified health data to sophisticated attacks and aid in the detailed interpretation of health imagery.

6. **Utilizing Beacons in Genetic and Health Data De-anonymization:**

   Beacons serve as crucial markers in genomic and health research, indicating specific genetic or health traits. These repositories, containing data on genetic marker frequencies or health patterns, are instrumental in de-anonymization attacks. By cross-referencing de-identified data with beacon databases, techniques like machine learning and deep learning algorithms [30] identify unique features, highlighting the vulnerability of de-identified data. Algorithms such as probabilistic models, decision trees, and ensemble methods like Gradient Boosting or Random Forests are used to analyze beacon responses, enhancing feature identification. Additionally, graph-based algorithms, including PageRank, help reveal complex interconnections within beacons, adding depth to the de-anonymization process. In beacon systems, which respond with the presence or absence of specific genetic variants, graph-based algorithms can analyze the network of responses to identify critical markers. Mapping the relationships between genetic markers or health traits as a graph helps in identifying which markers or traits are most crucial for distinguishing individuals. This integration of diverse algorithms demonstrates the sophisticated nature of de-anonymization, especially when datasets are linked to external repositories. [31]

5.2 Re-Identification attacks

In this section, we outline the specific de-anonymization attacks that have been identified as the most promising to be attempted on the target datasets, in order to determine their susceptibility to each given attack route. The focus is on membership attacks, inference matching attacks, attribute inference attacks, and linkage attacks. These attack strategies are selected based on a preliminary analysis of the potential vulnerabilities identified in the genomic, neuroimaging, and other health data under consideration. We fully expect only some of these attacks to be successful against some of the datasets: to imagine the opposite would be to essentially expect the current anonymization techniques to be utterly and completely broken, which clearly is not the case. T2.2 is focused on uncovering yet unidentified vulnerabilities that are likely to be present in some techniques against some types of attacks, with the purpose of proposing countermeasures.

5.2.1 Membership attacks

Membership attacks aim to ascertain whether a specific individual's data is included in a dataset, even after anonymization. Our research focuses on genomic data, utilizing variant calling algorithms such as GATK or Samtools for Whole Genome Sequencing (WGS) data [32]. Identified genetic variants undergo annotation with tools like ANNOVAR or VEP to understand their functional impact on genes and phenotypes. The process involves:

- Variant Calling using GATK or Samtools.
- Functional Annotation with ANNOVAR or VEP [33].
- Identification of Unique Genetic Identifiers

Algorithmic approaches include Maximum Entropy Models, as demonstrated in [32], using predictions from de-identified WGS to match with phenotypic and demographic information. This facilitates a comprehensive membership attack, assessing whether a specific record corresponds to an individual.

5.2.2 Inference attacks

Inference attacks, encompassing both Membership and Attribute Inference, involve the aggregation of information to deduce specific details about individuals within a dataset [34]. This process often incorporates statistical analyses, machine learning, or data linking techniques to infer personal data from the available dataset.

In the context of genomic data, the inference-matching attack unfolds with a strategic focus on predicting sensitive attributes and uniquely identifying individuals. Employing algorithms such as Maximum Entropy Models, Likelihood Ratio Tests, and Probabilistic Models, this attack delves into the reconstruction of individual genomes using clustering techniques, revealing distinctive genetic markers. The utilization of beacons becomes pivotal, with queries and clustering-based algorithms contributing to genome reconstruction. The [31] research delves into two attack strategies: Query Inference (QI) and Genome Inference (GI). QI attacks use Linkage Disequilibrium (LD) to infer beacon responses from SNP pairs, while GI attacks apply high-order Markov chains with beacon queries to reveal hidden SNPs. Key technical aspects encompass variant calling algorithms like GATK or Samtools for genetic variant identification, functional annotation tools like ANNOVAR or VEP, and the clustering of similar genetic markers for reconstruction.

The Attack is relevant for neuroimaging data in identifying discriminative features related to genetic markers and associated pathologies. Convolutional Neural Networks (CNNs) take the lead in extracting features from neuroimaging data, emphasizing patterns linked to genetic traits. Matrix analysis, employing score-based sampling methods, aids in pinpointing discriminative features. External data integration further enhances the attack's matching capabilities, exploring intersections with external sources like beacons. The technical intricacies encompass the application of CNNs for feature extraction, matrix analysis for discriminative feature identification, and the synergy of external sources for comprehensive matching.

Beyond genomic and neuroimaging, the Inference Matching Attack extends its reach to general health data (time series) like in [21], or DICOM files. In the last scenario, the attack focuses on identifying patterns indicative of unique attributes within health images. Image analysis techniques, potentially including Convolutional Neural Networks (CNNs), play a pivotal role in feature extraction from DICOM files. Specialized processing is undertaken to extract relevant features, emphasizing patterns associated with genetic markers or individual traits.

5.2.3 Linkage attacks

Linkage Attacks involve a multistep process to establish connections between records from different datasets, unravelling sensitive information across seemingly unrelated sources. In the context of genomic data, adversaries initiate the linkage attack by identifying shared genetic markers or traits across multiple beacons or genetic databases. The process often begins with extracting relevant features, emphasizing patterns indicative of genetic relationships.

- Feature Extraction: This step involves the identification of unique genomic features, markers, or phenotypic traits that are distinctive enough to be used as identifiers across datasets and discern which features can most effectively link records across seemingly unrelated information pools.
- Correlation Analysis: Analyze correlations between genetic variants and traits for linking patterns, and leveraging the inherent link between genetics and phenotypic expressions to trace and establish identities.
- Probabilistic Matching: Use probabilistic techniques for connections based on allele frequencies and shared traits as Bayesian networks and Hidden Markov Models (HMMs) are powerful in modeling the probabilistic relationships between genetic markers. This method stands out for its reliance on mathematical probabilities to make inferences [27], thereby amplifying the precision of the attack strategies.
- Data Integration: Synthesize information and accommodate variations for a comprehensive profile. By integrating information from diverse datasets, attackers can construct comprehensive profiles of individuals, significantly increasing the risk of privacy breaches.
- Pattern Recognition: Employ advanced algorithms for subtle connections, contributing to re-identification. DL networks (CNN for image-based phenotypic data as an example) or ensemble methods like Random Forests, are used to detect complex patterns and associations in the data.

5.2.4 Methodologies and attack overview:

Our review highlights a variety of methodologies employed in de-anonymization attacks, focusing on key areas such as ECG data, EHR data, and genomic data. This overview, while not exhaustive, aims to shed light on the strategies and techniques that have been explored in the research:

- **ECG Data:**
  - **Pattern Recognition Attack:** Employ LSTM [24] networks to identify unique ECG patterns [23].
  - **Temporal Matching Attack:** Use Dynamic Time Warping (DTW) [20] for matching ECG data with identified datasets [35].
  - **Feature Correlation Attack:** Apply SVMs to correlate ECG features with personal identifiers.

  Basic scenarios:

  1. **Heart Rate Variability Analysis:**
     - Dataset includes anonymized heart rate variability metrics in ECG recordings.
     - **Attack:** We assume an adversary's capacity to access supplementary data (Leaked Health Records and publicly available information - Social media and health blogs), then he compares these health data to the the extracted patterns to match individuals (The adversary's ability to obtain the external knowledge determines the actual risk).
  2. **Identification of Cardiac Conditions:**
     - Anonymized ECG data shows waveform patterns indicating cardiac conditions.
     - **Attack:** Attacker matches these patterns with known disease profiles for re-identification.
- **EHR Data:**
  - **Pattern Association Attack:** Identify patterns in EHR data like medication history and diagnostic codes.
  - **Cluster Analysis Attack:** Use clustering algorithms to identify unique patient groups.

  - **Feature Linking Attack:** Link anonymized EHR data with public health datasets [26].

  Basic Scenarios:

  1. **Medication and Diagnosis Correlation:**
     - Dataset includes anonymized records with medication and diagnosis codes.
     - **Attack:** Adversary correlates this data with public health databases to identify patients.
  2. **Hospital Visit Patterns:**
     - Anonymized EHR data shows patterns of hospital visits.
     - **Attack:** Attacker uses pattern recognition to match visit patterns for re-identification.

- **Genomic Data:**

  - **Membership Attack:** Researchers focus on reconstructing individual genomes using clustering techniques [30]. Beacon queries are employed, with a *Clustering-Based Algorithm for Genome Reconstruction Attack* [30], and queries are further used for membership detection [31].
  - **Inference Matching Attack:** Techniques involve identifying discriminative features using matrix analysis and leverage-score-based sampling methods [28]. Reconstruction clustering techniques are used for inference, associating genomic findings with personal information [30].
  - **Attribute Inference Attack:** Pathway analysis is employed to assess how sets of genes contribute to biological pathways [32]. Integrative genomic analysis combines WGS data with other omics data for a holistic view [32].

  Basic Scenarios:

  1. **Genetic Marker Identification:**
     - The released dataset includes genomic sequences and information about the presence or absence of specific genetic markers associated with rare diseases.
     - **Attack:** An adversary with access to external genetic databases or publicly available genomic datasets may attempt to identify unique genetic markers in the released dataset.
  2. **Inference of Disease Predispositions:**
     - The anonymized genomic data contains information on variations in certain genes linked to an increased risk of particular diseases.
     - **Attack:** An attacker could cross-reference the genomic data with publicly available information on disease-gene associations to infer the potential disease predispositions of individuals in the dataset.

- **Health Data in General:**

  - **Attribute Inference Attack:** Sensors general health data is subjected to feature analysis and classification for identification. Similarity-based attacks on general health time series combine blood volume pulse, electrodermal activity, body temperature, and acceleration data [21].

  Basic Scenarios

  1. **Specific Medical Condition Inference:**
     - A hospital releases de-identified aggregated health data (without personal identifiers) but includes information on the prevalence of certain medical conditions.
     - **Attack:** An adversary could attempt to link this released data with external information sources (publicly available data, previous leaks, or even other de-identified datasets) to identify individuals and associate them with specific medical conditions.

- **Neuroimaging Data:**

- **Inference Matching Attack:** CNN mostly are applied for advanced image analysis in neuroimaging data, focusing on extracting features and identifying discriminative patterns critical for re-identification. This method leverages external sources, such as publicly available neuroimaging databases and genomic databases, to enhance matching accuracy. The integration of external genomic information or detailed phenotypic data with neuroimaging patterns allows for a precise matching process. The promise of this approach lies in its ability to exploit the rich, yet subtle, information contained in neuroimaging data, which, when combined with complementary datasets, can significantly improve the accuracy of re-identification attacks. [28].

5.3 Samples of attack plans

In the landscape of digital health information, understanding the potential vulnerabilities and methods employed in de-anonymization attacks is crucial for developing robust privacy protection measures. Below, we outline a series of potential attack plans targeting various types of health data. While we will explore the identified attack scenarios, it is expected that only some of these scenarios will lead to successful re-identification, and on some given datasets. In the context of a risk-based approach, such as the one taken by the GDPR, not only the success of a specific attack on a given dataset, but also the overall probability of successful attacks on a data type should be evaluated.

5.3.1 ECG Data de-anonymization attack:

Table 6 – ECG Data De-anonymization Steps

Step	**Description**
Data Acquisition	Collect ECG data, both identified and anonymized, for analysis.
Feature Extraction	Transforms the preprocessed ECG signals into a structured features (QRS detection, heart rate variability), where the model will analyze these to capture and learn distinctive ECG signal patterns associated with individual identifiers.
Pattern Analysis	Use LSTM, DTW, and SVMs for capturing ECG signal patterns and individual identifiers.
Model Training	Deep learning models are trained on labelled data to recognize these patterns. This step differentiates from the initial analysis by focusing on constructing predictive models capable of identifying similar patterns in new, anonymized datasets, thus enabling the re-identification process.
Re-identification	Apply models to anonymized data for matching and re-identification, considering unique ECG characteristics.
Outcome	Successful identification of individuals from anonymized ECG datasets, highlighting the vulnerability to sophisticated techniques.

5.3.2 EHR Data de-anonymization attack:

Table 7 – EHR Data De-anonymization Steps

Step	**Description**
Pattern Association	Use data mining to identify unique EHR patterns like medication history or diagnostic codes.
Cluster Analysis	Separate out the likely target records for re-identification, use clustering algorithms to divide the EHR data into groups according to common health conditions, treatment outcomes, or demographic data (isolating potential target records for re-identification)
Feature Linking	cross-referencing the clustered anonymized EHR data with publicly available health datasets or leaked records to find matching profiles.
Medication-Diagnosis Correlation	Correlate medication prescriptions with diagnosis codes to identify patients.
Hospital Visit Patterns	Examining patterns of hospital or clinic visits that are recorded in the EHR data and contrasting them with data from any accessible datasets (insurance claim databases, health forums) to match with known patient histories for re-identification.
Outcome	Successful re-identification using pattern association, clustering, and feature linking techniques.

5.3.3 Genomic data de-anonymization attack:

Table 8 – Genomic Data De-anonymization Steps

Step	**Description**
Data Collection	Obtain genomic data. Preprocess using genomic data tools.
Feature Extraction	Extract features (e.g., SNPs) using tools like GATK, Samtools. Apply PCA or t-SNE for dimensionality reduction.
Pattern Matching	Use HMMs for identifying disease-related genetic characteristics.
Model and Decision	Train ML models (Random Forest, Gradient Boosting) for pattern recognition. Apply algorithms to unknown profiles for membership detection.
Membership Detection	Employ clustering and beacon queries for genome reconstruction and enhanced detection.
Outcome	Identify specific SNPs, reconstruct genomes, infer identities, detect memberships.

5.3.4 Neuroimaging data de-anonymization:

Table 9 – Neuroimaging Data De-anonymization Steps

Step	Description
Feature Extraction	Preprocess data. Use CNNs to extract discriminative features and apply transfer learning for phenotype prediction (VGGFace for phenotype prediction).
Clustering Analysis	Apply matrix analysis and leverage-score-based sampling for clustering brain signatures.
Matching Attack	Use external data or beacons to match clustered features, evaluate re-identification likelihood.
Outcome	Identify unique neuroimaging features, integrate genetic data from beacons for improved matching.

5.3.5 General Health data de-anonymization (sensors' time series):

Table 10 – General Health Sensor Data De-anonymization Steps

Step	Description
Data Collection	Acquire sensor data (heart rate, steps, sleep patterns) from smart devices (smartwatch in use case). Collect both identified and anonymized datasets.
Feature Extraction	Preprocess data to extract time series features. Use algorithms like PAA (Piecewise Aggregate Approximation) for dimensionality reduction.
Time Series Analysis	Apply time series analysis techniques like Dynamic Time Warping (DTW) for pattern recognition and correlation.
Pattern Matching	Use pattern matching algorithms to match anonymized data with identified datasets. Employ statistical methods to assess similarities.
Model and Decision	Train machine learning models (e.g., Random Forest, XGBoost) to recognize and match patterns between datasets.
Linkage Attack	Implement linkage attacks to connect anonymized records to identified ones based on unique behavioral and physiological patterns.
Outcome	Successfully link anonymized sensor data to identified records, or establish unique identifiers within the anonymized dataset.

5.4 Conclusion

In this Section, we have delved into the challenge of ensuring privacy in the health and GDPR age, specifically focusing on the vulnerabilities inherent in anonymized neuroimaging, genomic, and public health data. This exploration is crucial in the wake of increasing sophistication in data analysis techniques and the potential for misuse.

This section explores the weaknesses in anonymised ECG, neuroimaging, genomic, and public health data, showcasing a variety of datasets and possible attack strategies we aim to implement. The techniques being explored include methods such as CNNs and HMMs. Beacons, which stand in for external data sources, can be helpful for re-identification attempts.

The described attack plans and scenarios highlight the significance of taking preventative action to safeguard individual privacy in the context of health data by illustrating the multiple strategies used by adversaries.

A further contribution of this research, beyond engaging in direct attacks, is the analysis of probabilistic models that estimate the risk of re-identification of a single individual in anonymized datasets.

6 Private synthetic data generation techniques

Synthetic data generation aims to provide data that is statistically similar to the original data but does not convey information from the original data. These techniques can produce data that is similar to the original, but not completely equal to the original data which is promising as it would be possible to produce new data that could be shared without providing any sensitive information from the original patients. However this is on-going debated in the current state of the art as this data still needs to be checked regarding information leakage [36, 37]. In this sense, there must be a balance between usability of the data and privacy and, depending on the modeling techniques, the generated data might not be useful when the privacy is preserved [38]. This balance is a challenge that we will be addressing in the SECURED project. On the other hand, synthetic data generation can also be useful aside from the privacy preserving aspect, as it can be used to complement real data for tasks such as training *Machine Learning* models, specially *Deep Learning models*, which require as much data as possible to provide good and generalizable results.

As basis of this service we use the open source library MediGAN [39], which provides Deep Learning based models to generate different health image modalities from different studies and datasets. Moreover, we are still considering SVD [40] and Gretel [41] libraries for other data types such as EHR.

Going further, under this tasks we aim to offer a service that is able to generate the following data types:

- Mammographies
- Breast and Colorectal tissue
- Lung X-ray
- Fetal heartbeat series
- Respiratory series
- Missing slices from breast MRI

We aim to evaluate the first two image modalities within the Use Case 3 (Synthetic Data Generation for Education) with healthcare professionals from the Semmelweis University. These two types of data were prioritized as they were marked by the use case leaders as required for their case.

Given that there are no libraries yet to generate some of the types like the tissue data, we plan to develop new methodologies to produce that kind of data. Following this line, the work performed under SECURED for private synthetic data generation is directed in three different axes:

- Privacy: Provide data that does not leak any sensitive information and cannot be used to match the original data.
- Improve generation: Advance in the state of the art by improving the quality of the generated data (i.e., utility for the use case, for example improving the accuracy of a model by augmenting the dataset) or new data types that have not been completely explored.
- Functionality: Provide tools to the users that address specific requirements aside from general generation.

In the following subsections, these three axes will be explored with the following research lines (ordered by current maturity):

- Privacy: Generation of data with Differential Privacy
- Improval: Cancer tissue generation
- Improval: Healthcare time series generation
- Functionality: Generation parametrization
- Functionality: Missing MRI scan slice generation

6.1 Differential Privacy applied in generative models

Generative Adversarial Network (GAN) and its variants are a popular tool to generate data, e.g. images, time series, etc. However, one common issue in GAN is that the density of the learned generative distribution could concentrate on the training data points, meaning that they can easily remember training samples, in particular due to the high model complexity of deep networks. Moreover, one common default of GAN is mode collapse, when the generator generates data samples that are very similar or even identical.

It is a major concern when GAN are applied to private and/or sensitive data such as medical xray images. Indeed, the concentration of distribution can reveal critical patient information.

There are several interests to consider GAN when considering Privacy issues. In a GAN, only discriminator has access to sensivite data and only the generator is useful for the final users. Privacy can be introduced on the discriminator and propagated to the generator thanks to nice properties of, among others, Differential Privacy, explained in the Section 6.1.1. It allows to explore some Teacher-Students approaches as the ones proposed in PATE-GAN [42], G-PATE [43] and GS-PATE [44].

6.1.1 Explored techniques

6.1.1.1 Differential Privacy and Rényi Differential Privacy

Differential privacy guarantees individual privacy in statistical databases. Intuitively, it corresponds to ensuring that the output distribution of an algorithm won't be significantly different considering the presence or absence of one particular individual. An adversary with access to the algorithm won't be able to learn about individuals, but will only have access to the global knowledge of the algorithm among them, hence privacy is protected. The Privacy budget is given by (ε, δ), which are illustrated in the Figure 5. ε is the equivalent of the bounded ratio in Figure 5. It gives the albility to the algorithm to produce similar outputs distribution if the algorithm has been trained with or without one instance. δ corresponds to a probability that the privacy protection fails independently of the data considered. DP schemes are very popular because of the nice properties it allows in the algorithm verifying DP, with among others:

- Composition property insures that we can apply several times a DP algorithm while being still differentially private;
- Post-processing insures that a function can be applied to the outputs of a differentially private algorithm while preserving the differential privacy;
- Robustness to auxiliaries features insures that whatever the data the attacker has, it does not create new risk.

When $\delta = 0$, the DP guarantee asserts that the probability of observing a bad outcome will not change (either way) by more than a factor of e^{ϵ} whether anyone's record is part of the input or not (for appropriately defined adjacent inputs). Rényi differential privacy is an relaxation of *pure* ε-differential privacy. For the detail of the meaning of the relaxation, the reader can refer to [45].

The *simpler* method to add differential privacy into a classic network training via an empirical minimization of loss functions is the Differentially Private Stochastic Gradient Descent (DP-SGD) [46]. Indeed, this method keeps the classic training process of a network, and ensures the differential privacy by adding noise and clipping the gradients, i.e. by insuring that the norm of the gradients are bounded by the clip value, before the optimization step. First this principle for gradient clipping was used to improve deep learning based model robustness. Here, the idea is then to *obfuscate* the influence of each individual of the training dataset, which is exactly the idea behind the definition of differential privacy seen previously. Given a clipping bound on gradients and a noise distribution (in [46] they choose a Gaussian noise with variance proportional to the clipping bound), the main training scheme is then, for each batch of samples :

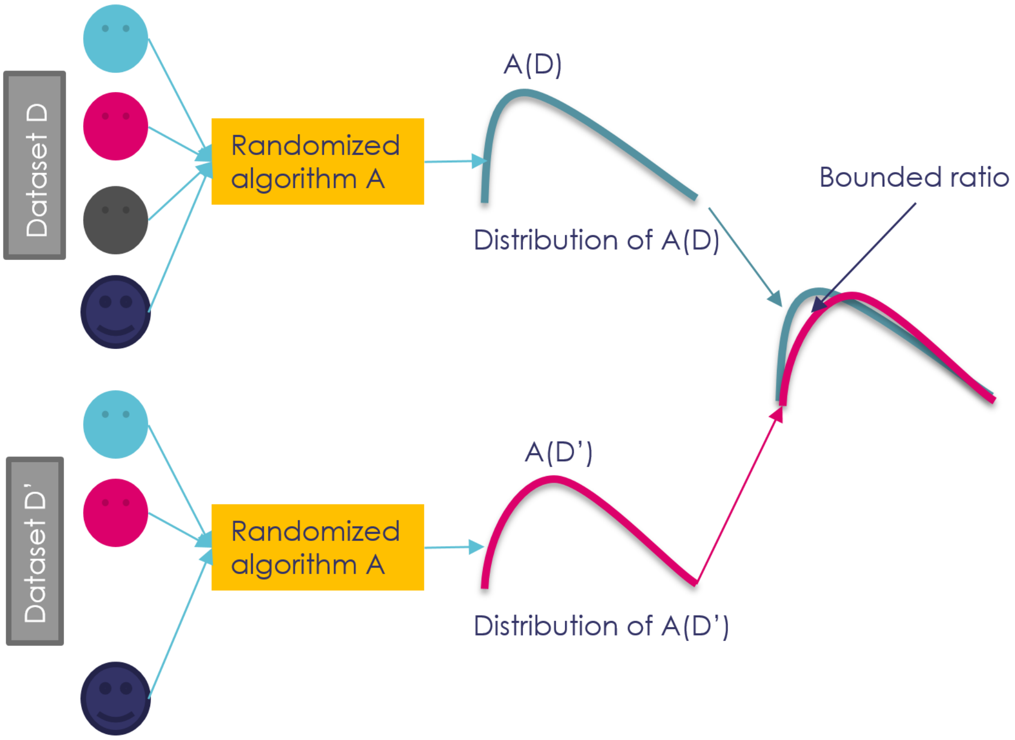

Figure 5 – Illustration of Differential Privacy.

1. Compute per-sample empirical loss and corresponding gradient;
2. Clip per-sample gradients to the minimum between the clipping value and their l2 norm;
3. Reduce gradients by computing the mean over samples;
4. Add noise to the reduced gradients;
5. Apply noisy reduced gradients through an optimization step.

6.1.1.2 Generative Adversarial Networks (GAN) and variants

In the GAN setting, the mapping between the data distribution and the chosen latent space (Uniform or Gaussian noise) is modeled by a neural network, called generator. The main idea of GAN [47] is to introduce a second neural network, the discriminator, only used during the training to do a competitive training. The discriminator is a classifier trained to distinguish real and generated (fake) samples.

The adversarial training then consists in training the discriminator to be able to distinguish the output of the generator (fake images) from the ones from the training dataset (real ones), while on the contrary training the generator to fool the discriminator.

Note that, in practice, such training is really unstable, precisely due to this adversarial formulation: for instance, the generator's training requires to have a sufficiently efficient discriminator to avoid gradient vanishing.

To make a link useful for interpretation, we can cite the following work expressing the link between GAN training and divergence minimization [48]. We intuitively want to have a generator able to model a distribution as close as possible from the real one of the data distribution.

GAN training is often unstable. To limit this instability, some GAN variants have been proposed. For example, Wasserstein-GAN (WGAN) [49] use Wasserstein distance that induces a weaker topology and so makes it easier for a sequence of distribution to converge, which will then ease the unstable training of GANs.

Another alternative can been used when learning a generative model from labeled data. In this context, we may want to control the modes of the generated data. For this purpose, GANs can be extended to a conditional formulation (cGAN) [50], where both the generator and the discriminator are given some extra information. It will be category labels in our case, but can also be some external features for instance.

6.1.1.3 Differentially Private GAN (DP-GAN)

When trying to incorporate differential privacy into a variant of GAN model, a first *simplest* approach is to directly use the DP-SGD method into the training. In GAN training, there are two adversarial models : the idea of Differentially Private GAN (DP-GAN) [51] [52] is to ensure differential privacy via the discriminator, applying the DP-SGD procedure during its updates.

Indeed, according to the post-processing properties, ensuring the differential privacy of the discriminator will ensure the differential privacy of the generator, which is trained on its outputs. In [51] and [52], the choice of applying DP-SGD to the discriminator instead of the generator is justified by an easiest computation of the privacy loss, the discriminator's architecture being often less complex (as its goal is to classify and not generate realistic samples), and it's the only one to access real sensitive data.

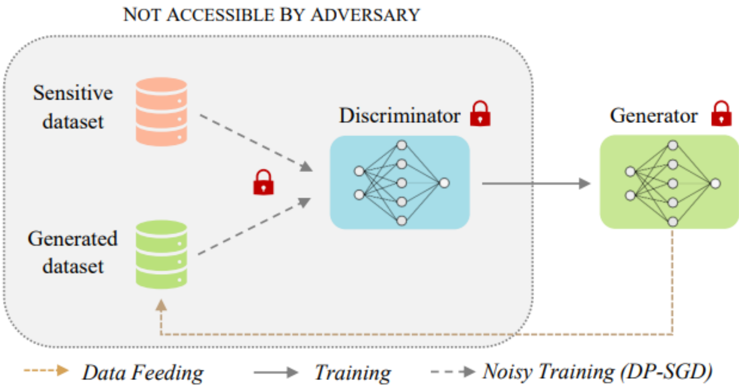

Figure 6 – Scheme of DP-GAN architecture. Differential privacy is added during the discriminator's update by a noisy gradient descent, using DP-SGD method. Thanks to the post-processing properties, generator privacy is ensured too.

6.1.1.4 Preliminary results: application on MNIST

MNIST is a labeled dataset containing grayscale images of handwritten digits, from 0 to 9. It then contains 10 classes, which are well balanced, and can be easily applied to classification tasks. Its images are of low dimension and so resolution, with shapes $28 \times 28 \times 1$. It was one of the earlier wide dataset and is still popular for benchmarks. The training and test sets are respectively composed of $60\,000$ and $10\,000$ images. We do not apply data augmentation or pre-processing, except standardizing the images, so that the pixels have values in $[-1, 1]$. We use a `tanh` last activation in the generator to respect this range of value.

We obtain images as shown in Figure 7. We consider 6 different privacy budgets. Stronger privacy budget means less privacy. In Figure 7, we provide some examples of images generated by generator respecting the privacy budget given. These examples illustrate the trade-off between privacy level and image quality.

Tensorflow has been used to implement the first results. The results are still experimental.

6.1.2 Status and future work

In Table 11, we provide the status of work on the different variants of GAN with differential privacy. This part of the task 2.3 will focus on the impact of the addition of differential privacy in the generator and future work will depend of the data and the potential existing generators that the use case provider can provide.

We will first use classical GAN architecture on open data avaible about xray, and compare the quality of data generated when introducing DP with metrics like Frechet Inception Distance [53] and its variants and Inception

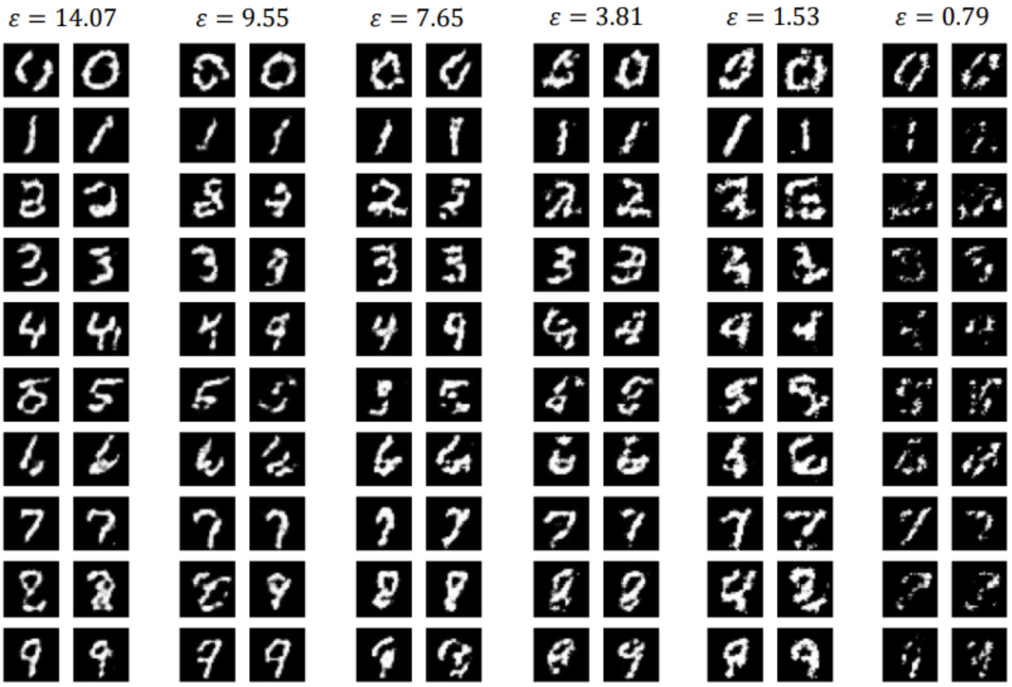

Figure 7 – Overview of the generated images obtained with our DP-GAN models on MNIST with their corresponding DP guarantees.

Main techniques explored	DP-GAN, DP-WGAN and DP-cGAN
Existing or new method	Existing methods
Relation with use case	• Collaboration under the task with Use Case 3; • Potential initial data type chest xrays data generation; • Initiate experiment on chest Open Xray that are avaible (see D4.1); • Potential improvement of security of other generators by the different partners of T2.3;
Progress status	Preliminary results

Table 11 – Differential Privacy for GAN status.

Score [54] and its variants. Moreover, when it is applicable and its exists some labeled images, we can evaluate the images generated utility by evaluating a classifier trained in three differents settings and comparing the loss of performance of the classifier:

- **Setting A**: Train on real data, Tested on real data;
- **Setting B**: Train on fake data, Tested on fake data;
- **Setting A**: Train on fake data, Tested on real data.

Moreover, evaluate the good privacy budgets is a tricky part. We propose to do it by using empricial inspection:

- **Nearest Neighbors**: We want our generator to be able to generate realistic images, which are new samples and not just copies of the training data. As an experiment to detect over-fitting, we can generate some samples, and look for its nearest neighbors in the real data used for the generator's training. We expect to find similar images but to see the ability of the generator to create new content;
- **Interpolations in the latent space**: try to understand the content learned by the generator via the space continuity of the generated images. An unwanted behaviour would be a generator memorizing a distinct

number of generated images, without any consistency in the latent space.

In both case, we study the impact of the introduction of DP. It is important to denote that the focus of this task is to introduce DP in data generators. As no efficient non private data generator is yet available, potentially the data generated will be of low quality.

We will follow the work plan below:

1. Select and collect Open Xray data with three potential sources:
   - **ChestX-ray14**[8]: medical imaging dataset which comprises 112,120 frontal-view X-ray images of 30,805 (collected from the year of 1992 to 2015) unique patients with the text-mined fourteen common disease labels, mined from the text radiological reports via NLP techniques;
   - **NODE21**[9]: consists of frontal chest radiographs with annotated bounding boxes around nodules. It consists of 4882 frontal chest radiographs where 1134 CXR images (1476 nodules) are annotated with bounding boxes around nodules and the remaining 3748 images are free of nodules hence represent the negative class;
   - **CheXpert**[10]: a large dataset of chest X-rays and competition for automated chest x-ray interpretation, which features uncertainty labels and radiologist-labeled reference standard evaluation sets.
2. Implementation of quality metrics and differential private generators;
3. Train, test and compare non private and private data generators;
4. Ease the potential reusability of the code;
5. Access to data and potentially existing generators (mandatory inputs from the use case provider);
6. Train, test and compare non private and private data generators on use case 3 provider data (if available);
7. (optional) Explore more complex GAN based architecture.

In this work plan, we focus on chest xray data generation. Morever, according the interest of the others partners to secure their generator, we would collaborate to improve privacy of their generators.

6.2 Cancer tissue generation

The ultimate goal is to process a given collection of histological images in order to generate new patterns that are not only pathologically accurate but also of high-quality. These generated images aim to be meticulously observed for relevant details and lesions by medical professionals and students. The objective is to offer readily available synthetic images for education and data augmentation purposes.

To this end, we developed a diffusion model for synthesizing histopathologic scans. In this initial stage, our primary focus is on accurate data generation without strict privacy guarantees. As the project progresses, we intend to enhance the accuracy of our technique through evaluation by medical experts (pathologists) and implement more formal privacy guarantees described in Section 6.1.

6.2.1 Background: Diffusion models

Diffusion models are similar to variational autoencoders [12]. First, the data set is unified, i.e. the images are transformed to have the same dimensions. The diffusion process then consists of two phases, called forward and reverse diffusion. In the training process, these two phases are performed on the input, and then a random noise with a standard normal distribution is given to the model from which it generates an image.

[8] https://paperswithcode.com/dataset/chestx-ray14

[9] https://node21.grand-challenge.org/Data/

[10] https://stanfordmlgroup.github.io/competitions/chexpert/

6.2.1.1 Forward diffusion

In the training dataset, noise is progressively added to the images until we achieve an image with a standard normal distribution. At this stage, a critical task is determining the optimal number and size of the iteration steps, as these factors significantly impact the model's outcomes. Choosing a higher number of steps comes with the trade-off of longer learning times and increased computational demands. Specifying the number of steps, or the amount of noise added in each step, is not straightforward. A kind of scheduling is employed where we don't follow a linear progression, which adds the same amount of noise at each step, but rather utilize more complex functions to shape this value. Fortunately, the distribution of our final noisy image can be computed from the initial image in a closed form. Therefore, operation does not introduce substantial computational complexity.

6.2.1.2 Reverse diffusion

The goal is to recover the original image from the noise generated through forward diffusion. This process is far from straightforward, so specific U-Nets are employed, which are specialized neural networks. A U-Net [55] is a type of neural network that, when given an image as input, first reduces its dimensions in multiple steps while increasing the channel number[11] It then passes the image through the middle block and finally brings it back to the original dimensions and channel number through upsampling. Our U-Net architecture is shown in Figure 8.

To tackle the challenge of recovering the original image from the noise, the noisy image produced by the forward diffusion undergoes multiple iterations through this U-Net. This iterative process gradually removes the noise, resulting in the complete generated image. The neural network predicts the expected value of the normal noise, which is then subtracted from the image with a fixed variance. The resulting image is fed into the next iteration.

6.2.2 Dataset

We used the publicly available PatchCamelyon Dataset [7] which consists of 327.680 color images (96 x 96px) extracted from histopathologic scans of lymph node sections in the Camelyon16 Dataset [56].

6.2.3 Diffusion model to synthesize histopathologic scans

The model incorporates a more complex U-Net structure based on [55] and illustrated in Figure 8, to predict the expected noise value during reverse diffusion. Both the Downsample and Upsample phases involve various layers such as Conv2d, ConvTranspose2d, Batchnorm2d, and Groupnorm2d, along with Swish activation functions. A Linear layer manages the sinusoidal embedding.

Additionally, in our advanced model, we have the option to include an Attention layer in each of these blocks. This layer is borrowed from the Transformer architecture [57].

6.2.3.1 Architecture

The U-Net architecture contains 4 downsampling and 4 upsampling Residual Blocks which all are composed of:

- A Convolutional layer which is responsible for upscaling/downscaling the channel numbers,
- A Time Embedding layer responsible for embedding the time information for the U-Net to know which iteration the loss is calculated for,

[11] Images are often organized into channels, where each channel represents a particular aspect of the image. For example, in a color image, the three primary channels are often red, green, and blue (RGB), where each channel carries information about the intensity of that color at each pixel. In grayscale images, there is typically only one channel representing the intensity of light.

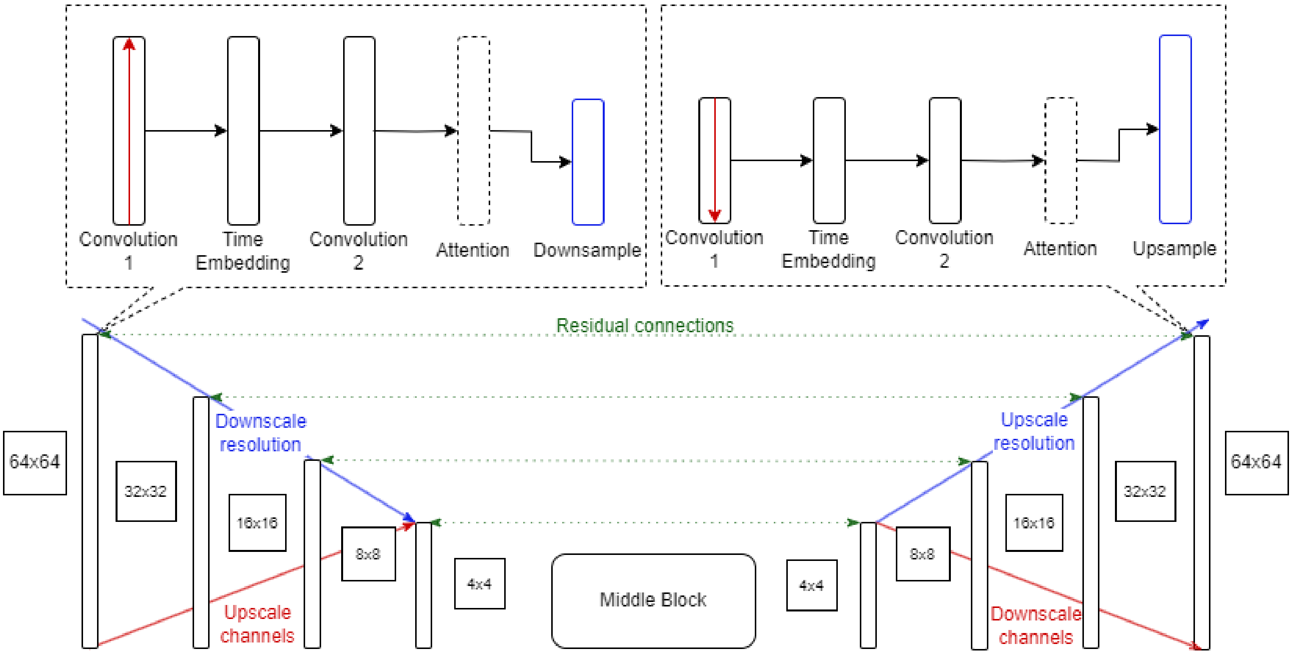

Figure 8 – Structure of our U-Net

- Another Convolutional layer,
- An optional Attention layer,
- A final layer for downsampling/upsampling the resolution of the images to the required form.

The channel numbers are scaled from 3 to 64, 128, 256 and 1024 in the Downsampling phase (and from 1024 to 256, 128, 64 and finally 3 in the Upsampling phase). The image resolutions are scaled from 96x96 to 48x48, 24x24, 12x12 and 6x6 in the Downsampling phase (and from 6x6 to 12x12, 24x24, 48x48 and finally 96x96 in the Upsampling phase. The optional Attention layers were only used in the 3rd and 4th blocks of the Downsampling phase and in the corresponding blocks (1st and 2nd) in the Upsampling phase.

6.2.3.2 Parameters

The number of steps T, i.e. the number of iteration steps in forward and reverse diffusion in which the image is decomposed into noise and back, is set to 1000. The input dimension of the images on which the model learns is 96x96. The batch size is chosen depending on the size of the dataset and the available video memory.

6.2.3.3 Testing

Though expert validation is essential to assess the fidelity of the generated images for pathological usage, the visual inspection by a layperson already reveals a high degree of similarity between the model-generated images and real images, as illustrated in Figure 9.

Moreover, we plan to calculate the FID score [58], a metric commonly used in generative models working with images, between a test dataset and a set of images generated by the model. The current FID value of 81.16 is not conclusive at this stage. It's crucial to note that the dataset preprocessing is currently in a rudimentary form, leading to many images in both the training and generated datasets that do not adequately represent histological patterns.

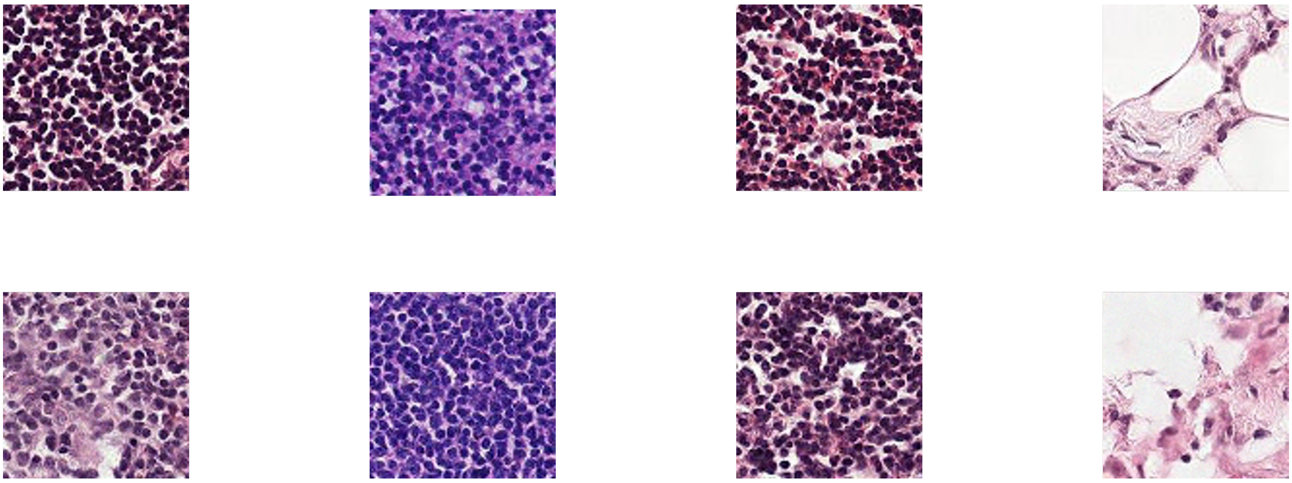

Figure 9 – Real samples in the top row and the synthetic samples generated by our model in the bottom row.

6.2.4 Next steps

Although the results with diffusion models are promising, there are still several unresolved issues. First, generating complete histopathologic scans appears to be very costly in terms of computation time, if not impractical. The current solution involves generating patches of the whole scan; however, pathologists typically work with complete, high-resolution scans. Second, the fidelity and practical usefulness of the generated samples remain unclear without expert evaluation. Lastly, generating a large number of patches of the whole scan (and not even complete scans) is still very costly and may be impractical with the commodity hardware that an average medical university tends to have.

In the upcoming months, our efforts will focus on evaluating vision transformers (ViT) to generate complete scans. The idea behind ViT models is that they generate larger images from smaller patches. We believe this approach should be more scalable and capable of generating images with better quality if the patches also exhibit high fidelity.

6.3 Fetal heartbeat time series generation

Cardiotocography (CTG) is a technique used to record the fetal heartbeat and uterine contractions during pregnancy. The device employed in CTG is known as a cardiotocograph. It involves placing two transducers onto the abdomen of a pregnant woman. One transducer records the fetal heart rate using ultrasound, while the other monitors uterine contractions by measuring the tension of the maternal abdominal wall, providing an indirect indication of intrauterine pressure. These transducers record two time series in parallel: one for fetal heartbeat and the other for uterine contractions over time. The recorded time series are then utilized to determine whether the pregnancy is categorized as high or low risk. This distinction is crucial, as it adds context to the CTG reading. For instance, if the pregnancy is classified as high-risk, the threshold for intervention may be lower.

Educational organizations often lack a sufficient amount of data to train medical personnel. Our goal is to perform data augmentation by generating synthetic time series of fetal heart rate and uterine contractions. As a first step, we are investigating autoregressive methods to generate the next value of the time series given the previous k measurements. The predicted value is then added to the generated time series, and this process is repeated until a sufficient number of consecutive measurements are generated to form a complete time series. Due to the complexity of the heart rate signal and the inherent noise in the measurement process, we have opted for a transformer-based model to iteratively predict each consecutive measurement of a synthetic time series. In general, any sequential models are applicable, but transformers have shown a great progress recently in the domain if Natural Language Processing which is also a type of sequential data.

6.3.1 An autoregressive approach based on discretization

An approach is investigated where all time series undergo initial discretization (quantization), followed by training a generative sequential model on these discretized data to produce discrete time-series. The generated sequences are then converted back to the continuous domain. The process begins by dividing all time-series into non-overlapping fixed-size subsequences. All possible subsequences are subsequently clustered into K clusters, with each cluster member represented by a single token. The subsequences are then mapped to their respective tokens, creating a token sequence or the discretized time-series. The generative model is trained on these tokenized time-series, and the trained model can generate tokenized time-series. These generated sequences are transformed back to the continuous domain by replacing each generated token with its corresponding clusterhead.

The underlying rationale is that discretization helps mitigate the exploding and vanishing gradients problem arising from extremely small and large values in the dataset. A smaller pool of possible values may also aid the model in generalizing better. The process introduces two sources of error: (1) the *representation (or tokenization) error*, resulting from mapping a token to the corresponding clusterhead, which imperfectly approximates the original subsequence, and (2) *the prediction error* of the generative model due to incorrectly predicting the next token. The total error is the sum of these two errors:

$$E_{total} = E_{token} + E_{model} \tag{1}$$

The tokenization error E_{token} increases when the alphabet size K decreases, while E_{model} decreases since the model may generalize better with a smaller pool of possible values. For instance, if a single cluster/token represents every possible subsequence $(K = 1)$, the tokenization error is maximal, but the model's prediction accuracy is perfect $(E_{model} = 0)$. Conversely, if there is a unique discrete value for every possible value in the dataset (i.e., K equals the number of unique subsequences), E_{token} is 0, while E_{model} is maximal, equivalent to a model without tokenization. The goal is to find the optimal trade-off between the tokenization error and the model's prediction error.

The tokenization should be optimized to enhance the model's performance while minimizing the representation error. To formalize this problem, the following equation needs to be addressed:

$$\mathcal{C}^* = \underset{\mathcal{C}}{\text{argmin}} \left(E_{token}^{M_\mathcal{C}} + \beta \cdot E_{model}^{M_\mathcal{C}} \right)$$

Here, $\mathcal{C}$ represents the tokenization function that maps a subsequence to a token, $M_\mathcal{C}$ denotes the model that takes the tokens produced by $\mathcal{C}$ as input, and β is a trade-off parameter. The objective is to find the optimal mapping $\mathcal{C}^*$ that minimizes the weighted sum of the tokenization error $E_{token}^{M_\mathcal{C}}$ and the model prediction error $E_{model}^{M_\mathcal{C}}$. While $\mathcal{C}^*$ could be determined by training a model and iteratively adjusting the tokenization, this approach may become prohibitively expensive depending on the model's complexity.

6.3.2 Next steps

Our preliminary results are not promising. Our current models are unable to generate synthetic data with high fidelity. One artifact of our current approach is that it starts to generate a constant signal after a few initial steps. Hence, our next step is to incorporate aggregated information about the entire time series when predicting the next value. One option is to first generate a coarser-grained representation of the entire time series and then use an autoregressive method to generate finer-level details. In the next few months, we will be investigating various standard time series representations, such as Fourier and wavelet transformations, that can capture high-level information like trends and periodicity more effectively. Subsequently, we plan to combine this synthetic data with a finer-grained representation using autoregressive or directed diffusion models

6.4 Generation parameterization: Conditioning mammography generation

Generation of mammographies is an already present in the State-of-the-art. For example, MediGAN [39] is able to generate mammographies in two projections as the ones seen in Figure 10: craniocaudal view (CC view) and mediolateral oblique view (MLO view). However, the images are generated completely at random. Therefore, to obtain an image with the desired characteristics the user would need to execute several times this generation process until the desired image would be generated. As we have seen, there is a gap to cover on the paremetrization/conditioning of the generation. This is specially interesting for Use Case 3, *Synthetic data generation for education*, as the lecturer might require the images to have a particular set of characteristics.

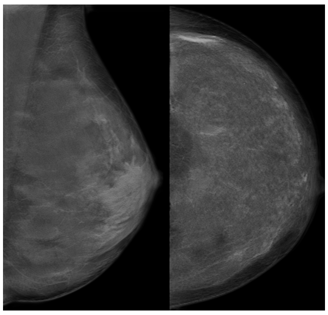

Figure 10 – Two different mammographies generated with MediGAN. Left image is an mediolateral oblique (MLO) view and the right image is a Craniocaudal (CC) view. Generated with the models trained with CSAW [4]

Given that, the problem to address would be defined by the following question: Is it possible to create a good enough model that generates high quality mammographies and that provides input parameters to modify the characteristics of the generated image?

6.4.1 Approach

Towards this end, we are exploring Conditional Generative Adversarial Networks (CGAN)[59], which takes as input the random noise for the generation, as a regular GAN does, plus attributes that condition this generation. In this way, both the image and tabular information are used as input of the models. By doing so the latent space of the generated images improves its separability with the provided information, as shown in Figure 11.

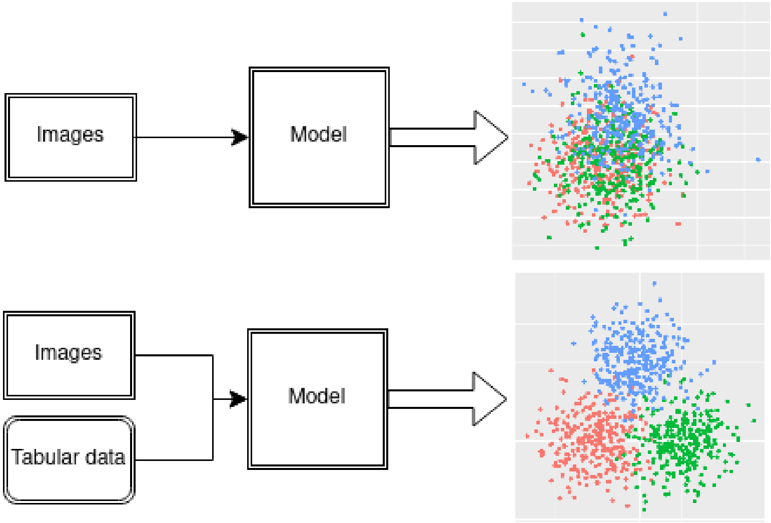

Figure 11 – Effect of conditioning the model on other information over a given mammography class, e.g. BI-RADS for scoring the mammographies to classify the status of the breast. When including it as part of the generation, the separation between classes is forced and, therefore, the generation of a given class is possible.

Currently, we have explored the generation capabilities of MediGAN and also explored which metrics are relevant for the problem. Moreover, we have obtained part of the datasets obtained in Section 3 and studied their properties. Figure 10 depicts two mammograms generated with this framework. In particular, we have found that metrics are usually divided in three different categories: performance metrics, security metrics and utility metrics. This idea is also shown in the work performed in the deliverable D4.1.

6.4.2 Next steps

Currently, our progress in this task is in a preliminary stage. Following the research performed our plan is summarised in the following steps:

- Define a subset of metrics appropriate for the problem. The subset will include privacy and accuracy metrics.
- Explore the potential of CGANs along with other similar neural network architectures to condition the generation
- Develop a model from the explored architectures with open data as an initial baseline
- Collaborate with Use Case 3 to define which are the relevant characteristics to use in the conditioning, e.g. age or BI-RADS score.

6.5 Missing MRI scan slice generation

The objective is the reconstruction of a missing slice within an MRI scan sequence. Breast and brain are the two candidates for this application. It is a challenging task for both cases, considering the expected accuracy and precision of the generated image. Thus, state-of-the-art approaches will be employed in an attempt to achieve the goal. Diffusion models will constitute the basis of the final mechanism, while other techniques, like tensor-based learning and GANs (Generative Adversarial Networks), could assist towards the common goal.

6.5.1 The Datasets

At least for the initial steps, open datasets will be preferably utilized in an aim to speed up the process of development. For the breast case, Duke-Breast-Cancer-MRI dataset [60] seems to be suitable, containing lot of MRI scans with plenty of additional information along with them. Accordingly, the BRATS 2018 dataset [61] serves as an optimal starting point for applying the developed algorithms. Of course, throughout the project, datasets from the collaborating partners will be utilized when they are available. A proper combination of the open datasets and the provided ones could, potentially, provide a more robust solution. The latter will come from the corresponding use cases of the project. Though, since the collection and the curation of data is an ongoing process, limitation could occur that are related to the availability or the usability of data. Aiming to tackle these issues, open datasets will be utilized at first place, in order to develop the Machine Learning models and prepare the corresponding pipelines.

6.5.2 The approach

During the forward pass of the proposed diffusion model, different noisy versions of the input MRI slice are generated at specified time steps. Following the backward pass, those generated, noisy images, along with the corresponding step, are given as input to a UNet that provides the estimated noise as output. Through the training process, the UNet takes into account, not only the noisy image (MRI slice) and the time step, but also information that comes from the original adjacent MRI slices. Reasonable loss function seems to be the Mean Squared Error (MSE) between the chosen noise and the predicted one. Except from the metrics that reveal the similarity of the generated slice to the original one, the result will be examined by proper medical experts as well, in order to determine the effectiveness of the proposed pipeline.

6.5.3 Next steps

Currently, our progress is at preliminary stage; a) we are researching similar challenges in the academic fields and, at the same time, b) we are exploring the available software libraries the are related to image generation and reconstruction. Our workplan could be summarized in the following steps.

- Explore similar challenges and related state-of-the-art of the art solutions,
- Design an end-to-end pipeline, meaning the data pre-processing stage, the training procedure the inference mechanism and the extraction of evaluations metrics,
- Use few data (probably open data) for ensuring the proper functionality of the developed pipeline,
- Prepare sets of MRI scans for creating training, evaluation and testing procedures,
- Train the model and evaluate its performance,
- Make proper adjustments and try alternative schemes (like pretrained models, transfer learning, etc.) if needed,
- Evaluate the performance and enrich the model knowledge, when data from the related use cases is available,
- Build a common pre-processing stage for all the available data, and
- Deliver the integrated pipeline as a standalone service.

7 Private data anonymization library

The SECURED project implements the novel techniques for dataset anonymization and privacy-preserving synthetic data generation through different tools and services in the SECURED Federation Infrastructure. An overview of the complete infrastructure is depicted in Figure 12. The focus of WP2, and the progress so far as reported in this deliverable, is on the development of the internal libraries that implement the necessary algorithms. Collectively, we refer to these implementations as *the library*[12]. The development of the high-level user-facing interfaces through which users interact, such as the SECURED Innohub, are not discussed in this deliverable. These efforts are covered in WP4. However, care must be taken that the design choices reported in this deliverable do not contradict the requirements of the Innohub.

This section is organized as follows. Subsection 7.1 discusses the components in the SECURED architecture that are relevant for this deliverable, listing the primary goals and expected input and output. Then, Subsection 7.2 reports on the guidelines we have established for the software development of the components. Finally, Subsection 7.3 argues that our development choices enable the scaling-up approach that needs to be applied on all components and describes how this can be performed in a general way.

7.1 Relevant components in the SECURED architecture

In this subsection, we go more in-depth on the specific components in the SECURED Federation Infrastructure, as depicted in Figure 12, that relate to dataset anonymization and privacy-preserving synthetic data generation. For each component, we list its primary goal, with which other SECURED components it interacts, and the expected formats of the input and output data.

7.1.1 Data Anonymization Toolset (DANS)

As discussed in Section 4, the Data Anonymization Toolset (DANS) has as its primary goal the implementation of algorithms to transform datasets to mitigate the risk of re-identification as much as possible. DANS is part of the **Innohub**, which means that the tool can be downloaded separately by the users and executed on premise. It is also part of the **Data Transformation Engine**, which is offered as a SECURED cloud-service to assess anonymity of datasets and automatically anonymize them, if deemed necessary. It also contains the **Anonymizing Service**, which implements the main anonymization algorithms. The user, so the medical practitioner, will also interact with the **Anonymization Decision Support**, which decides on the algorithm(s) to be used for a given dataset.

Input: The following use-cases will submit input datasets to DANS. The actual definition of the health data types listed below is outlined in Section 3.

- **UC-2**, telemonitoring for children, consists of datasets involving medical time series data, such as ECG, heart rate, oxygen saturation and respiratory state. For this use-case, the datasets are only going to be processed by DANS if re-identification was assessed to be too easy.

- **UC-3**, synthetic data generation for education, needs to have input datasets to be anonymized before being used for synthetic data generation. There are three main categories of data: Imaging (e.g. mammography images and MRI scans), time series (e.g. ECG) and Electronic Health Record (EHR) with text and tabular data.

- **UC-4**, genomic data, might want to anonymize the results of the genomic analysis so they can be shared with the scientific community in a privacy-preserving manner.

[12] The same applies to the library developed by WP3, which is discussed in parallel in Deliverable D3.1. We have not chosen for different names, because the actual development is done separately for each component in the SECURED architecture, rather than each WP creating a grand overarching library.

Output: The output of DANS consists of the anonymized datasets and the algorithms and other relevant parameters used to perform the anonymization.

7.1.2 De-Anonymization/Re-Identification (DEAN)

As discussed in Section 5, the De-Anonymization/Re-Identification (DEAN) component is responsible for analyzing and performing novel re-identification attacks to assess whether previously applied anonymization techniques succeeded. DEAN is part of the **Data Transformation Engine**, which is offered as a SECURED cloud-service to assess anonymity of datasets and automatically anonymize them, if deemed necessary. It contains the **Anonymization Assessment Service**, which actually performs the assessment.

Input: The following use-cases will submit input datasets to DEAN. The actual definition of the health data types listed below are outlined in Section 3.

- **UC-2**, telemonitoring for children, needs to have the anonymization being assessed by DEAN.
- **UC-3**, synthetic data generation for education, needs to have both the input dataset for and the output of the generation to be properly anonymized. As such, DEAN needs to perform the assessment on input and output datasets.
- **UC-4**, genomic data, needs to have an anonymization assessment by DEAN on datasets to be released to the scientific community.

Output: The output of DEAN consists of a report containing the scores for all metrics that are relevant for the re-identification and that have been computed for the given dataset.

7.1.3 Synthetic Data Generation (SDG)

As discussed in Section 6, the Synthetic Data Generation (SDG) has as its main purpose to generate synthetic data with similar information-statistical contribution and extending existing datasets with high-fidelity data of wide diversity. SDG is part of the **Innohub**, which means that the tool can be downloaded separately by the users and executed on premise. It is part of the **Synthetic Data Generator** and contains the **Synthesis Engine**.

Input: The following use-cases will submit input datasets to SDG. The actual definition of the health data types listed below are outlined in Section 3.

- **UC-2**, telemonitoring for children, might need to increase the size of their datasets by making SDG generating synthetic data.
- **UC-3**, synthetic data generation for education, already has SDG in its name.

Output: The output of SDG consists of the generated dataset in the same format as the input dataset, but then with synthetic data.

7.2 Software development

The source code of the implementations of the three aforementioned components are managed with the Git Version Control System (VCS)[13]. Each component is assigned a Git repository of which the hosting is managed by the UvA. All repositories are made accessible to all partners in the SECURED project and only verified project members can commit code and approve merge requests.

[13] https://git-scm.com/

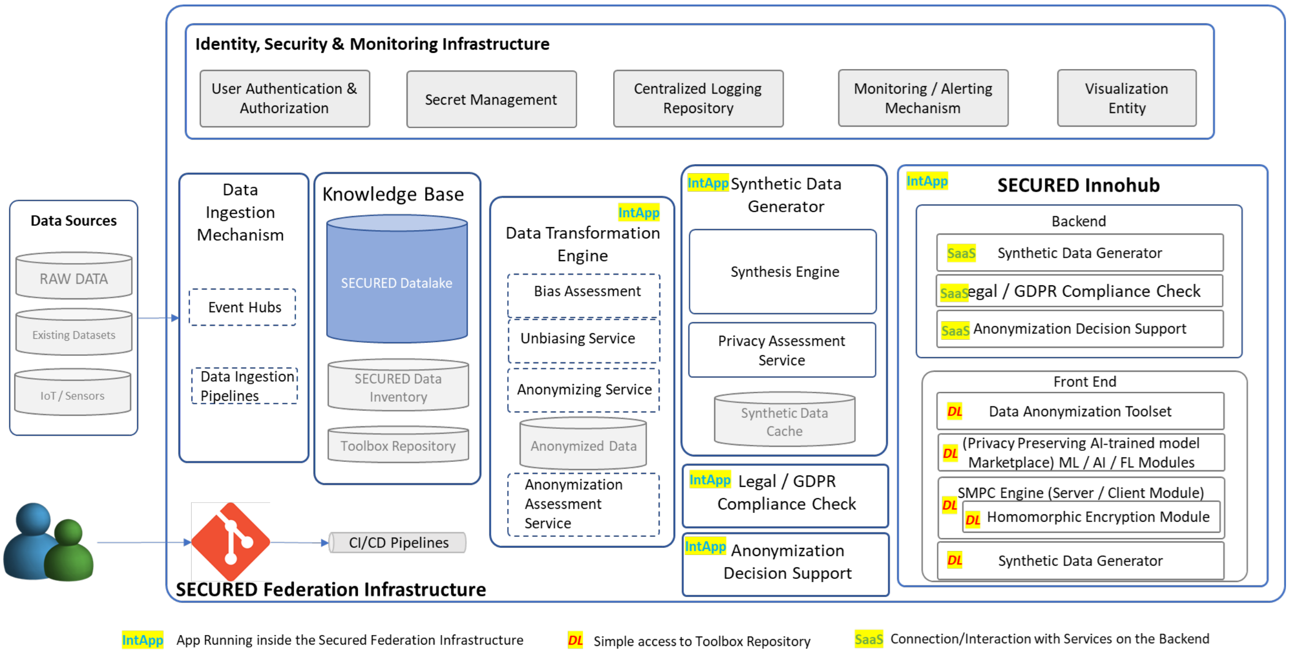
Identity, Security & Monitoring Infrastructure
User Authentication & Authorization
Secret Management
Centralized Logging Repository
Monitoring / Alerting Mechanism
Visualization Entity
Data Sources
RAW DATA
Existing Datasets
IoT / Sensors
Data Ingestion Mechanism
Event Hubs
Data Ingestion Pipelines
Knowledge Base
SECURED Datalake
SECURED Data Inventory
Toolbox Repository
IntApp
Data Transformation Engine
Bias Assessment
Unbiasing Service
Anonymizing Service
Anonymized Data
Anonymization Assessment Service
IntApp Synthetic Data Generator
Synthesis Engine
Privacy Assessment Service
Synthetic Data Cache
IntApp Legal / GDPR Compliance Check
IntApp Anonymization Decision Support
IntApp
SECURED Innohub
Backend
SaaS Synthetic Data Generator
SaaS Legal / GDPR Compliance Check
SaaS Anonymization Decision Support
Front End
DL Data Anonymization Toolset
DL (Privacy Preserving AI-trained model Marketplace) ML / AI / FL Modules
DL SMPC Engine (Server / Client Module)
DL Homomorphic Encryption Module
DL Synthetic Data Generator
CI/CD Pipelines
SECURED Federation Infrastructure
IntApp App Running inside the Secured Federation Infrastructure
DL Simple access to Toolbox Repository
SaaS Connection/Interaction with Services on the Backend

Figure 12 – Preliminary overview of the SECURED architecture, as presented in Deliverable D4.1.

This subsection describes how the software development of the components is managed in the Git repositories. To be able to properly manage a project of the scale of SECURED, it is paramount that unifying guidelines are set and followed for the implementations of the components. The following will delve into more detail about the programming language, documentation and testing.

7.2.1 Programming language, namespace and style

The main programming language for each component is Python[14]. After JavaScript, which is more focused on web applications, Python is the most popular high-level general-purpose programming language in 2023 on GitHub. As such, Python would be an ideal match for the developers in the SECURED project. It will also allow users of the SECURED Federated Infrastructure to easily understand the codebase and to easily adopt the SECURED Toolbox in their own software ecosystem. In case the requirements need parts of a component to be written in a different language, then this will be properly justified and documented.

All components fall under the `secured` Python package that represents the internal libraries of the SECURED project. Each component is assigned a unique name in this namespace. That is, the Data Anonymization Toolset is developed as the `secured.dans` package, De-Anonymization/Re-Identification as `secured.dean`, and Synthetic Data Generation as `secured.sdg`. Any helper functions that are developed independent of the components, such as functionality to parse datasets, fall under `secured.common`.

To streamline the development of the library based on the same environment and packages, a Python distribution such as Anaconda.Distribution[15] will be used. Anaconda is a distribution of the Python programming language, managing both package management and deployment to production. Most interestingly, Anaconda also allows non-Python dependencies to be included in the environment. This makes Anaconda a very versatile dependency manager and suitable for cases where components require system libraries for hardware support.

As potential users of the SECURED platform are medical practitioners who are not necessarily software developers themselves but do care about safeguarding their patients' privacy, the code needs to be understandable by itself. Besides selecting the right programming language, the code also needs to be written in a clear style. For this, we make use of Ruff[16]. Ruff is a Python linter and code formatter. That is, it can both detect code style

[14] https://www.python.org/
[15] https://www.anaconda.com/
[16] https://docs.astral.sh/ruff/

violations and fix them. Several plugins exist to integrate Ruff in modern-day code editors. Ruff's default rules will be consistently applied on the source code of all components.

7.2.2 Dependencies

After selecting the right algorithms for dataset anonymization and privacy-preserving synthetic data generation, these algorithms need to be efficiently implemented. Preference is given for already-established implementations that are compatible with SECURED's requirements. Even so, a wrapper or additional code changes might still be necessary to make it compatible with SECURED. Only if a base implementation is missing, should it be written completely from scratch.

All Anaconda-installable dependencies are listed in a `requirements.txt` in the root of the component's Git repository. This allows new adaptors of the SECURED library to easily initialize the Python environment in which the component should immediately be ready for operation. If external dependencies are required that cannot be installed by Anaconda, then this information needs to be documented in the `README` in the root of the Git repository.

Care must be taken that external dependencies are only used if their inclusion is absolutely necessary and if the developers are trusted. Using more dependencies increases the attack surface of the SECURED library. As the SECURED toolbox and services are used to process medical data that may not have been anonymized yet (because it still needs to be, for instance), this is vital to take into account. Preference should be given for packages with a strong and professional codebase that are actively maintained by a large community.

7.2.3 Relevant external libraries

Several external Python libraries will included/linked, to provide crucial routines for carrying out the anonymization, de-anonymization attacks and synthetic data generation over the discussed data types. The libraries play an important role in the pipeline for handling and analysing data. In general, Scipy, Numpy, and Pandas libraries offer the mathematical foundation for handling and preparing data for feature extraction and pattern analysis, and will be used for several data types. Other, more specialised libraries will be used for specific data types and formats, as described below.

ECG The WFDB library is pivotal for ECG data acquisition from repositories like PhysioNet and essential for collecting and processing ECG datasets for analysis. Matplotlib and Seaborn are essential tools for data visualisation, supporting initial analyses and interpreting model results.

EHR FHIR (Fast Healthcare Interoperability Resources) libraries are essential for dataset access and manipulation for EHR data, key to de-anonymization like pattern association. PyEHR facilitates EHR management and analysis, like medication-disease linkage and cluster analysis, which are essential for finding and isolating recognisable patterns in anonymized datasets.

Genomics For this kind of data we envision using the Plink format [17] for Single Nucleotide Polymorphism (SNP)s. For this kind of data, the package pandas-plink [18] offers all the functionallities to read this data and create a Pandas dataframe from it, which allows standard techniques to be applied over it.

Images There are two main imaging standards that are of interest in SECURED: DICOM and NIfTI. In order to read DICOM in python, Pydicom [19] comes handy as it provides functionalities to read both the image and the associated metadata (DICOM header) and these pieces of data can be converted later to more standard formats such as Numpy arrays. Alternatively, DICOM data can be read in matlab and exported to python for further processing. Data in NIfTI (Neuroimaging Informatics Technology Initiative) format can be read in matlab, and subsequently exported using matlab types (in .mat file format) for further processing by Python code. Matlab data files can be read in Python by using a library, such as scipy, which allows importing matlab

[17] https://www.cog-genomics.org/plink/1.9/formats

[18] https://pypi.org/project/pandas-plink/

[19] https://pydicom.github.io/

workspace variables and translating them into python variables. Such an approach is useful as it facilitates the utilization of available matlab code and implements a simple matlab-to-python interface. Furthermore, direct input and processing of NIfTI data is possible in python through the libraries nibabel [20] and nilearn [21].

7.2.4 Documentation

Each component needs to be documented in such a way that a prospective user of the SECURED platform, a medical practitioner who does not necessarily have a strong background in software engineering, should be able to understand the codebase from a high-level perspective. Especially for the SECURED toolbox, where practitioners can download the tools to run them on promise, the documentation must be complete and clear.

In the Python codebase, each file must contain a header with SECURED's copyright notice and the purpose of the file. Each function and class, if applicable, must be annotated by comments to explain its functionality. Any specific details about the input and output, such as any assumptions, must be documented in this way.

The documentation, i.e., the website and/or the PDF that contains all the information, is compiled by Sphinx[22]. When comments in the code are formatted in the reStructredText (RST) markup language, Sphinx is able to automatically create an overview of the Application Programming Interface (API). Additional pages can also be added as separate RST files to the Git repository. This is more suitable for e.g., tutorials.

7.2.5 Tests

The component's repository should contain a list of input datasets, component configuration and corresponding outputs to verify that the implementation in the user's environment is working as expected. Scripts to automate the verification should be part of the codebase as well. The documentation should properly state which output is expected and why. A measure of the quality of the tests, is the code coverage, i.e., which parts of the code are covered by the tests. This number should be as high as possible. Any part of the code not covered by the pre-supplied tests should be documented separately.

Ideally, external servers are setup that automatically perform these tests when changes to the codebase are submitted in the form of merge requests. This way, it can easily be verified whether old functionality is not being broken by the changes, or whether they introduce new performance regressions.

7.3 Scaling up the components

The first focus of the Python implementation of the three aforementioned components is on correct computation of the selected or crafted privacy-preserving algorithms. The selection of NumPy-native code should already allow SIMD optimizations to kick in while still maintaining correctness. The next step is to scale up the SECURED solution to be able to process more massive amounts of privacy-sensitive medical data. Although in the future algorithmic optimizations may seem necessary, several generic optimization techniques can be undertaken first. In this section, we discuss three techniques that are fully compatible with our Python-powered SECURED ecosystem: Process-based parallelism in Section 7.3.1, distributed computing in Section 7.3.2, and hardware accelerators in Section 7.3.3.

7.3.1 Process-based parallelism

Without depending on any additional hardware, Python has built-in support for the `multiprocessing` module[23] for multi-threading and multi-tasking. This allows the components to be developed as independent sub-tasks

[20] https://nipy.org/nibabel

[21] https://nilearn.github.io/

[22] https://www.sphinx-doc.org/

[23] https://docs.python.org/3/library/multiprocessing.html

that can be executed concurrently. The documentation also states that `multiprocessing` is not as much impacted by the Python Global Interpreter Lock, which generally hampers multi-threading[24].

7.3.2 Distributed computing

The distributed computing paradigm allows tasks that can be executed in a concurrent manner, to be executed on different physical nodes, in case the threat model of the medical practitioner allows for this. This enables the usage of microservices, such as depicted in Figure 2, with a package such as PyMS[25].

7.3.3 Hardware accelerators

Performance-critical parts of the code can be offloaded to specialized hardware for higher efficiency:

- Python bindings can be made with C and C++, from which specialized instructions such as SIMD support can be added, in case manual application is required.
- Through the same C and C++ bindings, accelerators can be connected that e.g., with High-Level Synthesis (HLS) can be explored on FPGAs.
- GPU bindings through e.g., `pycuda`[26]

[24] https://peps.python.org/pep-0703/

[25] https://python-microservices.github.io/home/

[26] https://documen.tician.de/pycuda/

8 Conclusions

This document summarises the current progress and next steps of Work Package 2 of the SECURED project. In line with the project structure, the work has been performed in a data-centric way, meaning that first the data types are located and then the related techniques are explored. This approach allowed us to identify which kind of data types or modalities are relevant and which kind of datasets are available to work with. Regarding the explored techniques, this deliverable presents progress on all the areas defined by the WP2 task: Anonymization, de-anonymization/re-identification, synthetic data generation and their efficient implementation.

Regarding data anonymisation, we have analysed several anonymisation tools and open-source libraries for creating the SECURED anonymisation toolset. In the first stage the DANS tool and Amnesia library are selected, providing k-anonymity and Differential privacy among others privacy models and mechanisms for anonymisation. These tools will be able to anonymise the different types of data such as tabular data, time-series data, image metadata and genomic data, relevant to the project. We have also provided the initial design of the architecture of the DANS 2.0 (SECURED anonymisation toolset), which integrates the different tools and open-source libraries to be deployed as microservices.

Complementary to the work on anonymization, the parallel strand of research investigating de-anonymization/re-identification attacks has progressed with the identification of the the data types (and associated target datasets, both within the project consortium and in public repositories) to be used. This has been followed by an analysis of the attack strategies to be pursued over the coming months, including membership attacks, inference matching attacks, attribute inference attacks, and linkage attacks.

Regarding synthetic data generation we have explored the relevant tools to create a baseline of tools to produce data. The current work was defined as three different axis: security, improvement and functionallity of the methods. These three axis have shown to be relevant for the task and for the related use cases. Security is shown to be important relevant as generated images can be very close to the original or convey patterns that can be later linked with the original. We have seen that Differential Privacy GANs show potential in this direction. On the other hand, to match with the use cases, generation of cancer tissue images and time series was explored. In particular, cancer tissue images pose a challenging problem due to their size. Finally, even though not as mature as the previous, two different functionallities are shown: Generation conditioning and missing MRI scan generation. These two functions will provide the user to address specific tasks and direct the generation given their requirements.

Finally, the private anonymization library key ideas, methodology and concepts were presented. This library will ensure that all components are written in an accessible way in Python, integrating it easily in widely-used data science tools in the medical sector. On the other hand, the setup will ensure to offer ways to scale up the components through different techniques such as parallelism, distributed computing or hardware accelerators.

Acknowledgements

Funded in part by the European Union (Grant Agreement Nr. 101095717, SECURED Project). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the Health and Digital Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

References

[1] K. Lekadir, R. Osuala, C. Gallin, N. Lazrak, K. Kushibar, G. Tsakou, S. Aussó, L. C. Alberich, K. Marias, M. Tsiknakis *et al.*, "Future-ai: guiding principles and consensus recommendations for trustworthy artificial intelligence in medical imaging," *arXiv preprint arXiv:2109.09658*, 2021.

[2] X. Li, P. S. Morgan, J. Ashburner, J. Smith, and C. Rorden, "The first step for neuroimaging data analysis: Dicom to nifti conversion," *Journal of neuroscience methods*, vol. 264, pp. 47–56, 2016.

[3] I. C. Moreira, I. Amaral, I. Domingues, A. Cardoso, M. J. Cardoso, and J. S. Cardoso, "Inbreast: toward a full-field digital mammographic database," *Academic radiology*, vol. 19, no. 2, pp. 236–248, 2012.

[4] "Csaw dataset," https://snd.gu.se/en/catalogue/dataset/2021-204-1.

[5] "Optimam dataset," https://www.cancerresearchhorizons.com/licensing-opportunities/optimam-mammography-image-database-omi-db.

[6] "Duke breast mri dataset," https://www.cancerimagingarchive.net/collection/duke-breast-cancer-mri/.

[7] "Patchcamelyon," https://patchcamelyon.grand-challenge.org/.

[8] "Chest x-ray 14 dataset," https://paperswithcode.com/dataset/chestx-ray14.

[9] "Chest x-ray node21 dataset," https://node21.grand-challenge.org/Data/.

[10] "Chest x-ray 14 dataset," https://stanfordmlgroup.github.io/competitions/chexpert/.

[11] "St. jude data repository," https://www.stjude.cloud/research-domains/pediatric-cancer/.

[12] SECURED project: D4.1-State of the Art and initial technical requirements, Fournaris, Apostolos Editor. 2023.

[13] L. Sweeney, "k-anonymity: A model for protecting privacy," *Int. J. Uncertain. Fuzziness Knowl. Based Syst.*, vol. 10, no. 5, pp. 557–570, 2002. [Online]. Available: https://doi.org/10.1142/S0218488502001648

[14] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "*L*-diversity: Privacy beyond *k*-anonymity," *ACM Trans. Knowl. Discov. Data*, vol. 1, no. 1, p. 3, 2007. [Online]. Available: https://doi.org/10.1145/1217299.1217302

[15] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in *Proceedings of the 23rd International Conference on Data Engineering, ICDE 2007, The Marmara Hotel, Istanbul, Turkey, April 15-20, 2007*, R. Chirkova, A. Dogac, M. T. Özsu, and T. K. Sellis, Eds. IEEE Computer Society, 2007, pp. 106–115. [Online]. Available: https://doi.org/10.1109/ICDE.2007.367856

[16] ARX - Data Anonymization Tool: A Comprehensive Software for Privacy-Preserving Microdata Publishing, 2022, Available online: https://arx.deidentifier.org, accessed on 01 January 2024.

[17] M. Terrovitis, N. Mamoulis, and P. Kalnis, "Privacy-preserving anonymization of set-valued data," *PVLDB*, vol. 1, no. 1, 2008.

[18] C. Gerlitz, A. Eriksson, and C. Hansson, "Anonymisation score for time series consumption data," in *27th International Conference on Electricity Distribution (CIRED 2023)*, vol. 2023. IET, 2023, pp. 428–432.

[19] M. Templ *et al.*, "Privacy of study participants in open-access health and demographic surveillance system data: Requirements analysis for data anonymization," *JMIR Public Health and Surveillance*, vol. 8, no. 9, p. e34472, 2022.

[20] J. Shen, S.-D. Bao, L.-C. Yang, and Y. Li, "The plr-dtw method for ecg based biometric identification," in *2011 33rd annual international conference of the IEEE engineering in medicine and biology society*. IEEE, 2011, pp. 5248–5251.

[21] L. Lange, T. Schreieder, V. Christen, and E. Rahm, “Privacy at risk: Exploiting similarities in health data for identity inference,” *ArXiv*, vol. abs/2308.08310, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:260926441

[22] B.-H. Kim and J.-Y. Pyun, “Ecg identification for personal authentication using lstm-based deep recurrent neural networks,” *Sensors*, vol. 20, no. 11, p. 3069, 2020.

[23] A. Fratini *et al.*, “Individual identification via electrocardiogram analysis,” *BioMedical Engineering OnLine*, vol. 14, no. 1, p. 78, December 2015. [Online]. Available: https://doi.org/10.1186/s12938-015-0072-y

[24] I. Landi, B. Glicksberg, H. Lee *et al.*, “Deep representation learning of electronic health records to unlock patient stratification at scale,” *npj Digital Medicine*, vol. 3, p. 96, 2020.

[25] J. Irvine and S. Israel, “A sequential procedure for individual identity verification using ecg,” *EURASIP Journal on Advances in Signal Processing*, vol. 2009, no. 1, 2009.

[26] N. Basil, S. Ambe, C. Ekhator, and E. Fonkem, “Health records database and inherent security concerns: A review of the literature,” *Cureus*, vol. 14, no. 10, p. e30168, Oct 2022.

[27] K. Goucher-Lambert and C. McComb, “Using hidden markov models to uncover underlying states in neuroimaging data for a design ideation task,” *Proceedings of the Design Society: International Conference on Engineering Design*, vol. 1, no. 1, p. 1873–1882, 2019.

[28] V. Ravindra and A. Grama, “De-anonymization Attacks on Neuroimaging Datasets,” in *Proceedings of the 2021 International Conference on Management of Data*. Virtual Event China: ACM, Jun. 2021, pp. 2394–2398. [Online]. Available: https://dl.acm.org/doi/10.1145/3448016.3457234

[29] R. Venkatesaramani, B. A. Malin, and Y. Vorobeychik, “Re-identification of individuals in genomic datasets using public face images,” *Science Advances*, vol. 7, no. 47, p. eabg3296, Nov. 2021. [Online]. Available: https://www.science.org/doi/10.1126/sciadv.abg3296

[30] K. Ayoz, E. Ayday, and A. E. Cicek, “Genome Reconstruction Attacks Against Genomic Data-Sharing Beacons,” *Proceedings on Privacy Enhancing Technologies*, vol. 2021, no. 3, pp. 28–48, Jul. 2021. [Online]. Available: https://petsymposium.org/popets/2021/popets-2021-0036.php

[31] N. Von Thenen, E. Ayday, and A. E. Cicek, “Re-identification of individuals in genomic data-sharing beacons via allele inference,” *Bioinformatics*, vol. 35, no. 3, pp. 365–371, Feb. 2019. [Online]. Available: https://academic.oup.com/bioinformatics/article/35/3/365/5056754

[32] C. Lippert, R. Sabatini, M. C. Maher, E. Y. Kang, S. Lee, O. Arikan, A. Harley, A. Bernal, P. Garst, V. Lavrenko, K. Yocum, T. Wong, M. Zhu, W.-Y. Yang, C. Chang, T. Lu, C. W. H. Lee, B. Hicks, S. Ramakrishnan, H. Tang, C. Xie, J. Piper, S. Brewerton, Y. Turpaz, A. Telenti, R. K. Roby, F. J. Och, and J. C. Venter, “Identification of individuals by trait prediction using whole-genome sequencing data,” *Proceedings of the National Academy of Sciences*, vol. 114, no. 38, pp. 10 166–10 171, Sep. 2017. [Online]. Available: https://pnas.org/doi/full/10.1073/pnas.1711125114

[33] E. Uffelmann, Q. Huang, N. Munung *et al.*, “Genome-wide association studies,” *Nat Rev Methods Primers*, vol. 1, p. 59, 2021.

[34] M. Gymrek, A. L. McGuire, D. Golan, E. Halperin, and Y. Erlich, “Identifying Personal Genomes by Surname Inference,” *Science*, vol. 339, no. 6117, pp. 321–324, Jan. 2013. [Online]. Available: https://www.science.org/doi/10.1126/science.1229566

[35] B. Nasri, M. Guennoun, and K. El-Khatib, “Using ecg as a measure in biometric identification systems,” in *Science and technology for humanity (TIC-STH), 2009 IEEE Toronto international conference*, 2009, pp. 28–33.

[36] K. P. Seastedt, P. Schwab, Z. O’Brien, E. Wakida, K. Herrera, P. G. F. Marcelo, L. Agha-Mir-Salim, X. B. Frigola, E. B. Ndulue, A. Marcelo *et al.*, “Global healthcare fairness: We should be sharing more, not less, data,” *PLOS Digital Health*, vol. 1, no. 10, p. e0000102, 2022.

[37] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramer, B. Balle, D. Ippolito, and E. Wallace, "Extracting training data from diffusion models," in *32nd USENIX Security Symposium (USENIX Security 23)*, 2023, pp. 5253–5270.

[38] T. Stadler, B. Oprisanu, and C. Troncoso, "Synthetic data–anonymisation groundhog day," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 1451–1468.

[39] R. Osuala, G. Skorupko, N. Lazrak, L. Garrucho, E. García, S. Joshi, S. Jouide, M. Rutherford, F. Prior, K. Kushibar *et al.*, "medigan: a python library of pretrained generative models for medical image synthesis," *Journal of Medical Imaging*, vol. 10, no. 6, p. 061403, 2023.

[40] "Sdv: Synthetic data vault," https://sdv.dev/.

[41] "Gretel: The multimodal synthetic data platform for developers," https://gretel.ai/.

[42] J. Jordon, J. Yoon, and M. van der Schaar, "PATE-GAN: generating synthetic data with differential privacy guarantees," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. [Online]. Available: https://openreview.net/forum?id=S1zk9iRqF7

[43] Y. Long, B. Wang, Z. Yang, B. Kailkhura, A. Zhang, C. A. Gunter, and B. Li, "G-pate: Scalable differentially private data generator via private aggregation of teacher discriminators," 2021.

[44] D. Chen, T. Orekondy, and M. Fritz, "GS-WGAN: A gradient-sanitized approach for learning differentially private generators," *CoRR*, vol. abs/2006.08265, 2020. [Online]. Available: https://arxiv.org/abs/2006.08265

[45] I. Mironov, "Rényi differential privacy," in *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*. IEEE, aug 2017. [Online]. Available: https://doi.org/10.1109%2Fcsf.2017.11

[46] 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*. ACM, oct 2016. [Online]. Available: https://doi.org/10.1145%2F2976749.2978318

[47] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014. [Online]. Available: https://arxiv.org/abs/1406.2661

[48] M. Shannon, B. Poole, S. Mariooryad, T. Bagby, E. Battenberg, D. Kao, D. Stanton, and R. Skerry-Ryan, "Non-saturating gan training as divergence minimization," 2020. [Online]. Available: https://arxiv.org/abs/2010.08029

[49] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," 2017. [Online]. Available: https://arxiv.org/abs/1701.07875

[50] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014. [Online]. Available: https://arxiv.org/abs/1411.1784

[51] L. Xie, K. Lin, S. Wang, F. Wang, and J. Zhou, "Differentially private generative adversarial network," 2018. [Online]. Available: https://arxiv.org/abs/1802.06739

[52] X. Zhang, S. Ji, and T. Wang, "Differentially private releasing via deep generative model (technical report)," 2018. [Online]. Available: https://arxiv.org/abs/1801.01594

[53] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, G. Klambauer, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a nash equilibrium," *CoRR*, vol. abs/1706.08500, 2017. [Online]. Available: http://arxiv.org/abs/1706.08500

[54] S. Barratt and R. Sharma, "A note on the inception score," 2018.

[55] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015 - 18th International Conference Munich, Germany, October 5 - 9, 2015, Proceedings, Part III*, ser. Lecture Notes in Computer Science, N. Navab, J. Hornegger, W. M. W. III, and A. F. Frangi, Eds., vol. 9351. Springer, 2015, pp. 234–241. [Online]. Available: https://doi.org/10.1007/978-3-319-24574-4_28

[56] “Camelyon16,” https://camelyon16.grand-challenge.org/.

[57] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.

[58] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 6626–6637. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/8a1d694707eb0fefe65871369074926d-Abstract.html

[59] M. Mehdi and O. Simon, “Conditional generative adversarial nets,” 2014, cite arxiv:1411.1784. [Online]. Available: http://arxiv.org/abs/1411.1784

[60] A. Saha, M. R. Harowicz, L. J. Grimm, C. E. Kim, S. V. Ghate, R. Walsh, and M. A. Mazurowski, “A machine learning approach to radiogenomics of breast cancer: a study of 922 subjects and 529 dce-mri features,” *British journal of cancer*, vol. 119, no. 4, pp. 508–516, 2018.

[61] S. Bakas, M. Reyes, A. Jakab, S. Bauer, M. Rempfler, A. Crimi, R. Shinohara, C. Berger, S. Ha, M. Rozycki *et al.*, “Identifying the best machine learning algorithms for brain tumor segmentation,” *progression assessment, and overall survival prediction in the BRATS challenge*, vol. 10, 2018.